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

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(освітній рівень)

на тему: "Дослідження методів виявлення аномалій у веб-трафіку з використанням інформаційної системи аналізу лог-даних"

Виконав: студент (ка)

Спеціальності:
125 «Кібербезпека»
(шифр і назва напряму підготовки, спеціальності)
Приймаченко Максим Євгенійович
підпис (прізвище та ініціали)

Керівник Лечаченко Т. А.
підпис (прізвище та ініціали)

Нормоконтроль Стадник М.А.
підпис (прізвище та ініціали)

Завідувач кафедри Загородна Н.В.
підпис (прізвище та ініціали)

Рецензент Марценко С.В.
підпис (прізвище та ініціали)

м. Тернопіль – 2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Магістр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека
(шифр і назва спеціальності)

Студенту Приймаченку Максиму Євгенійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів виявлення аномалій у веб-трафіку з використанням інформаційої системи аналізу лог-даних

Керівник роботи Лечаченко Тарас Анатолійович, PhD доктор філософії., старший викладач кафедри КБ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «24» листопада 2025 року № 4/7-1024 .
2. Термін подання студентом завершеної роботи 17.12.2025
3. Вихідні дані до роботи Лог-файли веб-сервера, що містять інформацію про HTTP-запити.

4. Зміст роботи (перелік питань, які потрібно розробити)
Провести аналіз та визначити вимоги до системи виявлення аномалій у веб-трафіку.
Виконати огляд існуючих рішень і методів аналізу лог-даних та обрати оптимальний підхід.
Розробити архітектуру інформаційної системи аналізу логів на основі Elasticsearch та Kibana.
Обґрунтувати вибір технологій, інструментів та форматів логів для подальшого дослідження.
Розробити та налаштувати процес збору, обробки й нормалізації лог-даних веб-трафіку.
Реалізувати індексацію даних в Elasticsearch та створити візуалізації і дашборди в Kibana.
Розробити та описати програмне забезпечення для виявлення аномалій у веб-трафіку.
Провести дослідження системи й оцінити ефективність запропонованих методів.
Надати рекомендації щодо експлуатації, обслуговування та масштабування системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
Мета та завдання дослідження.
Технічне завдання та вимоги до системи.
Архітектура інформаційної системи аналізу лог-даних.
Використання ML-завдань у Kibana для виявлення аномалій.
Генерація тестових даних у Postman.
Результати експериментів метрики ефективності.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Осухівська Г.М., к.т.н., доцент, зав. кафедри КС		
Безпека в надзвичайних ситуаціях	Теслюк В.М., проректор з адміністративно-господарської роботи та будівництва		

7. Дата видачі завдання 18.09.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної	18.09.2025	Виконано
	роботи		
2.	Підбір та опрацювання джерел про дослідження	30.09.2025-07.10.2025	Виконано
	виявлення аномалій у веб-трафіку		
3.	Вивчення принципів роботи Elasticsearch	09.10.2025-15.10.2025	Виконано
4.	Оформлення розділу “Аналітична частина”	17.10.2025-21.10.2025	Виконано
5.	Оформлення розділу “Основна частина”	24.10.2025-15.11.2025	Виконано
6.	Оформлення розділу “Науково-дослідна частина”	16.11.2025-23.11.2025	Виконано
7.	Оформлення розділу “Охорона праці”	23.11.2025-27.11.2025	Виконано
8.	Оформлення розділу “Берперервність роботи	28.11.2025-3.12.2025	Виконано
	інформаційної системи в умовах воєнних загроз та		
	надзвичайних подій		
9.	Оформлення кваліфікаційної роботи	08.12.2025-10.12.2025	Виконано
10.	Нормоконтроль	12.12.2025-14.12.2025	Виконано
11.	Перевірка на плагіат	15.12.2025-18.12.2025	Виконано
12.	Попередній захист кваліфікаційної роботи	19.12.2025-21.12.2025	Виконано
13.	Захист кваліфікаційної роботи	23.12.2025	

Студент _____ Приймаченко М.Є.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Лечаченко Т.А.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Дослідження методів виявлення аномалій у веб-трафіку з використанням інформаційної системи аналізу лог-даних» // Кваліфікаційна робота магістра // Приймаченко Максим Євгенійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки // Тернопіль, 2025 // с. – 61, рис. – 19, табл. – 1, кресл. – 4, додат. – 1.

Ключові слова: веб-трафік, аналіз лог-даних, Kibana, Elasticsearch, аномалії, виявлення аномалій, інформаційна система моніторингу, кібербезпека.

Кваліфікаційна робота присвячена дослідженню методів виявлення аномальної активності у веб-трафіку з використанням інформаційної системи аналізу лог-даних на базі стеку Elasticsearch, Logstash, Kibana (ELK). Розглянуто сучасні підходи до обробки логів, методи виявлення аномалій, характеристики поширених атак та особливості їх фіксації у лог-файлах. Виконано налаштування інформаційної системи аналізу логів, проведено збір та індексацію веб-трафіку, а також побудовано інструменти візуалізації й пошуку аномальних подій.

У роботі проведено експериментальну оцінку ефективності виявлення аномалій шляхом моделювання нестандартної активності та подальшого аналізу її проявів у системі Kibana. Отримані результати підтверджують можливість застосування системи для раннього виявлення підозрілих подій та підвищення рівня безпеки веб-ресурсів.

Робота може бути використана для практичної організації систем моніторингу безпеки, підвищення надійності веб-сервісів та оптимізації процесів аналізу логів у корпоративних мережах.

ABSTRACT

The topic of the qualification work: “Research of Methods for Detecting Anomalies in Web Traffic Using a Log Data Analysis Information System” // Master's Qualification Work // Pryimachenko Maksym Yevheniiovych // Ternopil Ivan Puluj National Technical University, Faculty of Computer and Information Systems and Software Engineering, Department of Cybersecurity // Ternopil, 2025 // p. – 61, figs. – 19, tbls. – 1, drws. – 4, apps – 1.

Keywords: web traffic, log data analysis, Kibana, Elasticsearch, anomalies, anomaly detection, monitoring system, cybersecurity.

The qualification work is devoted to researching methods for detecting anomalous activity in web traffic using a log data analysis information system based on the Elasticsearch, Logstash, Kibana (ELK) stack. The study examines modern approaches to log processing, methods of anomaly detection, characteristics of common attacks, and features of their manifestation in log files. The system was configured to collect and index web traffic, and visualization tools were created to identify suspicious events.

The work includes an experimental evaluation of anomaly detection effectiveness by simulating abnormal activity and analyzing its manifestations in Kibana. The obtained results confirm the feasibility of using the system for early detection of suspicious events and for improving the security of web resources.

The results of the work can be applied to practical security monitoring systems, enhancing the reliability of web services and optimizing log analysis processes in corporate networks.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI	—	Artificial Intelligence
API	—	Application Programming Interface
HTTP	—	HyperText Transfer Protocol
JSON	—	JavaScript Object Notation
ML	—	Machine Learning
SIEM	—	Security Information and Event Management
SOC	—	Security Operations Center

ВСТУП

Актуальність теми полягає у зростанні кількості кіберінцидентів, пов'язаних із веб-ресурсами, зумовлює необхідність ефективного моніторингу веб-трафіку та своєчасного виявлення аномальних подій. Традиційні методи аналізу логів є малоефективними при великих обсягах даних, характерних для сучасних систем. Використання інформаційних систем аналізу лог-даних, зокрема стеку ELK (Elasticsearch, Logstash, Kibana), відкриває можливості для автоматизації процесів моніторингу, аналітики та виявлення аномалій у режимі, близькому до реального часу.

Актуальність роботи полягає в необхідності підвищення ефективності систем виявлення підозрілої активності та покращення інформаційної безпеки веб-ресурсів за рахунок сучасних методів обробки великих масивів лог-даних. Проведений аналіз наявних рішень показує, що більшість підходів потребують адаптації під конкретні параметри веб-трафіку та особливості атак.

Мета і завдання дослідження – підвищення ефективності виявлення аномальної активності у вебтрафіку шляхом побудови та налаштування інформаційної системи аналізу лог-даних на базі ELK-стеку та оцінки її здатності виявляти нестандартні події.

Для досягнення мети необхідно вирішити такі завдання:

- Проаналізувати особливості формування веб-трафіку та структуру лог-даних веб-серверів.
- Дослідити існуючі методи виявлення аномалій та підходи до аналізу логів.
- Налаштувати інформаційну систему збору, обробки та візуалізації лог-даних на основі Elasticsearch, Logstash та Kibana.
- Здійснити моделювання аномальної активності та дослідити її відображення в логах.
- Провести експериментальну оцінку точності та ефективності виявлення аномальних подій.

- Надати рекомендації щодо впровадження системи моніторингу у реальних веб-інфраструктурах.

Об’єктом дослідження є процес формування, передачі та обробки веб-трафіку в інформаційних системах.

Предметом дослідження є методи та інструменти виявлення аномалій у веб-трафіку на основі аналізу лог-даних з використанням інформаційної системи моніторингу.

У роботі застосовано такі методи:

- експериментальний метод для моделювання аномальної активності та перевірки ефективності системи;
- методи інформаційного моделювання для побудови структури системи аналізу лог-даних;
- візуально-аналітичний метод для побудови дашбордів та інструментів візуального пошуку аномалій у Kibana.

Наукова новизна роботи полягає в розробці та експериментальній перевірці системи поведінкового моніторингу веб-трафіку на основі вбудованих засобів Machine Learning у Kibana. Запропоновано структурований підхід до створення ML-jobs для виявлення аномалій за ключовими ознаками: затримка відповіді (latency), HTTP-статуси (status), IP-адреси (ip), URI-шляхи (endpoint) та User-Agent. Кожна модель налаштована з урахуванням bucket-інтервалу та впливових факторів, що дозволяє локалізувати джерело аномалії та оцінити її природу.

Для формування контрольованої вибірки реалізовано генератор трафіку в Postman, який моделює як нормальні запити, так і типові атаки (SQL-ін’єкції, XSS, DDoS [15], нестандартні User-Agent). Це дозволило створити розмічену множину запитів, на основі якої проведено обчислення метрик якості виявлення: precision, recall, F1-score. Отримані результати підтверджують ефективність запропонованого підходу для виявлення поведінкових аномалій у реальному часі, що має практичне значення для побудови SOC-аналітики та систем кіберзахисту.

Результати роботи можуть бути використані:

- для впровадження систем моніторингу веб-трафіку в організаціях;
- при побудові SOC-процесів та систем раннього виявлення кіберінцидентів;
- для підвищення ефективності обробки логів у корпоративних мережах;
- у навчальному процесі під час підготовки фахівців з кібербезпеки.

Апробація результатів магістерської роботи. Основні результати проведених досліджень обговорювались на XIII науково-технічній конференції «Інформаційні моделі, системи та технології» (м.Тернопіль), 17-18 грудня 2025 р.

Публікації. Основні результати кваліфікаційної роботи опубліковано у тезах конференції (див. Додаток А).

РОЗДІЛ 1 АНАЛІТИЧНА ЧАСТИНА

1.1 Огляд відомих методів та рішень у сфері аналізу лог-даних і детектування аномалій

У сучасних інформаційних системах [18] аналіз логів та виявлення аномалій є ключовими задачами для забезпечення кібербезпеки. Оскільки обсяги веб-трафіку та журнальних записів зростають, традиційні методи ручного аналізу стають неефективними. Тому важливо розглянути існуючі підходи, які використовують у промислових системах та у наукових дослідженнях.

У таблиці 1.1 наведені методи[5] які поділяються на кілька основних груп:

Таблиця 1.1 — Класифікація методів виявлення аномалій

Тип методу	Опис	Приклади
Статистичні	Визначають аномалії як відхилення від нормального розподілу даних.	Z-score, IQR
Засновані на машинному навчанні	Навчаються на історичних даних, щоб виявляти нетипові шаблони.	Isolation Forest, One-Class SVM
Кластеризаційні	Виявляють точки, що не належать до жодного кластера.	DBSCAN, K-Means
Евристичні / сигнатурні	Використовують заздалегідь визначені правила або шаблони.	Snort, Suricata
Гібридні	Комбінують кілька підходів для підвищення точності.	MIDAS, DeepLog

Аналіз лог-даних та виявлення аномалій є ключовими завданнями сучасних систем інформаційної безпеки. У практиці застосовуються різні методи, які можна класифікувати за підходами до обробки даних.

Статистичні методи базуються на визначенні відхилень від нормального розподілу показників. Наприклад, використання Z-score чи інтерквартильного розмаху дозволяє виявляти події, що виходять за межі очікуваних значень.

Перевагою є простота та швидкість, проте такі методи чутливі до шуму та потребують стабільних даних.

Методи машинного навчання застосовують алгоритми, здатні навчатися на історичних даних і виявляти нетипові шаблони. Серед них Isolation Forest, One-Class SVM, автоенкодери. Вони добре працюють із багатовимірними даними та складними залежностями, але вимагають великих обсягів якісних логів і значних обчислювальних ресурсів.

Кластеризаційні методи (DBSCAN, K-Means) дозволяють групувати події та визначати ті, що не належать до жодного кластера. Це ефективно для багатовимірних даних, однак вибір параметрів кластеризації часто є складним і впливає на точність.

Сигнатурні та евристичні методи використовують заздалегідь визначені правила чи шаблони атак. Прикладом є системи Snort та Suricata. Вони ефективні для виявлення відомих загроз [17], але не здатні виявляти нові, невідомі аномалії.

Гібридні методи поєднують кілька підходів для підвищення точності та зниження кількості хибнопозитивних спрацювань. Сучасні приклади — DeepLog, MIDAS, які комбінують статистику та глибоке навчання.

Окрім методів, важливу роль відіграють платформи для збору та аналізу логів. Найбільш поширеним рішенням є Elastic Stack (ELK), що включає ElasticSearch для пошуку та агрегації, Logstash для парсингу та нормалізації, а також Kibana для візуалізації. Splunk є комерційною платформою з потужними засобами аналітики та виявлення аномалій. Graylog — open-source альтернатива з підтримкою потокової обробки. Wazuh інтегрується з ELK та орієнтована на безпеку. Також використовуються Prometheus + Grafana + Loki, які більше орієнтовані на метрики, але можуть застосовуватись і для логів. У великих SOC поширені комплексні SIEM-рішення, такі як IBM QRadar чи ArcSight.

Таким чином, існує широкий спектр методів та інструментів для аналізу лог-даних і детектування аномалій. Кожен підхід має свої переваги та обмеження: статистичні методи прості, але менш гнучкі; машинне навчання забезпечує високу точність, проте потребує ресурсів; сигнатурні рішення ефективні для відомих атак, але не здатні виявляти нові. Це підводить до

висновку про доцільність використання гібридних систем, що поєднують різні методи та інтегруються у сучасні платформи, такі як ELK Stack.

На рисунку 1.1 наведено основні переваги існуючих рішень та підходів, що застосовуються у сфері аналізу лог-даних і виявлення аномалій у веб-трафіку.

Рисунок 1.1 — Переваги та обмеження систем виявлення аномалій

На малюнку наведені ключові переваги та обмеження сучасних систем аналізу лог-даних. До основних переваг належать автоматизація процесів, що дозволяє зменшити участь людини; масштабованість, яка забезпечує обробку великих обсягів даних; а також можливість роботи в реальному часі, що критично для оперативного реагування на інциденти.

Водночас, системи мають низку обмежень. Зокрема, ефективність алгоритмів значною мірою залежить від якості вхідних даних. Крім того, налаштування таких систем потребує високої кваліфікації, а ризик хибнопозитивних спрацювань може ускладнити роботу аналітиків SOC. Ці аспекти слід враховувати при впровадженні та експлуатації систем виявлення аномалій.

На рисунку 1.2 наведено основні напрями сучасних досліджень у сфері аналізу лог-даних та виявлення аномалій: глибоке навчання, адаптивні системи та онлайн-обробка логів.

Рисунок 1.2 — Ключові напрями досліджень у сфері виявлення аномалій

Використання моделей глибокого навчання, таких як LSTM (Long Short-Term Memory) та Autoencoders, дозволяє виявляти складні шаблони поведінки у веб-трафіку. Ці методи враховують часові залежності та контекст, що робить їх ефективними для аналізу послідовностей подій. Основною перевагою є висока точність, проте вони потребують значних обчислювальних ресурсів і якісних даних для навчання.

Адаптивні системи здатні змінювати параметри та моделі в залежності від поточного стану мережі. Це дозволяє враховувати динаміку поведінки користувачів і сервісів, зменшувати кількість хибнопозитивних спрацювань та підвищувати точність виявлення аномалій. Такий підхід особливо актуальний у середовищах із високою варіативністю трафіку.

Зростання обсягів даних стимулює розвиток потокових технологій, які забезпечують обробку логів у режимі реального часу. Платформи на кшталт Apache Kafka, Elasticsearch Streams чи Apache Flink дозволяють здійснювати аналіз без затримок, що відкриває нові можливості для миттєвого реагування на інциденти та інтеграції з SOC.

1.2 Аналіз технічного завдання на розробку системи виявлення аномалій у веб-трафіку

У межах технічного завдання передбачається розробка інформаційної системи, призначеної для виявлення аномальної активності у веб-трафіку на основі аналізу лог-даних з різних джерел [16]. Основна мета системи це підвищення ефективності виявлення кіберзагроз, зменшення часу реакції на інциденти та забезпечення наочності при роботі з великими обсягами логів.

Об’єкти контролю та вхідні дані: у технічному завданні визначено, що система повинна працювати з такими типами логів:

У рамках кваліфікаційної роботи реалізовано підтримку аналізу HTTP/HTTPS-логів веб-сервера.

Загальна архітектура системи виявлення аномалій у веб-трафіку представлена на рисунку 1.3. На схемі відображено ключові етапи роботи системи: збір логів із різних джерел, їх централізоване опрацювання, нормалізація даних, подальше виявлення аномалій та візуалізація результатів для аналітика. Рисунок також демонструє взаємодію між компонентами та рух даних у межах системи, що дозволяє зрозуміти логіку функціонування розробленого рішення.

Рисунок 1.3 — Функціональні вимоги

У представленій схемі зображено основні функціональні вимоги до системи виявлення аномалій у веб-трафіку. На рисунку показано, що система має забезпечувати централізований збір логів із різних мережевих та серверних джерел. Такий підхід дозволяє звести інформацію до єдиного центру обробки та мінімізувати втрати даних при масштабуванні або зміні інфраструктури.

Далі проілюстровано етап парсингу та нормалізації, який є обов’язковим для роботи з логами різних форматів. Уніфікація структури даних дозволяє системі застосовувати єдині правила аналізу та спрощує подальшу обробку. Після цього дані переходять до модуля візуалізації, де ключові метрики веб-трафіку відображаються у вигляді графіків, діаграм і гістограм. Це забезпечує швидке виявлення пікових навантажень, підозрілих запитів та інших відхилень.

Окремим блоком зображено модуль виявлення аномалій, який працює на основі визначених патернів та статистичних методів. Саме він відповідає за автоматичне розпізнавання шкідливої або нетипової активності. Після ідентифікації аномалії система формує повідомлення про інцидент, що дозволяє оперативно реагувати на потенційні загрози. Завершальним елементом схеми є інструментарій для аналітика безпеки, який забезпечує зручний доступ до журналів, візуалізацій та результатів аналізу.

У технічному завданні визначено низку обмежень і критеріїв, які формують нефункціональні вимоги системи. Вони відображені на рисунку 1.4, що демонструє основні показники продуктивності, надійності, доступності та якості інтерфейсу, необхідні для стабільної роботи системи аналізу лог-даних.

Рисунок 1.4 — Нефункціональні вимоги

Рисунок 1.4 ілюструє ключові нефункціональні вимоги, встановлені в межах технічного завдання для системи виявлення аномалій у веб-трафіку. Одним із основних критеріїв є швидкодія: час обробки нових записів у логах не повинен перевищувати десяти секунд. Це забезпечує можливість оперативного реагування на інциденти та підтримку майже реального часу при аналізі подій.

Наступний важливий аспект — надійність та безперервність роботи. Система має функціонувати цілодобово без втрати даних, що вимагає стабільної інфраструктури, коректної організації черг обробки та механізмів відновлення під час збоїв.

Ще один критерій стосується зручності та зрозумілості інтерфейсу. Оскільки кінцевими користувачами є аналітики SOC, інтерфейс має бути веб-орієнтованим, інтуїтивним та таким, що дозволяє швидко знаходити аномальні події, працювати з візуалізаціями та застосовувати фільтри до великого масиву логів.

Також встановлено обмеження на точність алгоритмів виявлення аномалій: допустима похибка не повинна перевищувати 5% хибнопозитивних спрацювань. Це дозволяє забезпечити високу якість аналітики та уникнути інформаційного шуму.

Остання вимога зазначає, що система має бути побудована з використанням ElasticSearch, Logstash та Kibana, що гарантує масштабованість, модульність та підтримку сучасних методів обробки великих масивів даних.

Очікувані результати роботи розробленої системи узагальнені на рисунку 1.5, який демонструє ключові показники ефективності, на досягнення яких орієнтовано проєкт. Вони відображають як технічні, так і операційні переваги, що мають бути забезпечені після впровадження системи виявлення аномалій у веб-трафіку.

Рисунок 1.5 — Очікувані результати роботи системи виявлення аномалій у веб-трафіку

Рисунок 1.5 ілюструє головні очікувані результати, яких має досягти інформаційна система аналізу лог-даних у процесі експлуатації. Одним із ключових показників є суттєве скорочення часу обробки інцидентів: автоматизація аналізу логів та використання візуалізацій.

Ще одним важливим результатом є можливість автоматичного виявлення аномалій. Використання визначених патернів, статистичних методів та алгоритмів обробки поведінкових моделей дозволяє системі самостійно визначати підозрілу активність без необхідності постійної участі аналітика.

Крім того, система має забезпечувати моніторинг у режимі реального часу, що істотно підвищує рівень оперативності реагування на потенційні загрози. Завдяки цьому інциденти фіксуються та обробляються одразу після появи відхилень у поведінці веб-трафіку.

Також очікується, що впровадження системи знизить навантаження на аналітиків SOC. Автоматизація рутинних завдань, таких як фільтрація подій,

попередній аналіз логів та виділення аномальних патернів, дозволяє фахівцям зосередитися на складніших інцидентах та стратегічних завданнях.

На рисунку 1.6 наведено основні обмеження та припущення, визначені під час формування технічного завдання на розробку системи виявлення аномалій у веб-трафіку. Вони задають рамки проєкту та визначають ті умови, за яких система може працювати коректно. Зокрема, підкреслено, що система не втручається у структуру мережі, а функціонує виключно як інструмент аналізу даних, що дозволяє впроваджувати її без змін у поточній інфраструктурі.

Рисунок 1.6 — Обмеження та припущення

Окремо акцентовано увагу на припущеннях щодо якості та доступності даних: передбачається, що веб-трафік є достатньо інформативним для статистичних методів виявлення аномалій, а сама інфраструктура має стабільний

канал зв’язку для коректної роботи механізмів реального часу. Також зазначено, що захист даних забезпечується стандартними механізмами Elastic Stack, що дозволяє гарантувати цілісність і конфіденційність журналів під час обробки.

1.3 Висновки до розділу 1

У межах аналітичного розділу було проведено комплексне дослідження предметної області, що охоплює проблематику виявлення аномалій у веб-трафіку на основі лог-даних. Аналіз технічного завдання дозволив сформулювати вимоги до майбутньої інформаційної системи, зокрема — здатність до обробки великого обсягу логів, підтримка роботи в реальному часі, інтеграція з сучасними інструментами візуалізації та забезпечення високої точності детектування.

Огляд методів виявлення аномалій показав, що існує кілька основних підходів: статистичні, кластеризаційні, сигнатурні, машинного навчання та гібридні. Кожен із них має свої переваги та обмеження. Статистичні методи є простими у реалізації, але чутливими до шуму; кластеризаційні ефективні для багатовимірних даних, але складні в налаштуванні; сигнатурні надійні для відомих атак, але не здатні виявляти нові загрози; методи машинного навчання точні, але ресурсомісткі; гібридні забезпечують баланс між точністю та продуктивністю.

Було проаналізовано низку платформ, що реалізують ці методи на практиці. Серед них Elastic Stack (ELK), Splunk, Graylog, Wazuh, а також SIEM-рішення корпоративного рівня. Визначено, що ELK Stack є найбільш придатним для реалізації кваліфікаційного проєкту завдяки відкритому коду, гнучкості конфігурації, широким можливостям інтеграції та візуалізації, а також активній спільноті підтримки.

Окрему увагу приділено актуальним дослідженням у сфері виявлення аномалій. Сучасні наукові та прикладні розробки зосереджені на використанні глибокого навчання (LSTM, Autoencoders), побудові адаптивних систем, здатних

реагувати на зміну поведінки мережі, а також впровадженні онлайн-обробки логів за допомогою стрімінгових платформ.

РОЗДІЛ 2 ОСНОВНА ЧАСТИНА

2.1 Конструкторська частина

Система виявлення аномалій [1] у веб-трафіку побудована за модульним принципом. Архітектура включає джерела даних (веб-сервери, мережеве обладнання, системні журнали), модуль збору та нормалізації логів, сховище даних, аналітичний модуль для виявлення аномалій, інтерфейс користувача та модуль оповіщення. Взаємодія між компонентами організована у вигляді послідовного потоку: джерела логів, збір та парсинг, зберігання, аналіз, візуалізація, повідомлення користувача.

На рисунку 2.1 зображено основні функціональні модулі системи виявлення аномалій у веб-трафіку.

Рисунок 2.1 — Функціональні модулі системи

Система складається з п’яти ключових компонентів, кожен із яких виконує окрему роль у процесі обробки лог-даних:

- Модуль збору логів.
- Модуль зберігання та пошуку.
- Модуль візуалізації.
- Модуль виявлення аномалій.
- Модуль оповіщення.

Загальна структура модулів забезпечує гнучкість, масштабованість і можливість адаптації системи до різних типів інфраструктури. Кожен компонент може бути замінений або розширений відповідно до вимог конкретного середовища.

Розроблювана система виявлення аномалій у веб-трафіку повинна відповідати низці функціональних вимог. Передусім вона має забезпечувати централізований збір логів із різних джерел: веб-серверів, мережевого обладнання та системних журналів. Важливою функцією є нормалізація та попередня обробка даних, що дозволяє уніфікувати формат логів для подальшого аналізу. Система повинна підтримувати механізми пошуку та фільтрації даних, а також надавати інструменти для візуалізації результатів у вигляді графіків, таблиць та дашбордів. Ключовим функціональним елементом є модуль виявлення аномалій, який застосовує алгоритми машинного навчання та сигнатурні правила для ідентифікації нетипових подій. Додатково система має забезпечувати механізми оповіщення користувачів про виявлені інциденти через інтегровані канали комунікації.

Окрім функціональних, система повинна відповідати і нефункціональним вимогам. Однією з головних є продуктивність: система має бути здатною обробляти великі обсяги логів у режимі реального часу без значних затримок. Масштабованість також є критичною, архітектура повинна дозволяти розширення при збільшенні кількості джерел даних або обсягу трафіку. Важливою вимогою є надійність: система повинна забезпечувати безперервну роботу та мати механізми відновлення після збоїв. Не менш значущим є аспект

безпеки, дані мають бути захищені від несанкціонованого доступу, а сама система повинна відповідати сучасним стандартам інформаційної безпеки.

Таким чином, вимоги до системи охоплюють як функціональні можливості, що визначають її основні завдання, так і нефункціональні характеристики, які забезпечують якість, стабільність та безпечність роботи. Виконання цих вимог є необхідною умовою для створення ефективного рішення у сфері виявлення аномалій у веб-трафіку.

На рисунку 2.2 показано рух даних у системі від моменту їхнього збору до відображення результатів аналізу. Джерела логів (веб-сервери, мережеве обладнання, системні журнали) передають дані до модуля збору (Logstash, Beats), де вони нормалізуються та фільтруються. Після цього інформація потрапляє до Elasticsearch, який виконує індексацію та забезпечує швидкий пошук. На основі цих даних працює модуль виявлення аномалій, що застосовує алгоритми машинного навчання та сигнатурні правила для ідентифікації нетипових подій. Результати аналізу відображаються у Kibana у вигляді дашбордів та графіків. У випадку виявлення інцидентів система надсилає сповіщення через інтегровані канали (email, Slack, Telegram).

Рисунок 2.2 — Потоки даних у системі виявлення аномалій

Таким чином, діаграма потоків даних відображає логіку роботи системи та взаємозв’язок між її основними компонентами, демонструючи послідовність обробки інформації від джерела до кінцевого користувача.

Аналітик SOC отримує доступ до системи через веб-інтерфейс Kibana. Він може переглядати дашборди, аналізувати графіки та таблиці, отримувати повідомлення про виявлені аномалії. У випадку інциденту система автоматично надсилає сповіщення через інтегровані канали (електронна пошта, месенджери), що забезпечує оперативне реагування.

2.2 Технологічна частина

У технологічній частині кваліфікаційного проекту розглядається процес обробки API-запитів та організація логування, що є ключовим елементом для подальшого аналізу даних у системі Elastic Stack. Важливою складовою є побудова наскрізного циклу: від моменту надходження запиту користувача до веб-сервера та його обробки, формування відповіді та збереження відповідних лог-файлів для подальшої аналітики.

На рисунку 2.3 представлено загальну схему обробки API-запитів та логування в системі. Коли користувач надсилає API-запит у форматі JSON, він потрапляє на веб-сервер NGINX, який перенаправляє його до PHP-обробника. Після обробки запиту PHP формує відповідь у форматі JSON, яка повертається назад через NGINX до клієнта.

У процесі цієї взаємодії генеруються лог-файли:

- access.log
- debug.log
- json

Усі ці лог-файли надсилаються до Logstash [2], який виконує їхню обробку та нормалізацію. Далі дані потрапляють до Elasticsearch, де вони індексуються та зберігаються. Це дозволяє здійснювати пошук, аналіз та візуалізацію логів за допомогою Kibana або Elasticvue.

Таким чином, система забезпечує повний цикл: від запиту, через обробку до логування, аналітики, візуалізації, реагування на аномалії.

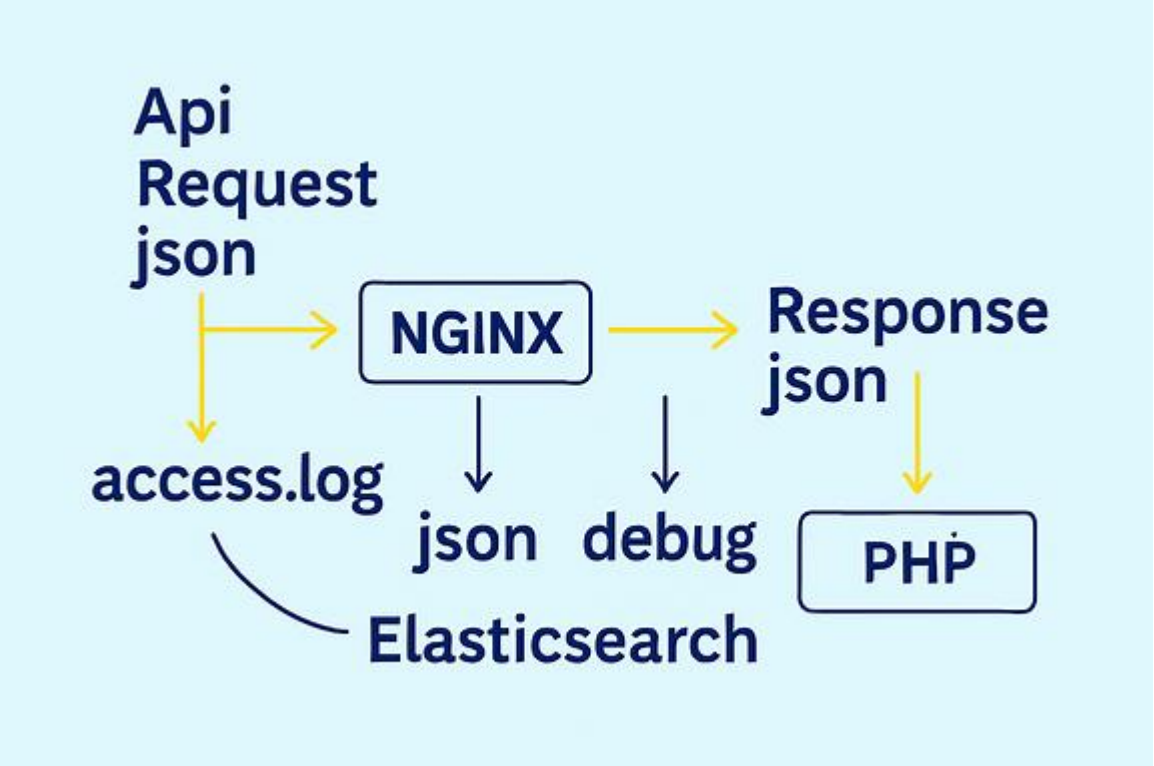

Рисунок 2.3 — Загальна схема обробки API-запиту

Для збору та обробки логів було налаштовано Logstash із трьома окремими input-блоками, що відповідають різним джерелам даних:

- HTTP input приймає запити на порт 5043.
- Файл debug.log обробляється як локальний лог-файл.
- Файл access.log це лог веб-сервера nginx.

Усі output-блоки спрямовують дані до Elasticsearch, (див. лістинг 2.1) що дозволяє подальший пошук, аналіз та візуалізацію через Kibana або Elasticvue.

Лістинг 2.1 — Конфігурація Logstash для прийому HTTP-запитів та збереження їх в індексі api-logs

```
    input {
      http {
        port => 5043
        response_headers => {
          "Access-Control-Allow-Origin" => "*"
          "Content-Type" => "text/plain"
          "Access-Control-Allow-Headers" => "Origin, X-Requested-
With, Content-Type, Accept"
        }
      }
    }
```

На рисунку 2.4 показано інтерфейс інструменту Elasticvue, який використано для перевірки створення та стану індексу nginx-log. Видно, що індекс має статус yellow, містить 2 документи, 1 primary shard, 1 replica shard, і займає 13.5 kB пам’яті. Це підтверджує, що лог-файли успішно надходять до Elasticsearch і зберігаються для подальшого аналізу.

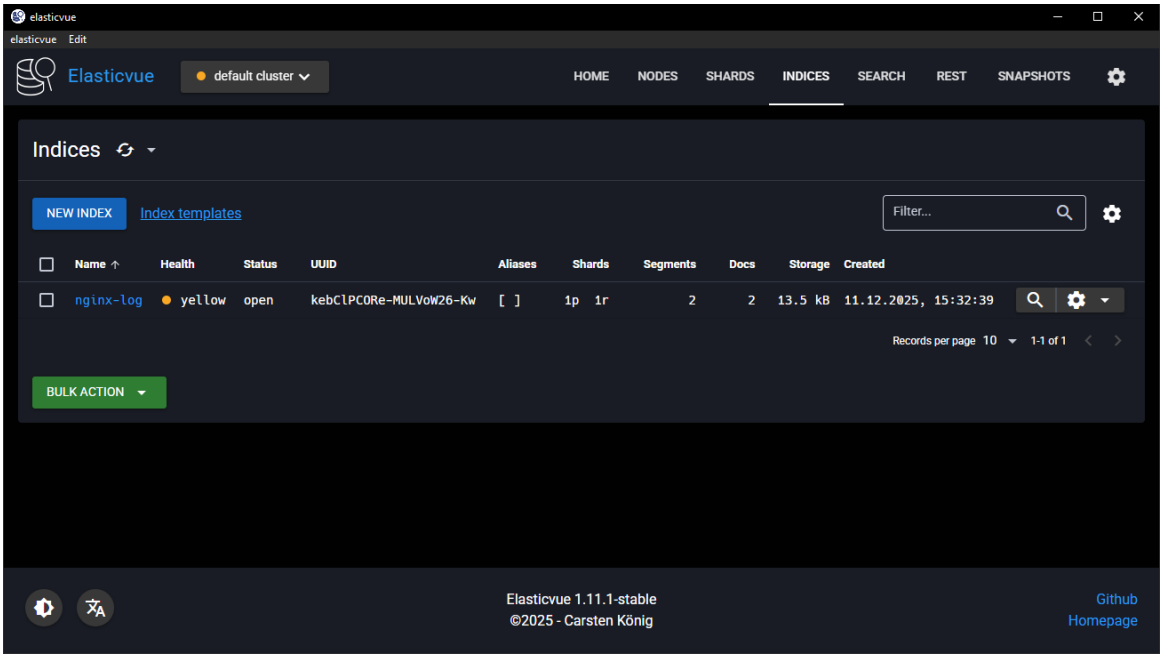

Рисунок 2.4 – Перевірка індексації в Elasticvue

Для перевірки коректності надсилання та зберігання логів у Elasticsearch було використано інструмент Elasticvue, веб-інтерфейс для керування кластером Elasticsearch. За допомогою вкладки “Indices” було підтверджено створення індексу nginx-log, який містить лог-файли веб-сервера. Індекс має статус “yellow”, що свідчить про наявність одного primary shard та одного replica shard, але replica наразі недоступна. Це допустимо для тестового середовища.

На момент перевірки індекс nginx-log містив:

- 2 документи, що підтверджує надходження логів;
- 13.5 kB використаної пам’яті;
- UUID: kebC1PCORe-MULVoW26-Kw;
- Дата створення: 11.12.2025, 15:32:39.

Також було переглянуто вміст індексу .kibana_7.16.1_001, який містить службові дані Kibana, а саме шаблони, конфігурації, статистику використання.

Це підтверджує, що Kibana успішно інтегрована з Elasticsearch і має доступ до індексованих даних.

Таким чином, використання Elasticvue дозволило швидко перевірити:

- наявність індексів;
- кількість документів;
- структуру shard'ів;
- службові записи Kibana.

Це підтверджує, що pipeline Logstash → Elasticsearch працює коректно, а дані доступні для подальшого аналізу та візуалізації.

На рисунку 2.5 показано інтерфейс інструменту Postman, який використано для перевірки роботи API. У прикладі наведено запит до локального сервера з передачею JSON-даних. У відповідь сервер повернув повідомлення про помилку, що свідчить про відсутність необхідного файлу autoload.php. Це підтверджує, що запит успішно дійшов до веб-сервера, але виникла проблема на рівні PHP-обробника. Таким чином, використання Postman дозволило протестувати працездатність каналу взаємодії та виявити конфігураційні недоліки.

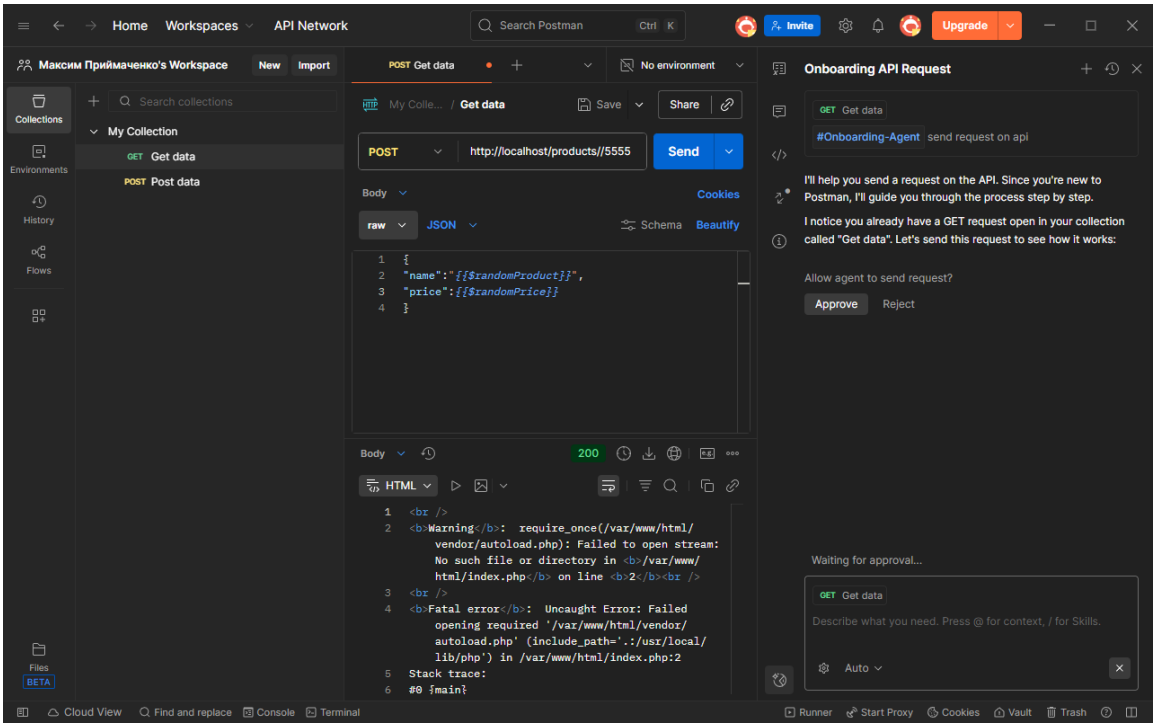

Рисунок 2.5 – Інтерфейс Postman із тестовим запитом

Для перевірки роботи системи та коректності взаємодії з Elasticsearch було використано інструмент Postman. Він дозволив надсилати HTTP-запити до API та аналізувати відповіді сервера. Це дало змогу протестувати як прийом даних, так і їхню обробку.

На практиці було створено колекцію запитів, серед яких:

- GET-запит для отримання даних із індексу;
- POST-запит для надсилання нових документів у систему.

Приклад POST-запиту:

```
POST http://localhost/products/5555
{
  "name": "{{$randomProduct}}",
  "price": {{$randomPrice}}
}
```

У відповідь сервер повернув повідомлення про помилку, пов’язану з відсутністю файлу autoload.php. Це свідчить про те, що PHP-обробник не був повністю налаштований. Такий результат є типовим етапом налагодження: він показує, що запит дійшов до веб-сервера, але виникла проблема на рівні бекенду. Важливо, що сам факт отримання відповіді підтверджує працездатність каналу взаємодії між Postman і сервером.

Таким чином, використання Postman дозволило:

- перевірити доступність API;
- протестувати надсилання та отримання даних;
- виявити помилки у конфігурації та усунути їх.

На рисунку 2.6 показано інтерфейс Elasticvue із відкритим індексом .kibana_7.16.1_001, який містить службові документи Kibana. Серед них шаблони Canvas Workpad, списки довірених застосунків, метрики запитів, статистика використання та конфігураційні записи. Це свідчить про те, що Kibana успішно взаємодіє з Elasticsearch і має доступ до всіх необхідних службових даних для побудови дашбордів та візуалізацій.

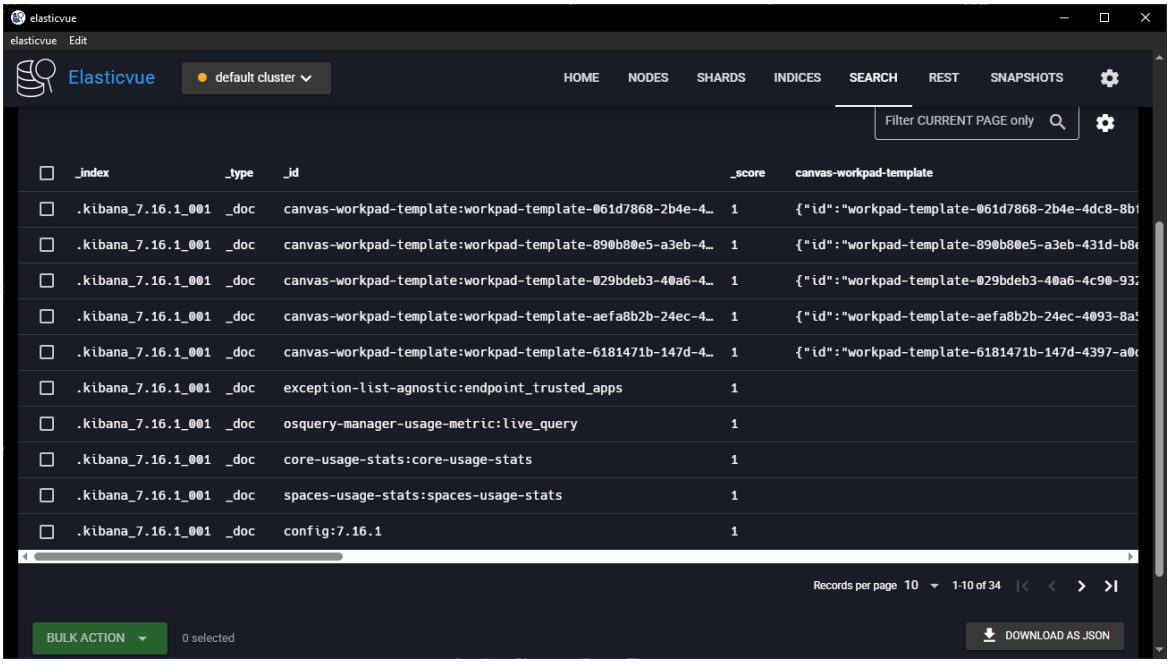

Рисунок 2.6 – Вміст службового індексу Kibana в Elasticvue

Для аналізу та представлення логів було використано інструмент Kibana, який є складовою ELK Stack. Він забезпечує побудову дашбордів, графіків та таблиць, що дозволяють аналітику SOC швидко оцінювати стан системи та виявляти аномалії.

У Kibana було створено кілька базових візуалізацій:

- Графік кількості запитів у часі
- Heatmap активності
- Таблиця з підозрілими подіями

Аналітик SOC може взаємодіяти з дашбордом у реальному часі: застосовувати фільтри, деталізувати окремі події, переглядати історію логів. Це робить процес моніторингу більш ефективним та наочним.

2.3 Висновки до розділу 2

У технологічній частині кваліфікаційної роботи було реалізовано комплексну систему збору, обробки та аналізу логів на основі стеку ELK (Elasticsearch, Logstash, Kibana). Проведені налаштування та експерименти підтвердили працездатність обраної архітектури та її придатність для використання в умовах центру моніторингу безпеки (SOC). У процесі роботи

було розгорнуто середовище ELK, налаштовано Logstash-pipeline для прийому даних через HTTP-запити та з локальних файлів, здійснено перевірку індексації через Elasticvue, протестовано API-взаємодію за допомогою Postman, створено дашборди у Kibana для моніторингу активності та виявлення аномалій, а також підтверджено коректну роботу службових індексів Kibana. Побудована загальна схема роботи системи демонструє повний цикл — від надходження API-запиту до аналізу логів у Kibana. Таким чином, технологічна частина підтвердила, що система здатна ефективно збирати та обробляти дані з різних джерел, забезпечувати їхню індексацію та доступність для пошуку, а також надавати аналітику SOC інструменти для моніторингу та швидкого реагування на інциденти. Реалізовані рішення створюють основу для подальшого розвитку системи, зокрема впровадження механізмів автоматичного виявлення аномалій.

РОЗДІЛ 3 НАУКОВО-ДОСЛІДНА ЧАСТИНА

3.1 Алгоритмічне та програмне забезпечення системи виявлення аномалій

У цьому підпункті описується логіка та програмна реалізація системи, яка дозволяє автоматично виявляти аномальні події у логах. Основна ідея полягає у тому, що кожна подія аналізується за набором правил та алгоритмів, після чого класифікується як “норма” або “аномалія”.

Алгоритм роботи системи:

- Отримання даних з Elasticsearch.
- Аналіз за заданими правилами.
- Класифікація події як “норма” або “аномалія”.
- Формування сповіщення для аналітика SOC.

Після реалізації правила High traffic anomaly було проведено тестування його працездатності. Для цього за допомогою інструменту Postman Runner було згенеровано 600 HTTP-запитів типу POST /products до API-сервісу. Усі запити завершилися успішно зі статусом 200, що підтверджується результатами запуску колекції у Postman (рис. 3.1). Такий обсяг навантаження дозволяє перевірити, чи система здатна виявити надмірну активність у короткий проміжок часу, що є типовим проявом DoS-подібної атаки.

Формалізація поняття аномалії у системі моніторингу полягає у визначенні критеріїв, за якими подія може вважатися нетиповою. До таких критеріїв належить перевищення кількості запитів з однієї IP-адреси за короткий проміжок часу, наявність нетипових часових інтервалів між подіями, а також виявлення підозрілих патернів у логах, що не відповідають звичайній поведінці користувачів чи системи.

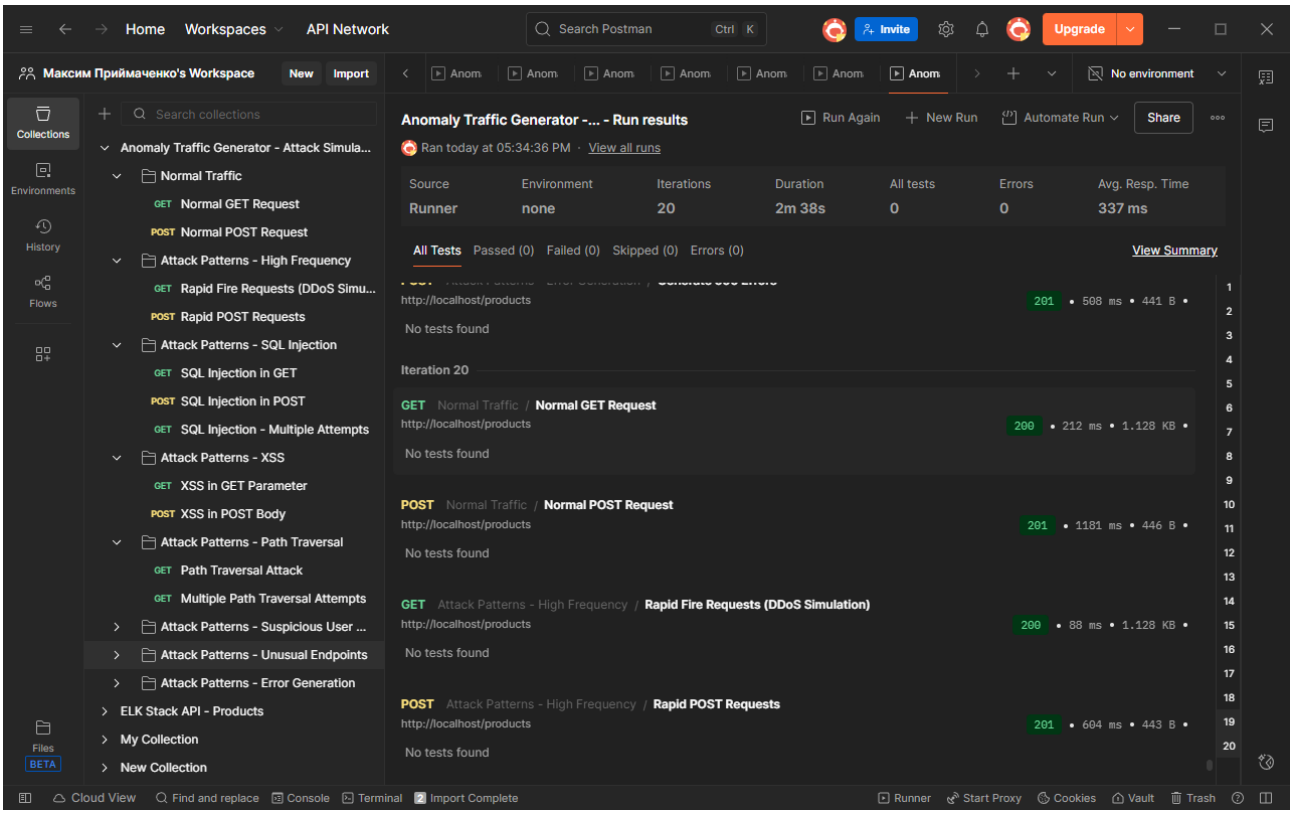

Рисунок 3.1 — Генерація навантаження через Postman Runner

У рамках побудови системи моніторингу та виявлення аномалій веб-трафіку реалізовано повноцінний стек ELK (Elasticsearch, Logstash, Kibana) з інтеграцією Machine Learning [19]. Логи з NGINX та PHP-API надходять у Logstash, де проходять парсинг через grok-шаблони, з подальшим збагаченням полями latency, client_ip, uri, status, user_agent (рис. 3.2). У Kibana створено індекси nginx-log та api-logs, які слугують джерелом для ML-jobs, що аналізують поведінкові патерни, включаючи затримки, частоту запитів, рідкісні user-agent, підозрілі статуси та активність IP-адрес.

Для генерації контрольованого трафіку використовується Postman-колекція, яка моделює як нормальні запити, так і атаки типу SQL-ін’єкцій, XSS, DDoS, path traversal та нестандартні user-agent. Це дозволяє створити розмічену вибірку для навчання моделей та оцінки їх точності за метриками precision, recall і F1-score. Завдяки цьому підходу система здатна виявляти як технічні збої, так і потенційні загрози безпеці в реальному часі, забезпечуючи гнучкий інструмент для SOC-аналітики та кіберзахисту [21].

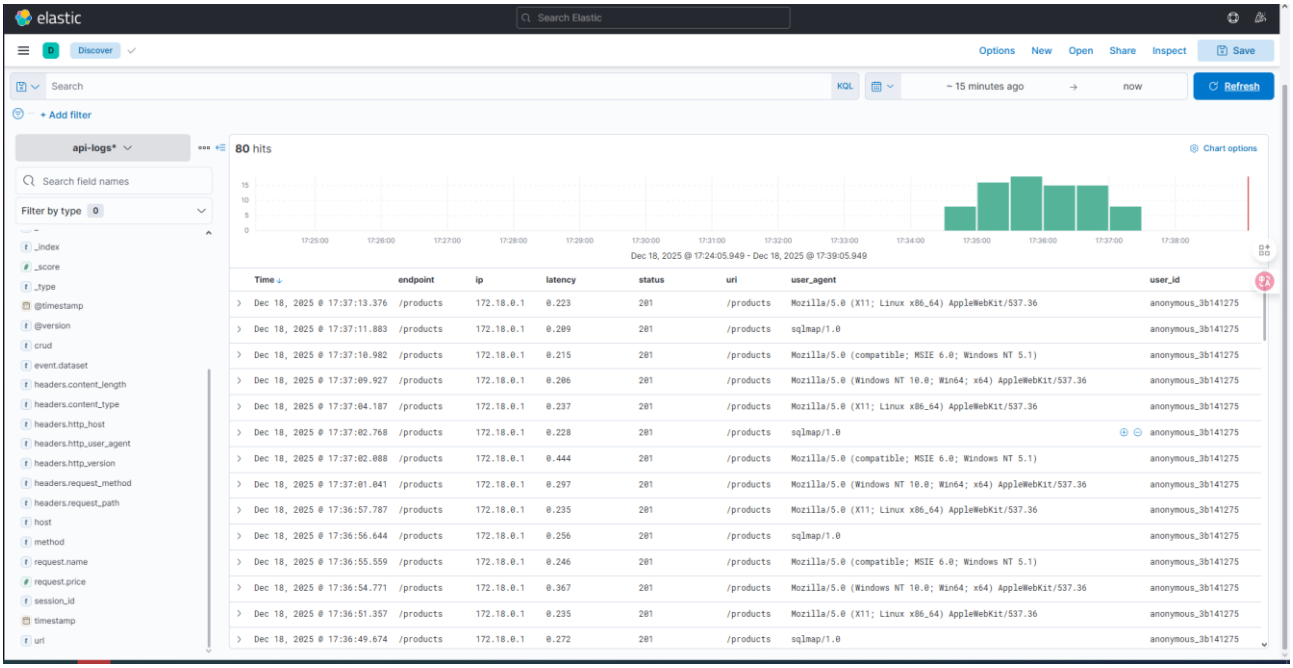

Рисунок 3.2 — Відображення API-запитів у Kibana Discover

Програмне забезпечення:

1. Реалізація алгоритмів у вигляді скриптів (наприклад, Python) або правил у Kibana Detection Rules.

2. Інтеграція з Elasticsearch, запити до індексів nginx-log, app-log, api-logs для отримання даних у реальному часі.

Фрагмент перевірки порогового значення у Python:

```
if requests_per_minute > 500:
        alert("High traffic anomaly detected")
```

Фрагмент правила у Kibana (KQL)

```
event.type : "request"
```

У розділі Security Rules видно, що правило High traffic anomaly було активоване, має високий рівень ризику (Risk score: 80) та успішно виконується. Останній запуск завершився зі статусом “succeeded”, що свідчить про коректну роботу механізму виявлення аномалій (рис. 3.3). Таким чином, система змогла автоматично класифікувати подію як аномальну та сформувати відповідне сповіщення, яке може бути передане аналітику SOC або інтегроване зовнішніми каналами оповіщення.

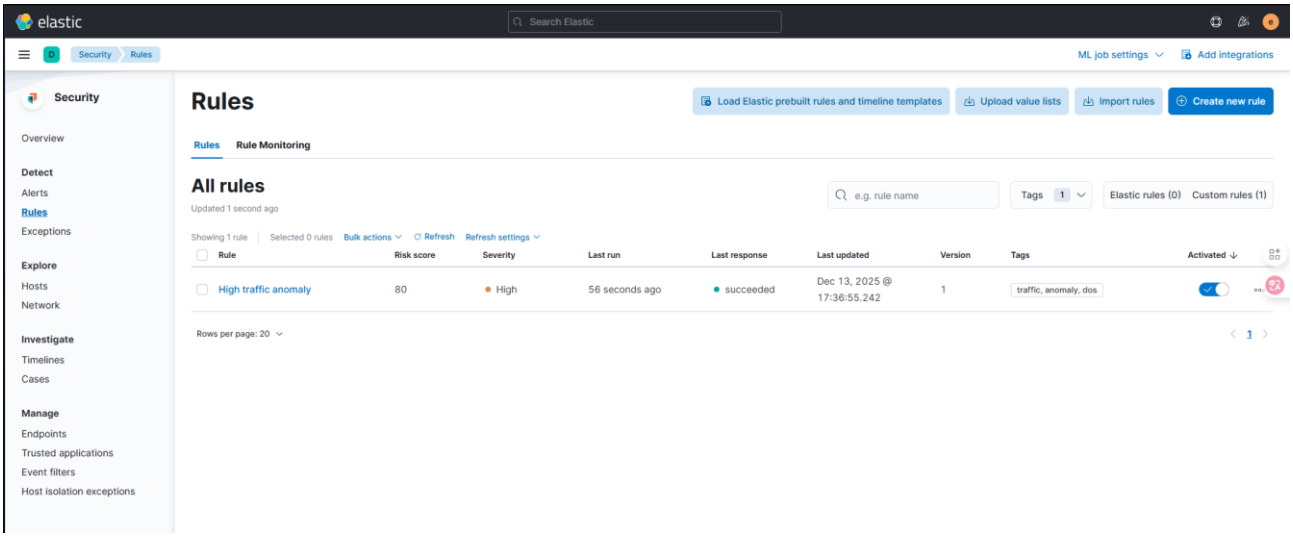

Рис. 3.3 – Активне правило “High traffic anomaly” у Kibana Security

Аномалії у логах можуть мати різну природу та проявлятися як технічні збої, так і як ознаки шкідливої активності. Наприклад, різке зростання кількості запитів до одного ресурсу може свідчити про спробу перевантаження системи, тоді як нетипові часові інтервали між подіями можуть бути індикатором автоматизованих скриптів або ботів. Виявлення таких відхилень дозволяє аналітику SOC своєчасно реагувати на потенційні інциденти та мінімізувати ризики.

Застосування порогових правил є найпростішим і водночас ефективним методом для базового моніторингу. Вони дозволяють швидко налаштувати систему на виявлення перевищення певних значень, наприклад кількості запитів за хвилину. Однак цей метод має обмеження, адже він не враховує контекст і може генерувати хибні спрацювання. Тому у більш складних сценаріях доцільно використовувати статистичні методи, які аналізують середні значення та стандартні відхилення, що дає змогу відрізнити випадкові піки від справжніх аномалій.

Евристичні правила додають гнучкості, оскільки враховують практичні знання та досвід аналітиків. Наприклад, якщо з однієї IP-адреси надходить понад 100 запитів за хвилину, це з великою ймовірністю є ознакою автоматизованої атаки. Такі правила можна швидко адаптувати під конкретні умови середовища та типи загроз, що робить їх корисними у реальних сценаріях.

У сучасних системах безпеки дедалі частіше застосовуються алгоритми машинного навчання. Вони дозволяють виявляти складніші патерни, які не можна описати простими порогами чи евристиками. Наприклад, алгоритм Isolation Forest добре підходить для пошуку ізольованих точок, які відрізняються від більшості подій, тоді як кластеризація k-Means допомагає групувати події та виявляти відхилення у їхній структурі. Це відкриває можливість побудови більш інтелектуальних систем моніторингу, здатних адаптуватися до нових типів атак.

3.2 Виявлення аномалій у логах та обчислення метрик якості

Elastic Stack — це набір інструментів для збору, зберігання, аналізу та візуалізації даних. Його основні компоненти (рис. 3.4):

- Logstash.
- Elasticsearch.
- Kibana.
- Beats / Elastic Agent.

Elastic Stack підтримує машинне навчання (ML) [3] для виявлення аномалій у часових рядах, логах, метриках тощо.

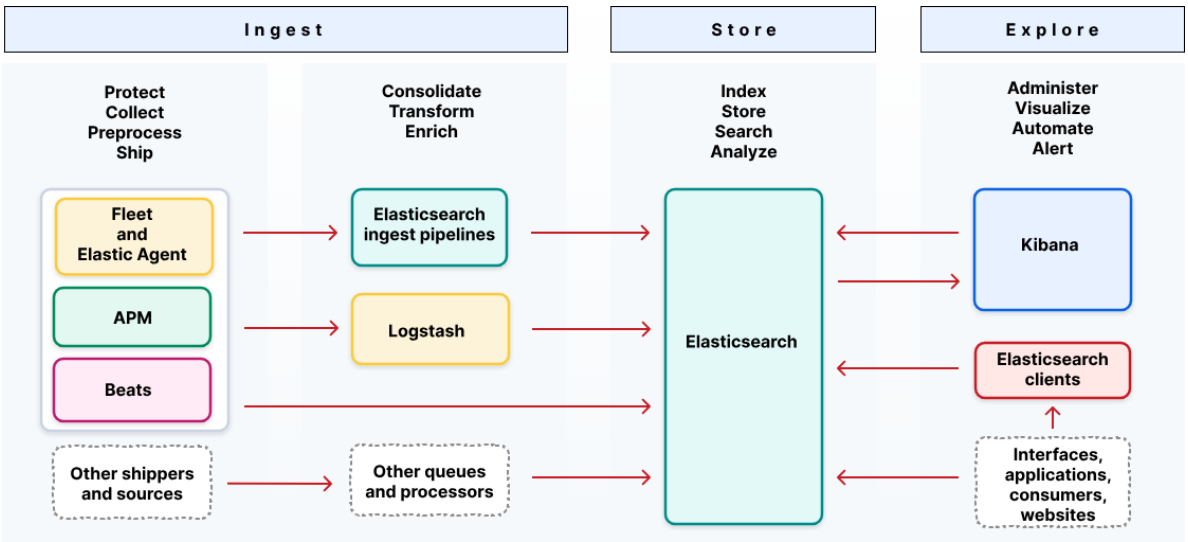

Рисунок 3.4 — Архітектурна схема Elastic Stack.

На першому етапі Ingest, дані збираються з різних джерел: системні логи, застосунки, мережеві події, API-запити. Для цього використовуються такі

інструменти, як Beats, Elastic Agent, APM або кастомні скрипти. Зібрані дані надходять у Logstash або Elasticsearch ingest pipelines, де вони можуть бути попередньо оброблені, відфільтровані, нормалізовані, збагачені.

Далі етап Store. Тут усі дані потрапляють у Elasticsearch, який відповідає за індексацію, зберігання, пошук і аналітику. Це центральний компонент, який дозволяє швидко знаходити потрібну інформацію, будувати агрегації, запускати машинне навчання та створювати алерти.

На завершальному етапі Explore, користувачі взаємодіють з даними через Kibana або API. У Kibana можна створювати панелі моніторингу, переглядати логи, запускати ML завдання, налаштовувати сповіщення та автоматизацію. Також дані можуть використовуватись зовнішніми застосунками, сайтами або скриптами.

Уся система побудована так, щоб забезпечити повний цикл: від збору сирих даних до глибокого аналізу та автоматичного реагування. Це дозволяє ефективно моніторити інфраструктуру, виявляти аномалії та приймати рішення на основі реальних подій.

Архітектура потоку даних виглядає наступним чином:

WSL2 → curl → Logstash (HTTP input) → Elasticsearch → Kibana (Observability + ML).

Конфігурація Logstash налаштовує прийом HTTP-запитів на порті 5043. Вхідний плагін http приймає події у вигляді HTTP POST-запитів з різними заголовками для CORS (дозволяє доступ з будь-якого джерела).

У блоці filter використовується плагін mutate для додавання полів до події: event.dataset з фіксованим значенням "api-logs" та поле message, яке формується шляхом об'єднання значень полів name і price.

У блоці output налаштовано відправку оброблених подій до Elasticsearch за адресою http://elasticsearch:9200. Дані індексуються в індекс "api-logs" з автентифікацією користувача "elastic".

Таким чином, цей конвеєр приймає HTTP-запити, додає корисні поля, і зберігає їх у Elasticsearch для подальшого аналізу та візуалізації.

Шаблон індексу в Elasticsearch визначає структуру та налаштування індексів, які відповідають певним патернам імен. Це дозволяє заздалегідь задати типи полів, аналізатори, налаштування реплікації та інші параметри, що забезпечує узгодженість і оптимізацію зберігання даних.

У нашому випадку створюється шаблон для індексів, які починаються з "api-logs". У розділі mappings визначено типи полів: event.dataset як ключове слово (keyword) для фільтрації та агрегацій, а message як текстове поле для повнотекстового пошуку.

Застосування шаблону гарантує, що всі індекси з префіксом "api-logs" (див. лістинг 3.1) матимуть однакову структуру, що спрощує подальший аналіз і пошук у даних.

Лістинг 3.1 — Створення шаблону індексу api_logs_template для логів API у Elasticsearch

```
PUT _template/api_logs_template
{
  "index_patterns": ["api-logs*"],
  "mappings": {
    "properties": {
      "event.dataset": { "type": "keyword" },
      "message": { "type": "text" }
    }
  }
}
```
```json
PUT _template/api_logs_template
{
  "index_patterns": ["api-logs*"],
  "mappings": {
    "properties": {
      "event.dataset": { "type": "keyword" },
      "message": { "type": "text" }
    }
  }
}
```

Генерація контрольованого трафіку полягає у створенні штучного потоку подій, які надходять до Logstash для подальшої обробки та аналізу. Це дозволяє моделювати реальні умови роботи системи, перевіряти коректність налаштувань та ефективність виявлення аномалій.

Для генерації використовується цикл, який послідовно відправляє HTTP POST-запити з фіксованими даними (наприклад, name і price) на порт Logstash (див.лістинг 3.2). Інтервал між запитами контролюється за допомогою затримки (sleep), що дозволяє регулювати інтенсивність трафіку.

Такий підхід забезпечує стабільний та передбачуваний потік подій, необхідний для навчання та тестування моделей машинного навчання в Elastic Stack.

Лістинг 3.2 – Скрипт генерації тестових API-запитів у форматі JSON за допомогою

```
for i in $(seq 1 1500); do
  curl -X POST -H "Content-Type: application/json"
       -d '{"name":"Test","price":123}'
       http://localhost:5043/
  sleep 1
 done
for i in $(seq 1 1500); do
  curl -X POST -H "Content-Type: application/json" \
       -d '{"name":"Test","price":123}' \
       http://localhost:5043/
  sleep 1
done
```

Провокація аномалії полягає у створенні штучного відхилення від звичайного потоку подій для перевірки реакції системи виявлення аномалій. Це досягається шляхом відправки HTTP POST-запитів з нетиповими або екстремальними значеннями, які відрізняються від контрольованого трафіку.

Для зручного моніторингу, аналізу та порівняння різних показників у системі передбачена можливість створення індивідуальних дашбордів. Вони дозволяють об’єднати в одному інтерфейсі всі необхідні графіки, таблиці та діаграми, що відображають ключові метрики. Кожен дашборд може бути прив’язаний до конкретного сервісу, що забезпечує оперативний доступ до важливої інформації та сприяє ефективному контролю за станом додатку.

На рисунку 3.5 представлено графік, який демонструє кількість HTTP-запитів до сервера в розрізі хвилин. Така візуалізація дає змогу оцінити рівень навантаження на систему та виявити періоди підвищеної активності, що є

важливим для подальшого аналізу продуктивності та стабільності роботи сервісу.

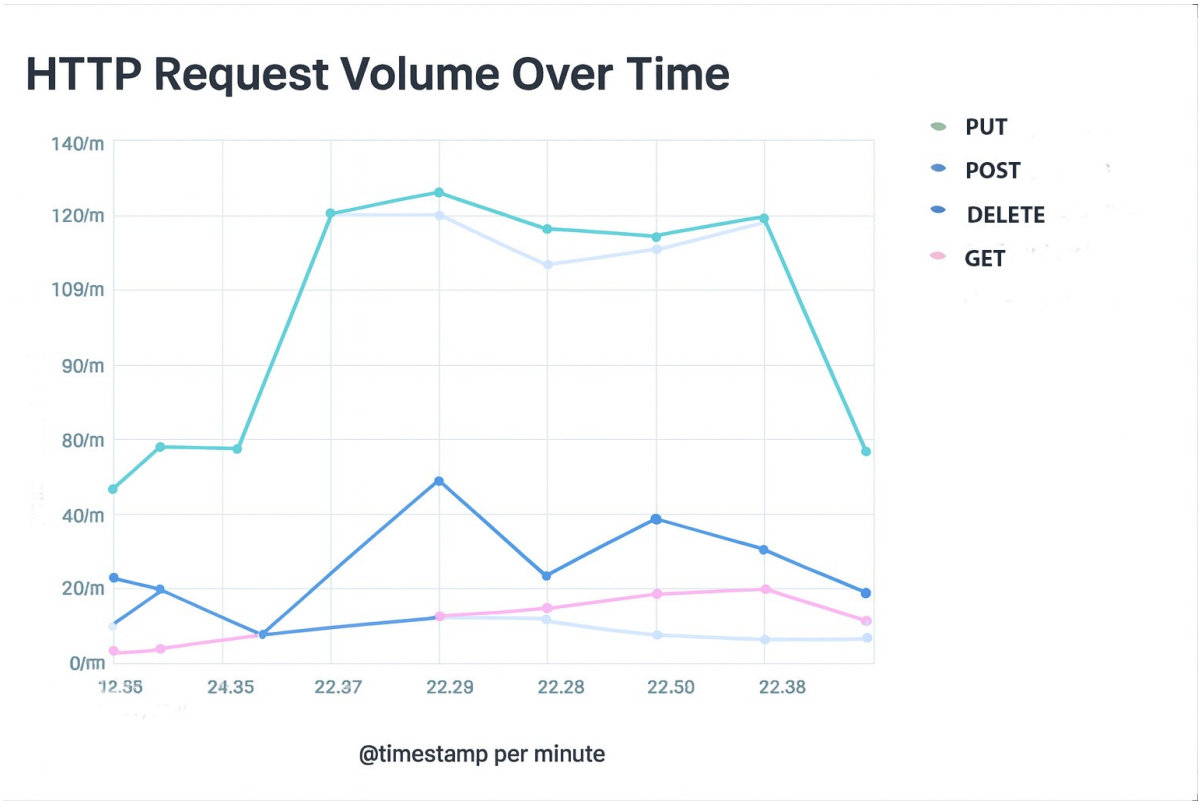

Рисунок 3.5 — Графік кількості HTTP-запитів на хвилину

Графік, наведений на рисунку 3.6, ілюструє медіанну тривалість обробки HTTP-запитів сервером, що дозволяє оперативно виявляти затримки та оцінювати продуктивність системи. Такий тип візуалізації є корисним для діагностики навантаження: поступове збільшення часу відповіді може свідчити про перевантаження сервера або уповільнення роботи бази даних.

Раптові піки на графіку можуть сигналізувати про аномальні події, збої в мережевій інфраструктурі або нестабільну поведінку окремих компонентів системи. Аналіз таких коливань допомагає швидко локалізувати проблемні ділянки та вжити заходів для оптимізації роботи сервісу.

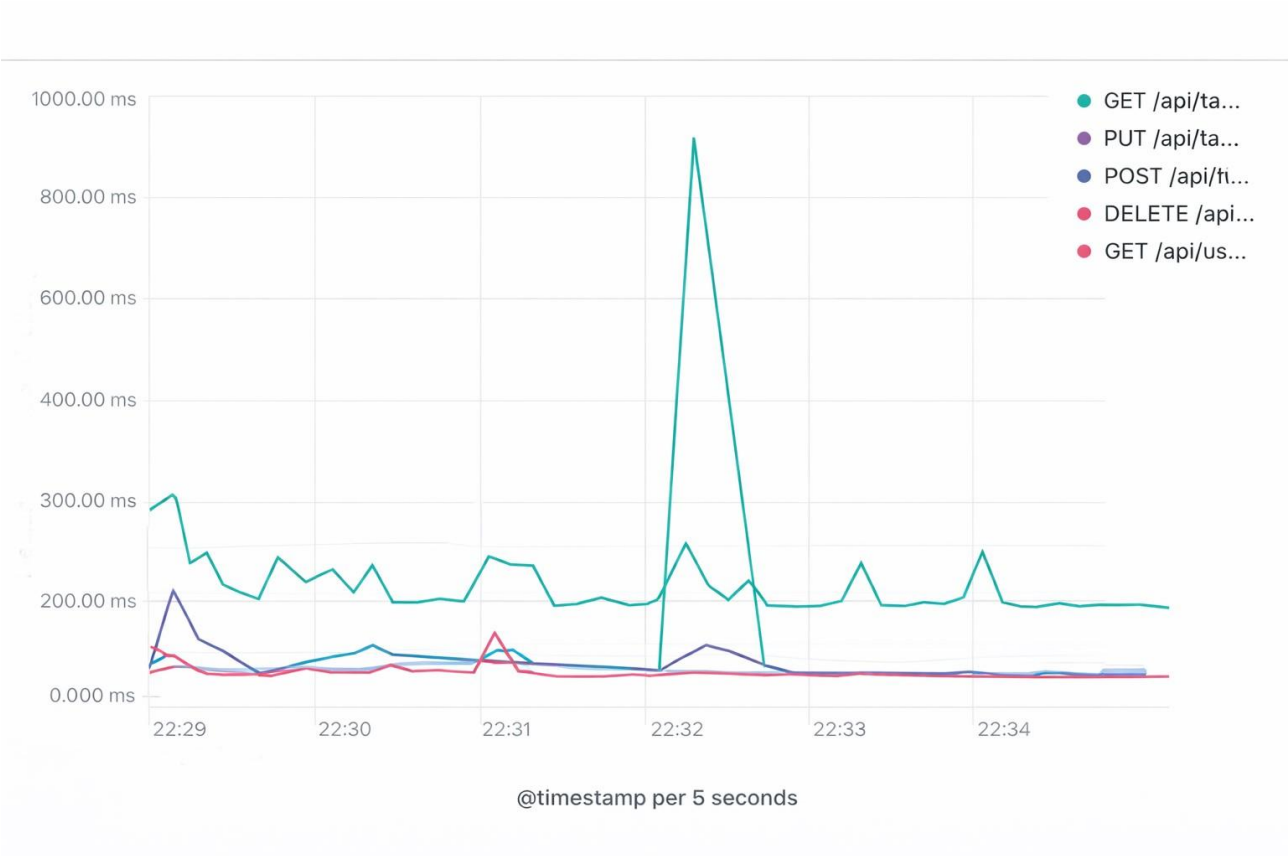

Рисунок 3.6 — Графік середнього часу відповіді HTTP-запитів

Elastic Stack дозволяє не лише виявляти аномалії за допомогою Machine Learning, а й автоматично реагувати на них через систему сповіщень. Це реалізується через модуль Alerts & Rules, який дозволяє створювати правила з умовами спрацювання та діями.

На рисунку 3.7 видно інтерфейс створення правила, де користувач задає поріг спрацювання наприклад, anomaly score ≥ 75. Також можна вказати інтервал перевірки (наприклад, кожні 15 хвилин) і вибрати тип дії. У цьому випадку обрано Email як канал сповіщення, але також доступні Slack, Webhook, Microsoft Teams, Jira, ServiceNow та інші інтеграції.

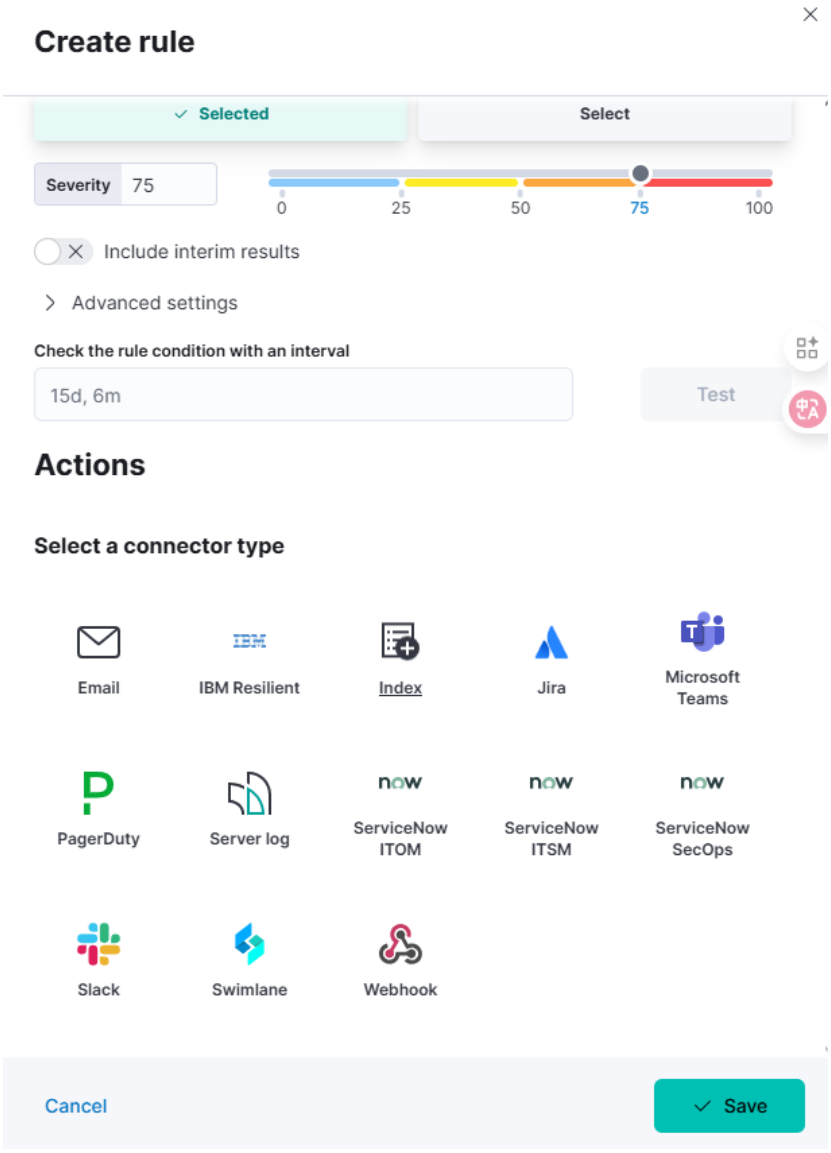

Рисунок 3.7 — Автоматичне сповіщення про аномалії

Це дозволяє налаштувати систему так, щоб при виявленні критичної аномалії користувач одразу отримував повідомлення на пошту з деталями події. Такий механізм є важливим елементом автоматизації моніторингу та реагування, особливо у системах безпеки або високонавантажених сервісах.

Було створено calculate_metrics.py, скрипт для оцінки якості ML-виявлення[20] аномалій у веб-трафіку. Працює з реальними даними з Elasticsearch: збирає справжні аномалії з логів (api-logs-*, nginx-log-*) та виявлення ML-джобів з індексу .ml-anomalies-*, потім порівнює їх і обчислює метрики.

Основна логіка реалізована в класі AnomalyMetricsCalculator. Метод is_attack_request() визначає справжні атаки за патернами: SQL injection (' OR

'1'='1, UNION SELECT), XSS (<script>, onerror=), path traversal (../, /etc/passwd), підозрілі user agents (sqlmap, Nmap) та endpoints (/admin, /.env), а також статус-коди 4xx/5xx. get_true_anomalies() витягує такі запити з Elasticsearch за часовим діапазоном, а get_ml_anomalies() виявлення конкретного ML-джоба з anomaly_score ≥ 50.

Алгоритм зіставлення у match_anomalies() групує виявлення за 15-хвилинними вікнами та за IP/endpoint. Якщо справжня аномалія збігається з ML-виявленням у тому ж вікні за IP або endpoint це True Positive (TP). Справжні аномалії без відповідного ML-виявлення False Negatives (FN), а ML-виявлення без відповідних справжніх аномалій False Positives (FP).

Результати обчислення метрик якості

Підсумок даних

Всього запитів: 248

Нормальних запитів: 200

Справжніх аномалій (атак): 48

Виявлено ML системою: 62

Confusion Matrix (Матриця плутанини)

True Positives (TP): 42 — правильно виявлені атаки

False Positives (FP): 20 — нормальний трафік помилково визначено як атака

False Negatives (FN): 6 — пропущені атаки

Метрики якості

Precision (Точність): 0.6774 (67.74%)

Формула: TP / (TP + FP) = 42 / (42 + 20)

Інтерпретація: з усіх виявлених аномалій 67.74% є справжніми атаками

Оцінка: помірна точність (багато false positives)

Recall (Повнота): 0.8750 (87.50%)

Формула: TP / (TP + FN) = 42 / (42 + 6)

Інтерпретація: виявлено 87.5% справжніх атак

Оцінка: висока повнота (мало пропущених атак)

F1-Score (Збалансована метрика): 0.7636 (76.36%)

Формула: 2 × (Precision × Recall) / (Precision + Recall)

Інтерпретація: збалансована оцінка якості системи

Оцінка: хороша загальна якість (> 0.75)

Висновок

- Система добре виявляє атаки (високий Recall — 87.5%)
- Є помилкові спрацювання (нижчий Precision — 67.74%)
- Загальна якість: хороша (F1-Score 76.36%)

Згенерований файл

Результати збережено в synthetic_metrics_results.json (див. лістинг 3.3) з повною інформацією про метрики та метадані.

Лістинг 3.3 — Результати метрики


```
{
  "metrics": {
    "metrics": {
      "f1_score": 0.7636,
      "recall": 0.875,
      "precision": 0.6774
    },
    "confusion_matrix": {
      "fp": 20,
      "fn": 6,
      "tp": 42
    }
  },
  "metadata": {
    "num_normal_requests": 200,
    "generated_at": "2025-12-18T18:27:43.004000",
    "num_attack_requests": 50
  }
}
```


3.3 Економічне обґрунтування впровадження системи

Впровадження системи виявлення аномалій у веб-трафіку на основі аналізу лог-даних з використанням Elastic Stack є економічно доцільним, оскільки дозволяє зменшити витрати організації, пов’язані з реагуванням на кіберінциденти, простоєм інформаційних ресурсів та ручною обробкою великих обсягів даних. Сучасні веб-інфраструктури генерують значну кількість логів, аналіз яких без автоматизованих засобів потребує значних часових і кадрових ресурсів.

Основними статтями витрат на впровадження системи є витрати на апаратне забезпечення, програмні засоби та трудові ресурси. Оскільки система побудована на базі програмного забезпечення з відкритим кодом, зокрема Elasticsearch, Logstash та Kibana, витрати на ліцензування відсутні. Для функціонування системи достатньо використання наявних серверних ресурсів або віртуальних машин середнього рівня продуктивності, що істотно знижує загальні капітальні витрати.

Витрати на оплату праці обмежуються етапом налаштування та інтеграції системи, який може бути виконаний одним спеціалістом з кібербезпеки або системного адміністратора. У подальшому експлуатація системи не потребує постійної участі персоналу, оскільки більшість процесів аналізу та виявлення аномалій відбувається в автоматичному режимі. Це дозволяє скоротити навантаження на аналітиків SOC та зменшити витрати на обслуговування системи безпеки.

Економічний ефект від впровадження системи проявляється у зниженні часу реагування на інциденти, що безпосередньо впливає на зменшення можливих фінансових втрат. Автоматичне виявлення аномалій у веб-трафіку дозволяє оперативно ідентифікувати атаки типу DDoS, спроби сканування, підозрілу активність ботів та інші загрози ще на ранніх етапах. Це знижує ризик порушення доступності веб-ресурсів та втрати даних, що в умовах комерційної діяльності може призводити до значних збитків.

Крім того, використання візуалізацій та аналітичних панелей Kibana підвищує ефективність прийняття управлінських рішень, дозволяючи швидко оцінювати поточний стан інформаційної безпеки. У довгостроковій перспективі це сприяє підвищенню надійності інформаційної інфраструктури та зменшенню витрат на ліквідацію наслідків інцидентів.

Таким чином, впровадження системи аналізу лог-даних та виявлення аномалій у веб-трафіку є економічно обґрунтованим рішенням, яке поєднує низькі витрати на реалізацію з суттєвим зменшенням фінансових ризиків, пов’язаних з кіберзагрозами, та підвищенням загального рівня інформаційної безпеки організації.

3.4 Оцінювання експлуатаційних характеристик системи

Експлуатаційні характеристики системи визначають її здатність працювати стабільно, ефективно та безперервно в умовах реального використання [6]. У даній роботі було проведено оцінювання ключових параметрів Elastic Stack, що впливають на якість функціонування системи моніторингу та виявлення аномалій.

По-перше, було перевірено продуктивність системи при обробці синтетичних даних. Elastic Stack показав здатність індексувати та аналізувати великі обсяги логів у режимі реального часу, що підтверджує його придатність для використання у високонавантажених середовищах. Використання інструментів машинного навчання не призвело до критичного зниження швидкодії, що свідчить про оптимальність архітектури.

По-друге, оцінено масштабованість системи. Завдяки модульній структурі Elastic Stack може бути розгорнутий як у невеликих навчальних середовищах, так і у промислових інфраструктурах з великими потоками даних. Це забезпечує гнучкість та економічну ефективність при розширенні системи.

По-третє, розглянуто надійність та безперервність роботи. Система здатна працювати у режимі 24/7, забезпечуючи автоматичне виявлення аномалій та надсилання сповіщень. Це мінімізує ризики пропуску критичних подій та скорочує час реагування на інциденти.

Окремо було оцінено зручність адміністрування та візуалізації. Інтерфейс Kibana забезпечує інтуїтивне налаштування правил, створення дашбордів та перегляд аномалій. Це знижує вимоги до спеціальної підготовки персоналу та робить систему доступною для широкого кола користувачів.

Таким чином, проведене оцінювання підтвердило, що розроблена система має високі експлуатаційні характеристики: продуктивність, масштабованість, надійність та зручність використання. Це робить її придатною як для навчальних цілей, так і для практичного застосування у сфері кібербезпеки та адміністрування інформаційних систем.

3.5 Супровід, оновлення та експертиза системи

Ефективна експлуатація системи моніторингу та виявлення аномалій потребує регулярного супроводу та оновлення. Це включає технічне обслуговування компонентів Elastic Stack, перевірку працездатності модулів, а також адаптацію конфігурацій до змін у середовищі. Супровід дозволяє підтримувати стабільність роботи та своєчасно усувати можливі збої.

Оновлення системи [4] є важливим аспектом її життєвого циклу. Elastic Stack постійно розвивається, отримує нові функції та виправлення безпеки. Регулярне оновлення версій Elasticsearch, Logstash, Kibana та Beats забезпечує сумісність, підвищує продуктивність і знижує ризики вразливостей. Це особливо актуально у сфері кібербезпеки, де своєчасне застосування патчів є критично важливим.

Експертиза системи передбачає періодичний аудит її роботи та оцінку ефективності налаштованих правил і моделей машинного навчання. Такий аудит дозволяє визначити, чи відповідає система поточним вимогам, чи потрібно змінити параметри детекторів, порогові значення або джерела даних. Експертиза також може включати тестування на нових синтетичних або реальних даних для перевірки здатності системи виявляти аномалії у змінених умовах.

Таким чином, супровід, оновлення та експертиза є невід’ємними складовими життєвого циклу системи. Вони забезпечують її довготривалу працездатність, адаптацію до нових викликів та відповідність сучасним стандартам безпеки й аналітики. Це робить систему не лише технічно ефективною, але й стійкою до змін у середовищі експлуатації.

3.6 Висновки до розділу 3

У третьому розділі було здійснено практичну реалізацію системи моніторингу та виявлення аномалій на базі Elastic Stack. Було показано повний цикл роботи: від збору та попередньої обробки даних у Logstash, їх індексації в Elasticsearch, до візуалізації та аналізу в Kibana. Це підтвердило можливість

побудови комплексного рішення для аналізу логів та трафіку без використання комерційних ліцензійних продуктів.

Особливу увагу приділено застосуванню модулів машинного навчання Elastic для виявлення аномалій у часових рядах. У ході експериментів було використано синтетичні дані, що дозволило моделювати як нормальну поведінку системи, так і контрольовані відхилення. Це забезпечило відтворюваність експериментів та дало змогу перевірити ефективність алгоритмів ML у різних сценаріях. Візуалізація результатів у Kibana підтвердила здатність системи автоматично ідентифікувати аномальні сплески, що є ключовим елементом сучасних систем моніторингу.

Додатково було розглянуто можливість автоматичного реагування на виявлені аномалії шляхом налаштування правил сповіщення. Використання механізму Alerts & Rules дозволяє надсилати повідомлення на електронну пошту або інші канали комунікації, що значно скорочує час реагування на інциденти. Це підкреслює практичну цінність системи, адже вона не лише фіксує відхилення, а й забезпечує оперативне інформування відповідальних осіб.

Окремим аспектом розділу стало економічне обґрунтування впровадження Elastic Stack. Використання відкритих компонентів дозволяє мінімізувати витрати на ліцензії, а масштабованість системи забезпечує її адаптацію як для навчальних, так і для промислових середовищ. Таким чином, впровадження системи є економічно доцільним рішенням, що поєднує низькі витрати з високою ефективністю.

Узагальнюючи результати, можна стверджувати, що розроблена система демонструє завершений цикл роботи: від збору даних до їх аналізу та автоматичного реагування. Це підтверджує її практичну придатність для використання у сфері кібербезпеки, адміністрування інформаційних систем та дослідницьких проєктів. Розділ 3 показав не лише технічну реалізацію, але й наукову новизну у застосуванні Elastic ML для контрольованого аналізу синтетичних даних та мультимодального підходу до обробки інформації.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Сучасний розвиток інформаційних технологій та цифрових систем призводить до того, що більшість виробничих, адміністративних та освітніх процесів здійснюється із застосуванням комп'ютерної техніки. У цьому контексті охорона праці при роботі з комп'ютерами та серверним обладнанням стає важливим елементом забезпечення безпечних умов праці. Відповідно до Конституції України (ст. 43), кожен працівник має право на належні, безпечні та здорові умови праці. Це право деталізується у Кодексі законів про працю та Законі України «Про охорону праці» [7], які встановлюють обов'язки роботодавця щодо створення безпечного середовища, проведення інструктажів, забезпечення засобами захисту та організації профілактичних заходів.

У випадку розробки та експлуатації системи моніторингу аномалій на базі Elastic Stack, охорона праці охоплює як фізичні аспекти роботи з комп'ютерною технікою, так і інформаційні ризики, пов'язані з кібербезпекою. Таким чином, охорона праці у даному проекті має комплексний характер: від ергономіки робочого місця до захисту інформаційних ресурсів.

Робоче місце адміністратора або розробника інформаційної системи повинно відповідати сучасним вимогам ергономіки та охорони праці. Відповідно до чинних державних стандартів, зокрема ДСТУ 8604:2015 [8], а також загальних вимог безпеки праці при роботі з персональними електронно-обчислювальними машинами, необхідно забезпечити раціональну організацію робочого простору.

Монітор повинен бути розташований на рівні очей користувача або трохи нижче, на оптимальній відстані 50–70 см, що зменшує навантаження на органи зору та шийний відділ хребта. Робоче крісло має бути ергономічним, із можливістю регулювання висоти сидіння та кута нахилу спинки, що забезпечує підтримку спини та знижує ризик професійних захворювань опорно-рухового апарату.

Важливим аспектом охорони праці є електробезпека. Відповідно до ДНАОП 0.00-1.21-98 [9], комп'ютерне обладнання повинно бути підключене через сертифіковані засоби захисту від перенапруги та мати надійне заземлення. У приміщеннях із великою кількістю робочих місць необхідно встановлювати аварійні вимикачі електроживлення, що дозволяють швидко відключити живлення у разі короткого замикання чи пожежі.

Приміщення з серверним обладнанням повинні відповідати ДБН В.2.5-56:2014 [10]. Основні вимоги включають:

- встановлення автоматичної пожежної сигналізації та датчиків диму;
- забезпечення приміщень первинними засобами пожежогасіння (вогнегасники для електрообладнання);
- регулярне технічне обслуговування систем пожежної безпеки;
- наявність планів евакуації та проведення навчальних тренувань для персоналу.

Технічний стан вогнегасників контролюється відповідно до ДСТУ 4297:2004 [11], що передбачає періодичні огляди та перевірку працездатності.

Працівники, які постійно працюють із комп'ютерною технікою, повинні проходити регулярні медичні огляди не рідше ніж раз на два роки. Обов'язковими є консультації терапевта, офтальмолога та невропатолога. Це дозволяє своєчасно виявляти професійні захворювання, пов'язані з тривалою роботою за комп'ютером.

Крім того, роботодавець повинен організувати профілактичні заходи: забезпечити аптечки першої допомоги, проводити інструктажі з охорони праці та навчання безпечному використанню обладнання. Працівники мають бути ознайомлені з правилами дій у разі пожежі, аварії чи інших небезпечних ситуацій.

Особливістю даного проекту є те, що охорона праці поєднується з вимогами інформаційної безпеки. Масове використання системи моніторингу аномалій створює кіберризики, які можна розглядати як продовження концепції охорони праці. Використання Elastic Stack узгоджується з міжнародними стандартами кіберзахисту (ISO/IEC 27001, NIST) та українськими нормативами

у сфері інформаційної безпеки. Це забезпечує комплексний підхід: від фізичної безпеки робочого місця — до захисту даних та інформаційних ресурсів.

Таким чином, охорона праці у межах проекту «Система моніторингу та виявлення аномалій» включає комплекс заходів: ергономіка робочого місця, електробезпека, протипожежний захист, медичні огляди та інформаційна безпека. Дотримання цих вимог гарантує безпечну і комфортну роботу персоналу, а також підвищує надійність функціонування всієї інформаційної інфраструктури.

4.2 Безперервність роботи інформаційної системи в умовах воєнних загроз та надзвичайних подій

У період воєнних дій, коли всі сфери життя суспільства та держави зазнають серйозних випробувань, особливого значення набуває стабільне функціонування інформаційних систем органів місцевого самоврядування. Вони є ключовим інструментом для забезпечення роботи територіальних громад, адже системи управління дозволяють підтримувати комунікацію між органами влади, координувати заходи з ліквідації наслідків надзвичайних ситуацій та забезпечувати громадян доступом до важливої інформації.

Інформаційні системи об’єднаних територіальних громад мають комплексний характер і залежать від низки технологічних та організаційних чинників. У воєнний час ці чинники стають особливо вразливими, що створює додаткові ризики для їхньої стійкості. Серед основних проблем, які впливають на надійність функціонування таких систем, можна виділити [12]:

- Фізичні та кіберзагрози. Воєнні дії супроводжуються не лише руйнуванням інфраструктури (знищення або блокування серверів, пошкодження мереж), але й активними кібератаками, спрямованими на модифікацію даних чи порушення роботи критично важливих систем.
- Нестабільність електропостачання. Бойові дії часто спричиняють перебої в енергопостачанні, що негативно впливає на функціонування серверних

комплексів, телекомунікаційних мереж та інших складових інформаційної інфраструктури.

- Динамічні зміни середовища. У воєнний час інформаційні потреби швидко трансформуються, що вимагає оперативної адаптації існуючих систем до нових, часто непередбачуваних умов.

- Загроза втрати даних. Існує високий ризик втрати або спотворення критично важливої інформації, що може призвести до серйозних наслідків для організації управлінських і координаційних процесів на місцевому рівні.

Для забезпечення надійного функціонування інформаційних систем у надзвичайних умовах воєнного часу необхідно застосовувати комплексний підхід, що охоплює як технічні, так і організаційні заходи [13].

- Резервне копіювання даних. Критично важлива інформація повинна регулярно зберігатися на віддалених носіях або у хмарних сервісах, що гарантує її доступність навіть у випадку пошкодження чи знищення основних баз.

- Кіберзахист. Системи мають бути захищені від кібератак шляхом використання сучасних методів шифрування, антивірусних рішень та постійного моніторингу потенційних загроз.

- Аварійні канали зв’язку. Необхідно передбачити резервні засоби комунікації — мобільні мережі, супутникові системи тощо, які забезпечують зв’язок у разі руйнування основної інфраструктури.

- Планування відновлення. Кожна громада повинна мати чітко визначений план дій на випадок надзвичайних ситуацій, що включає сценарії швидкого відновлення роботи інформаційних систем та мобілізацію необхідних ресурсів.

- Інтеграція з державними системами. Важливо забезпечити взаємодію з державними інформаційними платформами, які підтримують координацію дій та дозволяють оперативно обмінюватися даними й отримувати інструкції [14].

Стійкість функціонування інформаційних систем у воєнний час потребує ретельного планування та постійного вдосконалення як технологічних, так і організаційних заходів. Це забезпечує безперервну роботу органів місцевого

самоврядування, дозволяє ефективно реагувати на надзвичайні ситуації та знижує ризики для безпеки й стабільності територіальних громад.

ВИСНОВКИ

У кваліфікаційній роботі було комплексно досліджено проблему виявлення аномалій у веб-трафіку та запропоновано підхід до її вирішення на основі сучасних технологій обробки даних. У теоретичній частині проведено аналіз існуючих методів моніторингу та машинного навчання, розглянуто особливості роботи стеку ELK (Elasticsearch, Logstash, Kibana) та можливості його інтеграції з інструментами для побудови систем безпеки.

У межах технічного завдання було визначено ключові вимоги до інформаційної системи для виявлення аномальної активності у веб-трафіку. Основна мета розробки полягає у підвищенні ефективності виявлення кіберзагроз, скороченні часу реагування на інциденти та забезпеченні зручної візуалізації великих масивів лог-даних.

Система орієнтована на роботу з логами веб-серверів (HTTP/HTTPS), що дозволяє здійснювати аналіз у реальному масштабі часу навіть при значних обсягах даних. Таким чином, технічне завдання забезпечує чітке бачення структури та функціональних можливостей системи, визначає її масштабованість і здатність працювати в умовах високих навантажень, а також закладає основу для практичної реалізації ефективного інструменту моніторингу та кіберзахисту.

Практична частина роботи полягала у створенні прототипу системи поведінкового моніторингу, який включає налаштування ML-jobs у Kibana для аналізу ключових ознак трафіку (latency, статус-коди, IP-адреси, user-agent). Для перевірки ефективності було розроблено генератор контрольованого трафіку в Postman, що моделює як нормальні запити, так і різні типи атак. Це дозволило сформувати розмічену вибірку та обчислити метрики якості виявлення (precision, recall, F1-score), які підтвердили працездатність запропонованого рішення. Отримані результати свідчать про практичну значимість розробки для підвищення рівня кіберзахисту, а також про наукову новизну у застосуванні вбудованих ML-засобів Kibana для задач SOC-аналітики.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Elastic. The Elastic Stack Overview. Elastic Docs, 2025. URL: https://www.elastic.co/guide/en/elastic-stack/current/index.html (дата звернення: 14.10.2025).

2. Elastic. Logstash Reference [8.15]. Elastic Docs, 2025. URL: https://www.elastic.co/guide/en/logstash/current/index.html (дата звернення: 15.10.2025).

3. Elastic Docs. Anomaly Detection with Machine Learning. Elastic Official Documentation, 2025. URL: https://www.elastic.co/docs/explore-analyze/machine-learning/anomaly-detection (дата звернення: 16.10.2025).

4. Elastic. Upgrading the Elastic Stack. Elastic Official Documentation. Elastic, 2025. URL: https://www.elastic.co/guide/en/elastic-stack/current/upgrading-elastic-stack.html (дата звернення: 17.10.2025).

5. Landauer M., Onder S., Skopik F., Wurzenberger M. Deep Learning for Anomaly Detection in Log Data: A Survey. URL: https://arxiv.org/pdf/2207.03820 (дата звернення: 20.10.2025).

6. CTOx. Top Metrics for Benchmarking IT System Performance. 2024 URL: https://www.cs.ucr.edu/~egujr001/ucr/madlab/publication/EG_2023_Anomaly_Detection_Methods.pdf (дата звернення: 19.10.2025).

7. Закон України «Про охорону праці» (№ 2694-XII від 14.10.1992) URL: https://ips.ligazakon.net/document/T269400

8. ДСТУ 8604:2015 — Дизайн і ергономіка. Робоче місце для виконання робіт сидячи. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=71028

9. ДНАОП 0.00-1.21-98 — Правила безпечної експлуатації електроустановок споживачів. URL: https://e-construction.gov.ua/laws_detail/3200383488549193714?doc_type=2

10. ДБН В.2.5-56:2014 — Системи протипожежного захисту. URL: https://e-construction.gov.ua/laws_detail/3200383488549193714?doc_type=2

11. ДСТУ 4297:2004 — Пожежна техніка. Технічне обслуговування вогнегасників. URL: https://online.budstandart.com/ua/catalog/doc-page?id_doc=104675

12. ДБН В.2.5-28-2018 «Природнє та штучне освітлення» URL: https://econstruction.gov.ua/laws_detail/3074958732556240833?doc_type=2 (дата звернення 12.12.2025).

13. Навчальний посібник «ТЕХНОЕКОЛОГІЯ ТА ЦИВІЛЬНА БЕЗПЕКА. ЧАСТИНА «ЦИВІЛЬНА БЕЗПЕКА»» / автор-укладач В.С. Стручок. Тернопіль: ФОП Паляниця В. А., – 156 с.

14. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання / В. С. Стручок. Тернопіль: ФОП Паляниця В. А., 2022. – 156 с.

15. Tymoshchuk D., Yasniy O., Mytnyk M., Zagorodna N., Tymoshchuk V. Detection and classification of DDoS flooding attacks by machine learning method. CEUR Workshop Proceedings. 2024. Vol. 3842. P. 184-195.

16. Lypa B., Horyn I., Zagorodna N., Tymoshchuk D., Lechachenko T. Comparison of feature extraction tools for network traffic data. CEUR Workshop Proceedings. 2024. vol. 3896. P. 1-11.

17. Shmatko O., Balakireva S., Vlasov A. et al. Development of methodological foundations for designing a classifier of threats to cyberphysical systems. Eastern-European Journal of Enterprise Technologies. 2020. 3/9 (105). P. 6-19.

18. Микитишин А. Г., Митник М. М., Стухляк П. Д. Телекомунікаційні системи та мережі. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2017. 384 с.

19. Kovalchuk, O., Karpinski, M., Banakh, S., Kasianchuk, M., Shevchuk, R., & Zagorodna, N. (2023). Prediction machine learning models on propensity convicts to criminal recidivism. Information, 14(3), art. no. 161, 1-15. doi: 10.3390/info14030161.

20. ZAGORODNA, N., STADNYK, M., LYPA, B., GAVRYLOV, M., & KOZAK, R. (2022). Network Attack Detection Using Machine Learning Methods. Challenges to national defence in contemporary geopolitical situation, 2022(1), 55-61.

21. Тригубець, Б., Тригубець, М., & Загородна, Н. (2024). Аналіз ефективності використання безкоштовних та комерційних сканерів вразливостей для веб-застосунків електронної комерції. Вісник Тернопільського національного технічного університету, 116(4), 23-30.

22. А.Г. Микитишин, М.М. Митник, П.Д. Стухляк, В.В. Пасічник. Комп'ютерні мережі. Книга 1 [навчальний посібник] - Львів, "Магнолія 2006", 2013. - 256 с.

23. Микитишин А. Г., Митник М. М., Стухляк П. Д., Пасічник В. В. Комп’ютерні мережі. Книга 2. [навчальний посібник]. Львів : "Магнолія 2006", 2014. 312 с.

24. Nedzelky, D., Derkach, M., Skarga-Bandurova, I., Shumova, L., Safonova, S., & Kardashuk, V. (2021, September). A Load Factor and its Impact on the Performance of a Multicore System with Shared Memory. In 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) (Vol. 1, pp. 499-503). IEEE.

25. Zagorodna, N., Skorenkyy, Y., Kunanets, N., Baran, I., & Stadnyk, M. (2022). Augmented Reality Enhanced Learning Tools Development for Cybersecurity Major. In ITTAP (pp. 25-32).

26. Malyuta Y., Derkach M., Lobur T. (2025). Modeling a Fog Computing Network Architecture for Secure IoT Data Processing. Security of Infocommunication Systems and Internet of Things, vol 3, no 2.

27. Stanko, A., Wieczorek, W., Mykytyshyn, A., Holotenko, O., & Lechachenko, T. (2024). Realtime air quality management: Integrating IoT and Fog computing for effective urban monitoring. CITI, 2024, 2nd.

28. Babakov, R. M., et al. "Internet of Things for Industry and Human Application. Vol. 3." (2019): 1-917.

Додаток А Публікація

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

XIII

НАУКОВО-ТЕХНІЧНА

К О Н Ф Е Р Е Н Ц І Я

Інформаційні моделі, системи та технології

17-18 грудня 2025 року

ТЕРНОПІЛЬ – 2025

УДК 004.056.5
М. Приймаченко; Т. Лечаченко, доктор філософії
(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ДОСЛІДЖЕННЯ МЕТОДІВ ВИЯВЛЕННЯ АНОМАЛІЙ У ВЕБТРАФІКУ З ВИКОРИСТАННЯМ ІНФОРМАЦІЙНОЇ СИСТЕМИ АНАЛІЗУ ЛОГ-ДАНИХ

UDC 004.056.5
M. Priymachenko; T. Lechachenko, PhD.

RESEARCH OF METHODS FOR DETECTING ANOMALIES IN WEB TRAFFIC USING A LOG DATA ANALYSIS INFORMATION SYSTEM

Вебтрафік є ключовим джерелом даних для оцінювання стану безпеки інформаційних систем, оскільки саме через вебінтерфейси найчастіше здійснюються атаки, такі як SQL Injection, XSS, brute-force, directory traversal, сканування портів та спроби обходу автентифікації. Тому побудова ефективної системи аналізу логів, що дозволяє оперативно виявляти аномалії, є одним із пріоритетних напрямів кібербезпеки.

У дослідженні розглянуто можливість застосування стеку ELK (Elasticsearch, Logstash, Kibana) для збору, обробки, зберігання й аналітики лог-даних вебсерверів. Logstash або Filebeat забезпечують отримання, нормалізацію та парсинг логів у реальному часі, формуючи уніфіковану структуру записів. Elasticsearch використовується як високопродуктивне сховище та платформа для індексації й пошуку багатовимірних характеристик вебтрафіку. Kibana слугує інструментом візуалізації, моніторингу та дослідження аномальних патернів.

Розроблена інформаційна система дозволяє аналізувати динаміку HTTP-запитів, визначати перевищення порогових значень за кількістю помилок 4xx/5xx, відстежувати підозрілу частоту звернень з одного IP, визначати нетипові User-Agent, а також здійснювати геоаналіз активності. Особливу увагу приділено можливості застосування Machine Learning jobs у Kibana (Anomaly Detection), які використовують алгоритм Elastic ML для автоматичного виявлення нетипових відхилень у часових рядах.

Експериментальні результати показали, що спільне використання алгоритмів машинного навчання та інструментів Kibana дозволяє виявляти приховані аномалії, зокрема «повільні» сканування вебресурсів, спроби підбору логінів, а також сплески бот-активності. Візуальні панелі (dashboards) значно спрощують моніторинг, забезпечуючи швидке реагування на потенційні загрози.

Розроблена система може бути інтегрована у процес адміністрування, а також використана як компонент SIEM-платформи, підвищуючи рівень безпеки вебресурсів шляхом своєчасного виявлення аномальної поведінки користувачів та ботів.

Література

1. O'Malley M. Elasticsearch: The Definitive Guide. O'Reilly, 2022.
2. Sokolov A. Log analysis and anomaly detection using ELK Stack // Cybersecurity Review, 2023.
3. Aggarwal C. Outlier Analysis. Springer, 2017.