МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій,
обліку та фінансів
Кафедра комп'ютерних технологій
і моделювання систем

Кваліфікаційна робота
на правах рукопису

Чепіга Вадим Сергійович
(прізвище, ім'я, по батькові здобувача освіти)

УДК 004.056:004.73

КВАЛІФІКАЦІЙНА РОБОТА

Використання машинного навчання для виявлення аномалій у мережевому трафіку

(тема роботи)

125 Кібербезпека та захист інформації

(шифр і назва спеціальності)

Подається на здобуття освітнього ступеня магістр

кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи
Веретюк Сергій Михайлович
(прізвище, ім'я, по батькові)
к.т.н., доцент
(науковий ступінь, вчене звання)

Житомир – 2024

Висновок кафедри _____
за результатами попереднього захисту: _____

Протокол засідання кафедри _____
№ ____ від «_____» _____ 20____ р.

Завідувач кафедри _____

_____ _____ _____
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)
«_____» _____ 20____ р.

Результати захисту кваліфікаційної роботи

Здобувач вищої освіти _____
захистив (ла)
(прізвище ,ім'я, по батькові)
кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою _____
за шкалою ECTS _____
за національною шкалою _____

Секретар ЕК

_____ _____ _____
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)

АНОТАЦІЯ

Чепіга В.С. Використання машинного навчання для виявлення аномалій у мережевому трафіку. – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня магістр за спеціальністю 125 – Кібербезпека та захист інформації. – Поліський національний університет, Житомир, 2024.

Напрямок досліджень дипломної роботи пов'язаний із виявленням аномалій у мережевому трафіку та можливістю використання методу головних компонент для підвищення ефективності детекції. У процесі дослідження проведено аналіз існуючих методів виявлення аномалій, реалізовано метод головних компонент для зменшення розмірності даних, обрано оптимальний метод кластеризації для виявлення аномалій та розроблено практичні рекомендації щодо застосування методу виявлення аномалій.

Ключові слова: АНОМАЛІЇ, АНАЛІЗ МЕРЕЖЕВОГО ТРАФІКУ, МЕТОД ГОЛОВНИХ КОМПОНЕНТ, КЛАСТЕРИЗАЦІЯ, МАШИННЕ НАВЧАННЯ.

SUMMARY

Chepiga V.S. Using Machine Learning for Anomaly Detection in Network Traffic. - Qualification work in manuscript form.

Qualification work for the master's degree, specialty 125 - Cybersecurity and Information Protection. - Polissia National University, Zhytomyr, 2024.

The research direction of the thesis is related to anomaly detection in network traffic and the potential use of the principal component method to improve detection efficiency. During the research, an analysis of existing anomaly detection methods was conducted, the principal component method was implemented to reduce data dimensionality, an optimal clustering method for anomaly detection was selected, and practical recommendations for applying the anomaly detection method were developed.

Keywords: ANOMALIES, NETWORK TRAFFIC ANALYSIS, PRINCIPAL COMPONENT ANALYSIS, CLUSTERING, MACHINE LEARNING.

ЗМІСТ

ВСТУП

З розвитком інформаційних технологій та активним використанням мереж питання безпеки інформаційних систем набуває дедалі більшої актуальності. Мережі є основним середовищем для передачі даних, що робить їх уразливими до атак з боку зловмисників, які постійно шукають нові способи несанкціонованого доступу до конфіденційної інформації. Аномалії у мережевому трафіку є однією з ключових ознак порушення цілісності інформаційних систем або спроби несанкціонованого доступу, що можуть свідчити про кібератаки, зловмисні дії чи неправомірне використання ресурсів.

Традиційні методи виявлення аномалій, засновані на сигнатурах та правилах, не завжди виявляються ефективними у протидії новим типам атак, що постійно з'являються в сучасному кіберпросторі [1].

Актуальність проблеми визначається необхідністю впровадження більш ефективних підходів до виявлення аномалій, які б дозволяли не лише ідентифікувати вже відомі типи атак, а й передбачати нові загрози. Використання машинного навчання у цій сфері відкриває нові можливості, оскільки такі моделі здатні аналізувати великий обсяг даних, виявляти приховані патерни та швидко адаптуватися до змін у трафіку [2].

Наукова новизна полягає у розроблені методу виявлення аномалій в трафіку на основі поєднання методу головних компонент та методів кластеризації, що дозволяє спростити аналіз багатовимірних даних і підвищити ефективність виявлення загроз.

Практична цінність полягає у можливості інтеграції запропонованого методу у сучасні системи моніторингу та безпеки для автоматизованого виявлення аномального трафіку та підвищення рівня кібербезпеки.

За темою кваліфікаційної роботи було опубліковано наукові тезиси, а саме:

- Чепіга В. С. Підходи до аналізу аномалій у мережевому трафіку. Штучний інтелект і безпека : збірник матеріалів науково-

практичної конференції, 19-21 листопада 2024 р. Київ : ІПМЕ ім. Г.Є. Пухова, 2024. С. 33.

- Чепіга В. С. Застосування методу головних компонент для аналізу аномалій в мережевому трафіку. Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій : тези доповідей XII Міжнародної науково-практичної конференції, 10-12 грудня 2024 р. Запоріжжя : Національний університет «Запорізька політехніка», 2024. С. 319-321.
- Чепіга В. С. Реалізація методу PCA для виявлення аномального трафіку в мережі. Безпека, технології, інновації: нові горизонти : збірник праць учасників міжфакультетської науково-практичної інтернет-конференції здобувачів вищої освіти і молодих вчених, 12 листопада 2024 р. Житомир : Поліський національний університет, 2024. С. 19-22.

Об’єктом дослідження даного дипломного проекту є процеси виявлення аномалій в трафіку.

Предметом дослідження є методи та алгоритми машинного навчання для виявлення аномалій в мережевому трафіку.

Основною метою роботи є розроблення методу виявлення аномалій у мережевому трафіку на основі методу головних компонент.

Дипломний проект складається з наступних розділів: вступ, теоретичні основи виявлення аномалій у мережевому трафіку, виявлення аномалій в трафіку на основі методу головних компонент, дослідження ефективності методу виявлення аномалій в трафіку.

1 ТЕОРЕТИЧНІ ОСНОВИ ВИЯВЛЕННЯ АНОМАЛІЙ У МЕРЕЖЕВОМУ ТРАФІКУ

1.1 Поняття аномалії та її характеристики

Мережевий трафік — це сукупність даних, що передаються мережею за певний проміжок часу. Трафік може включати різні типи даних, як-от текстові повідомлення, файли, мультимедійний контент, запити до серверів та відповіді на них [3].

Трафік як процес має значну кількість властивостей, характеристик та ознак. Загалом, для аналізу трафіку використовують 41 широко розповсюджену ознаку трафіку [4].

Аномалія в мережевому трафіку – це дані або спостереження стану системи, що виходять за межі її локальної чи глобальної норми. Аномалія може проявлятися як рідкісна подія або відхилення від звичного шаблону в певний момент часу чи в конкретному контексті. Виникнення аномалії може бути зумовлене зовнішніми факторами, такими як помилки датчиків або кібератаки. Основна мета алгоритму виявлення аномалій – ідентифікувати такі відхилення та, за можливості, встановити їхні причини [5].

Для точного виявлення аномалій важливо визначити, що вважати нормальною поведінкою. Наприклад, висока активність мережі у певні години роботи компанії може виглядати як аномалія, якщо не враховувати контекст (робочі години, регулярні резервні копії тощо). Тому аномалія завжди визначається у співвідношенні з базовою моделлю нормальної поведінки мережі, яка враховує сезонні коливання, пікові години активності та природні зміни.

Для виявлення аномального трафіку важливо визначити, що слід вважати нормальною поведінкою. Наприклад, висока активність мережі у певні години роботи компанії може виглядати як аномалія, якщо не враховувати контекст (робочі години, регулярні резервні копії тощо). Тому аномалія завжди визначається у співвідношенні з базовою моделлю нормальної поведінки мережі,

яка враховує сезонні коливання, пікові години активності та природні зміни. Для цього було наведено 41 широко розповсюджену ознаку трафіку [4], див. табл. 1.

Таблиця 1 – Ознаки трафіку

Назва	Значення	Опис
f1	duration	Тривалість з'єднання
f2	protocol_type	Тип протоколу з'єднання: TCP, UDP, ICMP
f3	service	Сервіс: http, ftp, smtp, telnet тощо
f4	flag	Статус з'єднання: SF, S0, S1, S2, S3, OTH, REJ, RSTO, RSTOS0, SH, RSTRH, SHR
f5	src_bytes	Байти, відправлені під час з'єднання
f6	dst_bytes	Байти, отримані під час з'єднання
f7	land	Якщо IP-адреси та порти відправника і одержувача однакові, значення 1, інакше 0
f8	wrong_fragment	Сума пакетів з неправильними контрольними сумами у з'єднанні
f9	urgent	Сума термінових пакетів у з'єднанні (активований прапорець "urgent")
f10	hot	Сума "гарячих" дій у з'єднанні, таких як вхід у системні каталоги, створення або виконання програм
f11	num_failed_logins	Кількість невдалих спроб входу у з'єднанні
f12	logged_in	Якщо вхід успішний, значення 1, інакше 0
f13	num_compromised	Сума випадків появи помилки "не знайдено" у з'єднанні
f14	root_shell	Якщо root отримав shell, значення 1, інакше 0
f15	su_attempted	Якщо була використана команда "su", значення 1, інакше 0
f16	num_root	Сума операцій, виконаних від імені root у з'єднанні
f17	num_file_creations	Сума створених файлів у з'єднанні
f18	num_shells	Кількість входів звичайних користувачів
f19	num_access_files	Сума операцій у контрольних файлах у з'єднанні
f20	num_outbound_cmds	Сума вихідних команд під час FTP-сесії
f21	is_hot_login	Якщо користувач заходить як root або adm
f22	is_guest_login	Якщо користувач заходить як гість, анонім або відвідувач
f23	count	Сума з'єднань до однієї IP-адреси
f24	srv_count	Сума з'єднань до одного номера порту
f25	serror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): S0, S1, S2 або S3 серед з'єднань, об'єднаних у count (f23)

f26	srv_serror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): S0, S1, S2 або S3 серед з'єднань, об'єднаних у srv_count (f24)
f27	rerror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): REJ серед з'єднань, об'єднаних у count (f23)
f28	srv_error_rate	Відсоток з'єднань, у яких був активований прапорець (f4): REJ серед з'єднань, об'єднаних у count (f23)
f29	same_srv_rate	Відсоток з'єднань до одного й того самого сервісу серед з'єднань, об'єднаних у count (f23)
f30	diff_srv_rate	Відсоток з'єднань до різних сервісів серед з'єднань, об'єднаних у count (f23)
f31	srv_diff_host_rate	Відсоток з'єднань до різних цільових машин серед з'єднань, об'єднаних у srv_count (f24)
f32	dst_host_count	Сума з'єднань до однієї IP-адреси
f33	dst_host_srv_count	Сума з'єднань до одного номера порту
f34	dst_host_same_srv_rate	Відсоток з'єднань до одного й того самого сервісу серед з'єднань, об'єднаних у dst_host_count (f32)
f35	dst_host_diff_srv_rate	Відсоток з'єднань до різних сервісів серед з'єднань, об'єднаних у dst_host_count (f32)
f36	dst_host_same_src_port_rate	Відсоток з'єднань до одного й того самого джерела порту серед з'єднань, об'єднаних у dst_host_srv_count (f33)
f37	dst_host_srv_diff_host_rate	Відсоток з'єднань до різних цільових машин серед з'єднань, об'єднаних у dst_host_srv_count (f33)
f38	dst_host_serror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): S0, S1, S2 або S3 серед з'єднань, об'єднаних у dst_host_count (f32)
f39	dst_host_srv_serror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): S0, S1, S2 або S3 серед з'єднань, об'єднаних у dst_host_srv_count (f33)
f40	dst_host_rerror_rate	Відсоток з'єднань, у яких був активований прапорець (f4): REJ серед з'єднань, об'єднаних у dst_host_count (f32)
f41	dst_host_srv_error_rate	Відсоток з'єднань, у яких був активований прапорець (f4): REJ серед з'єднань, об'єднаних у dst_host_srv_count (f33)

1.2 Взаємозв'язок аномалій у трафіку та кібератак

DDoS (Distributed Denial of Service) є одним із найпоширеніших видів кібератак, які проявляються у вигляді аномалій у мережевому трафіку. Аномальний трафік, що супроводжує DDoS-атаку, складається з великої кількості запитів, які надсилаються до цільового вузла. Ці запити часто не несуть корисного навантаження, а їх кількість перевищує можливості обробки

системою, що призводить до заповнення пам'яті та зростання черги обслуговування. Як наслідок, відбувається різке зниження якості послуг мережі або навіть повна відмова в обслуговуванні.

Реалізація DDoS-атак значною мірою залежить від використання бот-мереж, які складаються з вузлів, заражених шкідливим програмним забезпеченням. Ці вузли автоматично генерують і надсилають «порожні» запити до цільового сервера, що створює лавиноподібне зростання трафіку. Така активність є типовим проявом аномалії, оскільки значно відрізняється від нормального мережевого трафіку а отже, DDoS-атаки можна розглядати як одну з основних причин аномалій у мережевому трафіку, а їх аналіз є важливим інструментом для своєчасного виявлення та нейтралізації подібних загроз.

Крім DDOS, є і інші типи атак, які можна виявити через аналіз аномалій, зокрема:

- брутфорс-атаки - супроводжуються великою кількістю спроб автентифікації за короткий час;
- використання експлойтів - змінює характер обміну даними між клієнтом і сервером;
- витік даних - може бути помічений через аномальні обсяги вихідного трафіку.

Компанією Cloudflare було опубліковано звіт про кампанію гіперо’ємних DDoS-атак за 2024 рік, де у першій половині 2024 року автономні системи захисту від DDoS-атак Cloudflare автоматично виявили та пом’якшили 8,5 мільйонів DDoS-атак: 4,5 мільйона в першому кварталі та 4 мільйони в другому кварталі. У третьому кварталі системи пом’якшили майже 6 мільйонів DDoS-атак, довівши загальну кількість DDoS-атак до 14,5 мільйонів з початку року. Це в середньому близько 2200 DDoS-атак щогодини [6].

З цих атак Cloudflare пом’якшив понад 200 гіпероб’ємних DDoS-атак мережевого рівня, швидкість яких перевищувала 1 Тбіт/с або 1 Біт/с. Максимальна швидкість найбільших атак становила 3,8 Тбіт/с і 2,2 Біт/с (див. рисунок 1.1) [6].

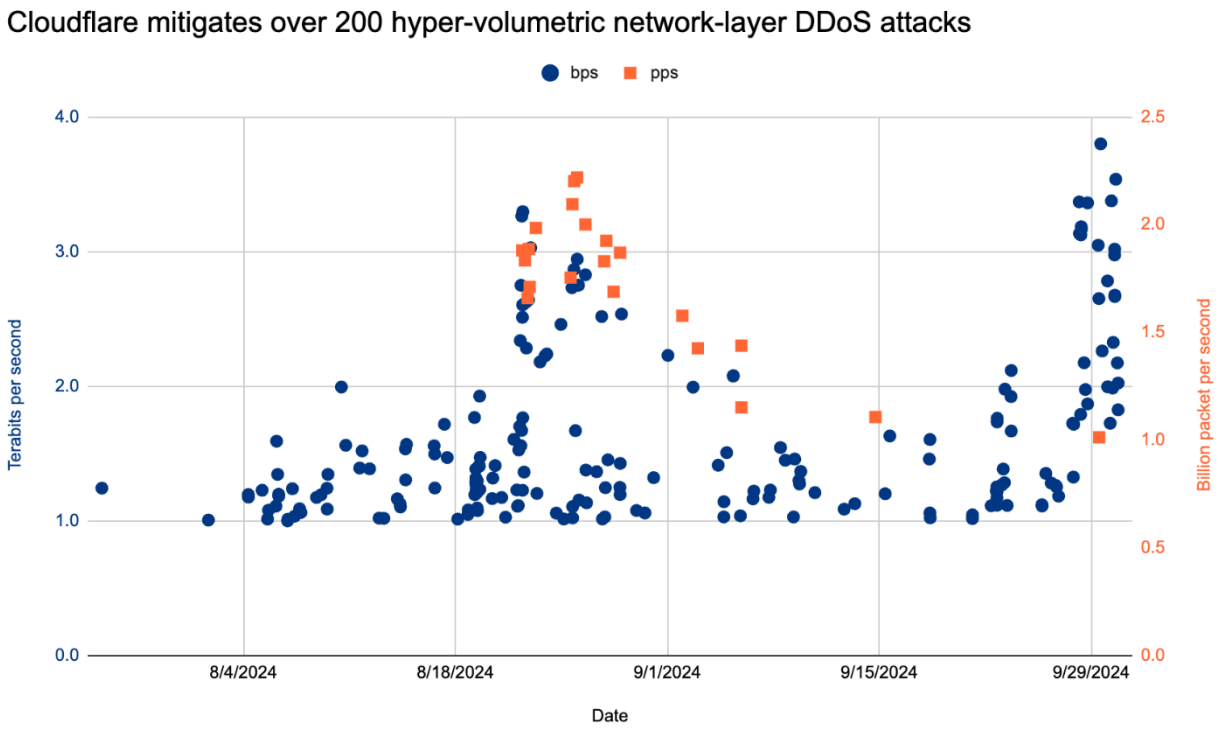

Рисунок 1.1 – Розподіл гіпероб'ємних DDoS-атак у часі

Cloudflare використовує XDP для вибіркового аналізу пакетів з метою виявлення підозрілих атрибутів, які можуть свідчити про атаку. Вибіркові дані включають такі поля, як IP-адреса джерела, порт джерела, IP-адреса призначення, порт призначення, протокол, TCP-прапори, порядковий номер, опції, швидкість передачі пакетів тощо. Цей аналіз здійснюється denial of service daemon (dosd). Dosd має безліч фільтрів, які на основі ретельно розроблених евристик визначають, коли слід починати заходи з пом'якшення наслідків [7].

Засоби захисту також включають розвідку про загрози в реальному часі, профілювання трафіку та класифікацію машинного навчання як частину адаптивного захисту від DDoS для пом'якшення аномалій трафіку [7].

Адаптивний захист від DDoS атак забезпечує такі типи захисту:

- Адаптивний захист від DDoS для користувацьких агентів: Виявляє та нейтралізує трафік, що відхиляється від основних користувацьких агентів, зафіксованих мережею Cloudflare. Профіль користувацьких агентів формується на основі даних всієї мережі Cloudflare, а не лише зони клієнта.

- Адаптивний захист від DDoS для протоколів: Виявляє та нейтралізує трафік, що відхиляється від протокольного профілю вашого трафіку. Профіль розраховується як глобальна швидкість для кожного з ваших префіксів.

1.3 Методи виявлення аномалій у мережевому трафіку

Дослідження наявних методів і розв'язків задачі детекції аномалій мережевого трафіку охоплює аналіз різних технологій і підходів, використовуваних для ідентифікації неправильних, незвичайних або зловмисних дій у мережах. Існує кілька методів детекції аномалій, серед яких виділяються статистичні методи, методи машинного навчання та гібридні методи, кожен з яких має свої переваги та недоліки.

Статистичні методи базуються на аналізі параметрів мережевого трафіку, де нормальний трафік описується за допомогою статистичних характеристик, таких як середнє значення, дисперсія, кореляції між різними параметрами. Аномалія визначається як відхилення від встановленої норми або ймовірнісного розподілу.

Типовими статистичними методами є:

- Метод контрольних карт (Control Chart) - використовується для моніторингу відхилень від встановлених меж.
- Метод порівняння середніх - застосовується для виявлення змін у середніх значеннях певних параметрів мережевого трафіку, накшталт кількості пакетів за одиницю часу.

Переваги:

- низька обчислювальна вартість;
- прості у впровадженні та інтерпретації результатів.

Недоліки:

- ефективність різко знижується при зростанні складності мережевих атак;
- потреба точних знань про "нормальний" трафік.

Методи машинного навчання дозволяють автоматично виявляти аномалії без потреби в жорстких припущеннях про характер "нормального" трафіку. Використовуються різноманітні алгоритми, які навчаються на великих обсягах даних, що дозволяє їм ефективно розпізнавати складні патерни аномалій.

Типовими методами машинного навчання є:

- Алгоритми кластеризації (K-means, DBSCAN) - використовуються для групування трафіку в класи, де кожен клас відповідає нормальному або аномальному поведінці;
- Методи глибинного навчання (нейронні мережі) - дозволяють побудувати складні моделі, здатні виявляти навіть складні аномалії, які важко визначити традиційними методами;
- Алгоритми на основі дерев рішень (Random Forest, Decision Trees) - використовуються для побудови моделей, які розпізнають аномальні патерни на основі наданих ознак.

Переваги:

- висока точність виявлення складних аномалій;
- здатність навчатися та адаптуватися до нових умов.

Недоліки:

- висока обчислювальна складність;
- потреба значних обсягів даних для навчання моделей.

Гібридні методи поєднують переваги статистичних і машинних методів, забезпечуючи більш точне і швидке виявлення аномалій.

Типовими гібридними методами є:

- використання алгоритмів машинного навчання для попередньої обробки даних та статистичних методів для подальшого аналізу
- комбінування кількох моделей машинного навчання для досягнення більшої стабільності та точності виявлення аномалій.

Переваги:

- здатність комбінувати точність моделей машинного навчання з простотою та ефективністю статистичних методів.

Недоліки:

- вимагають високих обчислювальних ресурсів.

Таким чином, виходить, що тема дослідження дійсно є актуальною з наступних причин:

- Кількість кібератак, що постійно зростає, зокрема, DDoS-атаки, які супроводжуються аномаліями в мережевому трафіку, залишаються однією з найпоширеніших загроз. За даними компанії Cloudflare спостерігались навіть гіпероб'ємні атаки, які досягали рекордної швидкості 3,8 Тбіт/с, що підтверджує необхідність аналізу таких аномалій для захисту систем.
- Аномалії у трафіку можуть бути ознакою не лише DDoS-атак, але й інших видів загроз, таких як брутфорс-атаки, витоки даних чи використання експлойтів. Вчасне їх виявлення допомагає мінімізувати потенційні втрати та підвищує кіберстійкість мереж.
- Еволюція сучасних методів аналізу, таких як машинне навчання і гібридні підходи, дозволяють значно покращити ефективність детекції аномалій.

Висновок до першого розділу

Було розглянуто теоретичні основи виявлення аномалій у мережевому трафіку, зокрема поняття аномалії, її характеристики та взаємозв'язок із кібератаками. Було проаналізовано основні параметри трафіку, які можуть свідчити про наявність аномалій, та встановлено зв'язок між такими відхиленнями і поширеними загрозами, як DDoS-атаки. Розглянуто сучасні підходи до детекції аномалій, включно зі статистичними, машинними та гібридними методами.

2 ВИЯВЛЕННЯ АНОМАЛІЙ В ТРАФІКУ НА ОСНОВІ МЕТОДУ ГОЛОВНИХ КОМПОНЕНТ

2.1 Використання методу головних компонент для кластеризації даних та для аналізу часових рядів

Метод головних компонент (Principal Component Analysis, PCA) є ефективним підходом для обробки високовимірних даних, дозволяючи зменшити розмірність простору даних, виділивши найменшу кількість головних компонент. Виділені головні компоненти можуть містити максимум характеристик вихідних даних і втрачати якомога менше інформації. PCA допомагає виділити основні напрямки змін у даних, що дає змогу виявити відхилення від стандартної поведінки. Використання PCA для аналізу мережевого трафіку дозволяє створити модель нормальної активності, а відхилення від цієї моделі вказують на можливу аномальну активність.

Вперше метод був застосований у стисненні даних, обробці зображень, нейронних мережах, інтелектуальному аналізі даних та розпізнаванні образів. Широке використання PCA в основному пов'язане з його трьома важливими характеристиками. По-перше, після стиснення даних високої розмірності в набір даних низької розмірності, середня квадратична похибка реконструйованих даних обернено пропорційна розмірності. По-друге, модель є стабільною без коригування параметрів у процесі роботи. По-третє, для заданих параметрів легко проводити компресію та декомпресію [8].

Основні етапи PCA:

- Стандартизація даних;
- Обчислення коваріаційної матриці;
- Знаходження власних векторів та власних значень;
- Вибір головних компонент;
- Трансформація даних.

Стандартизація являє собою перетворення кожної змінної так, щоб вона мала середнє значення 0 і стандартне відхилення 1. В результаті такої стандартизації кожна змінна матиме однакову вагу для подальшого аналізу.

Коваріаційна матриця (або коваріаційна таблиця) - це квадратна матриця, яка складена з попарних коваріацій і дисперсій двох або більше величин. Коваріаційна матриця демонструє, наскільки змінні змінюються разом [9].

Для обчислення власних значень і власних векторів використовується коваріаційна матриця. Власні значення визначають, яку частину загальної дисперсії даних пояснює кожен власний вектор. Власні вектори визначають напрямки дисперсії даних, вони ж головні компоненти. В PCA обираються власні вектори, що відповідають найбільшим власним значенням, для зменшення розмірності даних при збереженні максимальної варіації.

Щоб визначити, які компоненти залишити, обчислюється частка дисперсії, яку пояснює кожне власне значення. Після знаходження дисперсій головні компоненти, що мають високу частку, зберігають всі варіації даних, і нові значення даних у просторі головної компоненти, тоді як інші компоненти з малою часткою або її відсутністю можна ігнорувати, оскільки вони не додають інформації. Такий процес дозволяє спростити аналіз даних, зменшивши розмірність без втрати суттєвої інформації.

Після вибору головної компоненти ми можемо проектувати початкові стандартизовані дані у простір головних компонент. Замість декількох змінних тепер є одна змінна, що представляє дані у просторі головних компонент. Вся інформація про варіацію даних зберігається в одному значенні для кожного спостереження. Результуючі значення можна використати для побудови одновимірного графіка або кластеризації, зберігаючи основну інформацію про варіацію даних у значно спрощеному вигляді.

Переваги PCA для аналізу мережевого трафіку:

- Дозволяє обробляти великомасштабні дані, відкидаючи менш значущі компоненти без значних втрат інформації;

- Дозволяє відкинути незначні компоненти, які часто є джерелом шуму;
- Допомагає виявляти ключові ознаки аномалій, що спрощує їх класифікацію;
- Завдяки зменшенню кількості вимірів, алгоритми машинного навчання працюють швидше.

PCA може бути використаний як попередній етап перед кластеризацією. Зменшення розмірності даних дозволяє алгоритмам кластеризації, таким як K-means або DBSCAN, працювати швидше і точніше. Також, у задачах аналізу аномалій у мережевому трафіку, часові ряди (наприклад, зміни трафіку у часі) можуть бути складними. PCA допомагає зосередитися на найсуттєвіших змінах у поведінці мережі.

2.2 Збір та підготовка даних. Критерії належності до класу аномалії

Для проведення аналізу було використано набір даних Network Anomaly Detection Dataset, доступний на платформі Kaggle. Набір містить інформацію про різні аспекти мережевого трафіку, зокрема: обсяг вхідного та вихідного трафіку, кількість пакетів, помилки передачі тощо [10].

Набір даних включає 35 атрибутів (їхню візуалізацію дивитися на рис.2.1), що описують параметри мережевого трафіку, такі як:

- Отримані та передані байти;
- Tcp ознаки;
- Udp ознаки;
- Ip ознаки;
- ICMP ознаки.

Аналіз статистичних властивостей характеристик реального трафіку наведено на рис. 2.1 та табл.2.

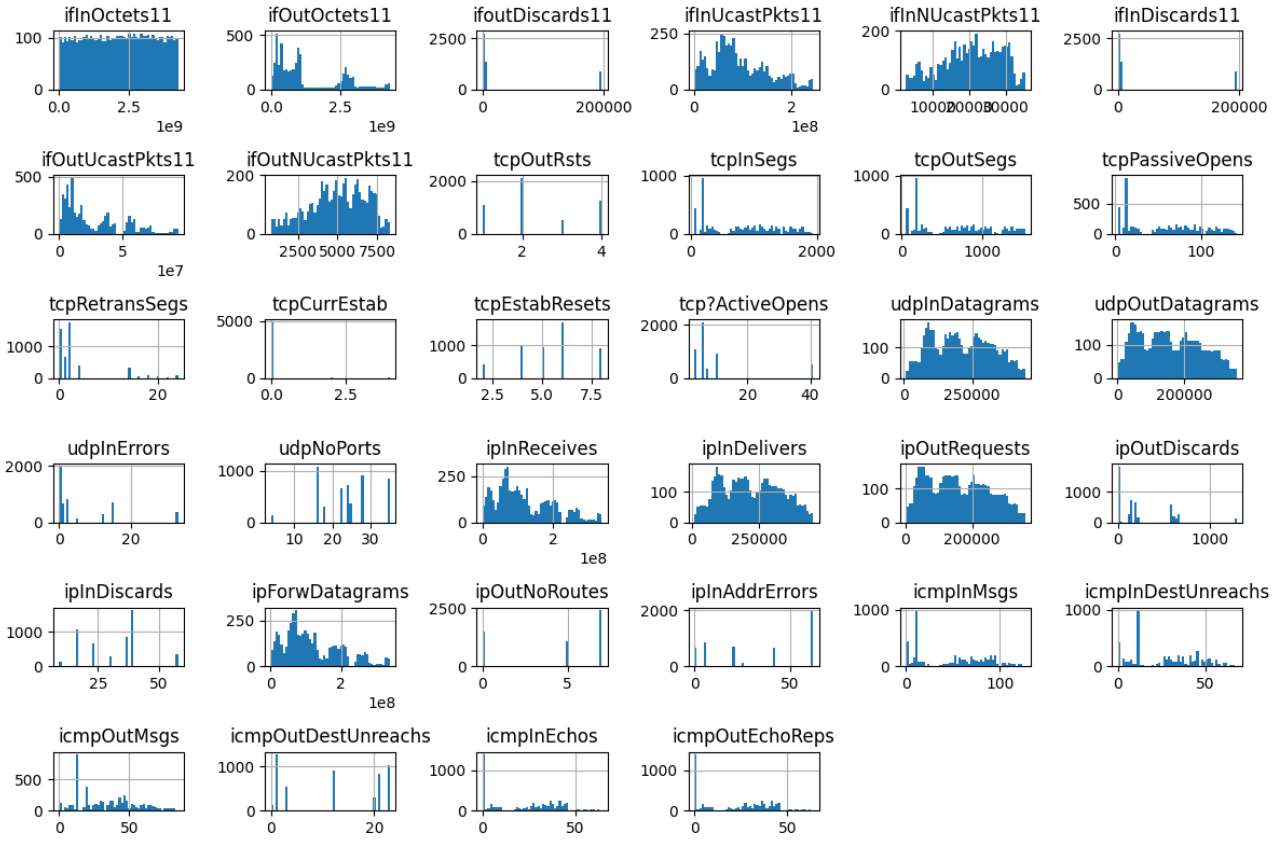

Рисунок 2.1 – Числові дані набору Kaggle у вигляді діаграм

Таблиця 2 – Статистичні властивості трафіку

Розподіл	Параметри
Рівномірний	ifInOctets11, ipInReceives, udpInDatagrams, ipOutRequests, ifOutNUcastPkts11, tcpInSegs, tcpOutSegs, ipForwDatagrams
Логнормальний	ifOutOctets11, ifOutUcastPkts11, tcpPassiveOpens, ipInDelivers
Мультимодальний	ifOutDiscards11, ifInUcastPkts11, icmpInMsgs, icmpInDestUnreachs
Нормальний	ifInNUcastPkts11
Вибірковий	ifInDiscards11, tcpOutRsts, ipOutDiscards, icmpOutMsgs, udpNoPorts, ipInAddrErrors
Експоненціальний	icmpOutMsgs, icmpOutEchoReps

Продемонстровані атрибути дозволяють отримати широкий спектр характеристик трафіку, необхідний для ідентифікації аномалій.

Етапи підготовки даних включали:

1. Перетворення міток класу

- Мітки класів були перетворені у числовий формат для зручності обробки (normal = 0, attack = 1).

2. Відокремлення ознак і міток

- Дані були розділені на дві частини(ознаки та мітки).

3. Стандартизація даних

- Для забезпечення коректності результатів PCA необхідно стандартизувати дані, щоб усі параметри мали середнє значення 0 та стандартне відхилення 1. Стандартизація виконувалася за допомогою класу StandardScaler з бібліотеки sklearn. Це гарантує, що параметри з різними одиницями виміру матимуть однакову вагу під час аналізу.

Для класифікації трафіку на основі методів кластеризації та PCA використовується підхід, що включає такі критерії:

1. Аналіз розподілу головних компонент

Метод PCA був застосований для зменшення розмірності ознак до двох головних компонент. Це дозволило побачити залежності між даними у спрощеному просторі.

2. Кластеризація методом K-середніх

На основі двох головних компонент було виконано кластеризацію із заданою кількістю кластерів (k=8). Один кластер відповідає нормальному трафіку, інші – аномальному.

3. Візуалізація результатів кластеризації

Побудова двовимірного графіка головних компонент з розфарбуванням точок у відповідності до їх належності до кластерів дозволяє наочно оцінити результат.

2.3 Реалізація методу головних компонент для вирішення задач класифікації

Після попередньої підготовки даних було реалізовано метод головних компонент, який передбачає виконання послідовності дій:

1. Стандартизація даних.

Для кожного значення x_{ij} змінно їj і зразка i:

$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$, де μ_j — середнє значення змінної j, а σ_j — її стандартне відхилення. Це дозволяє нормалізувати кожен параметр, щоб уникнути домінування ознак з великими значеннями.

2. Обчислення коваріаційної матриці.

Коваріаційна матриця показує, як зміни однієї ознаки співвідносяться зі змінами іншої. Кожен елемент матриці Σ розраховується як:

$\Sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^{n} z_{ij} \cdot z_{ik}$, де Σ_{jk} — коваріація між ознаками j і k, а n — кількість зразків у наборі даних.

3. Визначення власних векторів і власних значень.

Коваріаційна матриця розкладається на власні вектори і власні значення, що дозволяє визначити основні напрямки варіації в даних. Для цього розв'язують рівняння:

$\Sigma v = \lambda v$, де λ — власне значення, яке визначає, скільки варіації в даних пояснює відповідний власний вектор v. Власні вектори представляють головні компоненти, а власні значення вказують на їх відносну значущість.

4. Вибір головних компонент і зменшення розмірності**.**

Обирають k головних компонент, які пояснюють найбільше варіації, тобто компоненти з найбільшими власними значеннями. Отримана матриця перетворення W містить k головних компонент, які відповідають обраним власним векторам. Потім виконують проекцію стандартизованих даних на нові координати:

$Z' = Z \cdot W$, де Z'— проекція даних у просторі головних компонент. Ці нові координати представляють зменшену розмірність даних, зберігаючи основну інформацію про варіацію.

Розрахунок поясненої дисперсії для кожної компоненти показав, що:

- Компонента 1 пояснює 34.25% загальної дисперсії;
- Компонента 2 пояснює 27.42% загальної дисперсії;

- Компонента 3 пояснює 10.28% загальної дисперсії;
- Компонента 4 пояснює 7.56% загальної дисперсії;
- Компонента 5 пояснює 6.01% загальної дисперсії;
- Компонента 6 пояснює 3.34% загальної дисперсії;
- Компонента 7 пояснює 3.01% загальної дисперсії;
- Компонента 8 пояснює 2.83% загальної дисперсії;
- Компонента 9 пояснює 1.92% загальної дисперсії;
- Компонента 10 пояснює 1.73% загальної дисперсії;
- З 11 по 17 компоненту менше 1%;
- З 18 по 34 компоненту - 0%.

Завдяки PCA простір ознак було зменшено з 35 до 2 головних компонент, що дозволило візуалізувати багатовимірні дані у 2D-просторі (див. рис. 2.2).

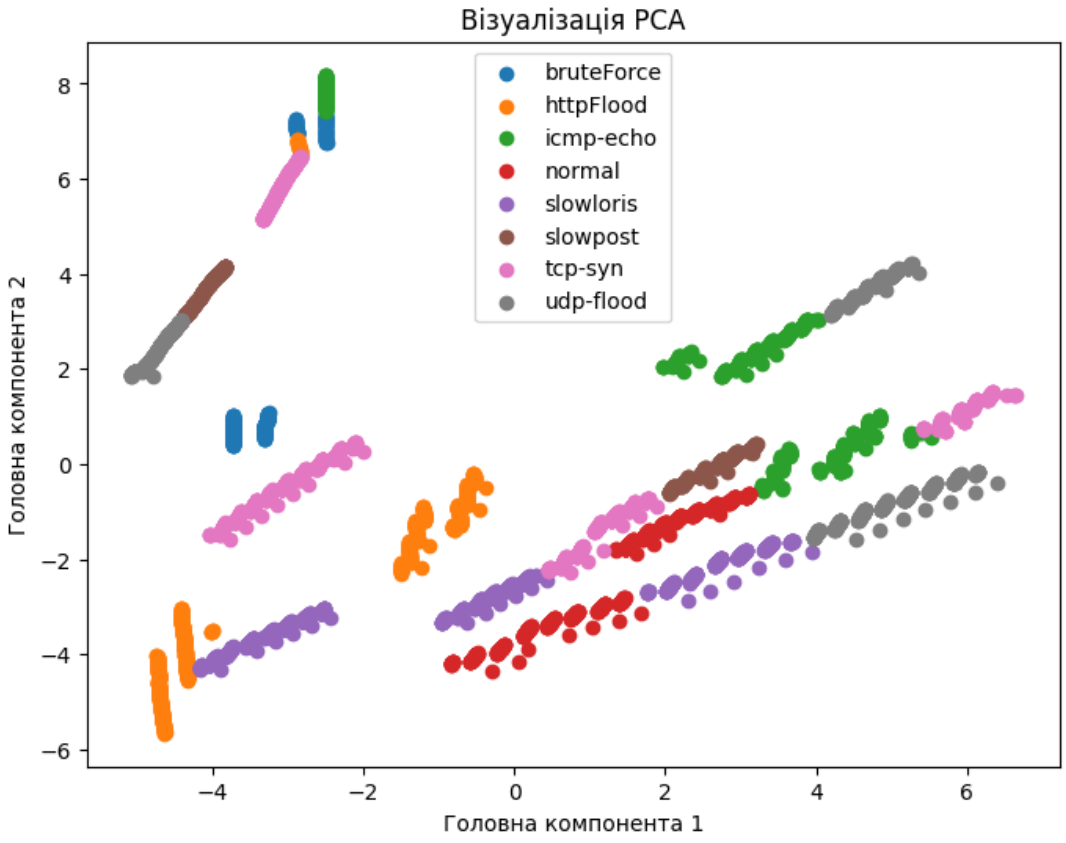

Рисунок 2.2 – Результат застосування PCA

Разом дві головні компоненти забезпечують більше 60% поясненої дисперсії, що дозволяє створити компактне представлення даних із частково втратою інформації.

Наступним етапом була кластеризація різними методами(див. рис. 2.3, 2.4, 2.5 та 2.6).

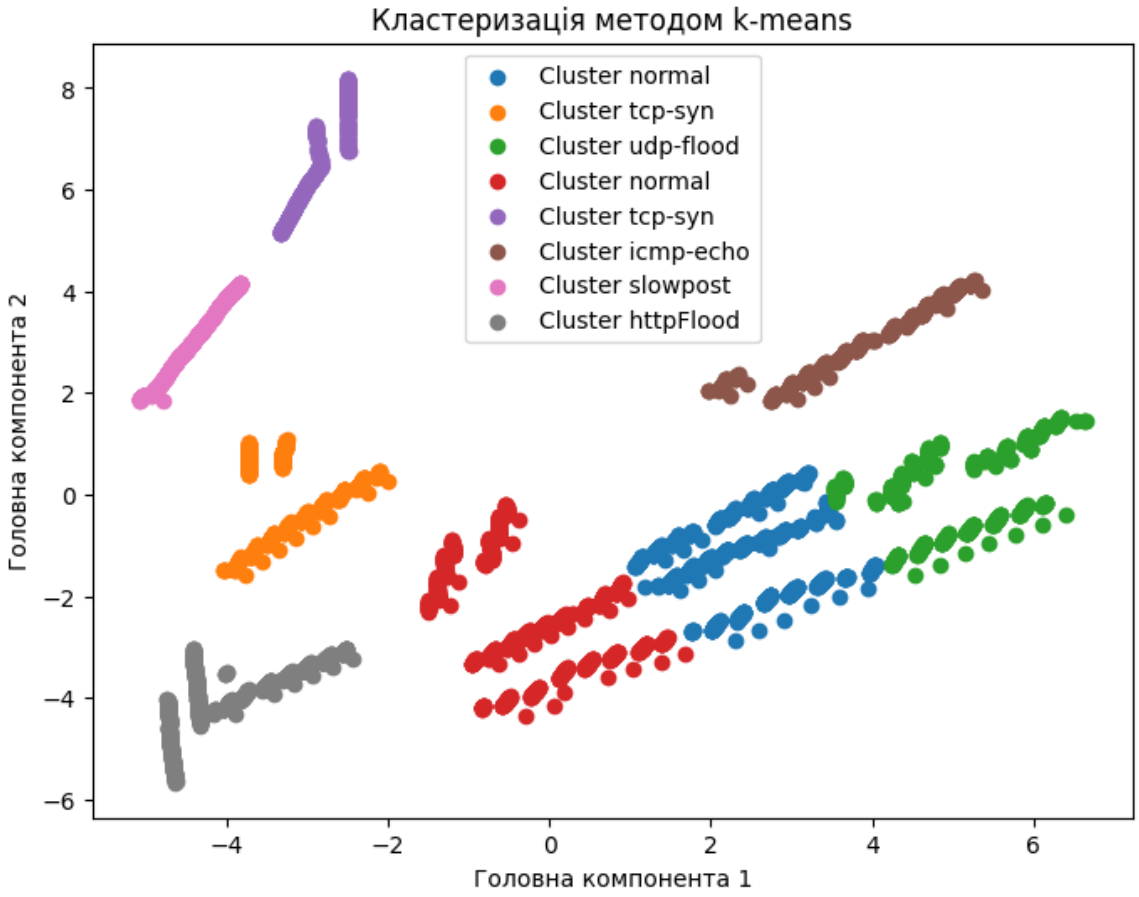

Рисунок 2.3 – Кластеризація методом k-means

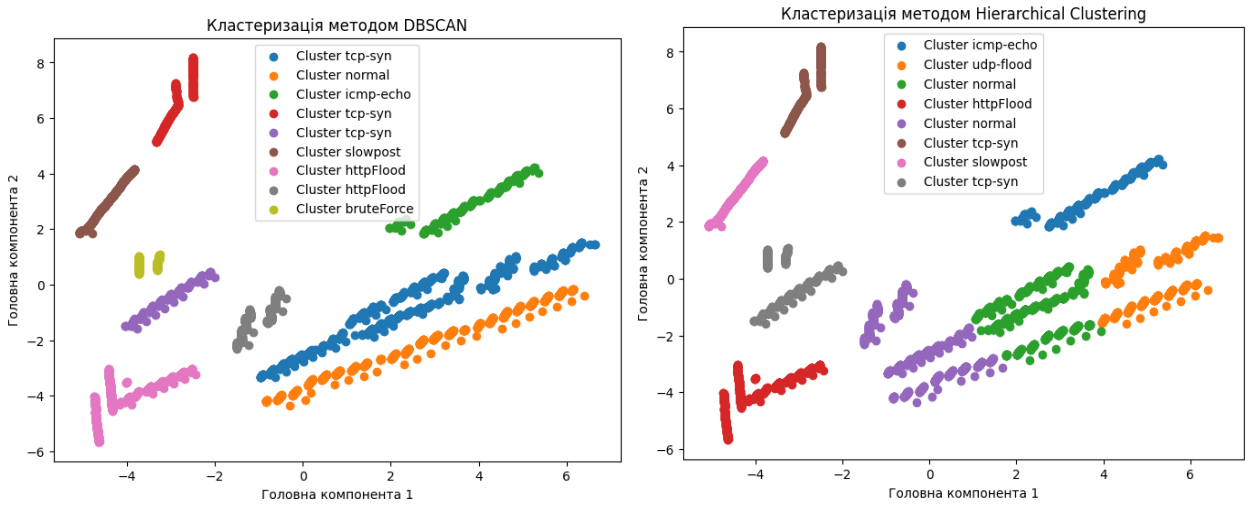

Рисунок 2.4 – Кластеризація методами DBSCAN та Hierarchical

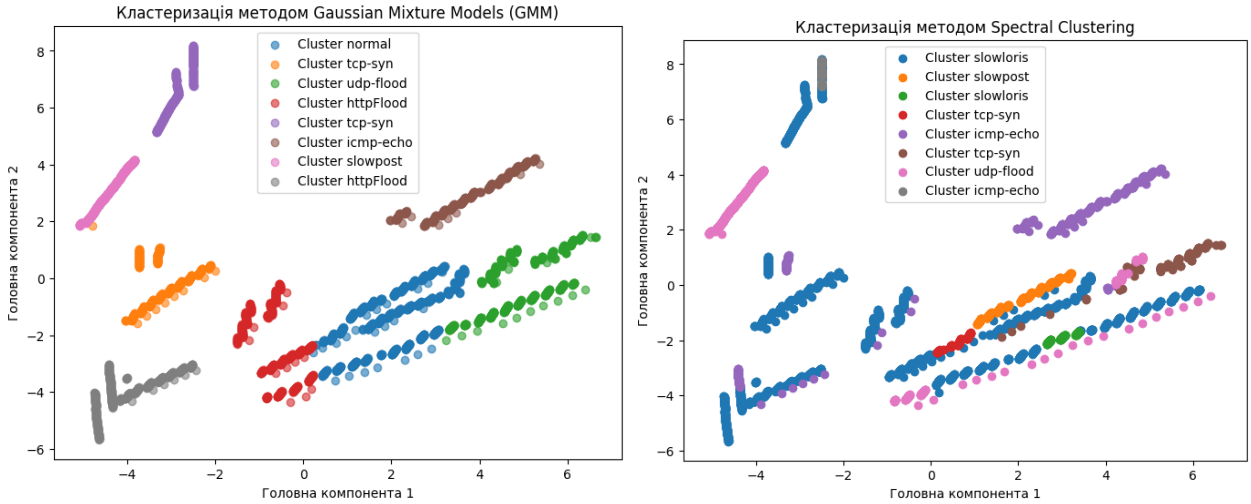

Рисунок 2.5 – Кластеризація методами GMM та Spectral

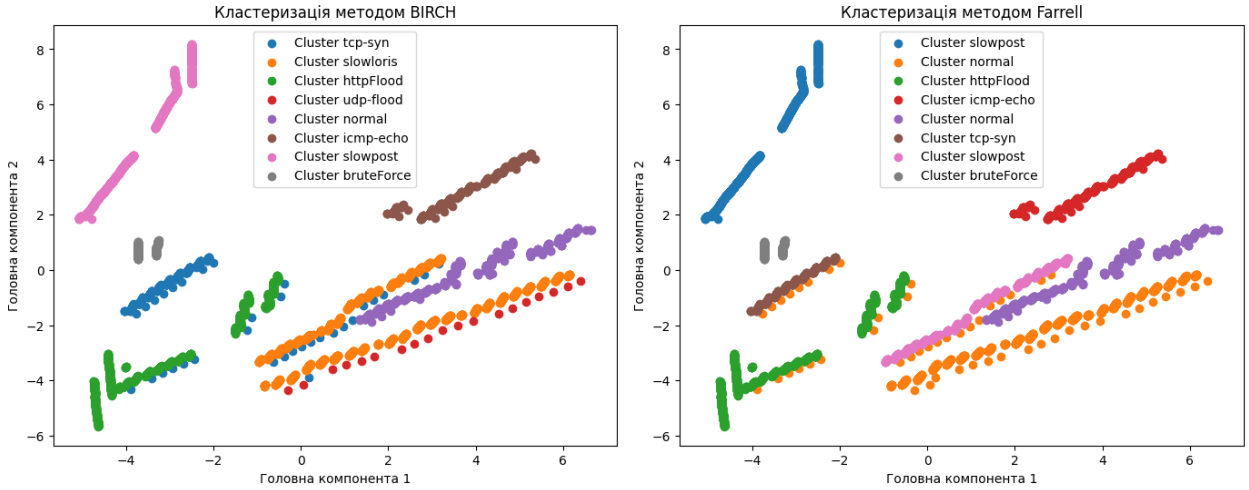

Рисунок 2.6 – Кластеризація методами BIRCH та Farrell

Проаналізувавши ряд методів кластеризації, а саме:

- k-means – коефіцієнт силуету 0.554;
- DBSCAN – коефіцієнт силуету 0.365;
- Hierarchical – коефіцієнт силуету 0.555;
- GMM (Gaussian Mixture Models) – коефіцієнт силуету 0.519;
- Spectral – коефіцієнт силуету -0.063;
- BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) – коефіцієнт силуету 0.468;
- Farrell – коефіцієнт силуету 0.501,

було обрано оптимальний по результатам коефіцієнта силуету, особливостей методу та часу виконання.

Силуетний аналіз можна застосовувати для оцінки якості кластеризації. Коефіцієнт силуета має діапазон значень від −1 до 1, де вищий коефіцієнт силуета вказує на модель із більш узгодженими кластерами. Іншими словами, коефіцієнти силуета, що близькі до 1 означають, що зразок знаходиться далеко від сусідніх кластерів. Значення 0 вказує на те, що зразок розташований на межі або дуже близько до межі між двома сусідніми кластерами. Нарешті, від'ємні значення вказують на те, що зразки могли бути потенційно віднесені до неправильного кластера [11].

Кластеризація методом K-means виявилася найкращою з огляду на її швидкість (особливо на великих обсягах даних) та відносно інших методів високу оцінку результатів кластеризації(коефіцієнт силуету 0.554)

Висновок до другого розділу

Було розглянуто підхід до виявлення аномалій у мережевому трафіку на основі методу головних компонент. Метод показав свою ефективність для зменшення розмірності даних та збереження основної інформації про їх варіації, що сприяє поліпшенню продуктивності алгоритмів кластеризації. Було зібрано та підготовлено дані, після чого реалізовано PCA, який дозволив виділити ключові компоненти з поясненням більше 60% дисперсії.

Для виявлення аномалій було обрано кластеризацію методом K-середніх, результати якої візуалізовано у просторі головних компонент. Проведені експерименти показали, що використання PCA у поєднанні з алгоритмом кластеризації забезпечує високу деталізацію і якість виявлення аномалій у мережевому трафіку, дозволяючи ідентифікувати відхилення від нормальної поведінки.

3 ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ МЕТОДУ ВИЯВЛЕННЯ АНОМАЛІЙ В ТРАФІКУ

3.1 Метрики ефективності

Для оцінки ефективності методу виявлення аномалій у мережевому трафіку використовуються різні метрики. Існує достатньо важливих метрик, такі як:

- Перехресна ентропія (cross-entropy);
- Точність (Accuracy);
- Повнота (Recall);
- Точність прогнозу (Precision);
- F-1 міра.

Однією з популярних метрик є перехресна ентропія, яка широко застосовується в задачах класифікації та оцінювання моделей машинного навчання. Вона дозволяє виміряти різницю між реальним розподілом класів у даних та прогнозованим розподілом, отриманим моделлю [12].

Перехресна ентропія визначається як математичне очікування ентропії реального розподілу щодо передбаченого розподілу. У формалізованому вигляді функція перехресної ентропії має вигляд [12]:

$H(p, q) = -\sum_{i=1} p_i \, log(q_i)$, де:

- p_i – ймовірність, що спостереження належить до класу;
- q_i – ймовірність, що модель передбачає належність до класу i;
- N – кількість класів.

Мінімальне значення перехресної ентропії досягається тоді, коли розподіл передбачень q_i максимально схожий на реальний розподіл p_i. У контексті виявлення аномалій у мережевому трафіку низьке значення перехресної ентропії свідчить про високу точність моделі при класифікації нормального та аномального трафіку.

Якщо метод виявлення аномалій базується на класифікації (наприклад, класифікатор на основі PCA або нейронної мережі), перехресна ентропія використовується для оцінювання коректності прогнозів. Також висока ентропія може свідчити про необхідність удосконалення моделі, таких як вибір інших гіперпараметрів або вдосконалення набору даних.

Перехресна ентропія дозволяє оцінити, наскільки точно модель прогнозує ймовірності. Це особливо важливо, якщо подальші дії залежать від передбачених імовірностей (наприклад, блокування підозрілих IP-адрес).

Хоча перехресна ентропія є важливою метрикою, але існують і інші метрики, точність одна з них. Точність є базовою метрикою для оцінювання ефективності алгоритмів класифікації, включаючи методи виявлення аномалій у мережевому трафіку. Вона визначає частку правильно класифікованих прикладів серед загальної кількості прикладів [13].

Формула розрахунку: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$, де:

- TP (True Positive) — кількість аномалій, які були правильно ідентифіковані як аномалії;
- TN (True Negative) — кількість нормального трафіку, який був правильно класифікований;
- FP (False Positive) — кількість нормального трафіку, помилково класифікованого як аномалії;
- FN (False Negative) — кількість аномалій, які не були виявлені.

Висока точність означає, що модель здебільшого правильно ідентифікує як нормальний, так і аномальний трафік. Низька точність свідчить про велику кількість помилок класифікації (FP або FN), що знижує ефективність виявлення аномалій.

Повнота (також відома як чутливість або Sensitivity) є ключовою метрикою для оцінювання здатності моделі правильно виявляти всі реальні аномалії у наборі даних. Вона вимірює частку правильно ідентифікованих аномалій серед усіх фактичних аномалій [14].

Формула розрахунку: $Recall = \frac{TP}{TP+FN}$

Висока повнота означає, що модель здатна виявляти більшість аномалій, навіть якщо при цьому вона допускає більше хибнопозитивних класифікацій (FP). Низька повнота свідчить про те, що модель пропускає значну кількість реальних аномалій, що може бути критичним у задачах безпеки.

Також повнота не враховує FP, тому модель з високою повнотою може бути недостатньо точна і часто класифікувати нормальний трафік як аномальний. Використання тільки повноти без додаткових метрик, таких як точність прогнозу (Precision), може призвести до необ'єктивної оцінки моделі [15].

Точність прогнозу (Precision) є важливою метрикою для оцінювання коректності класифікації аномального трафіку. Вона визначає частку правильних позитивних прогнозів (TP) серед усіх прогнозів, зроблених моделлю як аномальні (TP + FP) [15].

Формула розрахунку: $Precision = \frac{TP}{TP+FP}$

Висока точність прогнозу означає, що модель рідко помиляється, класифікуючи нормальний трафік як аномалії. Низька точність прогнозу свідчить про велику кількість FP, що може спричинити хибні тривоги.

Precision не враховує FN, тому модель з високою точністю прогнозу може пропускати багато реальних аномалій (низький Recall). У деяких системах, таких як системи моніторингу безпеки, важливіше мінімізувати FN, навіть якщо це призведе до зниження Precision.

F1-міра є комплексною метрикою, яка поєднує в собі точність прогнозу (Precision) і повноту (Recall) в одному числовому значенні. Вона використовується для оцінювання балансу між здатністю моделі уникати хибних тривог (FP) і пропусків реальних аномалій (FN) [16].

F1-міра визначається як гармонійне середнє Precision та Recall:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Високе значення F1-міри свідчить про те, що модель досягла позитивного балансу між Precision та Recall. Низьке значення F1-міри може свідчити про значні недоліки в одному або обох аспектах (низький Precision або низький Recall).

F1-міра не враховує загальну кількість прикладів (True Negatives, TN), тому вона не підходить для аналізу всього розподілу даних, коли важливі TN.

3.2 Розроблення методу виявлення аномалій в трафіку на тестових даних

Послідовність кроків методу виявлення аномального трафіку представлена на рисунку 3.1.

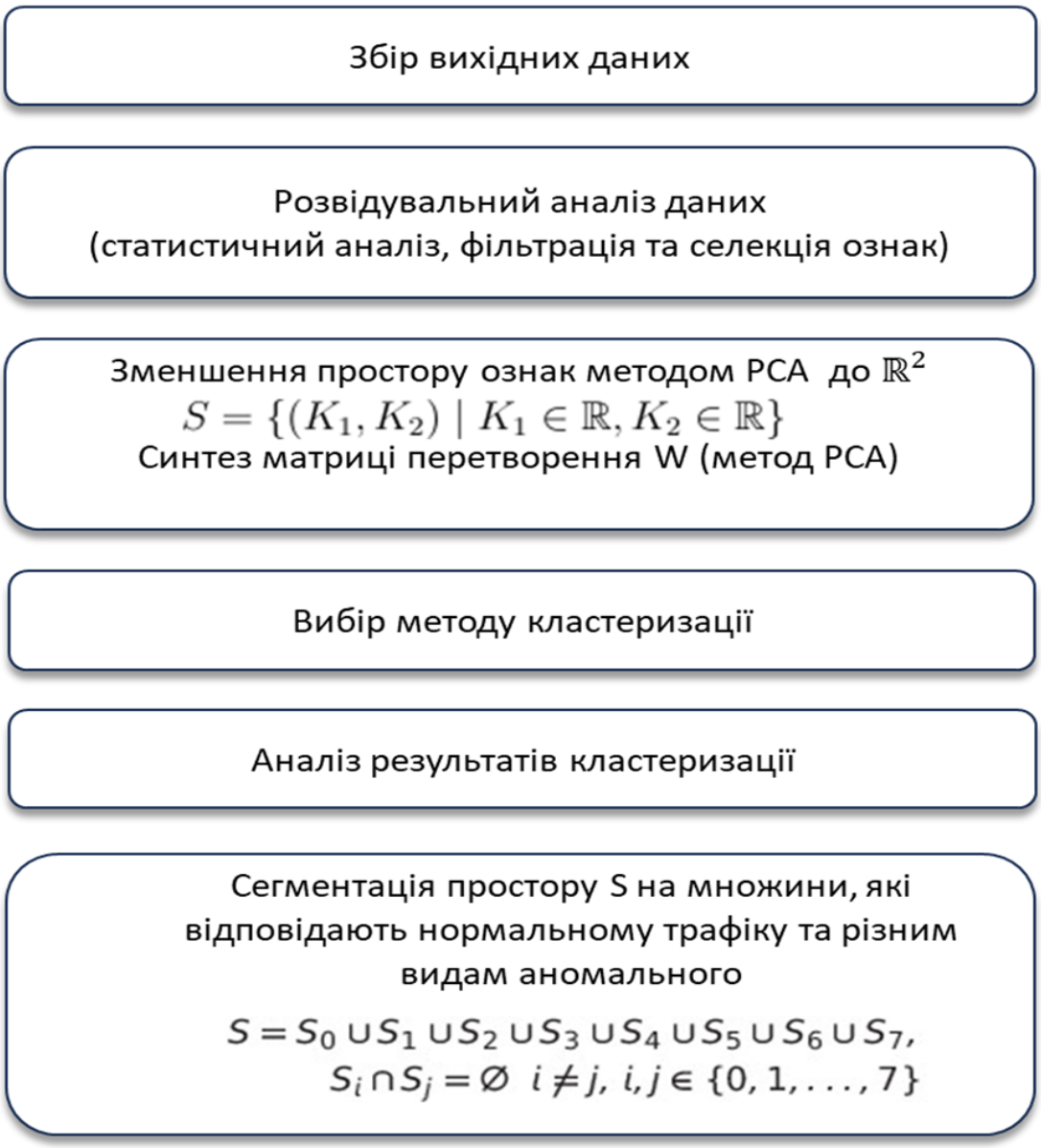

Рисунок 3.1 – Послідовність кроків методу виявлення аномального трафіку

1. Збір вихідних даних:

На цьому етапі відбувається отримання та накопичення вихідних даних про трафік (наприклад, логів, метрик, потоків пакетів).

Джерела даних можуть включати:

- файли логів;
- мережеві моніторингові інструменти (Snort, Wireshark);
- системи аналізу мережевого трафіку (NetFlow, IPFIX).

2. Розвідувальний аналіз даних (EDA – Exploratory Data Analysis)
    - Статистичний аналіз передбачає обчислення основних статистичних характеристик трафіку (середні значення, медіана, дисперсія, ковзні середні тощо).
    - Фільтрація та селекція ознак:
        - Вибираються найінформативніші ознаки, які можуть розрізняти нормальний і аномальний трафік;
        - Типові ознаки: швидкість передачі даних, кількість пакетів, розмір пакетів, частота з'єднань, затримки.
    - Аналіз залежностей між ознаками та виявлення можливих трендів.
3. Вибір методу кластеризації

Вибирається метод кластеризації для сегментації простору ознак S:

- Методи машинного навчання:
    - K-means;
    - DBSCAN (Density-Based Spatial Clustering of Applications with Noise);
    - Метод головних компонент (PCA);
    - Системи на основі нейронних мереж (Autoencoders).

4. Аналіз результатів кластеризації

Після виконання кластеризації проводиться аналіз отриманих кластерів:

- Визначаються ознаки, які відповідають нормальному трафіку;
- Аналізуються віддалені точки (outliers) та малі кластери, які можуть бути пов'язані з аномальним трафіком.

5. Сегментація простору S на множини, які не перетинаються

   Простір ознак S сегментується на дві основні групи:

   1. Нормальний трафік (основний кластер з більшістю точок).
   2. Аномальний трафік (всі інші групи або точки поза основним кластером).
   3. Множина аномального трафіку також може бути сегментована в залежності від типу аномального трафіку.

6. Розроблення методу виявлення аномалій

   - Використовується метод, що перевіряє приналежність точки до одного з кластерів;
   - Трафік позначається як аномальний у разі неналежності підмножині нормального (див рис. 3.2).

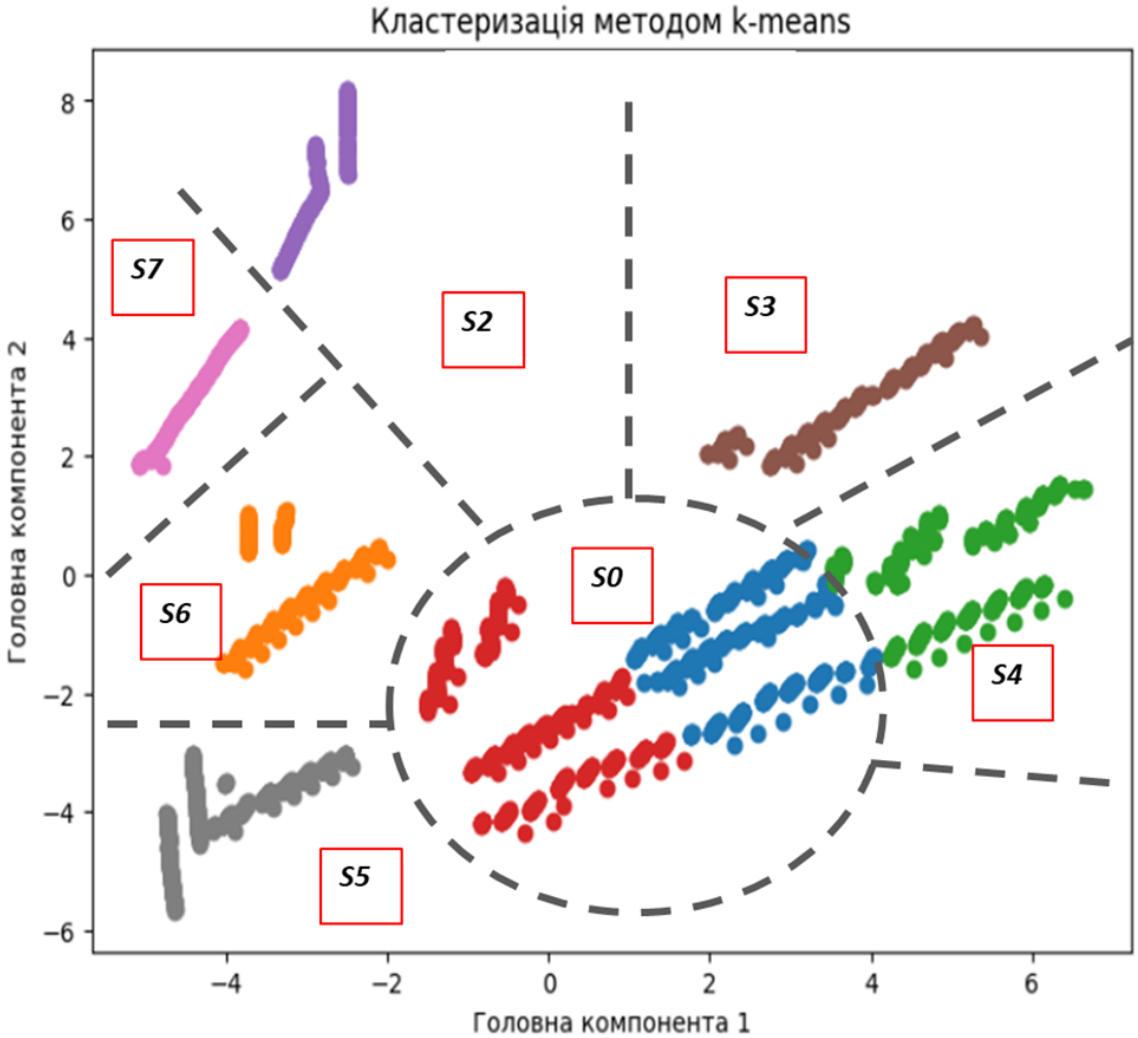

Рисунок 3.2 – Сегментація простору S на множини

7. Проекція в простір ознак

- Проекція в простір S:
    - Трафік (або ознаки трафіку) проектується в простір ознак S, де був виконаний попередній аналіз.
- Для нового, невідомого трафіку:
    - Перевіряється приналежність до сегмента нормального трафіку.

Висновки щодо трафіку:

1. Якщо трафік належить нормальному кластеру:
    - Висновок: трафік нормальний.
2. Якщо трафік не належить нормальному кластеру:
    - Висновок: трафік аномальний.
    - Проводиться подальший аналіз приналежності T* до однієї з підмножин аномального трафіку.

Подальший аналіз аномалій

- Виявлений аномальний трафік може бути класифікований для визначення типу аномалії.
- Аналіз включає:
    - Вивчення шаблонів (поведінка, розподіл у часі);
    - Кореляцію з попередніми інцидентами;
    - Розгляд логів і додаткових ознак для ідентифікації атаки.

Послідовність дій для застосування методу представлена на рисунку 3.3.

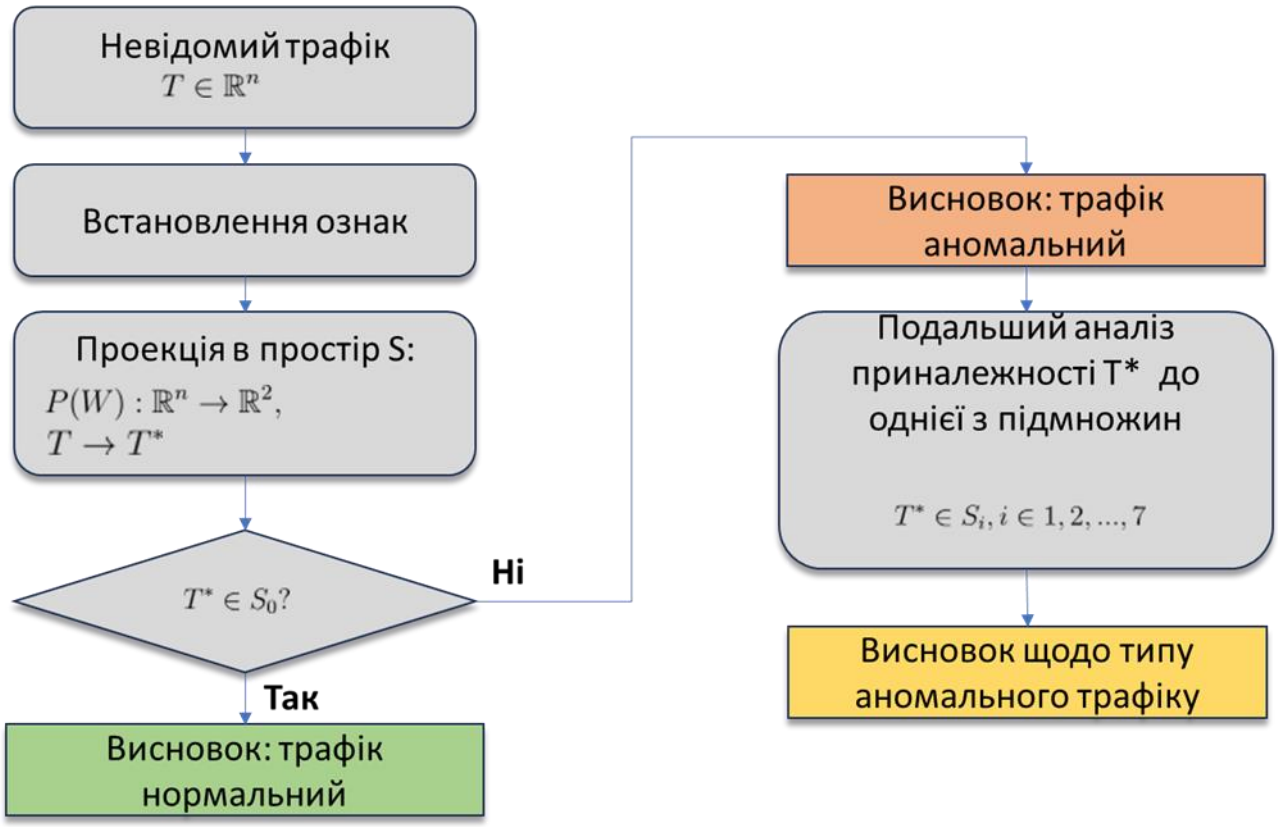

Рисунок 3.3 – Послідовність дій для застосування методу

3.3 Розроблення практичних рекомендацій щодо застосування методу виявлення аномалій в трафіку

Практичні рекомендації щодо застосування методів виявлення аномалій на основі машинного навчання можуть бути інтегровані у кількох ключових системах та напрямках, а саме:

1. Системи підтримки прийняття рішень

Системи підтримки прийняття рішень використовують дані для оптимізації процесів реагування на потенційні загрози у мережі. Методи машинного навчання, такі як PCA у поєднанні з кластеризацією, можуть забезпечувати:

- Швидке виявлення аномальних патернів у великих обсягах даних;
- Візуалізацію результатів у спрощеному вигляді (2D-простір), що дозволяє аналітикам легше інтерпретувати аномальні події.

Приклад використання:

- Моніторинг трафіку шляхом обробки мережевих логів у реальному часі, що дозволяє адміністраторам швидко визначати та оцінювати потенційні загрози;
- На основі виявлених аномалій (після кластеризації) система може автоматично рекомендувати блокування IP-адрес або активацію додаткових заходів безпеки.

2. Аналізатори мережевого трафіку

Аналізатори трафіку (на кшталт Wireshark або NetFlow) можуть бути доповнені методами PCA для швидкої ідентифікації відхилень. PCA зменшує розмірність даних, фокусуючись на суттєвих аномаліях у характеристиках трафіку.

Особливості застосування:

- Виявлення аномалій у реальному часі шляхом застосування PCA для моніторингу характеристик трафіку у реальному часі, таких як обсяг переданих даних, затримка між пакетами тощо;
- Візуалізація аномалій, тобто виявлення віддалених точок (outliers) на графі PCA дозволяє ідентифікувати підозрілі активності.

Аналізатори трафіку можуть виділяти підозрілі IP-адреси або нехарактерну активність на основі кластеризації після PCA, надаючи адміністраторам можливість детального вивчення підозрілих подій.

3. Системи захисту від DDoS-атак

DDoS-атаки проявляються у вигляді різкого збільшення трафіку. Використання PCA дозволяє:

- Виділити аномальну активність серед великого обсягу трафіку;
- Автоматизувати процес детектування атак шляхом ідентифікації кластерів з незвичними характеристиками.

Практична реалізація:

- Фільтрація бот-трафіку через PCA дозволяє виявити аномальний трафік, що був створений ботами, оскільки він суттєво відрізняється від нормального користувацького патерну;
- При виявленні підозрілого кластеру система може ініціювати блокування або перенаправлення трафіку.

4. Системи виявлення витоків даних

Витоки даних часто супроводжуються аномально високим вихідним трафіком. PCA дозволяє швидко аналізувати багатовимірні характеристики мережевого трафіку та виділяти відхилення.

Рекомендації:

- Інтеграція в системи DLP (Data Loss Prevention), де PCA використовується для моніторингу вихідного трафіку та фільтрації підозрілих з'єднань;
- Аналіз вихідних з'єднань допомагає зупинити витік даних шляхом виявлення великих обсягів незвичного трафіку за короткий час.

5. Захист IoT-пристроїв

IoT-пристрої часто є ціллю атак через слабкі механізми безпеки. Використання PCA допомагає:

- Виявити аномальну поведінку пристроїв;
- Запобігати компрометації пристроїв через аналіз характеристик з'єднань.

Можна застосовувати для моніторингу активності сенсорів у розумних будинках для виявлення нехарактерних патернів.

6. Системи виявлення брутфорс-атак

Брутфорс-атаки часто генерують велику кількість з'єднань з невдалими спробами автентифікації. PCA допомагає виявити ці аномалії за такими параметрами:

- Кількість спроб автентифікації;
- Інтенсивність запитів за короткий проміжок часу.

Системи можуть автоматично фільтрувати такі з'єднання, блокуючи IP-адреси та повідомляючи адміністратора.

7. Аналітичні системи для прогнозування загроз

PCA у поєднанні з алгоритмами машинного навчання може використовуватися для:

- Прогнозування майбутніх аномалій;
- Аналізу поведінки користувачів для виявлення потенційно шкідливих дій.

Рекомендації:

- Впровадження PCA як попереднього етапу для побудови моделей поведінкового аналізу.
- Використання методів прогнозування для виявлення трендів і можливих загроз.

Висновок до третього розділу

Було проведено дослідження ефективності методу виявлення аномалій у мережевому трафіку, визначено основні метрики ефективності методу, такі як точність (Accuracy), повнота (Recall), точність прогнозу (Precision), F1-міра та перехресна ентропія. Було розроблено послідовність кроків для застосування методу в реальних умовах, починаючи зі збору та підготовки даних і завершуючи аналізом результатів кластеризації та сегментації простору ознак. Метод показав свою ефективність у виявленні аномалій шляхом зменшення розмірності даних і виділення ключових компонент, що пояснюють найбільшу варіацію в поведінці трафіку. Також було розроблено практичні рекомендації щодо застосування методу.

Результати дослідження показали, що поєднання PCA з методами кластеризації є дієвим інструментом для ідентифікації аномалій у мережевому трафіку, підвищуючи ефективність систем кібербезпеки та мінімізуючи ризики порушень.

ВИСНОВКИ

Було досліджено проблему виявлення аномалій у мережевому трафіку із застосуванням машинного навчання. Аномалії у трафіку є ключовими ознаками різноманітних загроз, зокрема DDoS-атак, брутфорс-атак та витоків даних, що потребує ефективних методів їх своєчасного виявлення для забезпечення безпеки інформаційних систем.

У першому розділі було визначено основні поняття аномалій, їх характеристики та взаємозв'язок із сучасними кіберзагрозами. Проведений аналіз існуючих методів показав, що традиційні статистичні підходи поступаються ефективністю методам машинного навчання, особливо у випадках складних атак.

У другому розділі було реалізовано метод головних компонент для зменшення розмірності даних, що дозволило виділити ключові особливості трафіку. Після застосування кластеризації методом K-means трафік було успішно сегментовано на нормальний та аномальний, де коефіцієнт силуету склав 0.554, що підтвердило якість отриманих результатів.

Третій розділ присвячено оцінці ефективності запропонованого методу. Було використано основні метрики, зокрема точність, повноту та F1-міру, що забезпечило об'єктивну оцінку його продуктивності. Розроблено рекомендації щодо практичного застосування методу у системах підтримки прийняття рішень, аналізаторах трафіку, системах захисту від DDoS-атак та виявлення витоків даних, що підтверджує його універсальність.

Розроблений метод виявлення аномалій у мережевому трафіку показав свою результативність, зокрема завдяки можливості обробки великих обсягів даних та виділення ключових ознак аномальної поведінки. Подальші дослідження можуть бути спрямовані на інтеграцію глибинного навчання для підвищення точності та розробку рішень у реальному часі.

ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Signature vs. Anomaly-Based Detection: Which Is More Effective? : веб-сайт. URL: http://surl.li/fioodi (дата звернення: 05.10.2024)
2. Estévez-Pereira, J. J., Fernández, D., & Novoa, F. J. (2020, August). Network anomaly detection using machine learning techniques. In Proceedings (Vol. 54, No. 1, p. 8). MDPI.
3. Network traffic : веб-сайт. URL: https://en.wikipedia.org/wiki/Network_traffic (дата звернення: 10.10.2024)
4. Iglesias, F., & Zseby, T. (2015). Analysis of network traffic features for anomaly detection. Machine Learning, 101, 59-84.
5. Cook, A. A., Mısırlı, G., & Fan, Z. (2020). Anomaly Detection for IoT Time-Series Data: A Survey. IEEE Internet of Things Journal, 7(7), 6481–6494.
6. DDoS threat report for 2024 Q3 : веб-сайт. URL: https://radar.cloudflare.com/reports/ddos-2024-q3 (дата звернення: 20.10.2024)
7. How Cloudflare auto-mitigated world record 3.8 Tbps DDoS attack : веб-сайт. URL: https://blog.cloudflare.com/how-cloudflare-auto-mitigated-world-record-3-8-tbps-ddos-attack/ (дата звернення: 20.10.2024)
8. Ding, M., & Tian, H. (2016). PCA-based network traffic anomaly detection. Tsinghua Science and Technology, 21(5), 500-509.
9. Коваріаційна матриця : веб-сайт. URL: https://uk.wikipedia.org/wiki/Коваріаційна_матриця (дата звернення: 29.10.2024)
10. Набір даних: Network Anomaly Detection Dataset : веб-сайт. URL: https://www.kaggle.com/datasets/malkasasbeh/network-anomaly-detection-dataset/data (дата звернення: 03.11.2024)
11. Silhouette Coefficient - an overview | ScienceDirect Topics : веб-сайт. URL: https://www.sciencedirect.com/topics/computer-science/silhouette-coefficient (дата звернення: 10.11.2024)

12.Перехресна ентропія : веб-сайт. URL: https://jay.org.ua/перехресна-ентропія/ (дата звернення: 15.11.2024)

13. Accuracy and precision : веб-сайт. URL: https://en.wikipedia.org/wiki/Accuracy_and_precision (дата звернення: 17.11.2024)

14. Чутливість та специфічність : веб-сайт. URL: https://uk.wikipedia.org/wiki/Чутливість_та_специфічність (дата звернення: 19.11.2024)

15. Влучність та повнота : веб-сайт. URL: https://uk.wikipedia.org/wiki/Влучність_та_повнота (дата звернення: 21.11.2024)

16. Показник F1 у машинному навчанні : веб-сайт. URL: https://thetransmitted.com/adlucem/pokaznyk-f1-u-mashynnomu-navchanni/ (дата звернення: 23.11.2024)

17. Чепіга В. С. Підходи до аналізу аномалій у мережевому трафіку. Штучний інтелект і безпека : збірник матеріалів науково-практичної конференції, 19-21 листопада 2024 р. Київ : ІПМЕ ім. Г.Є. Пухова, 2024. С. 33.

18.Чепіга В. С. Застосування методу головних компонент для аналізу аномалій в мережевому трафіку. Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій : тези доповідей XII Міжнародної науково-практичної конференції, 10-12 грудня 2024 р. Запоріжжя : Національний університет «Запорізька політехніка», 2024. С. 319-321.

19. Чепіга В. С. Реалізація методу PCA для виявлення аномального трафіку в мережі. Безпека, технології, інновації: нові горизонти : збірник праць учасників міжфакультетської науково-практичної інтернет-конференції здобувачів вищої освіти і молодих вчених, 12 листопада 2024 р. Житомир : Поліський національний університет, 2024. С. 19-22.

ДОДАТКИ

ДОДАТОК А

```
import pandas as pd
import numpy as np
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from sklearn.metrics import silhouette_score
data = pd.read_csv("data.csv")
features = data.columns[:-1]
X = data[features].values
y = data['class']
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
print("Сума поясненої дисперсії двома головними компонентами:",
np.sum(pca.explained_variance_ratio_))
plt.figure(figsize=(8, 6))
for label in np.unique(y):
    plt.scatter(X_pca[y == label, 0], X_pca[y == label, 1], label=label)
plt.xlabel("Головна компонента 1")
plt.ylabel("Головна компонента 2")
plt.title("Візуалізація PCA")
plt.legend()
plt.show()
kmeans = KMeans(n_clusters=8, random_state=42)
clusters = kmeans.fit_predict(X_pca)
```

```
unique_clusters = np.unique(clusters)
cluster_labels = {}
for cluster in unique_clusters:
    class_indices = np.where(clusters == cluster)[0]
    most_common_class = y.iloc[class_indices].mode()[0]
    cluster_labels[cluster] = most_common_class
plt.figure(figsize=(8, 6))
for cluster in unique_clusters:
    cluster_points = X_pca[clusters == cluster]
    plt.scatter(cluster_points[:, 0], cluster_points[:, 1], label=f"Cluster
{cluster_labels[cluster]}")
plt.xlabel("Головна компонента 1")
plt.ylabel("Головна компонента 2")
plt.title("Кластеризація методом k-means")
plt.legend()
plt.show()
silhouette_avg = silhouette_score(X_pca, clusters)
print(f"Silhouette Score для кластеризації: {silhouette_avg}")
```