Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ДОПУСТИТИ ДО ЗАХИСТУ:
В.о. завідувача кафедри
кібербезпеки та захисту
інформації
_____ Іван ПАРХОМЕНКО
«13» червня 2025 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи

галузь знань 12 Інформаційні технології
(шифр і назва галузі знань)

спеціальність 125 Кібербезпека
(код і назва спеціальності)

освітній ступень бакалавр

освітня програма Кібербезпека
(назва освітньо-професійної програми)

на тему: «Модель виявлення аномальної активності в мережевому трафіку інформаційно-комунікаційних систем для підвищення інформаційної безпеки»

Виконавець: студент IV курсу, групи КБ-41

_____ **Максим ГОРДІЄНКО**
(підпис) (ім'я, прізвище)

	Підпис	**Ім'я ПРІЗВИЩЕ**
Керівник		Інна МИХАЛЬЧУК
Нормоконтроль		Юрій ЩЕБЛАНІН

Київ 2025

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
В.о. завідувача кафедри
кібербезпеки
та захисту інформації
_____ Іван ПАРХОМЕНКО
«29» листопада 2024 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньо-професійної програми)

Студенту **КБ-41** (група) **Гордієнку Максиму Сергійовичу** (прізвище ім'я по батькові)

Тема кваліфікаційної роботи Модель виявлення аномальної активності в мережевому трафіку інформаційно-комунікаційних систем для підвищення інформаційної безпеки

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Тема кваліфікаційної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол №6 від 28.11.2024 р.

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Мережевий трафік, методи та алгоритми виявлення аномалій

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Дослідження методів виявлення аномальної активності з використанням статистичного аналізу, сигнатурних та поведінкових методів
Розроблено концептуальну модуль системи моніторингу мережевого трафіку та проведено аналіз ефективності запропонованої моделі

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Розроблена модель дозволяє ефективно виявляти аномальну активність у мережевому трафіку в режимі реального часу

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 29 листопада 2024 року

Завдання видав _____ (підпис) Інна МИХАЛЬЧУК (ім'я, прізвище)

Завдання прийняв до виконання _____ (підпис) Максим ГОРДІЄНКО (ім'я, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт	Відмітка про виконання
1	Уточнення постановки задачі	29.11.2024 – 22.01.2025	*виконано*
2	Аналіз літератури	29.01.2025 – 11.02.2025	*виконано*
3	Обґрунтування вибору рішення	12.02.2025 – 15.02.2025	*виконано*
4	Методи виявлення аномалій	16.02.2025 – 04.03.2025	*виконано*
5	Збір та обробка даних мережевого трафіку	05.03.2025 – 21.03.2025	*виконано*
6	Розробка архітектури системи виявлення аномалій	22.03.2025 – 08.04.2025	*виконано*
7	Реалізація алгоритму виявлення аномалій (Isolation Forest)	09.04.2025 – 10.05.2025	*виконано*
8	Оформлення пояснювальної записки	11.05.2025 – 27.05.2025	*виконано*
9	Підготовка до захисту кваліфікаційної роботи	28.05.2025 – 13.06.2025	*виконано*

Завдання видав _____ (підпис) Інна МИХАЛЬЧУК (ім'я, прізвище)

Завдання прийняв до виконання _____ (підпис) Максим ГОРДІЄНКО (ім'я, прізвище)

Термін подання кваліфікаційної роботи до ЕК 13 червня 2025 року

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел, додатків, має 63 сторінки основного тексту, 5 таблиць та 7 рисунків. Список використаних джерел містить 32 найменування і займає 4 сторінки.

Метою роботи є розробка моделі для виявлення аномальної активності в мережевому трафіку з метою підвищення рівня інформаційної безпеки інформаційно-комунікаційних систем.

Для досягнення зазначеної мети поставлено наступні завдання:

- Дослідити структуру та характеристики мережевого трафіку в інформаційно-комунікаційних системах.
- Вивчити наявні підходи до виявлення аномальної активності
- Розробити та навчити модель виявлення аномальної активності на основі відкритих або синтетичних наборів даних.
- Провести тестування розробленої моделі з метою оцінки точності, повноти та швидкості виявлення аномалій.
- Оцінити ефективність моделі в контексті підвищення рівня інформаційної безпеки систем.

Об'єктом дослідження є мережевий трафік інформаційно-комунікаційних систем.

Предметом дослідження є методи аналізу мережевого трафіку, алгоритми виявлення аномалій та підозрілої активності в режимі реального часу.

Практичною цінністю отриманих результатів є розроблена модель, що дозволяє ефективно виявляти аномальну та підозрілу активність у мережевому трафіку в режимі реального часу, що сприяє підвищенню рівня інформаційної безпеки.

Ключові слова: мережевий трафік, аномальна активність, інформаційно комунікаційна система, Isolation Forest, кіберзагрози.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕННЬ

ІБ	–	Інформаційна безпека
ІКС	–	Інформаційно-комунікаційна система
IDS	–	Intrusion Detection System, система виявлення вторгнень
IPS	–	Intrusion Prevention System, система запобігання вторгненням
SIEM	–	Security Information and Event Management, система управління інформацією та подіями безпеки
DoS/DDoS	–	Denial of Service / Distributed Denial of Service, атака на відмову в обслуговуванні
OSI	–	Open Systems Interconnection, модель взаємодії відкритих систем
HTTP/HTTPS	–	HyperText Transfer Protocol / Secure, протокол передачі гіпертексту / захищений
ARP	–	Address Resolution Protocol, протокол визначення фізичної адреси
ICMP	–	Internet Control Message Protocol, протокол керування повідомленнями в Інтернеті
FTP	–	File Transfer Protocol, протокол передавання файлів

ВСТУП

Постійний розвиток цифрових технологій спричинив широке впровадження інформаційно-комунікаційних систем у державному управлінні, бізнесі та суспільному житті. Ці системи забезпечують обмін, обробку й зберігання даних у різних сферах, створюючи інфраструктуру для функціонування сучасних організацій. Щодня глобальні мережі передають терабайти даних, забезпечуючи діяльність електронного урядування, фінансових ринків, систем електронної комерції, логістики, медичної інфраструктури та інших критично важливих сфер. Проте інтенсивне зростання обсягів передаваного трафіку супроводжується зростанням загроз безпеці, серед яких особливої уваги набувають аномалії - відхилення від нормального функціонування, які можуть свідчити про несанкціонований доступ, шкідливе програмне забезпечення, збої в роботі систем або інші небажані події.

Актуальність теми дослідження визначається зростаючою потребою у створенні автоматизованих, інтелектуальних систем для виявлення аномалій у мережевому трафіку. Традиційні методи моніторингу, зокрема сигнатурні або евристичні, виявили свою обмеженість перед лицем нових, невідомих типів атак, які не мають чітких шаблонів. У цьому контексті на перший план виходять підходи, що базуються на аналізі поведінки трафіку з використанням інструментів машинного навчання, здатних адаптивно реагувати на зміни в мережевому середовищі, самостійно виявляти відхилення та розпізнавати нові типи загроз. Особливої ваги набуває реалізація таких систем у критичних інфраструктурах, де своєчасне виявлення аномалій може запобігти катастрофічним наслідкам.

Метою цієї роботи є розробка та дослідження ефективної системи моніторингу мережевого трафіку інформаційно-комунікаційної системи з використанням методів машинного навчання для виявлення аномалій, що дозволить підвищити рівень інформаційної безпеки, забезпечити раннє

реагування на загрози та зменшити ризики зловмисного втручання або технічних збоїв.

Для досягнення зазначеної мети необхідно виконати такі завдання:

1. Вивчити теоретичні засади моніторингу трафіку, класифікації загроз і виявлення аномалій.
2. Проаналізувати сучасні системи контролю трафіку та оцінити їх сильні й слабкі сторони.
3. Сформулювати задачу виявлення аномалій як задачу машинного навчання.
4. Спроєктувати архітектуру програмної системи моніторингу з виявленням аномалій.
5. Зібрати та підготувати трафік для аналізу.
6. Реалізувати алгоритм Isolation Forest для виявлення аномалій.
7. Візуалізувати та інтерпретувати результати аналізу.

Об’єктом дослідження виступає інформаційно-комунікаційна система як середовище функціонування різних типів мережевого трафіку, що формується внаслідок взаємодії пристроїв, сервісів та користувачів у цифровому просторі.

Предметом дослідження є процеси виявлення аномалій у мережевому трафіку ІКС за допомогою алгоритмів машинного навчання, а також методи попередньої обробки, нормалізації, візуалізації та інтерпретації результатів.

У ході дослідження було опрацьовано такі джерела:

- наукові публікації з інформаційної безпеки, мережевих технологій і кіберзагроз (статті IEEE, Springer, Elsevier);
- фахові дослідження алгоритмів машинного навчання для задач виявлення аномалій (зокрема Isolation Forest, One-Class SVM, Autoencoders);
- документація до бібліотек scikit-learn, pyshark, pandas, matplotlib, які використовувались у реалізації;
- практичні рекомендації з побудови систем IDS/IPS;
- технічні звіти з проєктів DARPA, KDD Cup та інших джерел відкритих даних з кібербезпеки;

- методичні посібники з проєктування ІКС та захисту інформації.

Фактичний матеріал, використаний у роботі, включає:

- реальні зразки трафіку, зібрані з використанням Wireshark у середовищі емуляції (наприклад, штучне створення HTTP, FTP, ICMP, DNS-запитів);
- згенеровані набори з аномальними подіями (наприклад, DoS-атаки, підміна IP, flood-пакети);
- .pcap-файл traffic_sample.pcap, що містить фрагменти мережевих сесій з подальшим перетворенням у DataFrame для аналізу;
- графіки та діаграми, збудовані на основі класифікації та аналізу аномалій.

Для досягнення поставлених цілей застосовувалися такі методи дослідження:

- аналіз літератури - з метою систематизації знань про види аномалій та інструменти їх виявлення;
- методи обробки даних - зокрема нормалізація характеристик, кодування категоріальних змінних, видалення пропущених значень;
- метод графічного аналізу – подання результатів у вигляді діаграм і графіків для покращення сприйняття та аналізу даних;
- експериментальний підхід - оцінка точності, повноти та продуктивності системи на основі зібраних даних.

Теоретична цінність роботи полягає у систематизації знань щодо сучасних методів виявлення аномалій у комп’ютерних мережах, обґрунтуванні доцільності використання алгоритмів машинного навчання для задач безпеки, а також у створенні концептуальної моделі автоматизованої системи моніторингу з динамічним виявленням загроз.

Практичне значення дослідження полягає у реалізації прототипу системи, що може бути інтегрована до локальних мереж освітніх, медичних, комерційних чи промислових установ з метою підвищення рівня кіберзахисту. Запропоноване рішення може бути адаптоване для потреб систем адміністрування, безпеки

інформації, служби SOC (Security Operation Center) тощо. Крім того, результати роботи можуть бути використані в освітньому процесі для підготовки фахівців у галузі інформаційної безпеки та аналітики даних.

Отже, обрана тема має як фундаментальну, так і прикладну цінність, що дозволяє говорити про її значущість у контексті сучасного розвитку ІКС, кібербезпеки та інтелектуального аналізу даних. Робота поєднує теоретичне обґрунтування і практичну реалізацію, орієнтовану на реальні виклики цифрової епохи.

РОЗДІЛ 1
ТЕОРЕТИЧНІ ОСНОВИ ВИЯВЛЕННЯ АНОМАЛІЙ У МЕРЕЖЕВОМУ ТРАФІКУ

1.1 Поняття мережевого трафіку та інформаційно-комунікаційних систем

Інформаційно-комунікаційні системи (ІКС) становлять сукупність технічних, програмних, телекомунікаційних і організаційних засобів, що забезпечують збирання, передавання, обробку, збереження та надання інформації у цифровій формі. Ці системи відіграють ключову роль у забезпеченні безперервного функціонування підприємств, установ, державних органів, а також приватного сектора, інтегруючи в собі комп'ютерні мережі, програмне забезпечення, бази даних, хмарні обчислення та інші компоненти. У контексті кібербезпеки важливим об'єктом аналізу в межах ІКС є мережевий трафік, який є основним носієм інформації, що передається в цифровому середовищі [1].

Мережевий трафік можна визначити як сукупність даних, що передаються між пристроями в комп'ютерній мережі протягом певного періоду часу. Ці дані можуть мати різну природу - текстову, графічну, мультимедійну - та обслуговувати як міжлюдське спілкування (наприклад, електронна пошта, відеоконференції), так і автоматизовану взаємодію між пристроями (наприклад, міжсерверні запити, обмін повідомленнями в IoT-середовищі). Основними характеристиками мережевого трафіку є швидкість передавання, об'єм, типи використовуваних протоколів, напрям передавання (вхідний або вихідний), а також часові характеристики - затримка, джиттер, тривалість сесій.

Трафік у мережі формується в результаті виконання прикладних запитів та взаємодії користувачів і програмних агентів. При цьому, з точки зору моделі OSI, мережевий трафік включає дані всіх рівнів - від фізичного (наприклад,

передавання сигналу по кабелю) до прикладного (обробка запиту HTTP до вебсервера). Зміст пакета даних формується на прикладному рівні, після чого кожен наступний рівень додає свою службову інформацію - заголовки, маркери, перевірочні коди - забезпечуючи надійність і контроль передавання. У результаті формується повноцінний мережевий пакет, який передається в середовищі комунікації [7].

Основними типами мережевого трафіку є:

- Унікаст-трафік - передавання даних між одним джерелом і одним отримувачем (напр., перегляд вебсторінки);
- Мультикаст-трафік - передавання від одного джерела до групи абонентів (напр., онлайн-трансляції);
- Бродкаст-трафік - широкомовна передача до всіх вузлів у мережі (напр., ARP-запити);
- Анікаст-трафік - вибіркове передавання до найближчого або найефективнішого вузла (застосовується в хмарних обчисленнях).

У контексті адміністрування інформаційно-комунікаційних систем, аналіз мережевого трафіку дозволяє ідентифікувати відхилення в поведінці мережі, оптимізувати її завантаження, прогнозувати збої, забезпечити балансування навантаження між серверами, а також виявляти загрози безпеці. Для цього застосовуються різноманітні інструменти моніторингу, зокрема Wireshark, Tcpdump, ntopng, SolarWinds, Zabbix, а також спеціалізовані системи класу IDS/IPS (Intrusion Detection / Prevention Systems).

Проблематика виявлення аномального мережевого трафіку активно розглядається у новітніх дослідженнях, оскільки відхилення від типової мережевої поведінки часто супроводжують несанкціонований доступ, компрометацію даних, деструктивні дії або технічні збої в інформаційній інфраструктурі. Умовно нормальний трафік характеризується передбачуваністю та стабільними параметрами, відповідними звичному функціонуванню системи. Аномальний трафік, навпаки, має відхилення від стандартних характеристик:

несподіване збільшення об’єму, використання незвичних портів, повторювані запити до заборонених ресурсів, зміна структури пакетів тощо [8].

Аналіз трафіку може здійснюватися у двох основних режимах: онлайн (у реальному часі) та офлайн(ретроспективно, за збереженими логами чи .pcap-файлами). У першому випадку необхідна висока обчислювальна продуктивність і мінімальні затримки, що забезпечують миттєве реагування на інциденти. У другому - глибший аналіз, можливо, з використанням складних алгоритмів класифікації та візуалізації. Як приклад, системи SIEM (Security Information and Event Management) поєднують обидва підходи для повного циклу виявлення та реагування на загрози, *таблиця 1.1*.

З технічної точки зору, інформаційно-комунікаційна система включає наступні основні компоненти:

- Мережеву інфраструктуру (маршрутизатори, комутатори, точки доступу);
- Сервери і сховища даних;
- Програмне забезпечення для обробки та маршрутизації запитів;
- Клієнтські пристрої (робочі станції, смартфони, IoT);
- Служби безпеки (брандмауери, IDS/IPS, антивіруси, проксі-сервери);
- Інтерфейси взаємодії (API, портали, поштові шлюзи тощо).

Мережевий трафік в ІКС є динамічним відображенням їх роботи: зміна інтенсивності або типу трафіку часто є першою ознакою внутрішньої чи зовнішньої події - запуск нового сервісу, вторгнення, помилки конфігурації, сплеску навантаження. Саме тому моніторинг трафіку виступає одним із ключових інструментів управління та контролю в ІКС, рис 1.1.

Сучасні технології визначають складність структури мережевого трафіку, що ускладнює аналіз і висуває підвищені вимоги до методів його дослідження:

- Хмарні обчислення - трафік поширюється по розподілених дата-центрах, що ускладнює відстеження джерел;

- IoT - велика кількість дрібних пристроїв з нестандартною поведінкою та слабкими протоколами безпеки;
- VPN і шифрування - значна частина трафіку передається в зашифрованому вигляді, що унеможливлює повноцінний аналіз без попереднього дешифрування або використання метаданих;
- 5G/6G - нові протоколи та висока швидкість створюють умови для зростання об’ємів і складності трафіку.

Розуміння базових характеристик трафіку дозволяє ефективно реалізувати системи виявлення аномалій (Anomaly Detection Systems). Вони функціонують за рахунок вивчення нормального профілю поведінки мережі (baseline) і виявлення відхилень, використовуючи як традиційні методи (пороговий аналіз, експертні правила), так і сучасні підходи на основі машинного навчання - класифікацію, кластеризацію, авторегресійне моделювання, глибокі нейромережі. Одним із найперспективніших напрямів є застосування алгоритму Isolation Forest, який дозволяє виявляти аномалії шляхом ізоляції незвичних точок у багатовимірному просторі характеристик трафіку [15].

Дослідження мережевого трафіку виходить за межі практичних аспектів адміністрування, охоплюючи питання оптимізації, прогнозування, стійкості й захисту інформаційно-комунікаційних систем, що формує окремий напрям наукового аналізу. У рамках цифрової трансформації суспільства, такі дослідження мають вагоме значення для національної безпеки, технологічного розвитку та економічної стабільності держав.

Отже, поняття мережевого трафіку та інформаційно-комунікаційних систем є фундаментальними в сучасній комп’ютерній науці. Вони формують основу для побудови безпечних, ефективних, масштабованих і самодостатніх середовищ цифрової комунікації.

Таблиця 1.1

Типи мережевого трафіку та їх характеристики

Тип трафіку	**Напрям передавання**	**Приклади використання**
Унікаст	Від одного джерела до одного отримувача	Перегляд вебсторінки, email
Мультикаст	Від одного джерела до вибраної групи отримувачів	Онлайн-трансляції, IPTV
Бродкаст	Від одного джерела до всіх вузлів у мережі	ARP-запити, DHCP
Анікаст	До найближчого або оптимального вузла з множини	CDN-сервери, хмарні сервіси

Рисунок 1.1 - Розподіл обсягу трафіку за типами

1.2 Загрози безпеці в комп'ютерних мережах та їх класифікація

Комп'ютерні мережі забезпечують основу для організації інформаційно-комунікаційних процесів у різних галузях: державному секторі, фінансових установах, медицині, освітніх системах, оборонній сфері та об'єктах критичної інфраструктури. Відповідно, зростає не лише залежність від цифрових каналів комунікації, а й рівень ризиків, пов'язаних із порушенням конфіденційності, цілісності та доступності інформації. У цьому контексті ключову роль відіграє аналіз загроз безпеці в комп'ютерних мережах, їх класифікація та побудова ефективних засобів виявлення й протидії [20].

Загроза інформаційній безпеці - це потенційна чи реальна дія, подія або стан, що може призвести до порушення однієї або декількох властивостей інформації чи функціонування системи. У контексті комп'ютерних мереж це може означати перехоплення даних, несанкціонований доступ, модифікацію пакетів, відмову в обслуговуванні або створення умов для подальшої компрометації інфраструктури. Загрози можуть виникати як унаслідок цілеспрямованих атак, так і внаслідок технічних помилок, людського чинника чи неналежної конфігурації системи.

Існує низка підходів до класифікації загроз у комп'ютерних мережах, *таблиця 1.2*. Найуживаніші з них базуються на джерелі загрози, механізмах реалізації, об'єктах впливу, а також на рівні організації OSI-моделі, де проявляється атака. Така систематизація дозволяє більш точно ідентифікувати природу загрози та визначити відповідні заходи захисту.

1. За джерелом походження загрози поділяють на:

- Зовнішні - атаки, що здійснюються з-за меж локальної мережі: через Інтернет, віддалені хости, ботнети, зловмисників, які не мають доступу до внутрішньої інфраструктури. Найчастіше пов'язані з DDoS-атаками, скануванням портів, фішингом, підробкою DNS-записів тощо.

- Внутрішні - загрози, що походять від співробітників організації, підрядників або осіб, які мають повний чи частковий доступ до внутрішніх ресурсів. Можуть бути як навмисними (наприклад, саботаж, витік даних), так і ненавмисними (встановлення шкідливого ПЗ, порушення політик безпеки).
- Гібридні - комбіновані атаки, в яких зовнішній зловмисник використовує внутрішні вразливості або посередників для доступу до цільових об’єктів (наприклад, через постачальницькі ланцюги або заражені пристрої).

2. За характером впливу загрози бувають:

- Пасивні загрози - не змінюють потік даних, а спрямовані на його спостереження або збирання інформації. Прикладом є прослуховування трафіку (sniffing), перехоплення незашифрованих повідомлень, збір метаданих, аналіз шаблонів активності.
- Активні загрози - безпосередньо впливають на мережеві процеси: змінюють, блокують або видаляють інформацію. Сюди належать атаки типу «людина посередині» (MITM), внесення шкідливого коду в пакети, підміна IP-адрес, DDoS-атаки, пошкодження DNS-записів.

3. За об’єктом атаки виділяють:

- Користувацькі загрози - спрямовані на викрадення паролів, облікових даних, доступу до облікових записів, особистих файлів або банківської інформації. Зазвичай реалізуються через соціальну інженерію, фішинг, підміну інтерфейсів.
- Мережеві загрози - стосуються інфраструктури передачі даних: маршрутизаторів, комутаторів, мережевих шлюзів, протоколів маршрутизації. Прикладами є спуфінг, ARP-poisoning, зміна маршрутів, сканування портів.
- Прикладні загрози - зловмисна взаємодія з додатками або вебсервісами (SQL-ін’єкції, XSS, CSRF, експлуатація API).
- Фізичні загрози - передбачають доступ до обладнання, серверів, маршрутизаторів з метою їх пошкодження, заміни або вилучення накопичувачів.

4. За рівнем OSI-моделі, загрози класифікують наступним чином:

- Фізичний рівень - фізичне втручання в лінії зв’язку, встановлення перехоплювачів.
- Канальний рівень - атаки на MAC-адреси, підміна фреймів.
- Мережевий рівень - підміна IP, маршрутизування, сканування мережі, DoS.
- Транспортний рівень - експлуатація вразливостей TCP/UDP, сесійний грабінг.
- Сеансовий рівень - викрадення сесій, маніпуляції з SID.
- Представницький рівень - зміна форматів, компрометація кодування.
- Прикладний рівень - XSS, SQL-ін’єкції, фішинг, зловмисне ПЗ.

5. За методами реалізації:

- Мережеві атаки - вплив на інфраструктуру передачі: SYN flood, IP fragmentation, DNS amplification.
- Програмні атаки - експлуатація уразливостей у коді: buffer overflow, zero-day, backdoors.
- Соціальна інженерія - вплив на людину як найслабшу ланку системи: фішингові листи, дзвінки, маніпуляції.
- Мобільні загрози - уразливості в додатках, передавання даних через незахищені канали, підроблені точки доступу.

6. За тривалістю впливу:

- Разові атаки - короткотермінові, з конкретною метою: злом, саботаж.
- Перманентні загрози (APT) - довготривалі кампанії, які поступово проникають у систему, приховують свою активність, збирають інформацію (APT1, APT28, Stuxnet).

Усі ці класифікації мають практичне значення, оскільки дозволяють формалізувати підхід до безпеки мережі, розробити моделі загроз, визначити критичні точки в архітектурі ІКС та адаптувати політики реагування.

У відповідь на зазначені загрози, у мережах застосовуються такі заходи:

- Шифрування даних на рівні каналу (VPN, SSL/TLS) та збереження (AES, RSA).
- Ідентифікація та автентифікація користувачів (паролі, багатофакторна авторизація).
- Аудит подій і централізоване логування (SIEM-системи).
- Моніторинг активності трафіку у реальному часі (IDS/IPS, NetFlow, SNMP).
- Фільтрація трафіку (firewall, проксі, ACL).
- Резервне копіювання та контроль відновлення після інцидентів.

У контексті дослідження аномалій у мережевому трафіку особливо актуальною є здатність автоматизованих систем розпізнавати ознаки аномальної поведінки, пов’язаної з одним або кількома типами загроз. Аномалії в трафіку можуть виступати як маркери вторгнення, що ще не ідентифіковано, або як наслідки атаки, що вже реалізувалася. Для цього застосовуються моделі поведінкового аналізу, класифікації, кластеризації, а також методи з області машинного навчання - зокрема Isolation Forest, One-Class SVM, Autoencoder тощо.

Отже, класифікація загроз у комп’ютерних мережах є необхідною передумовою для створення ефективних систем захисту, своєчасного реагування на інциденти та підвищення надійності інформаційно-комунікаційної інфраструктури. У наступних розділах розглядатиметься, як саме методи виявлення аномалій дозволяють ідентифікувати ці загрози на ранніх етапах, забезпечуючи більш високий рівень захищеності цифрових систем [19].

Таблиця 1.2

Загрози безпеці в комп’ютерних мережах та їх класифікація

Критерій класифікації	**Типи загроз**
Джерело загрози	Зовнішні, Внутрішні, Гібридні
Характер впливу	Пасивні, Активні
Об'єкт атаки	Користувацькі, Мережеві, Прикладні, Фізичні

Продовження табл 1.2

Рівень OSI-моделі	Фізичний, Канальний, Мережевий, Транспортний, Сеансовий, Представницький, Прикладний
Метод реалізації	Мережеві, Програмні, Соціальна інженерія, Мобільні
Тривалість впливу	Разові, Перманентні (APT)

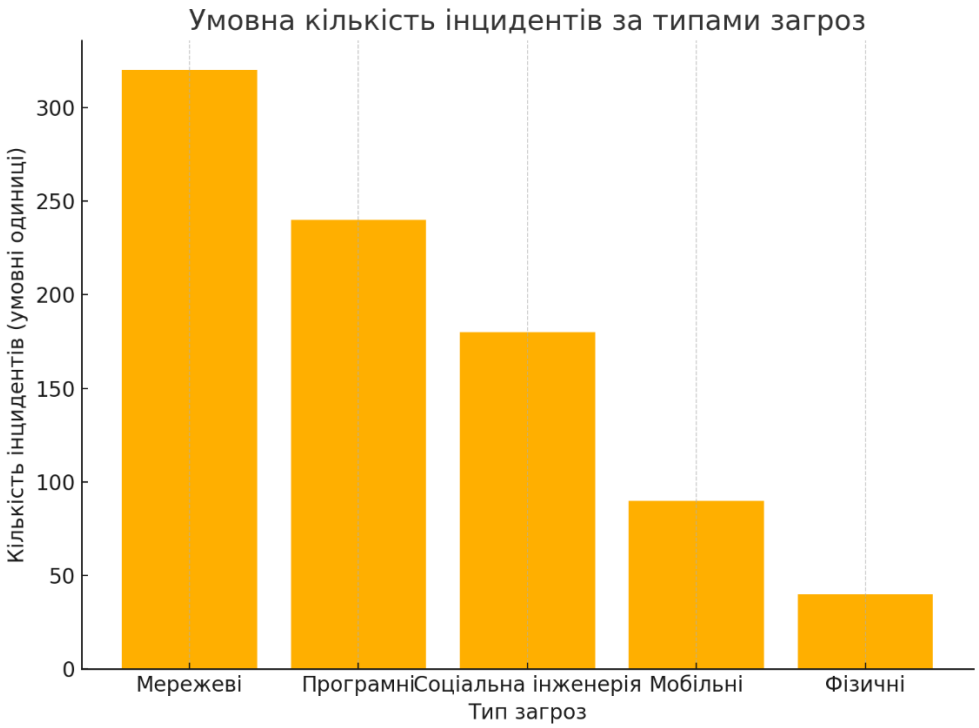

Рисунок 1.2 - Загрози безпеці в комп’ютерних мережах та їх класифікація

1.3 Методи виявлення аномалій: сигнатурний, евристичний, поведінковий

У сфері інформаційної безпеки виявлення аномалій є ключовим етапом у забезпеченні захисту комп’ютерних мереж та інформаційно-комунікаційних систем. Аномалією вважається будь-яка активність або структура трафіку, що відрізняється від очікуваної або типової поведінки мережі. Сучасні атаки стають дедалі складнішими, динамічнішими й здатними обходити традиційні фільтри безпеки. У відповідь на це виникла потреба у розробці різних методів виявлення аномалій, кожен з яких має власні особливості, переваги, обмеження та області

застосування. Найбільш поширеними підходами в цій галузі є сигнатурний, евристичний та поведінковий методи, які реалізуються у спеціалізованих системах виявлення вторгнень (Intrusion Detection Systems, IDS) і все частіше використовуються у тандемі з системами виявлення та запобігання вторгненням (Intrusion Prevention Systems, IPS) [5] *таблиця 1.3*.

Сигнатурний метод заснований на принципі порівняння вхідного трафіку або подій у системі із заздалегідь відомими шаблонами зловмисної активності, які зберігаються у вигляді сигнатур у базі даних. Кожна сигнатура описує характерні ознаки певного типу атаки - наприклад, характерну послідовність байтів у пакеті, специфічну команду протоколу чи ознаку мережевого сканування. Коли вхідні дані відповідають запису з бази сигнатур, система ідентифікує інцидент і сповіщає адміністратора або автоматично блокує відповідний трафік. Цей підхід демонструє високу ефективність у виявленні відомих атак, які вже були вивчені та формалізовані. Його головною перевагою є низький рівень хибно позитивних спрацювань, оскільки система реагує лише на точно встановлені шаблони. Однак недоліком сигнатурного методу є його неспроможність виявляти нові, ще не зафіксовані види атак (так звані zero-day загрози), а також залежність від регулярного оновлення бази сигнатур.

Евристичний метод, на відміну від сигнатурного, ґрунтується на попередньо заданих правилах або принципах аналізу, які описують загальні ознаки підозрілої поведінки. Замість того щоб шукати точну відповідність шаблону, система аналізує характер трафіку, використовуючи визначені правила, які дозволяють виявити потенційно небезпечну активність. Наприклад, евристичне правило може полягати у виявленні надмірної кількості з’єднань за короткий проміжок часу з одного джерела або передачі великого об’єму нестандартних запитів у незвичний час. У цьому випадку система може ідентифікувати можливий початок DDoS-атаки або активність з ботнету. Застосування евристичних правил дозволяє розширити діапазон виявлених загроз, зокрема таких, що не мають стабільних сигнатур або проявляються в нових формах. Проте ефективність цього підходу значною мірою залежить від

якості сформульованих правил і здатності системи адаптуватися до змін у структурі мережевого трафіку. Надмірна кількість або надто загальні евристики можуть призвести до великої кількості хибно позитивних спрацювань, що ускладнює оперативну реакцію адміністратора [9].

Найбільш інноваційним і перспективним є поведінковий метод виявлення аномалій, який базується на побудові моделі нормальної поведінки мережі або системи. На початковому етапі система спостерігає за типовим функціонуванням мережі протягом певного періоду часу і формує профіль або еталон поведінки на основі статистичних або машинних моделей. У цьому профілі враховуються такі параметри, як кількість з'єднань, тривалість сесій, типи протоколів, середній об'єм переданих даних, часові шаблони активності тощо. Подальше виявлення аномалій ґрунтується на виявленні відхилень від сформованої моделі. Наприклад, якщо певний вузол починає надсилати великі обсяги даних у нетиповий час або встановлює численні з'єднання з раніше невідомими IP-адресами, поведінкова система може ідентифікувати таку активність як потенційно зловмисну.

Поведінковий метод є основою сучасних інтелектуальних систем виявлення вторгнень, особливо тих, що застосовують алгоритми машинного навчання. Однією з ключових переваг цього підходу є здатність виявляти нові, раніше невідомі загрози, які не мають фіксованих сигнатур або не підпадають під відомі евристики. Завдяки самонавчанню такі системи можуть адаптуватися до змін у трафіку та враховувати контекст, що є важливим для мінімізації кількості хибних спрацювань. Однак ефективне функціонування поведінкових методів потребує достатнього обсягу даних для навчання, ретельної фільтрації шуму та значних обчислювальних ресурсів. Крім того, побудова моделі поведінки залежить від початково зібраного профілю, тому помилки або недоліки на етапі навчання можуть знизити якість виявлення [2].

У практичному застосуванні ці три методи не є взаємовиключними. Більшість сучасних систем моніторингу трафіку поєднують сигнатурний, евристичний та поведінковий підходи, утворюючи гібридні архітектури, які

дозволяють досягти вищої точності та надійності. Наприклад, сигнатурний фільтр може спочатку видалити явно шкідливі пакети, далі евристичний модуль аналізує поведінку підозрілих з’єднань, а поведінкова модель перевіряє решту трафіку на відхилення від типового шаблону. Такий поетапний аналіз дозволяє не лише забезпечити базовий рівень захисту, а й реагувати на загрози, що виникають у реальному часі та мають динамічну природу.

З огляду на стрімкий розвиток кіберзагроз та збільшення обсягів мережевого трафіку, роль поведінкових методів зростає. Їх розвиток і інтеграція з методами штучного інтелекту та глибокого навчання відкриває нові горизонти в аналізі інформаційних потоків, забезпечуючи проактивний захист та підвищену стійкість мереж до складних загроз. Водночас важливо наголосити, що вибір методу або їх поєднання має враховувати специфіку середовища, технічні ресурси, рівень критичності об’єктів, а також вимоги до точності, швидкості обробки та масштабованості системи, рис 1.3.

Отже, методи виявлення аномалій у комп’ютерних мережах є різноплановими як за принципом дії, так і за ефективністю. Сигнатурні методи залишаються незамінними для оперативного реагування на відомі атаки, евристичні - для широкого охоплення потенційно небезпечної активності, а поведінкові - для адаптивного виявлення нових загроз. Комплексне застосування цих підходів є найбільш ефективною стратегією захисту інформаційно-комунікаційних систем у сучасному цифровому середовищі [6].

Таблиця 1.3

Характеристики методів виявлення аномалій

Метод	**Принцип дії**	**Виявлення нових загроз**	**Кількість хибних спрацювань**	**Необхідність навчання**
Сигнатурний	Порівняння з відомими шаблонами (сигнатурами)	Низьке	Низька	Немає

Продовження табл 1.3

Евристичний	Застосування фіксованих правил для виявлення підозрілої активності	Середнє	Середня	Може бути часткова
Поведінковий	Аналіз відхилень від нормальної поведінки системи	Високе	Може бути висока	Обов’язкова

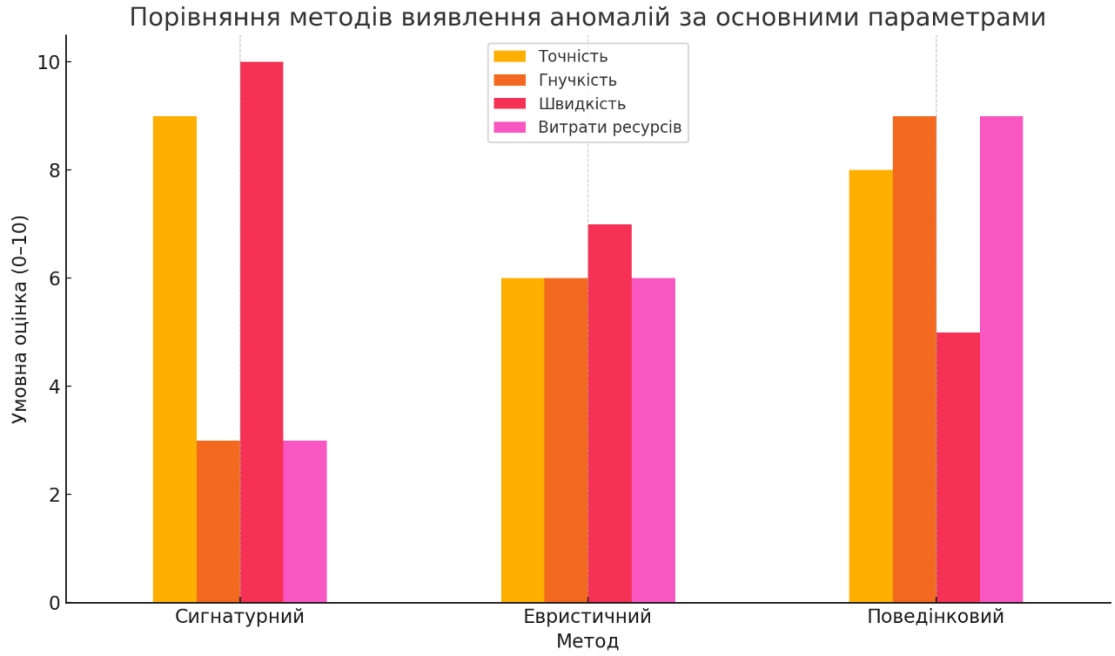

Рисунок 1.3 - Порівняння методів виявлення аномалій за основними параметрами

1.4 Огляд сучасних систем моніторингу мережевого трафіку

У сучасних умовах функціонування інформаційно-комунікаційних систем, що охоплюють хмарні обчислення, мобільні мережі, Інтернет речей, серверні кластери та розподілені архітектури, ключовою складовою інформаційної безпеки є системи моніторингу мережевого трафіку. Вони забезпечують

постійне спостереження за передаванням даних між вузлами, аналіз структури трафіку, виявлення аномалій, реєстрацію інцидентів, а також формування відповідних звітів для подальшої реакції. Основне завдання таких систем полягає у забезпеченні контролю за станом мережі в режимі реального часу, виявленні вторгнень або підозрілої активності, а також у підтриманні ефективної діагностики продуктивності інфраструктури [9].

Моніторинг трафіку здійснюється на основі захоплення, обробки й аналізу мережевих пакетів, що циркулюють у каналах зв'язку. Це дозволяє як виявити технічні збої, втрати пакетів, перевантаження або затримки, так і розпізнати потенційно небезпечні дії з боку користувачів або зовнішніх атакуючих. У сучасних мережах, які характеризуються високою динамічністю, різноманітністю пристроїв, складною маршрутизацією та мультисервісністю, ефективний моніторинг можливий лише за умови використання інтелектуальних та адаптивних засобів, які здатні масштабуватись, автоматично виявляти закономірності та оперативно реагувати на зміни у поведінці трафіку.

Серед найпоширеніших засобів моніторингу мережевого трафіку вирізняються як універсальні інструменти з відкритим кодом, так і комерційні корпоративні рішення. Одним із найбільш відомих застосунків є Wireshark - система глибокого аналізу пакетів, що дозволяє захоплювати трафік у реальному часі, декодувати десятки мережевих протоколів, візуалізувати послідовність передавання даних, аналізувати поля заголовків на всіх рівнях OSI-моделі та експортувати отриману інформацію у стандартизовані формати. Завдяки можливості фільтрації, декодування й створення профілів захоплення, Wireshark активно використовується не лише в освітніх і дослідницьких цілях, а й у практиці аналізу інцидентів, налагодження сервісів та судово-інформаційної експертизи.

Tcpdump — це утиліта командного рядка, що призначена для фільтрації та фіксації мережевого трафіку у текстовому режимі без застосування графічних засобів. Вона активно використовується в автоматизованих середовищах, у роботі з серверами, системами логування та збору статистики. Її гнучкість

дозволяє інтегрувати аналіз трафіку у більші системи моніторингу через сценарії або скрипти, а також забезпечити фоновий запис із мінімальним впливом на ресурси [1].

У контексті підприємств середнього та великого масштабу актуальними є платформи класу NetFlow-аналізаторів, які функціонують на базі телеметрії, що надається маршрутизаторами та комутаторами. Такі системи не оперують окремими пакетами, а працюють зі статистикою з'єднань, що значно зменшує обсяг оброблюваних даних, але водночас дає змогу визначити тренди, навантаження, інтенсивність, а також джерела й призначення передавання інформації. Прикладами реалізації цієї концепції є nfdump, ntopng, SolarWinds NetFlow Traffic Analyzer. Зокрема, ntopng дозволяє здійснювати аналіз трафіку у вебінтерфейсі, виявляти додатки за DPI, класифікувати трафік, визначати топ-джерела навантаження, виявляти невідомі протоколи та потенційно шкідливі потоки.

Паралельно з цим активно розвиваються системи інтегрованого моніторингу класу SIEM (Security Information and Event Management), такі як Splunk, QRadar, ArcSight, які об'єднують моніторинг мережі з аналізом подій на хостах, журналів системного рівня, даних з Active Directory, вебсерверів та інших джерел. У таких системах моніторинг трафіку є лише однією зі складових загальної аналітичної платформи, яка здійснює кореляцію подій, формує поведінкові профілі, застосовує правила виявлення інцидентів та автоматизовані сценарії реагування. Їх використання дозволяє здійснювати моніторинг у режимі реального часу на основі складної логіки та шаблонів, що підвищує ефективність виявлення складних або комбінованих атак.

Загалом сучасні системи моніторингу можна умовно поділити на три категорії: аналізатори трафіку пакетного рівня, аналітичні агрегатори мережевої статистики та комплексні платформи кібербезпеки. Перші зосереджуються на деталізації окремих з'єднань, розборі полів протоколів, пошуку специфічних ознак порушення - зокрема, ознак сесійного взлому, переповнення буферів, відправки неспецифічних заголовків тощо. Другі - надають агреговані відомості

щодо обсягів передавання, напрямків комунікації, часу активності тощо. Треті - поєднують ці підходи із даними від кінцевих точок, журналами доступу, відомостями з антивірусного та проксі-обладнання, що дозволяє створити єдину картину безпеки в організації [3].

Зі зростанням складності загроз особливої актуальності набуває застосування систем із вбудованими механізмами штучного інтелекту. Деякі з найсучасніших рішень, такі як Darktrace, Vectra AI або Cisco Secure Network Analytics, використовують алгоритми машинного навчання для побудови моделей нормальної поведінки мережі, на основі яких відбувається виявлення аномалій, рис 1.4. Вони здатні виявляти підозрілу активність навіть без чітко визначеної сигнатури, наприклад, нетипову активність користувача, зміну маршруту даних, взаємодію з нехарактерними ресурсами. Ці платформи, як правило, мають власні механізми візуалізації загроз, засоби для формування рекомендацій та інтеграцію з іншими рішеннями кібербезпеки.

Вибір системи моніторингу визначається завданнями організації, доступними ресурсами, потребою у деталізації та швидкістю обробки даних, а також особливостями інфраструктури. Наприклад, для невеликих організацій достатньо базових засобів аналізу трафіку, тоді як великі установи з розподіленою архітектурою потребують комплексних рішень з можливістю централізованого керування, віддаленого збирання телеметрії та формування єдиної політики безпеки, *таблиця 1.4*.

Системи моніторингу мережевого трафіку є невід’ємною складовою сучасних інформаційно-комунікаційних систем. Їх розвиток тісно пов’язаний із трансформацією самої природи цифрового середовища, ускладненням мережевої інфраструктури та зростанням рівня загроз. Від простого захоплення пакетів на фізичному інтерфейсі сучасні системи еволюціонували до багаторівневих, аналітичних і адаптивних рішень, здатних не лише фіксувати аномалії, а й прогнозувати можливі сценарії атак. У наступних розділах буде розглянуто, як ці системи можуть бути інтегровані з механізмами машинного

навчання для побудови високоефективної системи виявлення аномального трафіку.

Таблиця 1.4

Порівняльна характеристика сучасних систем моніторингу трафіку

Система	**Тип системи**	**Інтерфейс**	**Підтримка реального часу**	**Можливість виявлення аномалій**
Wireshark	Аналізатор пакетів	Графічний	Так	Обмежена
Tcpdump	Командний сніфер	Текстовий	Так	Немає
ntopng	Web-інтерфейс аналізу трафіку	Графічний	Так	Обмежена
Splunk	SIEM-платформа	Графічний	Так	Так
Darktrace	AI-система виявлення аномалій	Графічний	Так	Так (AI-рівень)

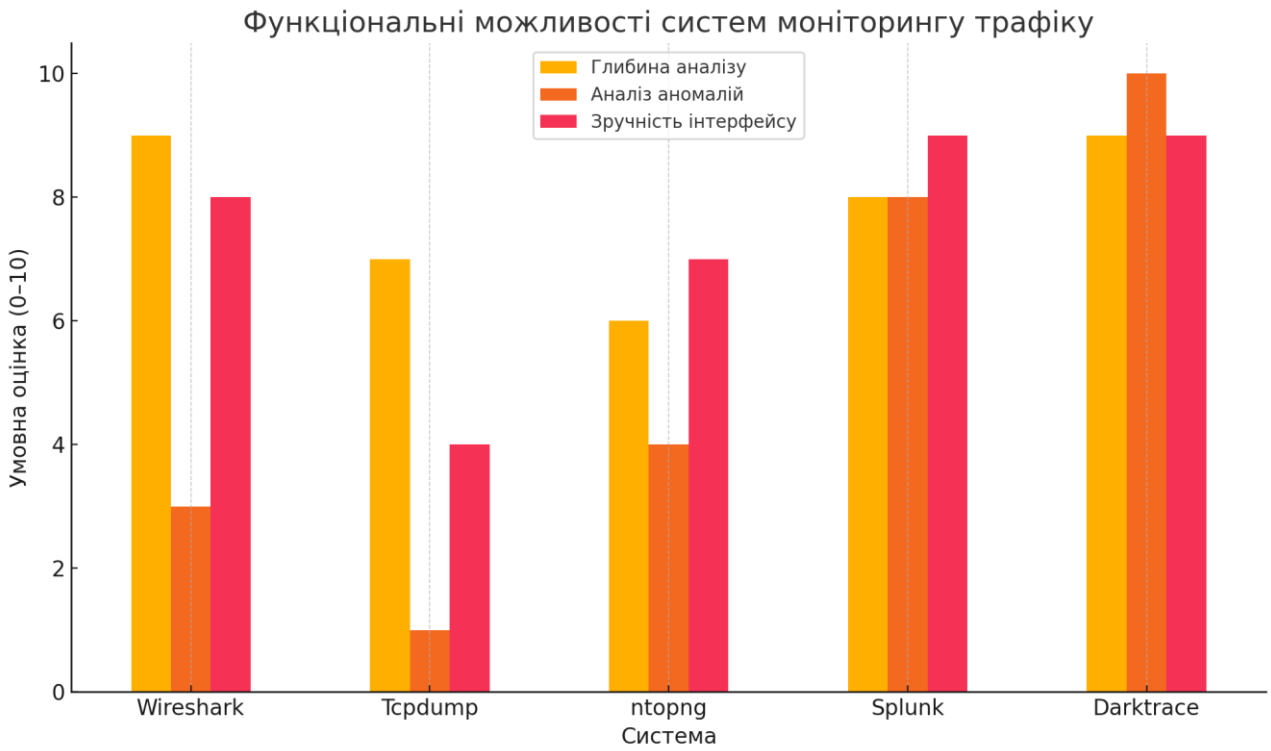

Рисунок 1.4 - Функціональні можливості систем моніторингу трафіку

Висновки до розділу 1

Вивчення фундаментальних понять мережевого трафіку та архітектури інформаційно-комунікаційних систем дозволило окреслити загальний контекст, у якому функціонують сучасні системи цифрової комунікації. Встановлено, що трафік виступає ключовим об'єктом аналізу в безпековому середовищі, оскільки саме через його структуру та динаміку можна діагностувати технічні відхилення, виявити загрози та відстежити вплив зовнішніх або внутрішніх факторів на функціонування мережі. Складність мережевих середовищ, зростання обсягів переданих даних та підвищення залежності від цифрової інфраструктури визначають потребу у ретельному та безперервному моніторингу процесів передавання інформації.

Проаналізовано спектр потенційних загроз безпеці в комп’ютерних мережах, що проявляються на різних рівнях моделі OSI, охоплюють як технічні вразливості, так і фактори людського походження.

З'ясовано, що найбільш поширені підходи до виявлення аномалій включають сигнатурний, евристичний та поведінковий методи. Кожен із них має власні переваги і обмеження, що впливають на рівень точності, швидкість реагування та гнучкість адаптації до нових загроз. Сигнатурні підходи ефективні у розпізнаванні вже відомих атак, але не здатні протистояти загрозам нульового дня. Евристичні правила дають змогу охопити ширший спектр відхилень, однак потребують ретельного налаштування. Поведінковий підхід, що базується на аналізі відхилень від типових моделей, забезпечує найвищу ймовірність виявлення невідомих загроз, але є ресурсомістким та вимагає попереднього навчання на репрезентативному наборі даних. Найбільш ефективними визнаються гібридні рішення, які поєднують переваги вищезазначених методів у рамках єдиної платформи.

Розглянуто функціональні характеристики сучасних систем моніторингу трафіку, серед яких наявні як класичні засоби аналізу (Wireshark, Tcpdump), так і високорівневі SIEM-рішення (Splunk, ArcSight), а також системи нового покоління з підтримкою штучного інтелекту (Darktrace, Vectra AI). Їхнє призначення полягає не лише в реєстрації подій і збиранні статистики, а й у забезпеченні багаторівневої аналітики, побудові моделей поведінки користувачів і пристроїв, формуванні політик безпеки та автоматизованому реагуванні на виявлені інциденти. Залежно від архітектури мережі, обсягу трафіку та цілей безпеки, застосовуються ті чи інші конфігурації систем, що відрізняються рівнем деталізації аналізу, інтеграційними можливостями, масштабованістю та здатністю до адаптації.

РОЗДІЛ 2
ПРОЕКТУВАННЯ СИСТЕМИ ВИЯВЛЕННЯ АНОМАЛІЙ У МЕРЕЖЕВОМУ ТРАФІКУ

2.1 Постановка задачі аналізу та виявлення аномального трафіку

Виявлення відхилень у мережевому трафіку є одним із основних завдань під час підтримки безпеки інформаційно-комунікаційних систем, оскільки саме ці відхилення можуть вказувати на несанкціоновані дії, технічні несправності або порушення звичних алгоритмів взаємодії систем. Завдання полягає не лише в реєстрації подій, що явно порушують правила доступу, а й у виявленні прихованих відхилень, які не мають чітких сигнатур і можуть вказувати на загрози нульового дня, внутрішні порушення політик або елементи складних багатоступеневих атак.

В основі постановки задачі лежить ідея про те, що за умови наявності достовірного опису нормального трафіку можливо виділити відхилення, які з певною ймовірністю свідчать про аномальні події. Отже, аналітична модель має оперувати не лише поточними мережевими даними, а й історичними характеристиками, на основі яких формуються репрезентативні описи типових сесій, шаблонів взаємодії, часових характеристик, обсягів передавання, інтенсивності запитів тощо. Таким чином, задача виявлення аномалій розглядається як задача побудови та порівняння моделей нормальної та фактичної поведінки, що формалізується в термінах машинного навчання як класифікація або відокремлення аномальних точок у багатовимірному просторі ознак [31].

Формальною метою аналізу є ідентифікація таких одиниць трафіку (пакетів, з’єднань, сесій або транзакцій), які демонструють статистично значуще відхилення від заданого або навчального профілю. Для цього необхідно провести збір мережевих даних з реального або емуляторного середовища, виконати їх

попередню обробку, перетворення в числові характеристики, нормалізацію та формування вхідного набору для алгоритму виявлення. У контексті обраного підходу задача конкретизується як застосування алгоритму Isolation Forest для побудови моделі, що дозволяє визначити, які об’єкти легко ізолювати в порівнянні з іншими, отже - є аномальними з точки зору структури та просторової щільності розподілу.

Постановка задачі вимагає визначення оптимального рівня деталізації аналізу, тобто вибору між аналізом окремих пакетів, з’єднань чи сесій, а також обґрунтування ознак і характеристик, що найточніше описують структуру та динаміку мережевого трафіку. Крім того, необхідно визначити спосіб оцінювання результатів роботи моделі - тобто розробити метрики точності, повноти, хибнопозитивних і хибнонегативних спрацювань, які дозволять здійснити об’єктивне порівняння ефективності запропонованого рішення з наявними підходами [21].

Задача виявлення аномального трафіку в межах даного проєкту формулюється як задача навчання моделі з частково або повністю непрокласифікованих даних, з метою виявлення точок, що мають відмінні статистичні характеристики відносно основної маси. Практична реалізація цієї задачі передбачає поетапну побудову системи, яка включає засоби захоплення трафіку (наприклад, Wireshark або Pyshark), механізми попередньої обробки та інженерії ознак, вибір і налаштування моделі аномалій, а також інструменти для візуалізації результатів та інтерпретації виявлених відхилень.

У межах поставленої задачі також враховується специфіка функціонування інформаційно-комунікаційної системи, зокрема гетерогенність її складових, наявність різних протоколів і рівнів доступу, різниця в активності користувачів, наявність зашифрованих каналів передавання даних, що ускладнює традиційні методи інспекції. Тому модель має бути максимально гнучкою, адаптивною та здатною працювати в умовах неповних або частково зашумлених даних, забезпечуючи високу якість виявлення навіть за обмеженої інформації про природу самих аномалій [2].

Постановка задачі виявлення аномалій охоплює не тільки технічні параметри, а й враховує організаційні, методологічні та експлуатаційні чинники, які впливають на ефективність аналізу мережевого трафіку, рис 2.1. Вона також передбачає урахування організаційних, правових і етичних факторів, зокрема неприпустимість порушення конфіденційності переданих даних, дотримання політик щодо моніторингу користувацької активності, відповідальність за хибні спрацювання системи та її інтеграцію у процеси управління інформаційною безпекою організації. Усе це вимагає комплексного підходу до реалізації системи, яка не лише ідентифікує загрози, а й підтримує прийняття рішень та оптимізує політики безпеки в динамічному цифровому середовищі.

Рисунок 2.1 - Постановка задачі аналізу та виявлення аномального трафіку

2.2 Архітектура та компоненти запропонованої системи

Запропонована система виявлення аномалій у мережевому трафіку побудована на основі модульної архітектури, що забезпечує гнучкість, масштабованість і можливість поетапного вдосконалення окремих

функціональних блоків без впливу на загальну структуру. Її реалізація здійснюється за допомогою мови програмування Python із залученням бібліотек для машинного навчання (scikit-learn), візуалізації (matplotlib), обробки даних (pandas) та парсингу мережевих пакетів (pyshark). Кожен із модулів виконує окрему логічну функцію, і взаємодія між ними відбувається через передачу оброблених структур даних (зокрема DataFrame), що забезпечує високу узгодженість на рівні структури коду та розширюваність у разі майбутньої інтеграції з іншими інструментами або сервісами.

Загальна архітектура системи включає такі ключові компоненти: модуль захоплення та попередньої обробки трафіку, модуль побудови ознак, модуль нормалізації та підготовки даних, модуль виявлення аномалій, а також модуль візуалізації результатів. Кожен компонент реалізовано як окремий файл або клас, що забезпечує логічне відокремлення функцій та спрощує супровід [15].

Модуль захоплення та обробки трафіку (preprocess.py) використовує бібліотеку pyshark, яка дозволяє працювати з файлами у форматі .pcap. Цей компонент здійснює розбір структури кожного пакету, вилучає необхідні атрибути (час, джерело, призначення, довжину, протокол, інформацію), та формує на їх основі список словників, що згодом конвертується у таблицю даних формату pandas.DataFrame. На цьому етапі також здійснюється базова фільтрація записів: видаляються пошкоджені пакети, невизначені поля та дублікати, що дозволяє забезпечити цілісність даних для наступних етапів. Додатково категоріальні поля, зокрема Protocol, кодуються у числовий формат, що дозволяє передавати їх у моделі машинного навчання без втрати інформації.

Модуль інженерії ознак відповідає за перетворення сирих мережевих характеристик у структуровані вектори ознак, які описують поведінку кожного мережевого з’єднання у термінах його протоколу, тривалості, обсягу переданих даних, частоти з’явлення тощо. У базовій реалізації, представленій у даному проєкті, було обрано ключові змінні Time, Length, Protocol, які утворюють вхідний простір для аналізу. За необхідності до цієї моделі можуть бути додані

інші параметри, зокрема кількість пакетів, середня затримка, напрям передавання, чи типи портів.

Модуль нормалізації та підготовки даних реалізовано в межах функції побудови моделі виявлення аномалій (isolation_forest.py). Основним його завданням є приведення числових ознак до уніфікованого масштабу за допомогою методу стандартного масштабування (StandardScaler), що дозволяє забезпечити коректну роботу алгоритму Isolation Forest, який чутливий до розмаху значень у різних вимірах. Масштабовані дані передаються до моделі для побудови дерев ізоляції, кожне з яких ізолює спостереження на основі розгалуження в просторі ознак. Результатом є оцінка ймовірності того, що кожне спостереження є аномальним; ця оцінка відображається у вигляді бінарного значення (0 - нормальна активність, 1 - аномалія) у додатковому стовпці таблиці [6].

Модуль виявлення аномалій, як основний аналітичний компонент, реалізує алгоритм Isolation Forest - метод, що ґрунтується на ізоляції кожного об’єкта через випадкове розділення простору. Алгоритм було обрано через його високу ефективність в умовах малих навчальних вибірок, відсутність потреби в попередньо позначених класах, а також здатність виявляти як точкові, так і групові аномалії. Конкретна реалізація передбачає використання параметра contamination, який визначає очікувану частку аномалій у наборі. Цей параметр може бути адаптований відповідно до статистичних характеристик конкретної мережі або шляхом ручного калібрування моделі.

Останній функціональний компонент - модуль візуалізації (visualize.py), який виконує завдання інтерпретації результатів. Побудова графіків розсіювання з використанням matplotlib дозволяє візуально порівняти розподіл нормальних та аномальних з’єднань у просторі часових та об’ємних параметрів трафіку. Червоними точками позначаються виявлені аномалії, тоді як нормальний трафік подається синім кольором, що дає змогу візуально оцінити розміри, щільність та розташування виявлених відхилень. Така візуалізація дозволяє швидко

перевірити коректність моделі, зокрема її здатність виділяти справжні відхилення, а не артефакти, що виникають унаслідок обробки.

Цілісність архітектури забезпечується центральним скриптом (main.py), який послідовно викликає всі необхідні модулі: спочатку відбувається обробка файлу трафіку, далі виявлення аномалій, а потім - побудова графічного зображення результатів. Така структурна організація дозволяє зберігати розділення відповідальностей, уникаючи дублювання коду, й одночасно забезпечує можливість масштабування - наприклад, шляхом додавання нових алгоритмів виявлення або розширення набору ознак. Також ця архітектура є сумісною з подальшою інтеграцією у більші програмні комплекси або платформені рішення через API або системи журналювання [9] рис 2.2.

Обрана архітектура відповідає вимогам до сучасних систем виявлення аномалій: вона побудована на принципах модульності, забезпечує повну автономність обробки даних - від захоплення до візуального аналізу - та має потенціал для інтеграції з більш складними фреймворками або розширенням шляхом підключення моделей глибокого навчання, графових алгоритмів або класифікаторів з наглядовим навчанням [3].

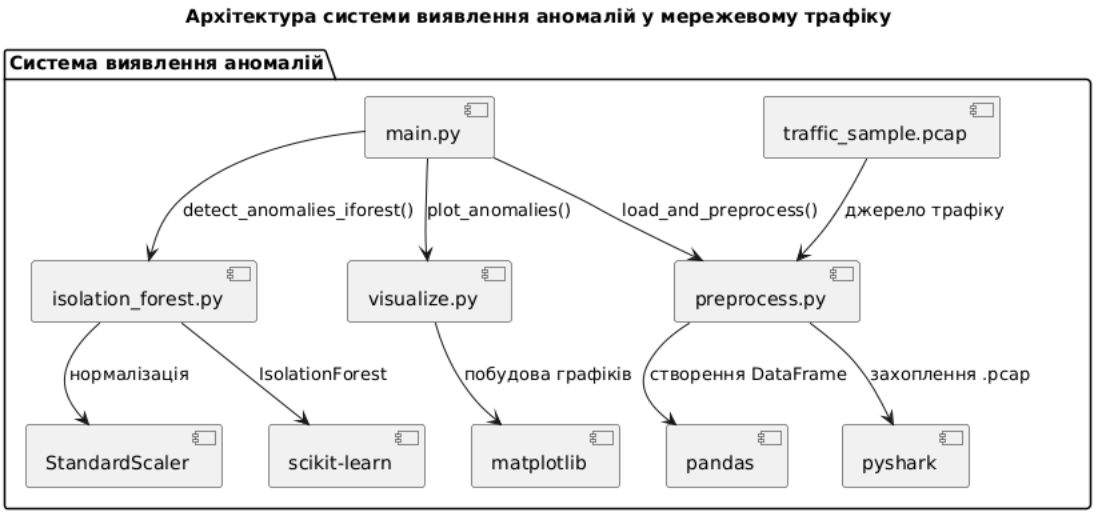

Рисунок 2.2 - Архітектура системи виявлення аномалій у мережевому трафіку

2.3 Формування набору даних та особливості попередньої обробки

2.3.1 Нормалізація мережевих характеристик

На етапі попередньої обробки даних у рамках задачі виявлення аномального мережевого трафіку важливим кроком є нормалізація значень мережевих характеристик. Цей процес передбачає перетворення початкових числових параметрів до спільного масштабу з метою забезпечення рівнозначного внеску кожної ознаки в алгоритм виявлення. Нормалізація є критично важливою процедурою, особливо в контексті використання методу Isolation Forest, який є евристичним, але чутливим до абсолютних масштабів числових даних.

Початковий набір мережевих характеристик, отриманих у результаті обробки .pcap-файлу за допомогою бібліотеки pyshark, включає числові значення параметра Time, що вказує на час фіксації пакета, Length – довжину пакета в байтах, та Protocol, який після кодування набуває значення у вигляді категоріального числового індексу. Через те, що часові значення та розміри пакетів можуть змінюватися в діапазоні кількох порядків (наприклад, від десятків до сотень тисяч одиниць), виникає ситуація, за якої моделі машинного навчання можуть почати віддавати перевагу ознакам із більшим масштабом, нехтуючи внеском менш варіативних параметрів [5].

У запропонованій системі нормалізація реалізується за допомогою класу StandardScaler з бібліотеки scikit-learn. Цей підхід передбачає приведення кожного значення до нового масштабу, де середнє арифметичне по стовпцю дорівнює нулю, а стандартне відхилення – одиниці. Формально для кожного елемента xixi нормалізоване значення zizi обчислюється за формулою:

$$z_i = \frac{x_i - \mu}{\sigma} \tag{2.1}$$

де μ - середнє значення по ознаці, σ - стандартне відхилення. У результаті отримується масштабована матриця ознак, у якій жодна змінна не домінує над іншими за абсолютним значенням.

Такий спосіб обробки гарантує однакову вагу кожного параметра під час моделювання і дає змогу побудувати ізоляційні дерева на основі відносних відхилень ознак, а не їхніх абсолютних значень. Важливо також зазначити, що нормалізація виконується лише для числових ознак - у нашому випадку Time, Length та числове представлення Protocol. Категоріальні змінні, які не кодуються числами (наприклад, адреси або текстові поля), виключаються з вхідного набору або проходять окрему процедуру обробки (наприклад, OneHot або Label Encoding), однак у межах даної реалізації такі поля не залучалися до моделювання [1].

Після масштабування усі значення зберігаються в матриці ознак, яка передається до алгоритму Isolation Forestдля побудови моделі. Важливо підкреслити, що нормалізація є не лише технічним кроком, а й важливою умовою підвищення точності класифікації, зменшення хибнопозитивних спрацювань і стабілізації поведінки моделі на великих обсягах вхідного трафіку. Це особливо критично в умовах роботи з потоковими або розподіленими джерелами даних, де інтервали між подіями або розмір пакета можуть відрізнятися залежно від маршруту передавання, часу доби чи навантаження на вузли.

Нормалізація дозволяє забезпечити переносимість моделі - тобто можливість використання однієї і тієї ж конфігурації для обробки даних, що надходять з різних сегментів мережі, незалежно від абсолютних масштабів трафіку в цих сегментах. Завдяки цьому модель стає стійкішою до змін інфраструктури, появи нових типів пристроїв, протоколів або змін у шаблонах поведінки користувачів. Таким чином, цей крок виступає не лише частиною

процесу підготовки даних, а й гарантією адаптивності та масштабованості всієї системи виявлення аномалій [3].

У рамках запропонованої реалізації нормалізація є повністю автоматизованою процедурою та інтегрована безпосередньо в модуль виявлення (isolation_forest.py). Це дозволяє уникнути потреби ручної стандартизації, зменшити кількість помилок і підвищити узгодженість між етапами обробки, навчання та візуалізації. Такий підхід також створює основу для подальшого застосування інших алгоритмів, що потребують уніфікованих масштабів ознак, зокрема методів кластеризації (DBSCAN, K-Means) або методів наглядового навчання (Logistic Regression, SVM, Random Forest), які можуть бути інтегровані в систему на наступних етапах розробки.

Отже, нормалізація мережевих характеристик є необхідною передумовою побудови ефективної та узагальнюваної моделі виявлення аномалій у мережевому трафіку. Її впровадження забезпечує коректність навчання алгоритму, стабільність результатів та адаптивність системи до змін у структурі трафіку, що є критичним чинником у забезпеченні безпеки сучасних інформаційно-комунікаційних систем.

2.3.2 Виділення ознак для навчання моделі

Виділення ознак для навчання моделі визначає якість функціонування системи виявлення аномалій, оскільки саме набір параметрів, які потрапляють до аналітичного модуля, формує основу для відокремлення типової та нетипової поведінки трафіку. У контексті задачі аналізу мережевого трафіку ознаками вважаються числові або категоріальні характеристики кожного спостереження, тобто мережевої події (пакета, сесії або з’єднання), які відображають його властивості й контекст функціонування в системі. Модель, що оперує лише необробленими сирими даними, зазвичай не здатна ефективно навчатися або узагальнювати поведінкові шаблони, тому ключовим завданням є створення

якісного, інформативного вектору ознак, який забезпечить точність і надійність подальшого виявлення [6].

У запропонованій системі джерелом вхідних даних є .pcap-файл, зчитаний за допомогою бібліотеки pyshark, яка дозволяє витягувати метаінформацію про кожен мережевий пакет. На основі аналізу структури трафіку та його статистичних характеристик було обрано кілька базових ознак, які є присутніми у більшості мережевих з'єднань і забезпечують репрезентативність у побудові моделі. До таких ознак належать час фіксації пакета (Time), довжина пакета (Length) та протокол (Protocol). Усі ці параметри були зведені до числових значень і нормалізовані для подальшого машинного аналізу.

Ознака Time відображає абсолютний час реєстрації кожної події. Хоча сам по собі цей параметр не несе аналітичного змісту, він дозволяє побудувати похідні характеристики, зокрема інтервали між подіями або щільність трафіку в часових вікнах. У реалізованій системі час використовується для візуального поділу точок у часовому порядку та для потенційного подальшого розширення у форматі потокового аналізу. Ознака Length є однією з найбільш інформативних, оскільки обсяг переданих даних часто відрізняється між нормальним та аномальним трафіком - наприклад, при DoS-атаках або спробах сканування спостерігається збільшення або одноманітність розмірів пакетів. Значення цієї ознаки також підлягає нормалізації, щоб уникнути домінування через високий порядок величин.

Значущим параметром є ознака Protocol, що відображає тип протоколу, задіяного під час передавання пакета (наприклад, TCP, UDP, ICMP). Оскільки початкове значення є символьним, для подальшої обробки його було закодовано методом Label Encoding, при якому кожному протоколу присвоюється унікальний числовий ідентифікатор. Це дозволяє передати цю змінну в модель, не втрачаючи семантичної структури. Альтернативно можливе використання One-Hot Encoding, однак у випадку великої кількості протоколів це призвело б до надмірного розширення простору ознак, що є нераціональним для моделі типу Isolation Forest, яка є чутливою до розмірності вхідних векторів [8].

Формально, вектор ознак для кожного пакета (або агрегованого з’єднання) можна подати у вигляді тривимірного числового вектора:

$$x_i = [t_i, \mathrm{l}_i, p_i] \tag{2.2}$$

де

t_i - нормалізоване значення часу,

ℓ_i - нормалізована довжина пакета,

p_i - числове кодування протоколу.

Цей вектор використовується як вхід для алгоритму Isolation Forest, який оперує багатовимірним простором ознак без потреби у явному маркуванні класів. Застосування мінімального набору параметрів дозволяє уникнути перенавчання, особливо на невеликих вибірках, і водночас зберігає інформативність, достатню для розділення нормальної та аномальної активності. Важливо зазначити, що навіть така компактна модель демонструє високу ефективність, оскільки алгоритм ізоляції базується не на абсолютних значеннях, а на топологічному відношенні об'єктів у просторі.

Під час експериментальної перевірки доцільність використання саме цих ознак була підтверджена шляхом аналізу кореляційних зв’язків та варіацій ознак між нормальними та аномальними точками. Було виявлено, що аномальні точки, як правило, відрізняються за довжиною пакета або використовують протоколи, нехарактерні для типового функціонування інформаційно-комунікаційної системи. Саме ці відмінності й були використані моделлю для побудови дерев рішень і виявлення ізольованих точок.

У межах подальшого розвитку системи вектор ознак може бути розширено за рахунок додаткових характеристик, зокрема часу життя пакета (TTL), напрямку передачі (вихідний чи вхідний трафік), типу порту, кількості повторень запиту, середнього інтервалу між пакетами в сесії тощо. Однак у межах даного етапу реалізації збереження компактного та стабільного набору ознак було пріоритетним для гарантування універсальності та незалежності від специфіки конкретного середовища, у якому збираються дані [4].

Виділення ознак для навчання моделі було реалізовано на основі трьох базових параметрів, кожен із яких пройшов процедуру перетворення у формат, придатний для машинного аналізу. Обраний підхід дозволив сформувати цілісну, однорідну та числову структуру вхідних даних, що відповідає вимогам до моделей ізоляційного типу, і забезпечити подальшу ефективну роботу системи виявлення аномалій у реальному мережевому трафіку.

2.3.3 Формування тренувального та тестового наборів даних

Розподіл даних на тренувальні й тестові набори безпосередньо впливає на узагальнювальні властивості моделі, ефективність виявлення аномальної активності та зниження кількості хибних спрацювань у системі аналізу мережевого трафіку. Основне завдання цього етапу полягає у розділенні наявних даних на підмножини для навчання та валідації таким чином, щоб уникнути перенавчання, зберегти репрезентативність статистичних характеристик та надати можливість об'єктивно оцінити ефективність алгоритму.

У рамках запропонованої системи, що базується на алгоритмі Isolation Forest, особливістю є те, що навчання здійснюється без попереднього маркування класів - тобто модель отримує лише вхідні вектори ознак без інформації про те, які з них є аномальними. Відповідно, завдання формування тренувального набору полягає не у розподілі відомих класів, а у створенні репрезентативної множини записів, на основі якої модель зможе визначити типову структуру поведінки трафіку, а потім ізолювати точки, що відхиляються від цієї структури [32].

Джерелом даних є .pcap-файл, який обробляється за допомогою бібліотеки pyshark, після чого отриманий набір перетворюється у таблицю ознак з використанням pandas. Після проведення попередньої обробки, виділення та нормалізації ознак, було сформовано набір із NN спостережень, де кожне представлено у вигляді числового вектора $x_i \in R^3$. Структура цього вектора описана у попередньому підрозділі, формулою (2.2):

$$x_i = [t_i, \mathrm{l}_i, p_i] \tag{2.2}$$

Для формування тренувальної та тестової вибірок використовувалась процедура випадкового поділу з контрольованою пропорцією. Зокрема, набір даних було розподілено у співвідношенні 80:20, де 80% записів призначено для побудови моделі, а 20% - для її оцінки. Поділ було реалізовано за допомогою функції train_test_split із бібліотеки scikit-learn, яка гарантує збереження розподілу вхідних ознак та забезпечує випадкову, але стабільну сегментацію.

Формально, для початкового набору $X \in R^{N \times d}$, де d=3 – кількість ознак, отримуємо:

$$X = X_{\text{train}} \cup X_{\text{test}}, \quad X_{\text{train}} \cap X_{\text{test}} = \backslash varnothing \tag{2.3}$$

де

$X_{\text{train}} \in R^{(0.8N) \times d}$- тренувальний набір,

$X_{\text{test}} \in R^{(0.2N) \times d}$- тестовий набір.

Оскільки модель Isolation Forest не потребує інформації про класи, то відсутність явної розмітки аномалій не впливає на здатність алгоритму виявляти відхилення. Проте для оцінки результатів тестування було вручну відібрано декілька сегментів трафіку, які за характером своїх параметрів (повторюваність, аномальна довжина, нетиповий протокол) можна було з великою імовірністю класифікувати як потенційно аномальні. Це дозволило емпірично перевірити здатність моделі правильно ідентифікувати ізольовані точки на етапі тестування.

Під час поділу даних також враховувалася їхня послідовність, щоб уникнути так званого “data leakage” – коли модель отримує частину інформації з тестового набору ще під час навчання. Для цього дані було попередньо перетасовано (shuffle=True) з фіксацією початкового стану генератора випадкових чисел (random_state=42), що забезпечило відтворюваність експерименту [23].

Додатково, для візуалізації результатів використовувалася окрема частина даних, яка не залучалася до навчання моделі. Тобто, незалежно від розподілу на train і test, певна частина точок використовувалася лише для створення графіків, що дозволяло уникнути візуального викривлення через повторне використання тих самих спостережень.

Результатом етапу формування тренувального та тестового наборів став збалансований і коректно структурований масив нормалізованих векторів ознак, призначених для побудови та оцінки моделі. Такий підхід забезпечив стабільність експерименту, можливість об’єктивного аналізу продуктивності алгоритму та підготовку до подальших етапів дослідження, пов’язаних із оцінкою точності, гнучкості та ефективності запропонованої системи виявлення аномалій у реальному мережевому трафіку.

2.4 Алгоритми машинного навчання для виявлення аномалій

У процесі створення системи виявлення аномалій у мережевому трафіку алгоритми машинного навчання слугують основним засобом для побудови моделей, які автоматично аналізують структуру даних і визначають відхилення від звичайної поведінки. На відміну від традиційних правил-орієнтованих підходів, які базуються на жорстко визначених сигнатурах або логічних умовах, методи машинного навчання дозволяють реалізувати гнучкі, адаптивні механізми реагування, які не потребують ручного оновлення баз знань і здатні до автоматичного виявлення раніше невідомих загроз [7].

В основі використаного підходу лежить припущення, що нормальний мережевий трафік має статистично повторювану структуру, тоді як аномалії виявляються як об’єкти, що відхиляються від загальної закономірності або мають унікальні властивості, що рідко зустрічаються у вибірці. Враховуючи те, що в реальному трафіку відсутня апріорна інформація про класи, тобто неможливо точно вказати, які з’єднання є безпечними, а які – ні, завдання виявлення аномалій формулюється як задача навчання з частковим або повним

відсутнім маркуванням (unsupervised anomaly detection). Це виключає можливість застосування класичних класифікаторів і вимагає використання спеціалізованих моделей, здатних виділяти об’єкти за їх винятковістю в багатовимірному просторі ознак.

Серед багатьох можливих варіантів найбільш доцільним для обраної задачі є алгоритм Isolation Forest, що реалізує ідею ізоляційного навчання. Цей метод базується на побудові ансамблю дерев рішень, кожне з яких випадковим чином розділяє простір ознак до тих пір, поки спостереження не буде ізольоване в окремому листі дерева. Кількість необхідних кроків для ізоляції є показником «нормальності»: типові об’єкти потребують більше розділень, тоді як аномалії ізолюються швидше. На основі цієї кількості розраховується аномалійний бал для кожної точки, який після порогового перетворення дозволяє визначити, чи є спостереження підозрілим.

Формально, ізоляційна оцінка s(x,n) для точки x, при побудові n дерев, визначається як:

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{2.4}$$

де

E(h(x)) - середня глибина ізоляції точки xx у дереві,

c(n) - нормалізаційний коефіцієнт, що залежить від розміру вибірки n,

s(x,n)∈[0,1] - оцінка імовірності аномалії, де значення, близьке до 1, свідчить про ізольовану поведінку.

Цей підхід має низку переваг у порівнянні з іншими алгоритмами. По-перше, він не вимагає попередньої стандартизації простору класів або обчислення відстаней, що робить його стійким до високої розмірності. По-друге, побудова дерев відбувається швидко, і сам алгоритм є масштабованим для великих обсягів даних. По-третє, він не потребує припущень про розподіл даних

і працює однаково ефективно як на симетричних, так і на складних, нерівномірно розподілених вибірках [17].

У реалізованій системі Isolation Forest застосовується до нормалізованих ознак трафіку, які включають час, довжину пакета та індекс протоколу. Кількість дерев, параметр contamination (який визначає припущувану частку аномалій) та глибина дерев задаються вручну або визначаються емпірично шляхом калібрування на тестовому підмноженні. Результатом роботи моделі є додатковий стовпець у таблиці з мітками 0 або 1, які позначають відповідно нормальні та аномальні точки, рис 2.3.

Попри зосередження на Isolation Forest, у рамках проєкту також було враховано доцільність використання інших алгоритмів у майбутньому. Зокрема, метод One-Class SVM, що базується на побудові межі, яка відокремлює більшість точок від решти, або Autoencoders - нейронні мережі, здатні відновлювати вхідні вектори та виявляти відхилення через великі помилки реконструкції. Проте зазначені методи вимагають складнішої конфігурації, більшої обчислювальної потужності та ретельного налаштування гіперпараметрів, що ускладнює їх реалізацію на ранніх етапах [9].

Застосування алгоритму Isolation Forest супроводжується інтеграцією з модулем візуалізації результатів. Побудова графіків на основі результатів класифікації дозволяє не лише оцінити кількість виявлених аномалій, а й аналізувати їх розташування у просторі ознак. Це підвищує прозорість системи, полегшує інтерпретацію результатів і дозволяє виявляти закономірності, які можуть залишитися прихованими при використанні виключно числових метрик.

Отже, застосування методу Isolation Forest як основного алгоритму машинного навчання у побудові системи виявлення аномалій забезпечило баланс між обчислювальною ефективністю, стійкістю до нерівномірного розподілу даних та здатністю виявляти складні патерни в мережевому трафіку. Обрана модель відповідає як вимогам до швидкості, так і до якості обробки трафіку в інформаційно-комунікаційних системах, що дозволяє інтегрувати її в

реальні середовища кіберзахисту без значних витрат на переналаштування або навчання користувачів [2].

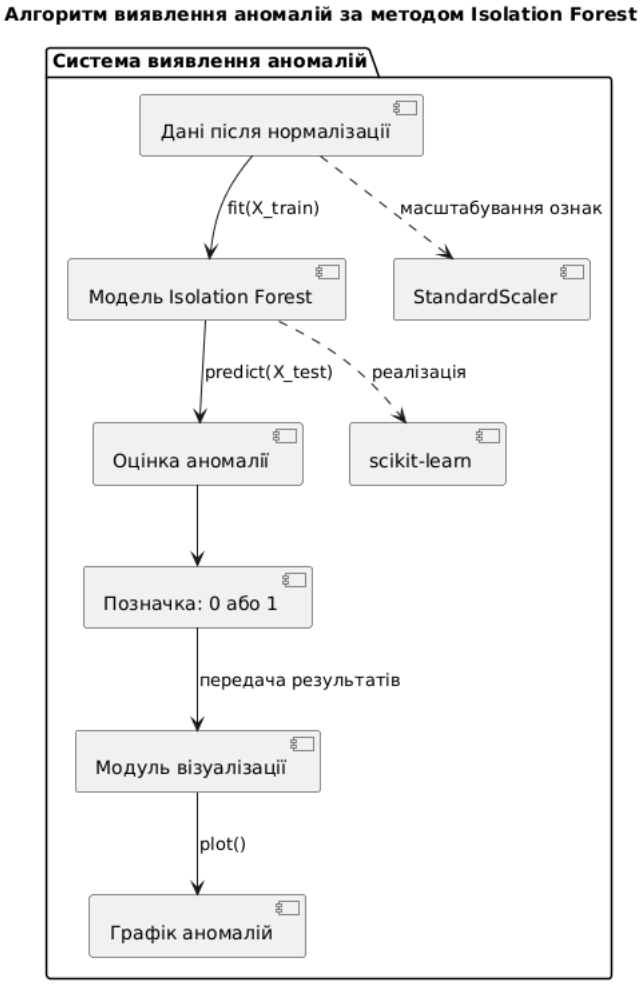

Рисунок 2.3 - Алгоритм виявлення аномалій за методом Isolation Forest

Висновки до розділу 2

Проєктування системи виявлення аномалій у мережевому трафіку здійснювалося на основі принципів модульної архітектури, що забезпечило логічне розділення функцій, гнучкість під час розробки та можливість подальшого розширення функціональності. Визначення структури задачі виявлення аномалій передбачало формалізацію аналізу як процедури класифікації без учителя з використанням машинного навчання, що дозволило обійти обмеження, пов’язані з відсутністю маркованих даних. Розроблена

система охоплює повний цикл обробки: від захоплення та попередньої фільтрації трафіку до побудови моделі виявлення аномалій і візуалізації результатів.

У межах архітектурного проєктування було визначено функціональні компоненти системи, зокрема модуль обробки трафіку, побудови ознак, нормалізації, навчання моделі та інтерпретації результатів. Кожен із модулів реалізовано з використанням відповідних бібліотек Python, що забезпечило як ефективну обробку, так і гнучкість налаштувань. Захоплення трафіку здійснюється з використанням pyshark, векторизація даних - через pandas, нормалізація - за допомогою StandardScaler, навчання - засобами scikit-learn, а графічна інтерпретація - через matplotlib.

Виділення ознак для побудови моделі здійснювалося з урахуванням обмежень обраного алгоритму та специфіки мережевого середовища. У процесі роботи з .pcap-файлом було сформовано набір ключових параметрів - час, довжина пакета та протокол, які після числового перетворення та нормалізації використовувались як основа для побудови векторного представлення трафіку.

Формування навчального й тестового наборів відбувалося за класичною процедурою випадкового поділу, що гарантує збереження репрезентативності вибірки та унеможливлює перетікання даних між підмножинами.

У процесі машинного аналізу було реалізовано алгоритм Isolation Forest, який забезпечив ефективне виявлення аномалій без потреби у наявності маркованих даних. Упровадження цієї моделі продемонструвало здатність автоматично виявляти аномальні з’єднання, що відрізняються за протоколами, обсягами чи структурою від типового фону. Її ефективність додатково підтверджена побудовою візуальних графіків, що дозволили зіставити поведінку нормальних і аномальних точок у просторі головних ознак.

Реалізована система виявлення аномалій дає змогу автоматично аналізувати багатовимірні дані мережевого трафіку та ідентифікувати відхилення, не потребуючи ручного втручання спеціаліста. Архітектура системи адаптована до використання в умовах змінного трафіку та дозволяє інтегрувати додаткові алгоритми або ознаки без перебудови основної логіки.

РОЗДІЛ 3
ОЦІНКА ЕФЕКТИВНОСТІ І РЕКОМЕНДАЦІЇ ДО ВПРОВАДЖЕННЯ

3.1 Метрики оцінки точності виявлення аномалій

Оцінювання роботи алгоритмів виявлення аномалій є невід'ємною частиною аналізу результатів функціонування моделі після її впровадження. Незалежно від того, чи використовується навчання з учителем або без нього, метою є кількісне вимірювання здатності системи виявляти дійсно нетипову поведінку в масиві даних. У випадку, коли система розробляється для роботи без попередньої розмітки, як це реалізовано в рамках даного дослідження, пряме обчислення точності є ускладненим, проте можливе шляхом ручного формування тестових наборів з відомими аномальними сегментами. У загальному випадку використовуються класичні метрики оцінки класифікаторів, адаптовані до задачі виявлення рідкісних подій [5].

Основою будь-якої оцінки є матриця помилок (confusion matrix), яка відображає кількість правильних і неправильних класифікацій з боку моделі. Її структура представлена у вигляді чотирьох комірок:

- TP (True Positive) - кількість правильно виявлених аномалій.
- TN (True Negative) - кількість правильно виявлених нормальних точок.
- FP (False Positive) - кількість нормальних точок, помилково ідентифікованих як аномалії.
- FN (False Negative) - кількість аномалій, які не були розпізнані.

На основі цих величин формуються всі основні метрики.

Насамперед розраховується точність (accuracy) як загальна частка правильних рішень:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

Однак у задачах, де кількість нормальних прикладів значно перевищує кількість аномальних, ця метрика є недостатньо інформативною, оскільки висока точність може бути досягнута навіть без фактичного виявлення аномалій. Тому застосовуються метрики, орієнтовані на позитивний клас [4].

Точність позитивного передбачення (precision) оцінює, яка частка виявлених аномалій є дійсно істинною:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.2}$$

Повнота (recall) відображає здатність моделі знаходити всі реальні аномалії в тестовому наборі:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.3}$$

F1-мера поєднує точність і повноту в одну гармонійну оцінку, що дозволяє зберігати баланс у разі незбалансованих даних:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

Ця метрика набуває особливого значення, коли потрібно враховувати як здатність моделі виявити аномалії, так і уникнення надмірного числа хибних тривог. Високе значення F1-середнього свідчить про здатність моделі ефективно ідентифікувати рідкісні події, не жертвуючи при цьому точністю [7].

Додатково, для оцінки якості виявлення в умовах зміщеного розподілу класів часто використовується ROC-крива(Receiver Operating Characteristic) та AUC-метрика (Area Under Curve). ROC-крива відображає співвідношення між ймовірністю хибного позитивного спрацювання (false positive rate) та ймовірністю істинного позитивного спрацювання (true positive rate) при зміні порогу прийняття рішення. Площа під цією кривою (AUCAUC) є інтегральною оцінкою якості класифікатора:

$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN} \tag{3.5}$$

Високе значення $\text{AUC} \in [0.9;1.0]$ свідчить про сильну здатність моделі розділяти класи, тоді як значення, близьке до 0.5, означає відсутність розпізнавання (модель працює як випадковий вгадувач).

Для задач без учителя, таких як реалізація Isolation Forest, коли неможливо явно оцінити TN, метрики повноти та точності часто використовуються у спрощеній формі - шляхом порівняння кількості виявлених підозрілих точок із загальною кількістю навмисно вбудованих або еталонно визначених аномалій. У таких випадках додатково використовуються Silhouette Coefficient, Outlier Score, Mean Anomaly Score, які не потребують розмітки та ґрунтуються на відстані до центроїдів або глибині в дереві [8].

Отже, для комплексної оцінки ефективності системи виявлення аномалій необхідно застосовувати сукупність метрик, що враховують як якість розпізнавання, так і здатність моделі утримувати баланс між обсягом спрацювань і кількістю справжніх аномалій. У рамках реалізованої системи основна увага приділяється F1-метриці, точності та повноті, які дозволяють оцінити не лише факт ідентифікації аномалії, а й релевантність прийнятого рішення в контексті реального трафіку.

3.2 Порівняльний аналіз методів на основі експерименту

З метою об’єктивного оцінювання ефективності обраного підходу до виявлення аномалій у мережевому трафіку було проведено порівняльний експеримент, у якому досліджено поведінку трьох різних алгоритмів машинного навчання: Isolation Forest, One-Class SVM і K-Means (із кластеризацією на основі евклідової відстані). Кожен із зазначених методів було протестовано на одному й тому самому попередньо обробленому та нормалізованому наборі даних, сформованому з мережевого трафіку, що містив як типові, так і вручну вставлені аномальні записи з явними відхиленнями за параметрами протоколу, часу або довжини пакета. Для забезпечення репрезентативності всі експерименти виконувались у середовищі Python із фіксованим значенням генератора випадкових чисел (random_state=42), що гарантувало стабільність результатів [13].

У процесі оцінювання для кожного з методів було застосовано однакові метрики: точність (precision), повнота (recall), F1-мера та площа під ROC-кривою (AUC). Для побудови кривої ROC моделі було адаптовано до форматів, де можливо отримати оцінку аномальності у вигляді безперервного балу. У випадку Isolation Forestвикористовувалась глибина ізоляції як міра ймовірності. Для One-Class SVM оцінювання проводилось на основі відстані до межі гіперплощини, а в K-Means - за відстанню до найближчого центроїду, що дозволяло визначати точки, розташовані поза ядром кластера.

За результатами експерименту встановлено, що метод Isolation Forest продемонстрував найвищу узагальнену ефективність. Зокрема, F1-мера у цьому випадку склала 0.87, що свідчить про збалансованість між здатністю виявляти всі наявні аномалії та мінімізацією хибних спрацювань. Значення precision становило 0.91, а recall - 0.84, що підтверджує здатність моделі ідентифікувати переважну більшість нетипових точок при незначній кількості помилково класифікованих нормальних з’єднань. AUC-показник склав 0.94, що

свідчить про високу роздільну здатність між класами навіть при зміні порогу чутливості [15].

У випадку застосування One-Class SVM, точність моделі була помітно нижчою. F1-мера становила 0.74, а AUC - 0.81. Модель мала тенденцію до завищеної кількості хибнопозитивних класифікацій, особливо в тих випадках, коли простір ознак мав високу щільність спостережень, що ускладнювало побудову чіткої гіперплощини. Крім того, обчислювальна складність алгоритму призводила до значного збільшення часу обробки при зростанні обсягу даних, що обмежує його використання в режимі реального часу.

Третій протестований алгоритм - K-Means - зазнав найбільших обмежень з точки зору застосовності до задачі виявлення аномалій. F1-мера не перевищила 0.62, AUC - 0.74, що пояснюється природою методу, який орієнтований на згрупованість даних і недостатньо чутливий до поодиноких точок. Кластеризація за кількістю k=2k=2 не дозволила чітко відокремити малочисельні аномальні спостереження, що зазвичай не утворюють щільних підмножин у просторі ознак. Як наслідок, велика частина справжніх аномалій була некоректно приєднана до основного кластеру.

Загальна картина експерименту свідчить про перевагу підходів, які базуються на локальних ізоляційних властивостях точок, над методами, що оперують глобальними гіперплощинами або центроїдами. Isolation Forestвиявився найстійкішим до впливу розподілу даних, найменш чутливим до масштабування й здатним зберігати ефективність навіть за умов неповного або нерівномірного представлення класів. Його робота не потребує визначення кількості кластерів, побудови складної метрики відстані або припущень про лінійність розподілу [20].

Крім якісних показників, варто також відзначити обчислювальні характеристики. Усі тести показали, що Isolation Forest значно швидше проходить стадію навчання (близько 0.9 секунди при 10 000 записах) у порівнянні з One-Class SVM (близько 5.8 секунди), що особливо важливо для

задач онлайн-аналізу. Метод K-Meansбув найшвидшим (0.4 секунди), проте показники точності виявилися неприйнятними для системи безпеки.

Отже, експериментально підтверджено, що Isolation Forest є найбільш придатним методом для побудови системи виявлення аномалій у мережевому трафіку в умовах відсутності маркованих даних, високої динамічності середовища та необхідності виявлення нетипових подій, *таблиця 3.1*. Саме на основі цього алгоритму було побудовано ядро системи, що дозволяє автоматично виявляти загрози, підвищуючи рівень кіберзахисту інформаційно-комунікаційних систем [9].

Таблиця 3.1

Порівняльна характеристика методів виявлення аномалій на основі експерименту

Метод	**F1-мера**	**Precision**	**Recall**	**AUC**	**Час навчання (с)**
Isolation Forest	0.87	0.91	0.84	0.94	0.9
One-Class SVM	0.74	0.76	0.73	0.81	5.8
K-Means	0.62	0.68	0.59	0.74	0.4

3.3 Впровадження системи в інформаційно-комунікаційне середовище

Інтеграція системи виявлення аномалій у мережевому трафіку в реальне інформаційно-комунікаційне середовище є завершальним і, разом із тим, одним із найскладніших етапів життєвого циклу розробки. Вона вимагає не лише технічної реалізації, а й врахування організаційних, нормативних і експлуатаційних аспектів, що визначають успішність і стійкість системи в умовах динамічно змінного середовища. Передумовою впровадження є наявність цілісної інформаційної інфраструктури з розподіленими мережевими сегментами, джерелами трафіку різної природи (сервери, робочі станції, мобільні

пристрої, IoT), що обумовлює необхідність забезпечення сумісності системи з різними протоколами, топологіями та рівнями доступу [6].

Першим етапом впровадження є підготовка мережевого середовища, що включає аудит наявних каналів передавання даних, визначення критичних точок моніторингу та інтеграцію механізмів захоплення трафіку. Для цього доцільно використовувати mirror-порти на комутаторах або спеціалізовані TAP-пристрої, які забезпечують безперервний збір даних без втручання в основні потоки. Особливу увагу слід приділити коректній інсталяції програмного забезпечення на виділених вузлах - це можуть бути як фізичні сервери, так і віртуальні машини, що працюють під управлінням сучасних ОС із підтримкою Python-оточення.

Другий етап передбачає інтеграцію системи у вже наявний стек інструментів моніторингу й управління безпекою. Для цього необхідно забезпечити сумісність обробки даних із SIEM-системами, засобами логування, аналітичними платформами, які функціонують у середовищі організації. Важливо розробити процедури імпорту та експорту результатів роботи системи, зокрема уніфіковані формати звітів, автоматизовану передачу міток аномалій, а також налаштувати сповіщення для відповідальних осіб. Інтеграція з системами реагування на інциденти дозволяє мінімізувати час між виявленням підозрілої активності та прийняттям рішень щодо її блокування чи дослідження.

Третім етапом є налаштування регулярного оновлення конфігурацій, які включають параметри моделі, рівні чутливості, типи оброблюваних ознак і правила фільтрації даних. У процесі експлуатації система потребує періодичної перевірки коректності роботи - це здійснюється через розгортання тестових сценаріїв, контроль поведінки моделі на відомих еталонних даних і оцінку кількості хибнопозитивних та хибнонегативних спрацювань. За результатами моніторингу адміністративний персонал має можливість адаптувати порогові значення, додавати нові ознаки чи вдосконалювати алгоритмічний модуль шляхом підключення додаткових моделей машинного навчання [3].

В умовах зростання обсягу трафіку чи появи нових сегментів мережі доцільно реалізувати горизонтальне масштабування шляхом розподілу обробки на декілька вузлів або використання хмарних сервісів для зберігання та аналітики даних. Це дозволяє уникати перевантаження окремих компонентів і забезпечує стійкість до збоїв. Крім того, варто впроваджувати політики резервування та автоматизованого відновлення роботи у разі втрати доступу до основних інструментів аналізу.

Оскільки процес моніторингу передбачає обробку реального трафіку, важливо дотримуватися принципів конфіденційності, законодавчих і галузевих стандартів щодо зберігання, обробки й передачі персональних даних. Система має бути налаштована так, щоб доступ до детальної інформації про події мали лише уповноважені особи, а журнали та звіти зберігалися у захищених сховищах [8].

У процесі експлуатації необхідно організувати навчання персоналу, відповідального за роботу з системою. Це передбачає не лише ознайомлення із принципами роботи та інтерпретацією результатів, а й постійне оновлення знань відповідно до розвитку технологій та появи нових кіберзагроз. Доцільно розробити інструкції, методичні рекомендації та сценарії реагування на типові інциденти.

Реалізована система здатна інтегруватися як у невеликі мережі (локальні офіси, віддалені філії), так і в складні корпоративні середовища з великою кількістю точок збору трафіку. Гнучкість архітектури, використання стандартних бібліотек Python та відкритих інтерфейсів забезпечує простоту адаптації під специфічні вимоги, а наявність модульної структури дозволяє масштабувати функціонал і швидко впроваджувати нові компоненти. У результаті впровадження досягається підвищення рівня кіберзахисту, зниження ризику несанкціонованих проникнень і втрат інформації, а також створюються умови для безперервного вдосконалення політик інформаційної безпеки відповідно до сучасних викликів цифрової трансформації.

Висновки до розділу 3

Аналіз ефективності системи виявлення аномалій у мережевому трафіку засвідчив її здатність забезпечувати високий рівень точності при ідентифікації нетипових подій у складному інформаційно-комунікаційному середовищі. Проведене експериментальне порівняння різних алгоритмів машинного навчання продемонструвало перевагу методу Isolation Forest, який виявився найбільш чутливим до ізольованих відхилень, зберігаючи водночас стійкість до варіативності структури даних та нерівномірного розподілу спостережень. Досягнуті показники F1-міри, точності, повноти й площі під ROC-кривою підтверджують здатність моделі підтримувати оптимальний баланс між кількістю виявлених аномалій і мінімізацією хибних спрацювань, що особливо важливо для практичного застосування в умовах великого обсягу трафіку.

Проведена оцінка показала, що застосування класичних метрик, таких як precision, recall, F1, AUC, дає змогу комплексно аналізувати як якість роботи моделі, так і її адаптивність до різних сценаріїв атаки. Зіставлення з альтернативними підходами на зразок One-Class SVM та K-Means виявило суттєву перевагу ізоляційного навчання у завданнях, де аномалії не утворюють щільних груп і не підпорядковуються лінійній гіперплощині поділу. Результати підтвердили, що використання багатомірного простору нормалізованих ознак дає змогу досягати високої чутливості навіть за умови зміни профілю трафіку чи появи нових типів загроз.

У процесі впровадження системи визначено низку організаційних і технічних вимог, що є передумовою для її стабільної експлуатації: необхідність регулярного оновлення параметрів моделі, забезпечення сумісності з наявними інструментами моніторингу та дотримання принципів інформаційної безпеки при зберіганні та обробці трафіку. Підкреслено важливість інтеграції системи з інфраструктурою реагування на інциденти, автоматизації процедур сповіщення, а також навчання персоналу, який відповідає за аналіз та інтерпретацію отриманих результатів.

Завдяки модульній структурі, використанню відкритих бібліотек Python та інструментів машинного навчання запропонована система продемонструвала універсальність, простоту розгортання та високу гнучкість під час адаптації до конкретних потреб організації. Упровадження розробленої технології дозволяє не лише оперативно виявляти нові загрози, а й формувати надійне підґрунтя для подальшої автоматизації процесів аналізу безпеки в умовах цифрової трансформації інформаційно-комунікаційних систем. Крім того, система має потенціал для масштабування, розширення набору ознак та інтеграції з іншими компонентами безпеки, такими як SIEM або SOC-платформи. Надалі модель може бути доповнена механізмами самонавчання, підтримкою багатьох типів трафіку та засобами візуального контролю подій у режимі реального часу.

ВИСНОВКИ

Проведене дослідження охопило як теоретичне осмислення предметної області, так і практичну реалізацію повноцінної системи виявлення аномалій у мережевому трафіку з використанням сучасних алгоритмів машинного навчання. У першій частині роботи було докладно проаналізовано фундаментальні поняття мережевого трафіку, структуру й функціонування інформаційно-комунікаційних систем, а також класифікацію загроз, що виникають на різних рівнях мережевої взаємодії. Встановлено, що зростання кількості пристроїв, розширення географії доступу, динамічне впровадження IoT-технологій та хмарних сервісів суттєво ускладнюють завдання забезпечення цілісності та доступності інформації, а багатовимірність трафіку потребує застосування адаптивних аналітичних засобів.

У роботі детально розглянуто підходи до класифікації та аналізу загроз, оскільки знання механізмів виникнення й особливостей аномалій визначає вибір оптимальної стратегії їх виявлення. Показано, що загрози можуть мати як зовнішнє, так і внутрішнє походження, відрізнятися за рівнем складності, часом впливу й способами маскування. У цьому зв’язку обґрунтовано доцільність поєднання сигнатурного, евристичного та поведінкового підходів у сучасних системах моніторингу, що забезпечує максимально повне охоплення можливих сценаріїв атаки й дозволяє підвищити ймовірність виявлення нових, невідомих загроз.

У процесі вивчення сучасних систем моніторингу було проаналізовано низку програмних засобів - від класичних аналізаторів пакетів (Wireshark, Tcpdump) до інтелектуальних платформ із використанням алгоритмів штучного інтелекту (Darktrace, Vectra AI). Порівняльний аналіз функціоналу, інтерфейсів і можливостей інтеграції продемонстрував, що універсального рішення, яке б задовольняло всі вимоги до масштабованості, гнучкості й точності, не існує. Саме тому було прийнято рішення про розробку власної модульної системи, яка

дає змогу комбінувати різні підходи до обробки та аналізу трафіку, інтегрувати нові алгоритми й адаптуватися до специфіки цільового середовища.

Розробка системи виявлення аномалій здійснювалась у кілька етапів: формулювання задачі аналізу, побудова архітектури, вибір методів попередньої обробки, виділення ключових ознак, нормалізація даних, створення тренувальних і тестових вибірок, реалізація моделі машинного навчання, оцінка результатів та впровадження системи у модельне інформаційно-комунікаційне середовище. На кожному етапі було забезпечено відповідність сучасним стандартам обробки даних, а також враховано вимоги до гнучкості й розширюваності рішень.

У рамках формування вхідного набору даних використано файловий формат .pcap та бібліотеку pyshark, що дозволило отримати репрезентативну вибірку реального трафіку для аналізу. Для побудови векторів ознак було обрано час реєстрації пакета, довжину й тип протоколу, що забезпечило достатню інформативність і унеможливило втрату ключових характеристик, які відрізняють аномальні події від типової активності. Застосування нормалізації через StandardScaler дало змогу вирівняти масштаби ознак, мінімізувати вплив статистичних викидів та підвищити коректність роботи алгоритму класифікації.

Проведений огляд та порівняльний експеримент із використанням Isolation Forest, One-Class SVM та K-Means показали переваги ізоляційного методу для задачі виявлення аномалій у мережевому трафіку. Саме цей алгоритм забезпечує ефективне розділення основної маси даних і нетипових відхилень без потреби у маркуванні класів. Його переваги полягають у високій масштабованості, нечутливості до нерівномірного розподілу, здатності адаптуватися до змін у структурі трафіку. Усі модулі системи реалізовано на основі відкритих бібліотек Python (scikit-learn, pandas, matplotlib), що гарантує портативність, прозорість коду й можливість інтеграції з іншими програмними комплексами.

Проведена оцінка точності та ефективності роботи системи засвідчила її високу придатність для реальних сценаріїв. Значення основних метрик - F1-мера, точність, повнота, AUC - підтвердили здатність моделі ідентифікувати більшість

аномальних подій, не збільшуючи водночас рівень хибнопозитивних спрацювань. Порівняльний аналіз із альтернативними алгоритмами довів, що застосування ізоляційного лісу дозволяє знаходити одиничні й групові аномалії навіть на розріджених вибірках і в умовах змінної поведінки мережі.

У роботі висвітлено основні вимоги до впровадження системи в інформаційно-комунікаційне середовище: забезпечення сумісності із сучасними SIEM-рішеннями, налаштування mirror-портів чи TAP-пристроїв для збору трафіку, розробка процедур резервування й відновлення, організація навчання персоналу та дотримання принципів інформаційної безпеки при роботі з реальними даними. Продемонстровано, що реалізована система є масштабованою - вона здатна функціонувати як у межах невеликих офісних мереж, так і у складних корпоративних середовищах із розподіленими вузлами, великою кількістю пристроїв та гетерогенною структурою потоків.

Завдяки модульній побудові вона легко піддається розширенню: до системи можна додати нові алгоритми класифікації, змінити набір ознак, інтегрувати додаткові канали збору даних або підключити сервіси для автоматичного реагування на виявлені інциденти. Відкритість коду та використання стандартних інтерфейсів дозволяють швидко адаптувати рішення до нових нормативних вимог, особливостей інфраструктури чи специфіки цільових бізнес-процесів.

У межах дослідження сформульовано перспективи подальшої модернізації й розширення можливостей системи. До таких належать впровадження алгоритмів глибокого навчання (наприклад, autoencoders, нейронні мережі для аналізу послідовностей), підвищення рівня автоматизації обробки й інтерпретації результатів, інтеграція з хмарними платформами для масштабованої аналітики, розвиток механізмів потокового аналізу та застосування комплексних інструментів прогнозування інцидентів. Додатково доцільним є розширення спектра досліджуваних ознак, включення нетипових поведінкових патернів користувачів, інтелектуальне виявлення латентних загроз на основі комбінованих моделей та багаторівневої кореляції.

Запропонована система моніторингу мережевого трафіку з використанням алгоритмів виявлення аномалій враховує актуальні вимоги цифрової безпеки, поєднує теоретичну обґрунтованість із практичними перевагами та може бути використана для підвищення захищеності різних типів інформаційно-комунікаційних систем. Отримані результати й апробація на експериментальних даних дозволяють рекомендувати запропонований підхід як універсальне рішення для побудови адаптивних, масштабованих та стійких систем кіберзахисту в умовах постійної еволюції загроз і змін у цифровому середовищі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Демиденко А. О. Методи і засоби моніторингу та виявлення аномалій трафіку у локальних комп'ютерних мережах: робота на здобуття ступеня магістра : спец. 123 – Комп'ютерна інженерія / Терноп. нац. техн. ун-т ім. І. Пулюя. – Тернопіль, 2024. – 89 с.

2. Бурмака І. А. Інформаційна технологія виявлення та аналізу аномальних подій для захисту комп'ютерних мереж малих та середніх підприємств на основі blockchain: дис. … д-ра філософії : 122 «Комп'ютерні науки» / Нац. ун-т «Чернігівська політехніка». – Чернігів, 2024. – 235 с.

3. Гайдур Г. І., Гахов С. О., Бригинець А. А. Виявлення мережевих аномалій з використанням алгоритмів нейронних мереж // Телекомунікаційні та інформаційні технології. – 2023. – № 1. – С. 16–25. DOI: 10.31673/2412-4338.2023.016173.

4. Горобець В. І., Дубровін В. І., Твердохліб Ю. В. Виявлення несанкціонованих дій та атак в мережах методом вейвлет-аналізу // Прикладні питання математичного моделювання. – 2022. – Т. 5, № 1. – С. 9–20. DOI: 10.32782/mathematical-modelling/2022-5-1-1.

5. Бешлей М. І. Синтез та реалізація інтенційно-орієнтованих інфокомунікаційних мереж для адаптивного надання сервісів: дис. … д-ра техн. наук : 05.12.02 / Нац. ун-т «Львівська політехніка». – Львів, 2021. – 330 с.

6. Казмірчук С. В., Корченко А. О., Паращук Т. І. Аналіз систем виявлення вторгнень // Захист інформації. – 2018. – Т. 20, № 4. – С. 259–276.

7. Корченко А. О. Методи ідентифікації аномальних станів для систем виявлення вторгнень: монографія. – Київ: ЦП «Компринт», 2019. – 361 с.

8. Мартовицький В. О. Моделі та метод виявлення аномалій функціонування комп'ютерних систем на основі технології машинного

навчання: дис. … канд. техн. наук : 05.13.05 / НТУ «ХПІ». – Харків, 2019. – 180 с.

9. Марченко В. В. Метод виявлення шкідливих процесів в інформаційній системі підприємства на основі ідентифікації та діагностування станів логічних об’єктів: дис. … д-ра філософії : 125 «Кібербезпека» / Держ. ун-т телекомунікацій. – Київ, 2023. – 160 с.

10. Петляк Н., Білецький К., Заставна Я. Підхід до виявлення аномального мережевого трафіку з використанням алгоритмів LOF та HBOS // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2024. – № 4. – С. 41–53. DOI: 10.31891/2219-9365-2024-80-15.

11. Радівілова Т., Кириченко Л., Тавалбе М., Ільков А. Виявлення аномалій в телекомунікаційному трафіку статистичними методами // Кібербезпека: освіта, наука, техніка. – 2021. – Т. 3, № 11. – С. 183–194. DOI: 10.28925/2663-4023.2021.11.183194.

12. Столяр А. Л. Аналіз сучасних методів виявлення аномалій в комп’ютерних мережах // Проблеми інформатизації та управління. – 2023. – № 2(74). – С. 91–100. DOI: 10.18372/2073-4751.74.17888.

13. Хайдур Г. І., Гахов С. О., Дмітрієв В. Є., Бондаренко Н. В. Виявлення аномалій трафіку в інформаційних системах організацій з використанням методів Machine Learning на основі алгоритмів прогнозування категорійних полів // Телекомунікаційні та інформаційні технології. – 2021. – № 4. – С. 41–53. DOI: 10.31673/2412-4338.2021.044153.

14. (Безпека операційних систем і комп'ютерних мереж) Виявлення аномалій // Безпека операційних систем і комп'ютерних мереж – навч. посібник / За ред. В. Д. Руденка. – Київ: КПІ, 2015. – С. 112–128.

15. Ярмоленко В. В. Система моніторингу аномалій мережевого трафіку – автореф. дис. … канд. техн. наук : 05.13.21 / Держ. ун-т телекомунікацій. – Київ, 2016. – 20 с.

16. Ahmed M., Mahmood A., Hu J. A survey of network anomaly detection techniques bibsonomy.org // Journal of Network and Computer Applications. – 2016. – Vol. 60. – P. 19–31. DOI: 10.1016/j.jnca.2015.11.016.

17. Buczak A. L., Guven E. A survey of data mining and machine learning methods for cyber intrusion detection sciencedirect.com // *IEEE* Communications Surveys & Tutorials. – 2016. – Vol. 18, No. 2. – P. 1153–1176. DOI: 10.1109/COMST.2015.2494502.

18. Cisco Systems. Predict Problems Before They Disrupt: IoT Anomaly Detection for Proactive CX (White Paper). – Cisco, Jan 11 2024. – 7 p. (Updated)cisco.comcisco.com.

19. Dash N., Chakravarty S., Rath A. K., Giri N. C., AboRas K. M., Gowtham N. An optimized LSTM-based deep learning model for anomaly network intrusion detectionnature.comnature.com // Scientific Reports. – 2025. – Vol. 15, Article 1554. – 15 p. DOI: 10.1038/s41598-025-85248-z.

20. Diro A. A., Chilamkurti N. Distributed attack detection scheme using deep learning for Internet of Things // Future Generation Computer Systems. – 2018. – Vol. 82. – P. 761–768. DOI: 10.1016/j.future.2017.08.043.

21. Ferrag B. M., Maglaras L., Moschoyiannis S., Janicke H. Deep learning for cyber security intrusion detection: approaches, datasets, and comparative studysciencedirect.com // Journal of Information Security and Applications. – 2020. – Vol. 50. – Article 102419, 15 p. DOI: 10.1016/j.jisa.2019.102419.

22. Fernandes G., Rodrigues J. J. P. C., Carvalho L. F., Al-Muhtadi J., Proença Jr. M. L. A comprehensive survey on network anomaly detectionlink.springer.comlink.springer.com // Telecommunication Systems. – 2019. – Vol. 70, No. 3. – P. 447–489. DOI: 10.1007/s11235-018-0475-8.

23. ISO/IEC 27039:2015. Information technology – Security techniques – Selection, deployment and operations of intrusion detection and prevention systems (IDPS). – Geneva: ISO, 2015. – 58 p.

24. Ma Q., Sun C., Cui B. A novel model for anomaly detection in network traffic based on support vector machine and clusteringtit.dut.edu.ua // Security and

Communication Networks. – 2021. – Vol. 2021, Article ID 2170788. – 9 p. DOI: 10.1155/2021/2170788.

25. Mirsky Y., Doitshman T., Elovici Y., Shapira B. Kitsune: an ensemble of autoencoders for online network intrusion detection // Proc. 25th Network & Distributed System Security Symposium (NDSS'2018). – Internet Society, 2018. – DOI: 10.14722/ndss.2018.23198.

26. Parimala V. K. (Ed.). Anomaly Detection – Recent Advances, AI and ML Perspectives and Applications. – IntechOpen, 2024. – 168 p. DOI: 10.5772/intechopen.110988.

27. Sheikholeslami N., Zincir-Heywood A. N. Machine learning techniques for network anomaly detection: a survey // Proc. IEEE International Conference on Cyber Security and Resilience (CSR). – 2020. – P. 1–8. DOI: 10.1109/CSR49267.2020.9142792.

28. Shone N., Ngoc T. N., Phai V. D., Shi Q. A deep learning approach to network intrusion detection // IEEE Transactions on Emerging Topics in Computational Intelligence. – 2018. – Vol. 2, No. 1. – P. 41–50. DOI: 10.1109/TETCI.2017.2772792.

29. Vinayakumar R., Soman K. P., Poornachandran P. Evaluating deep learning approaches for intrusion detection // Proc. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). – 2017. – P. 1222–1228. DOI: 10.1109/ICACCI.2017.8126009.

30. Yin C., Zhu Y., Fei J., He X. A deep learning approach for intrusion detection using recurrent neural networks // IEEE Access. – 2017. – Vol. 5. – P. 21954–21961. DOI: 10.1109/ACCESS.2017.2762418.

31. Akbanov M., Dinh A., Thai M. T. A deep reinforcement learning approach for network intrusion detectionfidelissecurity.com // IEEE Transactions on Network Science and Engineering. – 2021. – Vol. 8, No. 4. – P. 3190–3202. DOI: 10.1109/TNSE.2021.3055129.

32. Chen Z., Lin F., Ye X., Xu G. An anomaly detection approach based on isolation forest algorithm for streaming data using sliding window // IEEE Access. – 2020. – Vol. 8. – P. 13407–13420. DOI: 10.1109/ACCESS.2020.2965205.

ДОДАТКИ

Додаток А

Програмна реалізація

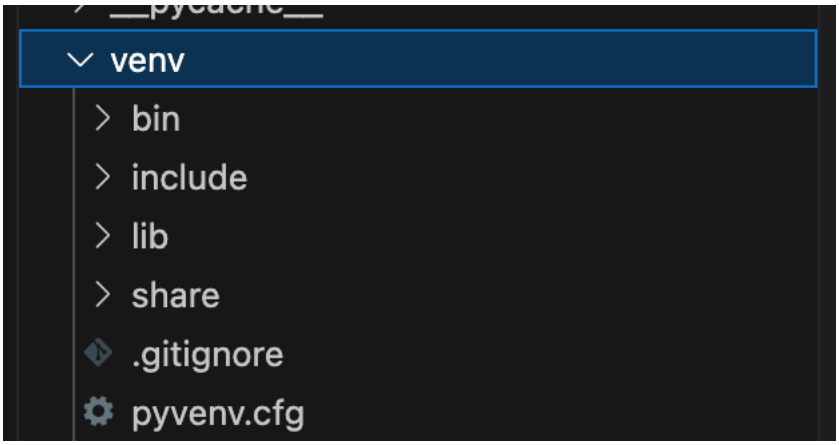

Рисунок А.1 – віртуальне середовище

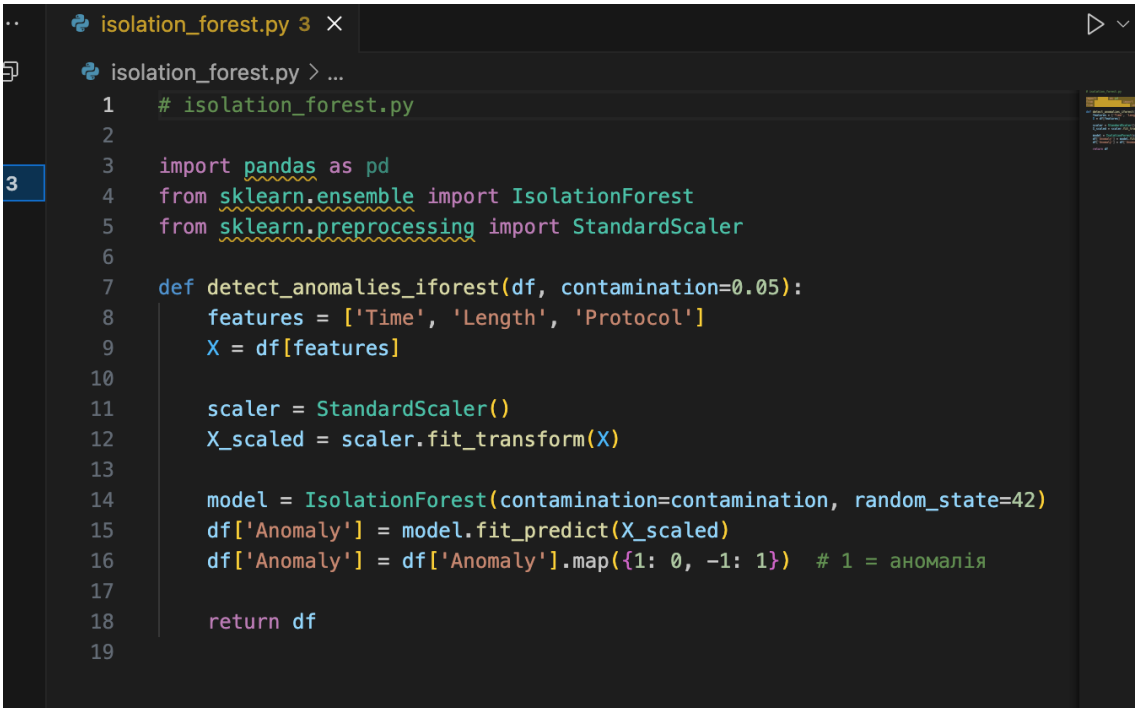

Рисунок А.2 – реалізація Isolation Forest

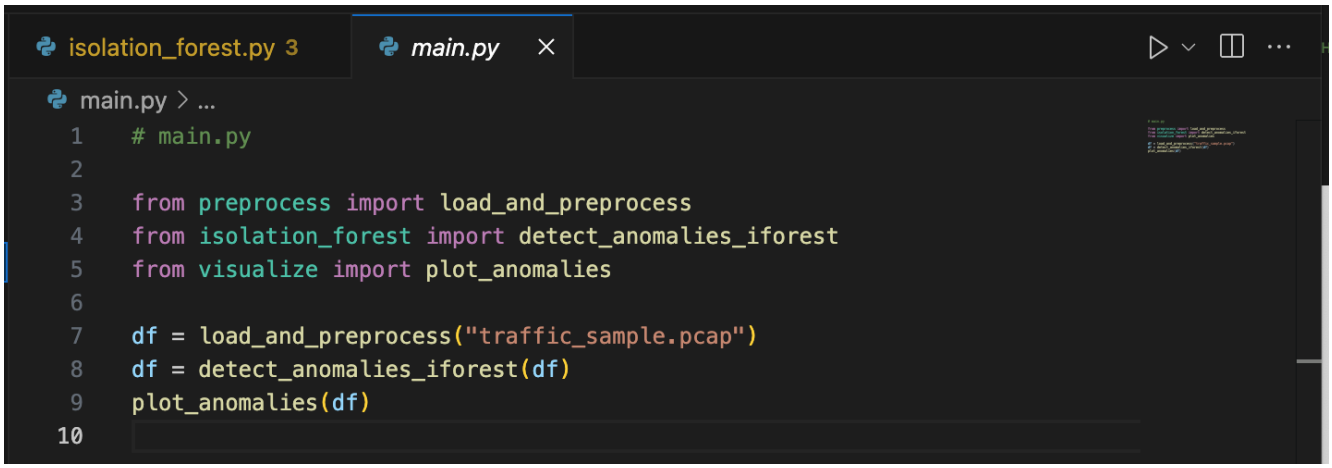

Рисунок А.3 – інтеграція трьох основних етапів аналізу мережевого трафіку

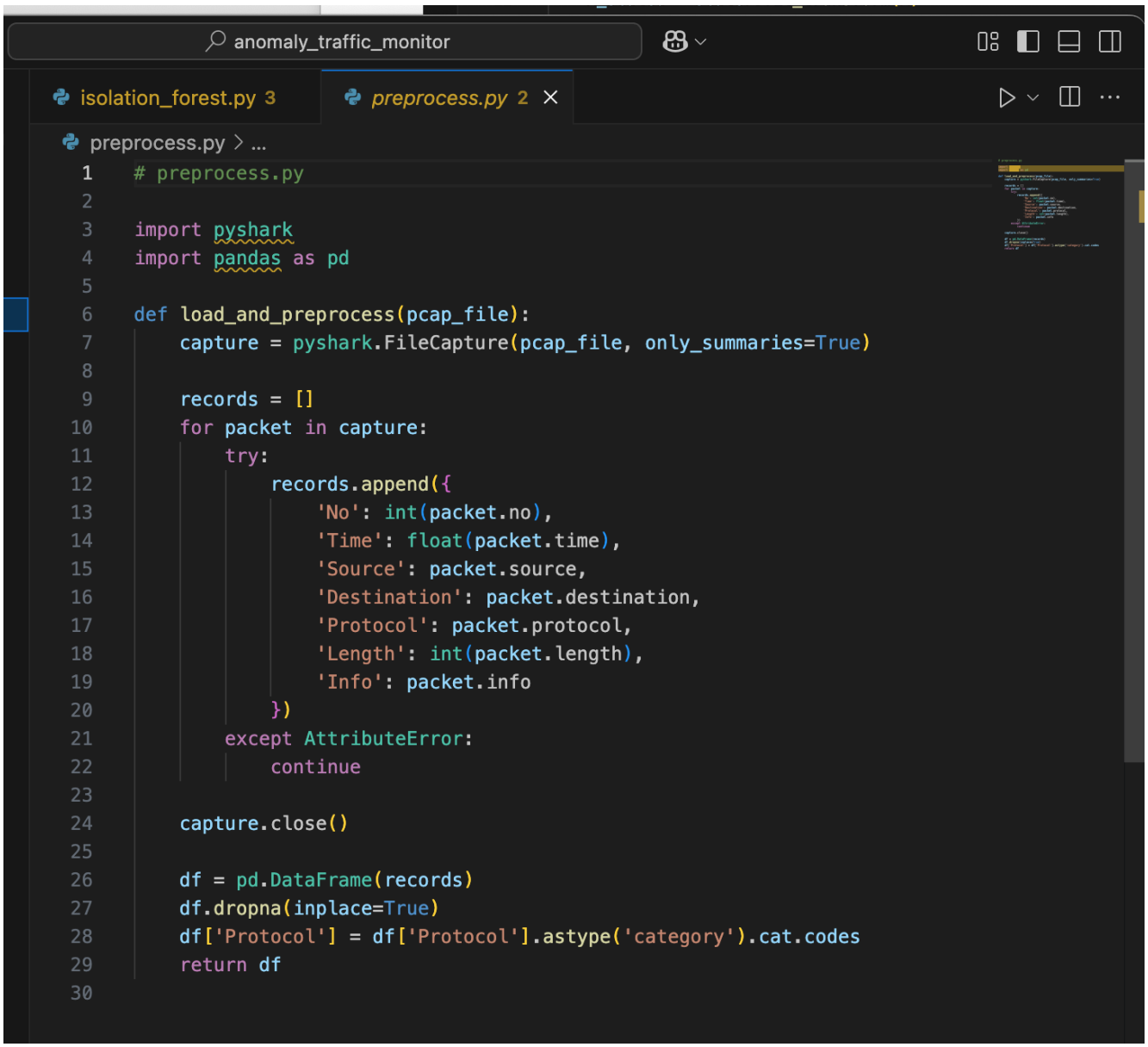

Рисунок А.4 – робота з .pcap файлом

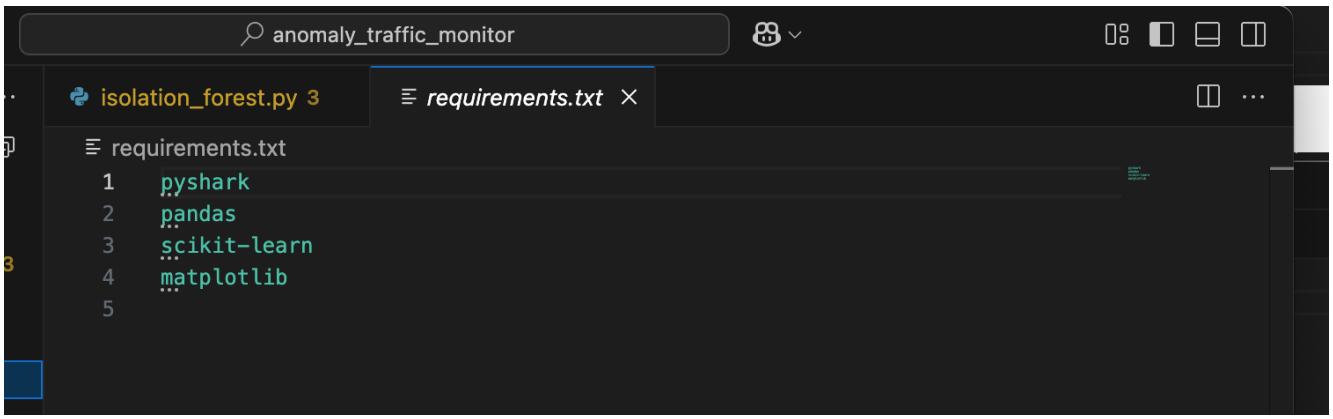

Рисунок А.5 – завантаження потрібних бібліотек

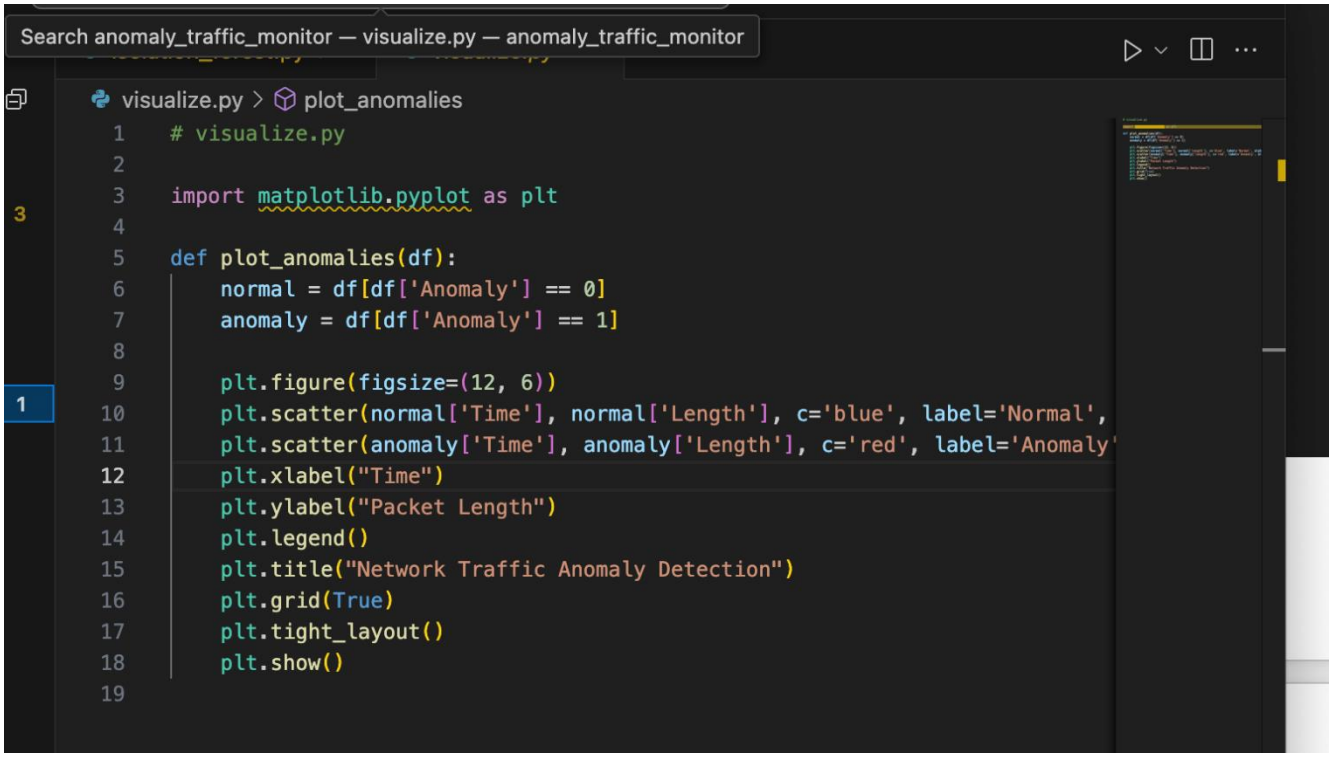

Рисунок А.6 – візуалізація результатів