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

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: Аналіз методів виявлення аномалій у журналах подій інформаційних систем

Виконав: студент IV курсу, групи СНз-41
спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)

_____ Козій О.І.
(підпис) (прізвище та ініціали)

Керівник _____ Литвиненко Я.В.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ Шимчук Г.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент _____ Голотенко О.С.
(підпис) (прізвище та ініціали)

Тернопіль 2026

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

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

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Боднарчук І.О.
(підпис) (прізвище та ініціали)
«_16_» _червня_ 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні наукиї
(шифр і назва спеціальності)

Студенту Козій Олег Іванович
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз методів виявлення аномалій у журналах подій інформаційних систем

Керівник роботи Литвиненко Ярослав Володимирович, д.т.н., проф., професор кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*14*» травня 2026 року № 4/9-238

2. Термін подання студентом завершеної роботи 16.06.2026р.

3. Вихідні дані до роботи Журнали подій в інформаційних системах

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. РОЗДІЛ 1. Аналіз літературних джерел у напрямку виявлення аномалій у журналах подій інформаційних систем 1.1 Інформаційні системи та журнали подій 1.2 Журнали подій інформаційних систем та їх роль у моніторингу й забезпеченні безпеки 1.3 Аномалії в журналах подій: поняття, класифікація та причини виникнення 1.4 Традиційні методи виявлення аномалій у журналах подій 1.5 Сучасні методи виявлення аномалій на основі машинного навчання та штучного інтелекту 1.6 Порівняльний аналіз існуючих методів виявлення аномалій та обґрунтування вибору підходу для дослідження 1.7 Висновок до першого розділу. РОЗДІЛ 2. Розробка програми для виявлення аномалій у журналах подій інформаційних систем. 2.1 Постановка задачі виявлення аномалій та формування вимог до програмного забезпечення 2.2 Проєктування архітектури програми та структури обробки журналів подій 2.3 Розробка алгоритму виявлення аномалій у журналах подій інформаційних систем 2.4 Реалізація програмного забезпечення та опис основних модулів системи 2.5 Тестування програми та аналіз результатів виявлення аномалій 2.6 Висновок до другого розділу РОЗДІЛ 3. Безпека життєдіяльності, основи охорони праці. 3.1 Безпека життєдіяльності. Мета та завдання. 3.2 Інформаційне забезпечення БЖД. 3.3 План ліквідації аварій на виробничому об'єкті. 3.4 Висновок до третього розділу. Висновки. Перелік джерел

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

Слайди: Тема 1. Мета та задачі 2. Актуальність роботи 3. Порівняльний аналіз систем для збирання журналів подій 4. Порівняння методів машинного навчання для виявлення аномалій у журналах 5. Блок схема алгоритму роботи системи 6. Результати роботи. Файл виявлених аномалій 7. Приклад виявлення аномалій в журналі подій 8. Приклад оцінки аномальності записів за допомогою Isolation Forest 9. Висновки 10.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці	к.т.н., доц. Гурик Олег Ярославович		

7. Дата видачі завдання 26 січня 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	26.01.2026	*Виконано*
2.	Підбір джерел необхідних для розробки програми	27.01.2026-16.01.2026	*Виконано*
3.	Опрацювання джерел по темі кваліфікаційної роботи	17.01.2026-10.05.2026	*Виконано*
4.	Виконання дослідження щодо розробки програми	11.05.2026-17.05.2026	*Виконано*
	Розроблення програми		
5.	Оформлення розділу «Аналіз літературних джерел у напрямку виявлення аномалій у журналах подій інформаційних систем»	18.05.2026-24.05.2026	*Виконано*
6.	Оформлення розділу «Розробка програми для виявлення аномалій у журналах подій інформаційних систем»	25.05.2026-31.05.2026	*Виконано*
7.	Виконання завдання до підрозділу «Безпека життєдіяльності»	01.06.2026-08.06.2026	*Виконано*
8.	Виконання завдання до підрозділу «Основи охорони	01.06.2026-08.06.2026	*Виконано*
	праці»		
9.	Оформлення кваліфікаційної роботи	09.06.2026-11.06.2026	*Виконано*
10.	Нормоконтроль	12.06.2026-15.06.2026	*Виконано*
11.	Перевірка на плагіат	15.06.2026	*Виконано*
12.	Попередній захист кваліфікаційної роботи	16.06.2026	*Виконано*
13.	Захист кваліфікаційної роботи	18.06.2026	

Студент _____ Козій О.І
(підпис) (прізвище та ініціали)

Керівник роботи _____ Литвиненко Я.В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Аналіз методів виявлення аномалій у журналах подій інформаційних систем // Кваліфікаційна робота освітнього рівня «Бакалавр» // Козій Олег Іванович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СНз-41 // Тернопіль, 2026 // С. , рис. – , табл. – , кресл. – , додат. – , бібліогр. – .

Ключові слова: журнал, метод, інформаційна система, аномалія.

Кваліфікаційна робота присв'ячена аналізу методів виявлення аномалій у журналах подій та розробці програми для виявлення аномалій у журналах подій інформаційних систем. Описано основні моменти щодо журналів подій в інформаціних системах їх роль у забезпеченні безпеки, розглянуті причини винекнення аномалій, проведений аналіз методів їх виявлення.

У першому розділі роботи розглянуто поняття інформаційних систем і журналів подій, їх структуру, класифікацію та роль у моніторингу й забезпеченні інформаційної безпеки. Описано основні види аномалій, причини їх виникнення та можливі наслідки для функціонування систем. Виконано огляд традиційних методів виявлення аномалій, а також сучасних підходів на основі машинного навчання та штучного інтелекту. Проведено порівняльний аналіз існуючих методів і обґрунтовано вибір алгоритму Isolation Forest як найбільш доцільного для дослідження. У другому розділі розроблено програмне забезпечення для автоматичного виявлення аномалій у журналах подій інформаційних систем. Сформовано функціональні та нефункціональні вимоги до програми, спроєктовано її архітектуру, структуру обробки даних та алгоритм роботи. Також описано основні програмні модулі, етапи аналізу журналів подій і реалізацію алгоритму Isolation Forest для виявлення аномальної активності.

У третьому розділі кваліфікаційної роботи висвітлено питання забезпечення безпеки життєдіяльності, визначено її мету, завдання та значення в сучасних умовах. Розглянуто порядок дій під час виникнення та ліквідації аварій

на об’єктах господарської діяльності. Крім того, здійснено аналіз основних видів катастроф і аварій, а також охарактеризовано типові заходи реагування та аварійно-відновлювальні роботи, що проводяться для усунення їх наслідків і відновлення нормального функціонування об’єктів.

Об’єкт дослідження: Процес виявлення аномалій у журналах подій інформаційних систем.

Предмет дослідження: Методи аналізу подій у журналах інформаційних систем.

Практичне значення одержаних результатів полягає у стовренні програми для виявлення аномалій у журналах подій інформаційних систем.

ANNOTATION

Analysis of methods for detecting anomalies in event logs of information systems // Qualification work of the educational level "Bachelor" // Koziy Oleg Ivanovych // Ivan Pulyuy Ternopil National Technical University, Faculty of Computer and Information Systems and Software Engineering, Department of Computer Science, Group SNz-41 // Ternopil, 2026 // C. , fig. – , tab. – , drawing – , append. – , bibliography – .

Keywords: log, method, information system, anomaly.

The qualification work is devoted to the analysis of methods for detecting anomalies in event logs and the development of a program for detecting anomalies in event logs of information systems. The main points regarding event logs in information systems and their role in ensuring security are described, the reasons for the occurrence of anomalies are considered, and the methods for their detection are analyzed.

The first section of the work considers the concepts of information systems and event logs, their structure, classification and role in monitoring and ensuring information security. The main types of anomalies, the causes of their occurrence and possible consequences for the functioning of the systems are described. A review of traditional methods for detecting anomalies, as well as modern approaches based on machine learning and artificial intelligence, is carried out. A comparative analysis of existing methods is carried out and the choice of the Isolation Forest algorithm as the most appropriate for the study is justified. In the second section, software is developed for automatic detection of anomalies in event logs of information systems. Functional and non-functional requirements for the program are formed, its architecture, data processing structure and operation algorithm are designed. The main software modules, stages of event log analysis and implementation of the Isolation Forest algorithm for detecting anomalous activity are also described. The third section of the qualification work highlights the issue of ensuring the safety of life, defines its purpose, objectives

and significance in modern conditions. The procedure for actions during the occurrence and elimination of accidents at business facilities is considered. In addition, the main types of disasters and accidents are analyzed, as well as typical response measures and emergency recovery work carried out to eliminate their consequences and restore normal functioning of facilities are characterized.

Object of research: The process of detecting anomalies in event logs of information systems.

Subject of research: Methods of analyzing events in event logs of information systems.

The practical significance of the results obtained is in creating a program for detecting anomalies in event logs of information systems.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БЖД – Безпека життєдіяльності.

ІС – Інформаційна система.

Event Logs – Інформаційні системи та журнали подій.

ERP (Enterprise Resource Planning) – Система планування ресурсів підприємства.

CRM (Customer Relationship Management) – Система управління взаємовідносинами з клієнтами.

SIEM (Security Information and Event Management) – Система управління подіями та інформацією безпеки.

UEBA (User and Entity Behavior Analytics) – Аналіз поведінки користувачів та сутностей.

ELK (Elasticsearch, Logstash, Kibana) – Платформа централізованого збору та аналізу журналів подій.

IP (Internet Protocol) – Мережевий протокол передачі даних, IP-адреса вузла мережі.

ЗМІСТ

ВСТУП

Актуальність теми. Актуальність теми дослідження зумовлена стрімким розвитком інформаційних технологій та постійним зростанням кількості інформаційних систем, що функціонують у різних сферах діяльності. У процесі роботи таких систем генеруються значні обсяги журналів подій, які містять інформацію про стан системи, дії користувачів, мережеві взаємодії та можливі збої. Аналіз цих даних є важливим інструментом для забезпечення надійності, безпеки та ефективності функціонування інформаційних систем.

Одним із ключових завдань аналізу журналів подій є виявлення аномалій, які можуть свідчити про кіберінциденти, несанкціонований доступ, технічні несправності, помилки конфігурації або інші відхилення від нормального режиму роботи. Своєчасне виявлення таких аномалій дає змогу оперативно реагувати на потенційні загрози, мінімізувати ризики порушення роботи систем та запобігати втраті даних.

Сучасні методи виявлення аномалій охоплюють широкий спектр підходів, включаючи статистичний аналіз, методи машинного навчання, алгоритми кластеризації та глибокі нейронні мережі. Водночас значні обсяги даних, їх різнорідність і динамічність створюють додаткові виклики щодо точності, швидкодії та адаптивності таких методів. Тому дослідження та аналіз сучасних підходів до виявлення аномалій у журналах подій інформаційних систем є актуальним науковим і практичним завданням.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є аналіз методів виявлення аномалій у журналах подій інформаційних систем. Для досягнення поставленої мети потрібно виконати ряд завдань, зокрема:

1. Провести аналіз літературних джерел з метою виявлення існуючих методів пошуку аномалій, визначити їх переваги, недоліки та особливості застосування;

2. Розробити архітектуру програмного забезпечення для автоматизованого аналізу журналів подій, визначивши структуру модулів та етапи обробки даних;

3. Здійснити розробку програми для виявлення аномалій на основі алгоритму Isolation Forest для автоматичного пошуку нетипових подій у журналах інформаційних систем;

4. Провести тестування та оцінювання ефективності розробленої програми, забезпечивши візуалізацію та аналіз отриманих результатів виявлення аномалій.

Практичне значення одержаних результатів.

Даний програмний продукт може бути використано на практиці для задачі виявлення аномалій у журналах подій інформаційних систем.

РОЗДІЛ 1. АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ У НАПРЯМКУ ВИЯВЛЕННЯ АНОМАЛІЙ У ЖУРНАЛАХ ПОДІЙ ІНФОРМАЦІЙНИХ СИСТЕМ

1.1. Інформаційні системи та журнали подій

Інформаційні системи та журнали подій (Event Logs) - це один із ключових напрямків адміністрування, кібербезпеки, моніторингу та аналізу роботи програмних систем.

Інформаційна система (ІС) - це сукупність: апаратного забезпечення; програмного забезпечення; баз даних; мережевої інфраструктури; користувачів; процедур обробки інформації.

Основне призначення ІС - збір, зберігання, обробка, передавання та аналіз даних. До таких систем належать ERP, CRM, банківські системи, державні реєстри, веб-платформи та корпоративні мережі.

Журнал подій (Event Log) - це хронологічний запис подій, які відбуваються в інформаційній системі. Події можуть генеруватися операційною системою, прикладним програмним забезпеченням, мережевими пристроями або користувачами.

Типовий запис журналу містить:

- дату та час події;
- тип події;
- джерело події;
- рівень важливості;
- ідентифікатор події;
- IP-адресу;
- ім'я користувача;
- опис події.

Основні типи журналів подій

- Системні журнали.

Фіксують роботу операційної системи: запуск і завершення роботи; помилки драйверів; зміни конфігурації; системні збої.

- Журнали безпеки.

Містять інформацію про: входи користувачів; невдалі спроби автентифікації; зміни прав доступу; події аудиту безпеки.

- Журнали прикладних програм

Фіксують: помилки програм; виконання бізнес-процесів; роботу сервісів та API; взаємодію з базами даних.

- Мережеві журнали.

Записують: мережеві з'єднання; мережеві атаки; маршрутизацію трафіку; роботу міжмережевих екранів.

Структура запису журналу подій

Приклад:

2026-05-20 10:00:01

INFO

user1

192.168.1.15

login_success

Рівні важливості подій

У більшості систем використовуються такі рівні подані в таблиці 1.1

Таблиця 1.1 - Рівні важливості подій

Рівень	**Значення**
INFO	Інформаційна подія
DEBUG	Діагностична інформація
WARNING	Попередження
ERROR	Помилка
CRITICAL	Критична помилка

Подібна класифікація використовується і в Windows Event Log.

Значення журналів подій

Журнали подій використовуються для:

- Моніторингу систем. Дозволяють контролювати стан серверів і сервісів.
- Аудиту безпеки. Допомагають виявляти: несанкціонований доступ; - підозрілу активність; спроби злому.
- Діагностики помилок. Адміністратори можуть визначати причини: відмов сервісів; збоїв програм; перевантажень системи.
- Аналізу бізнес-процесів. Журнали використовуються у Process Mining для відновлення та аналізу реальних бізнес-процесів на основі історичних подій.

Централізоване збирання журналів

У сучасних інформаційних системах застосовуються платформи:

- Elastic Stack (ELK)
- OpenSearch
- Splunk
- Graylog
- Wazuh
- Microsoft Sentinel

Такі системи забезпечують: централізоване зберігання логів; пошук подій; візуалізацію; автоматичне виявлення інцидентів. Їх порівняльний аналіз подано у таблиці 1.2

Таблиця 1.2 - Порівняльний аналіз систем для збирання журналів подій

Система	**Основне призначення**	**Переваги**	**Недоліки**	**Рівень складності**
Elastic Stack (ELK)	Збір, зберігання та аналіз логів	Безкоштовна, гнучка, потужна візуалізація через Kibana	Складне налаштування, потребує значних ресурсів	Високий

Прод. табл. 1.2

OpenSearch	Аналіз логів та пошук даних (форк Elasticsearch)	Відкритий код, сумісність з ELK, активний розвиток	Менша екосистема порівняно з ELK	Середній
Splunk	Корпоративний аналіз журналів та SIEM	Висока продуктивність, зручний інтерфейс, розвинена аналітика	Дуже висока вартість ліцензії	Низький–середній
Graylog	Централізоване збирання та аналіз логів	Просте розгортання, зручний веб-інтерфейс	Менше можливостей аналітики, ніж ELK або Splunk	Середній
Wazuh	Моніторинг безпеки та SIEM	Безкоштовний, виявлення загроз, контроль цілісності файлів	Орієнтований переважно на кібербезпеку	Середній
Microsoft Sentinel	Хмарна SIEM/SOAR-платформа	Інтеграція з Azure та Microsoft 365, AI-аналітика	Залежність від Azure, платна модель використання	Низький

Коротке узагальнене порівняння зведено у таблиці 1.3

Таблиця 1.3 - Узагальнене порівняння

Критерій	**ELK**	**OpenSearch**	**Splunk**	**Graylog**	**Wazuh**	**Sentinel**
Відкритий код	✓	✓	✕	✓	✓	✕
SIEM-функції	Частково	Частково	✓	Частково	✓	✓
Візуалізація	✓✓✓	✓✓✓	✓✓✓	✓✓	✓✓	✓✓✓
Виявлення загроз	✓	✓	✓✓✓	✓	✓✓✓	✓✓✓
Хмарна підтримка	✓	✓	✓	✓	✓	✓✓✓
Вартість	Низька	Низька	Висока	Низька	Низька	Висока

Проведемо короткий аналіз систем:

ELK – найкращий вибір для гнучкого аналізу журналів подій та дослідницьких проєктів.

OpenSearch – сучасна безкоштовна альтернатива ELK.

Splunk – корпоративний стандарт для великих організацій.

Graylog – просте рішення для централізованого збору логів.

Wazuh — оптимальний вибір для задач кібербезпеки та виявлення інцидентів.

Microsoft Sentinel — найкраще рішення для організацій, що використовують екосистему Microsoft Azure.

Журнали подій та виявлення аномалій

Для задач кібербезпеки журнали подій є основним джерелом даних для алгоритмів виявлення аномалій.

Ознаками аномалій можуть бути: велика кількість невдалих входів; доступ у нетиповий час; зміна прав доступу; підозріла активність IP-адрес; різке зростання кількості подій; нетипова поведінка користувачів.

У сучасних дослідженнях для аналізу журналів використовуються:

- Isolation Forest;
- One-Class SVM;
- Autoencoders;
- кластеризація;
- Process Mining;
- кореляційний аналіз подій.

Для виявлення аномалій у журналах подій інформаційних систем використовуються різні підходи, які відрізняються принципом роботи, вимогами до даних та ефективністю. Одним із найпоширеніших методів є Isolation Forest, який належить до алгоритмів машинного навчання без учителя. Його принцип полягає в ізоляції аномальних об'єктів за допомогою випадкових дерев рішень. Метод характеризується високою швидкодією, добре працює з великими

обсягами даних та не потребує попереднього маркування записів, що робить його особливо придатним для аналізу журналів подій.

One-Class SVM також є методом навчання без учителя, який будує межу навколо нормальних даних і визначає об'єкти за її межами як аномальні. Цей алгоритм забезпечує високу точність для складних наборів даних, проте має значно вищу обчислювальну складність порівняно з Isolation Forest і менш ефективний при роботі з дуже великими журналами подій.

Autoencoders належать до методів глибокого навчання. Вони навчаються відновлювати нормальні дані, а значна похибка відновлення свідчить про наявність аномалії. Автоенкодери здатні виявляти складні та приховані закономірності у великих багатовимірних даних, однак потребують значних обчислювальних ресурсів, тривалого навчання та якісної підготовки даних.

Методи кластеризації (K-Means, DBSCAN та інші) групують схожі записи у кластери, а об'єкти, які не належать до жодної групи або розташовані далеко від центру кластера, розглядаються як аномальні. Перевагою кластеризації є можливість виявлення нових типів аномалій без розмічених даних, проте результати значною мірою залежать від правильного вибору параметрів алгоритму.

Process Mining відрізняється від класичних методів машинного навчання тим, що аналізує послідовності подій і відновлює реальні бізнес-процеси на основі журналів подій. Аномаліями вважаються відхилення від очікуваного сценарію виконання процесу. Метод особливо ефективний для корпоративних систем, де важливий контроль бізнес-процесів, але менш придатний для аналізу окремих технічних подій.

Кореляційний аналіз подій базується на пошуку взаємозв'язків між різними записами журналів. Окремі події можуть виглядати нормальними, однак їхня певна послідовність або комбінація може свідчити про атаку чи збій системи. Такий підхід широко використовується в SIEM-системах для виявлення складних інцидентів безпеки, але потребує наявності правил кореляції або механізмів аналізу залежностей між подіями.

Таким чином, Isolation Forest є оптимальним вибором для автоматичного виявлення аномалій у великих журналах подій завдяки високій швидкодії та відсутності потреби в розмічених даних. One-Class SVM забезпечує високу точність, але потребує більше ресурсів. Autoencoders дозволяють виявляти складні приховані аномалії, проте є найбільш ресурсомісткими. Кластеризація ефективна для пошуку нових типів відхилень, Process Mining орієнтований на аналіз бізнес-процесів, а кореляційний аналіз подій найкраще підходить для виявлення складних інцидентів інформаційної безпеки.

Журнали подій є критично важливим компонентом інформаційних систем, оскільки забезпечують контроль роботи системи, аудит безпеки, аналіз інцидентів та підтримують алгоритми виявлення аномалій. У сучасних корпоративних і державних інформаційних системах централізоване збирання та аналіз логів є основою побудови ефективних систем моніторингу та кіберзахисту.

У сучасному суспільстві інформаційні системи є невід'ємною складовою діяльності підприємств, державних установ, освітніх закладів та інших організацій. Вони забезпечують збирання, зберігання, обробку, передачу та аналіз інформації, необхідної для підтримки бізнес-процесів і прийняття управлінських рішень. З розвитком цифрових технологій та збільшенням обсягів даних зростає складність інформаційних систем, що потребує ефективних засобів контролю їхнього функціонування та забезпечення інформаційної безпеки.

Інформаційна система являє собою сукупність програмних, технічних, організаційних і людських ресурсів, призначених для автоматизації процесів обробки інформації. До складу інформаційної системи входять сервери, бази даних, мережеве обладнання, програмне забезпечення, користувачі та механізми взаємодії між ними. У процесі своєї роботи такі системи постійно генерують значну кількість службових повідомлень, які фіксують різноманітні події, що відбуваються в системі.

Для реєстрації та збереження інформації про події використовуються журнали подій (event logs). Журнал подій являє собою структурований набір записів, кожен із яких містить інформацію про певну дію або стан системи у конкретний момент часу. Такі записи можуть створюватися операційними системами, вебсерверними платформами, базами даних, мережевими пристроями, прикладними програмами та іншими компонентами інформаційної інфраструктури.

Типовий запис журналу подій містить такі атрибути:

- дата та час виникнення події (timestamp);
- рівень важливості повідомлення (INFO, WARNING, ERROR, CRITICAL тощо);
- ідентифікатор користувача або процесу;
- IP-адреса джерела події;
- тип події;
- текстовий опис події.

Приклад запису журналу подій:

2026-05-20 10:00:25, ERROR, user3, 192.168.1.22, db_timeout

У цьому записі зафіксовано помилку доступу до бази даних, яка виникла 20 травня 2026 року о 10:00:25 під час роботи користувача user3.

Журнали подій виконують низку важливих функцій в інформаційних системах. Насамперед вони використовуються для моніторингу роботи програмного забезпечення та апаратних засобів. Аналіз журналів дозволяє своєчасно виявляти помилки, несправності та відмови компонентів системи. Крім того, журнали подій є важливим джерелом інформації для аудиту безпеки, розслідування інцидентів та контролю дій користувачів.

Особливого значення журнали подій набувають у сфері кібербезпеки. Більшість сучасних атак залишають характерні сліди у журналах системи у вигляді невдалих спроб автентифікації, підозрілих мережевих з'єднань, несанкціонованого доступу до ресурсів або аномальної активності користувачів.

Аналіз журналів подій дозволяє виявляти такі загрози на ранніх етапах та оперативно реагувати на них.

Сучасні інформаційні системи можуть генерувати мільйони записів щоденно. Через значні обсяги даних ручний аналіз журналів стає практично неможливим. Це обумовлює необхідність використання автоматизованих методів обробки даних, технологій аналізу великих даних, машинного навчання та штучного інтелекту для виявлення аномалій та потенційних загроз.

Таким чином, журнали подій є одним із ключових джерел інформації про функціонування інформаційних систем. Вони забезпечують можливість моніторингу, аудиту, діагностики несправностей та виявлення порушень безпеки. Ефективний аналіз журналів подій дозволяє підвищити надійність, продуктивність і захищеність інформаційних систем, що робить даний напрямок актуальним для сучасних наукових досліджень та практичних розробок.

1.2. Журнали подій інформаційних систем та їх роль у моніторингу й забезпеченні безпеки

Журнали подій (event logs) є невід'ємною складовою сучасних інформаційних систем і відіграють важливу роль у забезпеченні їх надійного функціонування, моніторингу стану та захисту від кіберзагроз. У процесі роботи інформаційної системи постійно відбувається велика кількість подій, пов'язаних із виконанням програм, діями користувачів, мережевими з'єднаннями, змінами конфігурації та роботою апаратних компонентів. Для фіксації таких подій використовуються спеціальні журнали, які зберігають інформацію про всі важливі дії та зміни в системі.

Журнал подій являє собою впорядкований набір записів, що містять інформацію про час виникнення події, її тип, джерело, рівень важливості та додаткові параметри. Типовий запис журналу може включати дату і час події, ім'я користувача, IP-адресу, ідентифікатор процесу, код помилки та текстовий

опис події. Такі дані забезпечують можливість відстеження роботи системи та аналізу її поведінки в різних умовах експлуатації.

Однією з головних функцій журналів подій є моніторинг стану інформаційної системи. Завдяки журналам адміністратори можуть отримувати актуальну інформацію про роботу серверів, мережевого обладнання, баз даних і прикладних програм. Аналіз журналів дозволяє своєчасно виявляти збої, помилки програмного забезпечення, перевантаження ресурсів та інші проблеми, які можуть негативно вплинути на продуктивність або доступність системи. На основі даних журналів можуть формуватися автоматичні сповіщення про критичні події, що забезпечує оперативне реагування на інциденти та мінімізацію можливих наслідків.

Важливою сферою застосування журналів подій є забезпечення інформаційної безпеки. Журнали містять відомості про автентифікацію користувачів, спроби входу до системи, зміни прав доступу, використання привілейованих облікових записів, доступ до конфіденційних даних та інші дії, що мають безпекове значення. Аналізуючи такі записи, фахівці з кібербезпеки можуть виявляти несанкціоновані спроби доступу, атаки типу brute force, внутрішні порушення політик безпеки та інші потенційні загрози.

Особливого значення журнали подій набувають під час розслідування інцидентів інформаційної безпеки. Вони виступають джерелом цифрових доказів, які дозволяють відновити послідовність дій користувачів або зловмисників, визначити причини виникнення інциденту та оцінити масштаби його впливу. Наявність детальних журналів є необхідною умовою проведення якісного аудиту безпеки та забезпечення відповідності вимогам міжнародних стандартів і нормативних документів.

У сучасних інформаційних системах обсяг журналів подій може досягати мільйонів записів на добу. Тому для їх обробки використовуються спеціалізовані системи централізованого збору та аналізу логів, такі як Elastic Stack (ELK), Splunk, Graylog, Wazuh та Microsoft Sentinel. Ці платформи забезпечують

автоматичне агрегування даних з різних джерел, пошук подій, побудову аналітичних звітів та візуалізацію показників функціонування системи.

Останніми роками значної популярності набули методи машинного навчання для аналізу журналів подій. Використання алгоритмів виявлення аномалій дозволяє автоматично знаходити нетипові або підозрілі події без необхідності формування великої кількості правил. Наприклад, алгоритм Isolation Forest може використовуватися для виявлення аномальних дій користувачів, підозрілих мережевих з'єднань або нетипової активності в інформаційній системі. Такий підхід підвищує ефективність моніторингу та дозволяє швидше реагувати на потенційні загрози.

Таким чином, журнали подій є одним із найважливіших джерел інформації про функціонування інформаційних систем. Вони забезпечують контроль роботи програмних та апаратних компонентів, підтримують процеси моніторингу, аудиту та розслідування інцидентів, а також слугують основою для побудови сучасних систем виявлення аномалій і кіберзахисту. Ефективне використання журналів подій дозволяє підвищити надійність, безпеку та керованість інформаційних систем у сучасному цифровому середовищі.

1.3. Аномалії в журналах подій: поняття, класифікація та причини виникнення

У сучасних інформаційних системах журнали подій є одним із основних джерел інформації про функціонування програмного забезпечення, серверів, мережевої інфраструктури та дії користувачів. Постійний моніторинг журналів подій дозволяє контролювати стан системи, виявляти помилки та забезпечувати належний рівень інформаційної безпеки. Однак серед великої кількості записів можуть виникати події, які суттєво відрізняються від нормальної поведінки системи. Такі події називаються аномаліями.

Аномалія в журналі подій – це запис або сукупність записів, характеристики яких істотно відрізняються від типових шаблонів

функціонування інформаційної системи. Аномальні події можуть свідчити про технічні несправності, помилки конфігурації, порушення політик безпеки, спроби несанкціонованого доступу або кібератаки. Виявлення таких подій є важливим завданням систем моніторингу та кіберзахисту, оскільки дозволяє своєчасно реагувати на потенційні загрози та запобігати негативним наслідкам.

Залежно від характеру прояву аномалії поділяють на кілька основних типів. Найпростішими є точкові аномалії, які являють собою окремі записи, що значно відрізняються від решти даних. Наприклад, багаторазові невдалі спроби входу до системи з однієї IP-адреси або запуск процесу, який раніше ніколи не використовувався в системі. Такі аномалії зазвичай легко виявляються за допомогою статистичних методів або алгоритмів машинного навчання.

Іншим видом є контекстні аномалії. У цьому випадку подія може бути нормальною за своїми характеристиками, але аномальною в конкретному контексті. Наприклад, успішний вхід користувача до системи є звичайною подією, проте якщо він відбувається о третій годині ночі або з іншої країни, така активність може розглядатися як підозріла. Для виявлення контекстних аномалій необхідно враховувати додаткові параметри, такі як час, місце або роль користувача.

Окрему групу становлять колективні аномалії, які проявляються лише при аналізі послідовності подій. Кожен окремий запис може виглядати нормальним, однак їхня сукупність свідчить про нетипову поведінку системи. Прикладом є серія послідовних спроб авторизації з різних облікових записів або поступове збільшення навантаження на сервер, що може бути ознакою підготовки до кібератаки. Виявлення таких аномалій потребує аналізу часових залежностей та взаємозв'язків між подіями.

З точки зору джерела виникнення аномалії можна поділити на технічні, організаційні та безпекові. Технічні аномалії виникають унаслідок відмов обладнання, помилок програмного забезпечення, перевантаження ресурсів або некоректної конфігурації системи. До таких подій належать аварійне завершення

процесів, помилки доступу до баз даних, переповнення дискового простору чи нестабільна робота мережевих сервісів.

Організаційні аномалії пов'язані з помилками користувачів або адміністраторів. Вони можуть виникати через неправильне налаштування програмного забезпечення, порушення встановлених процедур роботи, випадкове видалення даних або використання системи не за призначенням. Хоча такі події не завжди мають ознаки кіберзагроз, вони можуть призводити до порушення функціонування інформаційної системи.

Безпекові аномалії є найбільш критичними, оскільки часто свідчать про наявність зовнішніх або внутрішніх атак. До них належать численні невдалі спроби автентифікації, несанкціоноване підвищення привілеїв, доступ до конфіденційної інформації, виконання підозрілих команд, а також аномальна мережна активність. Саме такі події найчастіше аналізуються системами SIEM (Security Information and Event Management) та засобами виявлення вторгнень.

Причини виникнення аномалій можуть бути різноманітними. Однією з найпоширеніших причин є людський фактор. Помилки користувачів під час введення даних, неправильні налаштування програм або випадкові дії часто призводять до появи нетипових записів у журналах подій. Іншою поширеною причиною є технічні несправності, пов'язані зі збоєм обладнання, програмними помилками або проблемами мережевої інфраструктури.

Значну частину аномалій спричиняють кіберінциденти та атаки. Спроби підбору паролів, розповсюдження шкідливого програмного забезпечення, атаки типу «відмова в обслуговуванні» (DoS/DDoS), експлуатація вразливостей або внутрішні загрози можуть генерувати характерні аномальні шаблони поведінки, які фіксуються в журналах подій. Вчасне виявлення таких аномалій дозволяє швидко локалізувати загрозу та мінімізувати її наслідки.

У зв'язку зі стрімким зростанням обсягів журналів подій традиційні методи аналізу стають недостатньо ефективними. Тому для виявлення аномалій дедалі частіше використовуються методи машинного навчання та інтелектуального аналізу даних. Серед найбільш поширених алгоритмів застосовуються Isolation

Forest, One-Class SVM, Local Outlier Factor, кластеризація та нейронні мережі. Такі підходи дозволяють автоматично знаходити приховані закономірності та виявляти аномалії навіть у великих масивах даних без необхідності формування великої кількості правил.

Отже, аномалії в журналах подій є важливими індикаторами відхилень у роботі інформаційних систем. Їх своєчасне виявлення та аналіз сприяють підвищенню рівня надійності, безпеки та ефективності функціонування інформаційної інфраструктури. Розуміння класифікації аномалій та причин їх виникнення є необхідною передумовою для розроблення сучасних систем моніторингу та виявлення кіберзагроз.

1.4. Традиційні методи виявлення аномалій у журналах подій

Виявлення аномалій у журналах подій є одним із ключових завдань моніторингу та забезпечення безпеки інформаційних систем. Аномалії можуть свідчити про технічні несправності, помилки конфігурації, порушення політик безпеки або кіберінциденти. До появи сучасних методів машинного навчання аналіз журналів подій здебільшого здійснювався за допомогою традиційних підходів, які базуються на правилах, статистичних методах та експертних знаннях. Незважаючи на розвиток інтелектуальних систем аналізу даних, традиційні методи й сьогодні залишаються важливим інструментом моніторингу інформаційних систем завдяки своїй простоті, зрозумілості та відносно невеликим вимогам до обчислювальних ресурсів.

Одним із найпоширеніших традиційних підходів є аналіз на основі правил (rule-based analysis). Суть цього методу полягає у створенні набору правил або шаблонів, які описують нормальну або небажану поведінку системи. Наприклад, система може автоматично генерувати попередження у випадку перевищення певної кількості невдалих спроб входу до облікового запису за визначений проміжок часу або при спробі доступу до захищених ресурсів користувачем без відповідних прав. Такі правила формуються експертами з кібербезпеки та

адміністраторами систем на основі накопиченого досвіду. Перевагою методу є висока точність для відомих сценаріїв порушень, однак він практично не здатний виявляти нові або невідомі типи аномалій.

Іншим поширеним підходом є пороговий аналіз (threshold-based detection). У цьому випадку для певних параметрів встановлюються допустимі межі значень. Якщо значення показника виходить за встановлений поріг, подія вважається аномальною. Наприклад, різке зростання кількості запитів до сервера, перевищення допустимого навантаження на процесор або незвично велика кількість мережевих з'єднань можуть сигналізувати про наявність проблеми чи атаки. Простота реалізації робить цей метод популярним у системах моніторингу, проте його ефективність значною мірою залежить від правильного вибору порогових значень.

Важливе місце серед традиційних методів займають статистичні методи аналізу даних. Вони базуються на припущенні, що нормальна поведінка системи характеризується певними статистичними закономірностями. Для кожного параметра можуть обчислюватися середні значення, дисперсія, стандартне відхилення та інші статистичні характеристики. Події, що суттєво відхиляються від типових значень, розглядаються як потенційні аномалії. Наприклад, якщо кількість подій певного типу значно перевищує середній рівень за попередній період, система може згенерувати попередження про можливу аномальну активність.

Одним із найпростіших статистичних методів є використання Z-оцінки (Z-score). Даний підхід дозволяє визначити, наскільки конкретне значення відрізняється від середнього значення вибірки. Формула Z-оцінки має вигляд:

$$Z = \frac{x-\mu}{\sigma}, \tag{1.1}$$

де x – поточне значення показника, μ – середнє значення, а σ – стандартне відхилення. Якщо абсолютне значення Z-оцінки перевищує встановлений поріг, подія може бути класифікована як аномальна.

Для аналізу часових характеристик журналів подій широко застосовуються методи аналізу часових рядів. Вони дозволяють виявляти нетипові зміни інтенсивності подій у часі. Наприклад, якщо система щодня реєструє приблизно однакову кількість запитів, а раптово спостерігається різке зростання або падіння активності, це може свідчити про збій чи атаку. До таких методів належать ковзне середнє, експоненціальне згладжування та інші статистичні моделі прогнозування.

Ще одним традиційним підходом є сигнатурний аналіз, який широко використовується в системах виявлення вторгнень. Його принцип полягає у порівнянні подій із заздалегідь відомими шаблонами атак або небажаної активності. Якщо послідовність подій відповідає одній із сигнатур, система повідомляє про можливий інцидент безпеки. Такий підхід демонструє високу ефективність щодо відомих загроз, але не дозволяє виявляти нові види атак, сигнатури яких ще не були додані до бази знань.

Для складніших сценаріїв використовуються методи кореляції подій. Вони аналізують взаємозв'язки між різними записами журналів і дозволяють виявляти підозрілі послідовності дій. Наприклад, окремі події входу до системи, зміни прав доступу та доступу до конфіденційних даних можуть виглядати нормальними, однак їхня певна послідовність може свідчити про компрометацію облікового запису. Саме тому кореляційний аналіз є важливим елементом сучасних систем SIEM.

Основними перевагами традиційних методів є простота реалізації, прозорість прийняття рішень та можливість швидкого впровадження без значних обчислювальних витрат. Адміністратори можуть легко пояснити причини спрацювання системи та налаштувати необхідні параметри відповідно до особливостей конкретної інформаційної системи. Крім того, такі методи добре працюють у середовищах зі стабільними та передбачуваними процесами.

Разом із тим традиційні підходи мають низку суттєвих недоліків. Вони потребують постійного оновлення правил і сигнатур, мають обмежену здатність адаптуватися до змін поведінки системи та часто генерують значну кількість

хибних спрацювань. Зі збільшенням обсягів журналів подій та складності сучасних інформаційних систем ефективність таких методів поступово знижується. Саме тому останніми роками вони все частіше доповнюються або замінюються алгоритмами машинного навчання, які здатні автоматично виявляти приховані закономірності та нові типи аномалій.

Отже, традиційні методи виявлення аномалій у журналах подій є фундаментом сучасних систем моніторингу та кібербезпеки. Незважаючи на певні обмеження, вони залишаються важливим інструментом аналізу подій і часто використовуються у поєднанні з сучасними інтелектуальними методами для підвищення ефективності виявлення загроз та забезпечення надійності функціонування інформаційних систем.

1.5. Сучасні методи виявлення аномалій на основі машинного навчання та штучного інтелекту

Стрімкий розвиток інформаційних технологій, збільшення обсягів даних та ускладнення архітектури інформаційних систем призвели до суттєвого зростання кількості подій, що реєструються в журналах. Традиційні методи аналізу, засновані на правилах та сигнатурах, часто виявляються недостатньо ефективними для обробки великих потоків даних і виявлення нових типів загроз. У зв'язку з цим особливої актуальності набувають методи машинного навчання та штучного інтелекту, які дозволяють автоматизувати процес аналізу журналів подій, виявляти приховані закономірності та знаходити аномалії без необхідності формування великої кількості правил.

Машинне навчання являє собою напрям штучного інтелекту, що дозволяє комп'ютерним системам самостійно навчатися на основі даних та вдосконалювати результати аналізу без прямого програмування кожного сценарію. У задачах виявлення аномалій алгоритми машинного навчання використовуються для побудови моделей нормальної поведінки системи та подальшого визначення подій, які суттєво відхиляються від цієї моделі.

Одним із найпоширеніших підходів є навчання без учителя (unsupervised learning). У більшості реальних інформаційних систем відсутня достатня кількість розмічених даних, що містять приклади аномалій. Тому алгоритми без учителя дозволяють аналізувати журнали подій без попереднього маркування записів. Такі методи самостійно виявляють структуру даних і визначають об'єкти, які суттєво відрізняються від загальної сукупності.

Серед алгоритмів навчання без учителя особливе місце займає Isolation Forest. Даний метод був розроблений спеціально для задач виявлення аномалій і базується на принципі ізоляції об'єктів. Алгоритм будує множину випадкових дерев рішень і визначає кількість кроків, необхідних для відокремлення конкретного об'єкта від інших. Аномальні записи зазвичай ізолюються швидше, ніж нормальні, тому отримують вищий показник аномальності. Основними перевагами Isolation Forest є висока швидкодія, можливість роботи з великими наборами даних та відсутність необхідності в попередньому маркуванні інформації. Саме тому цей алгоритм широко використовується для аналізу журналів подій інформаційних систем.

Ще одним популярним методом є One-Class Support Vector Machine (One-Class SVM). Даний алгоритм формує межу, яка описує область нормальної поведінки системи. Події, що знаходяться за межами побудованої області, розглядаються як аномальні. One-Class SVM демонструє хороші результати на складних наборах даних, проте потребує значних обчислювальних ресурсів і може бути менш ефективним при роботі з дуже великими журналами подій.

Для пошуку аномалій також широко використовуються кластеризаційні методи, зокрема алгоритми K-Means та DBSCAN. Принцип їх роботи полягає у групуванні схожих записів у кластери. Події, які не належать до жодного кластера або розташовані далеко від центрів кластерів, можуть вважатися аномальними. Кластеризація дозволяє ефективно виявляти приховані структури даних і знаходити нові типи відхилень, які раніше не спостерігалися в системі.

Важливу роль у сучасних системах аналізу журналів подій відіграють методи глибокого навчання (Deep Learning). Вони базуються на використанні

багатошарових нейронних мереж, здатних автоматично виділяти складні закономірності в даних. Одним із найпоширеніших підходів є використання автоенкодерів (Autoencoders). Автоенкодер являє собою нейронну мережу, яка навчається стискати та відновлювати вхідні дані. Якщо під час відновлення виникає значна похибка, це може свідчити про наявність аномального запису. Автоенкодери особливо ефективні для роботи з великими та багатовимірними наборами даних.

Для аналізу послідовностей подій у журналах активно застосовуються рекурентні нейронні мережі (RNN) та їх удосконалені варіанти, такі як Long Short-Term Memory (LSTM). Дані моделі враховують часові залежності між подіями та дозволяють виявляти аномалії в поведінці системи на основі аналізу послідовностей дій користувачів або роботи програмних компонентів. Наприклад, LSTM можуть прогнозувати наступні події в журналі та сигналізувати про відхилення від очікуваного сценарію функціонування системи.

Останніми роками значного поширення набули моделі на основі трансформерів (Transformers), які успішно використовуються для аналізу великих потоків текстових і часових даних. Завдяки механізму уваги (attention mechanism) трансформери здатні ефективно аналізувати довгі послідовності подій та виявляти складні взаємозв'язки між ними. Це робить їх перспективним інструментом для аналізу журналів подій у великих корпоративних та хмарних інформаційних системах.

Крім окремих алгоритмів, у сучасних системах моніторингу активно використовуються комплексні платформи штучного інтелекту, які поєднують декілька методів аналізу. Такі рішення реалізуються в системах класу SIEM та UEBA (User and Entity Behavior Analytics). Вони дозволяють формувати профілі поведінки користувачів і пристроїв, аналізувати історичні дані та автоматично виявляти відхилення від нормальної активності. Використання таких систем значно підвищує ефективність виявлення внутрішніх загроз, складних атак та аномальної поведінки користувачів.

Основними перевагами методів машинного навчання та штучного інтелекту є здатність працювати з великими обсягами даних, автоматично адаптуватися до змін у поведінці системи, виявляти нові типи аномалій та зменшувати залежність від ручного налаштування правил. Водночас такі методи потребують значних обчислювальних ресурсів, якісної підготовки даних та ретельного налаштування параметрів моделей. Крім того, складні моделі штучного інтелекту часто характеризуються низькою інтерпретованістю результатів, що може ускладнювати аналіз причин виявлених аномалій.

Таким чином, сучасні методи виявлення аномалій на основі машинного навчання та штучного інтелекту є перспективним напрямом розвитку систем моніторингу та кібербезпеки. Вони дозволяють значно підвищити ефективність аналізу журналів подій, забезпечують своєчасне виявлення потенційних загроз і створюють основу для побудови інтелектуальних систем захисту інформаційних ресурсів. Особливо актуальним є застосування алгоритмів Isolation Forest, автоенкодерів та моделей глибокого навчання, які демонструють високу ефективність при роботі з великими масивами даних журналів подій сучасних інформаційних систем.

1.6. Порівняльний аналіз існуючих методів виявлення аномалій та обґрунтування вибору підходу для дослідження

Виявлення аномалій у журналах подій інформаційних систем є важливою складовою забезпечення інформаційної безпеки, надійності та стабільності функціонування програмних комплексів. Аномалії можуть свідчити про несанкціонований доступ, кібератаки, збої програмного забезпечення, відмови обладнання або помилки користувачів. Для їх виявлення застосовуються різноманітні методи, які можна умовно поділити на статистичні, кластеризаційні, методи машинного навчання та методи глибокого навчання.

Статистичні методи

Статистичні методи базуються на припущенні, що нормальна поведінка системи підпорядковується певному статистичному розподілу. Аномальними вважаються події або послідовності подій, які суттєво відхиляються від встановлених закономірностей. До таких методів належать аналіз середніх значень, стандартного відхилення, Z-оцінка, міжквартильний розмах (IQR) та інші.

Основними перевагами статистичних методів є простота реалізації, низькі обчислювальні витрати та зрозумілість результатів. Проте вони ефективні лише за умови відносно стабільного характеру даних і часто не здатні виявляти складні або приховані аномалії в багатовимірних наборах даних.

Методи кластеризації

Методи кластеризації дозволяють групувати схожі об'єкти у кластери без попереднього маркування даних. Найпоширенішими алгоритмами є K-Means, DBSCAN та Hierarchical Clustering. У процесі аналізу журнальних записів аномальними вважаються об'єкти, які не належать до жодного кластера або розташовані далеко від центрів кластерів.

Перевагою кластеризаційних методів є можливість працювати з немаркованими даними та виявляти нові типи аномалій. Недоліками є залежність результатів від вибору параметрів алгоритму та складність обробки великих обсягів журналів подій у режимі реального часу.

Методи машинного навчання

Методи машинного навчання сьогодні є найбільш поширеним підходом до виявлення аномалій. Вони можуть бути реалізовані як навчання з учителем, без учителя або напівконтрольоване навчання.

Для аналізу журналів подій часто використовуються такі алгоритми:

- Isolation Forest;
- One-Class SVM;
- Local Outlier Factor (LOF);
- Random Forest;
- Support Vector Machine (SVM).

Алгоритм Isolation Forest є одним із найбільш популярних рішень для виявлення аномалій у великих наборах даних. Його принцип роботи ґрунтується на випадковому розділенні простору ознак. Аномальні об'єкти ізолюються значно швидше, ніж нормальні, оскільки вони зазвичай розташовані далеко від основної маси даних.

До переваг Isolation Forest належать:

- висока швидкість роботи;
- відсутність необхідності у маркованих даних;
- ефективність при роботі з великими обсягами журналів подій;
- стійкість до шуму;
- можливість роботи з багатовимірними даними.

Недоліком є необхідність налаштування окремих параметрів моделі та певна складність інтерпретації результатів порівняно зі статистичними методами.

Методи глибокого навчання

Сучасні дослідження все частіше використовують нейронні мережі для аналізу журналів подій. Найбільш поширеними є:

- Autoencoder;
- LSTM (Long Short-Term Memory);
- Transformer-моделі;
- Deep Neural Networks.

Такі методи здатні виявляти складні часові залежності та приховані закономірності у послідовностях подій. Вони демонструють високу точність при аналізі великих масивів даних, однак потребують значних обчислювальних ресурсів, великої кількості навчальних даних та складного налаштування.

Порівняння основних підходів наведено у таблиці 1.4

Таблиця 1.4 - Порівняння методів машинного навчання для виявлення аномалій у журналах

Метод	**Необхідність розмічених даних**	**Швидкодія**	**Складність реалізації**	**Якість виявлення складних аномалій**
Статистичні методи	Ні	Висока	Низька	Низька
K-Means, DBSCAN	Ні	Середня	Середня	Середня
One-Class SVM	Ні	Низька	Висока	Висока
Isolation Forest	Ні	Висока	Середня	Висока
Autoencoder, LSTM	Частково	Низька	Висока	Дуже висока

Обґрунтування вибору підходу для дослідження

У межах даної роботи для реалізації програмного засобу виявлення аномалій у журналах подій інформаційних систем обрано алгоритм Isolation Forest. Такий вибір обумовлений низкою переваг даного методу.

По-перше, журнали подій інформаційних систем зазвичай не містять попередньо розмічених аномалій, тому використання алгоритмів навчання без учителя є найбільш доцільним. По-друге, Isolation Forest ефективно працює з великими обсягами даних та забезпечує високу швидкість обробки, що є важливим для практичного використання в системах моніторингу. По-третє, алгоритм добре виявляє як одиничні аномальні записи, так і нетипові шаблони поведінки користувачів або сервісів.

Крім того, Isolation Forest реалізований у бібліотеці Scikit-learn, що значно спрощує його інтеграцію в програмний продукт, розроблений мовою Python. Використання цього алгоритму дозволяє отримати прийнятний баланс між точністю виявлення аномалій, швидкодією та складністю реалізації програмного забезпечення.

Таким чином, аналіз існуючих методів показав, що алгоритм Isolation Forest є найбільш доцільним вибором для задачі виявлення аномалій у журналах подій інформаційних систем та буде використаний як основний метод у подальшому дослідженні й розробці програмного забезпечення.

1.7. Висновок до першого розділу

У першому розділі проведено аналіз літературних джерел з питань виявлення аномалій у журналах подій інформаційних систем. Розглянуто поняття інформаційних систем і журналів подій, їх структуру, основні типи та роль у моніторингу функціонування й забезпеченні інформаційної безпеки. Проаналізовано види аномалій, причини їх виникнення та можливі наслідки для роботи інформаційних систем. Досліджено традиційні методи виявлення аномалій, а також сучасні підходи на основі машинного навчання та штучного інтелекту. Проведено порівняльний аналіз існуючих методів і встановлено, що алгоритм Isolation Forest забезпечує оптимальне співвідношення між швидкодією, точністю та складністю реалізації. На основі проведеного аналізу обґрунтовано доцільність використання даного алгоритму як основного методу для розробки програмного засобу виявлення аномалій у журналах подій інформаційних систем.

РОЗДІЛ 2. РОЗРОБКА ПРОГРАМИ ДЛЯ ВИЯВЛЕННЯ АНОМАЛІЙ У ЖУРНАЛАХ ПОДІЙ ІНФОРМАЦІЙНИХ СИСТЕМ

2.1 Постановка задачі виявлення аномалій та формування вимог до програмного забезпечення

У сучасних інформаційних системах журнали подій є одним із основних джерел даних для моніторингу стану програмного забезпечення, мережевої інфраструктури та дій користувачів. Аналіз журналів подій дозволяє своєчасно виявляти відхилення від нормального функціонування системи, що можуть свідчити про збої обладнання, помилки програмного забезпечення, несанкціонований доступ або кіберінциденти.

Метою розробки програмного забезпечення є автоматизація процесу виявлення аномалій у журналах подій інформаційних систем із використанням сучасних методів аналізу даних та машинного навчання. Програма повинна забезпечувати обробку журналів подій, виявлення аномальних записів та формування результатів аналізу у зручному для користувача вигляді.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- виконати аналіз структури журналів подій інформаційних систем;
- реалізувати механізм завантаження та попередньої обробки журналів подій;
- здійснити виділення інформативних ознак із записів журналів;
- реалізувати алгоритм виявлення аномалій;
- забезпечити візуалізацію результатів аналізу;
- провести тестування програмного забезпечення на реальних або модельних даних.

В якості середовища розробки обрано мову програмування Python, яка має широкий набір бібліотек для аналізу даних та машинного навчання. Для реалізації системи можуть бути використані бібліотеки Pandas для обробки даних, NumPy для математичних обчислень, Scikit-learn для побудови моделей

машинного навчання, Matplotlib та Seaborn для візуалізації результатів.

Основними функціональними вимогами до програмного забезпечення є:

- завантаження журналів подій у форматах CSV та TXT;
- попередня обробка даних та усунення пропущених значень;
- формування набору ознак для аналізу;
- автоматичне виявлення аномалій за допомогою алгоритмів машинного навчання;
- відображення результатів аналізу у табличному та графічному вигляді;
- збереження результатів у файл.

Нефункціональні вимоги включають:

- простоту використання;
- швидкодію під час обробки великих обсягів даних;
- масштабованість програмного забезпечення;
- можливість інтеграції з іншими системами моніторингу;
- кросплатформеність та підтримку сучасних операційних систем.

Для реалізації алгоритму виявлення аномалій доцільно використати метод Isolation Forest, який демонструє високу ефективність при роботі з великими наборами даних та не потребує попереднього маркування аномальних записів. Результатом роботи програми є список подій, які класифіковані як аномальні, а також статистична інформація щодо їх кількості та характеристик.

2.2 Проєктування архітектури програми та структури обробки журналів подій

Ефективність виявлення аномалій у журналах подій значною мірою залежить від правильно спроєктованої архітектури програмного забезпечення. Архітектура програми повинна забезпечувати можливість обробки великих обсягів даних, їх аналізу та візуалізації результатів у зручному для користувача вигляді.

Основним призначенням розроблюваної системи є автоматизований аналіз

журналів подій інформаційних систем з метою виявлення аномальних записів, які можуть свідчити про збої в роботі системи, помилки програмного забезпечення або потенційні загрози інформаційній безпеці.

Архітектура програмного забезпечення побудована за модульним принципом, що забезпечує простоту супроводу, масштабованість та можливість подальшого розширення функціональності системи. Основні модулі програми наведено на рисунку 2.1.

Основні модулі програмної системи

1. Модуль завантаження даних

Призначений для зчитування журналів подій із зовнішніх файлів. Підтримуються формати CSV, TXT та LOG. Після завантаження дані передаються до модуля попередньої обробки.

2. Модуль попередньої обробки даних

Виконує підготовку журналів подій до подальшого аналізу. Основними функціями модуля є:

очищення даних від дублікатів;

обробка пропущених значень;

перетворення часових міток;

нормалізація числових параметрів;

кодування категоріальних ознак.

3. Модуль формування ознак

На цьому етапі з журналів подій виділяються характеристики, які можуть бути використані для навчання моделі виявлення аномалій. До таких ознак належать: час виникнення події; рівень критичності повідомлення; тип події;

джерело події; частота виникнення подій; тривалість між послідовними подіями.

4. Модуль виявлення аномалій

Центральний компонент системи, який реалізує алгоритм машинного навчання для пошуку аномальних записів. У роботі планується використання алгоритму Isolation Forest, що дозволяє ефективно виявляти аномалії без

необхідності попереднього маркування даних.

Модуль отримує підготовлений набір ознак та формує результат класифікації кожного запису як нормального або аномального.

5. Модуль візуалізації та формування звітів

Забезпечує відображення результатів аналізу у вигляді: таблиці виявлених аномалій; статистичних показників; графіків розподілу подій; діаграм аномальної активності.

Також модуль дозволяє експортувати результати у файли CSV або Excel.

Структура обробки журналів подій

Процес аналізу журналів подій складається з таких етапів:

- Завантаження журналу подій.
- Попередня обробка даних.
- Виділення інформативних ознак.
- Формування навчального набору даних.
- Виявлення аномалій за допомогою алгоритму Isolation Forest.
- Аналіз результатів класифікації.
- Візуалізація та збереження результатів.

Структуру роботи системи можна подати у вигляді такої послідовності поданої на рискнку 2.1.

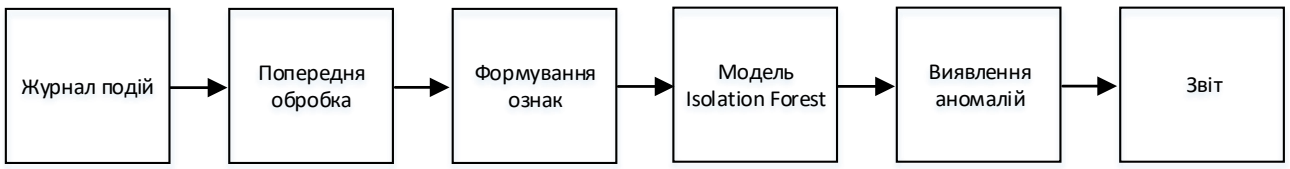

Рисунок 2.1 - Структуру роботи системи

Запропонована архітектура забезпечує гнучкість програмного забезпечення та дозволяє в майбутньому замінювати або доповнювати алгоритми виявлення аномалій іншими методами машинного навчання без суттєвої зміни структури системи.

2.3 Розробка алгоритму виявлення аномалій у журналах подій інформаційних систем

Одним із ключових етапів створення програмного забезпечення для аналізу журналів подій є розробка алгоритму виявлення аномалій. Основним призначенням алгоритму є автоматичне визначення подій, які суттєво відрізняються від нормальної поведінки інформаційної системи та можуть свідчити про виникнення помилок, збоїв, спроб несанкціонованого доступу або інших небажаних ситуацій.

Для реалізації системи виявлення аномалій було обрано алгоритм Isolation Forest, який належить до методів машинного навчання без учителя (Unsupervised Learning). Перевагою даного алгоритму є можливість виявлення аномальних об'єктів без необхідності попереднього маркування даних, що є особливо актуальним для журналів подій інформаційних систем.

Принцип роботи алгоритму Isolation Forest

Основна ідея алгоритму полягає в тому, що аномальні записи трапляються значно рідше за нормальні та мають відмінні характеристики. Тому для їх ізоляції потрібно виконати меншу кількість поділів даних порівняно зі звичайними записами.

Алгоритм будує множину випадкових дерев рішень, у яких кожний вузол містить випадково обрану ознаку та випадкове значення розділення. Для кожного запису визначається довжина шляху від кореня дерева до листка. Якщо середня довжина шляху є малою, об'єкт вважається аномальним.

Етапи роботи алгоритму

Розроблений алгоритм складається з таких основних етапів:

Етап 1. Завантаження журналів подій

На першому етапі виконується зчитування журналів подій із файлів формату CSV або TXT. Отримані записи зберігаються у вигляді таблиці даних.

Етап 2. Попередня обробка даних

Перед початком аналізу здійснюється підготовка журналів подій: видалення порожніх записів; усунення дублікатів; перетворення часових міток; нормалізація числових параметрів; кодування текстових полів.

Етап 3. Формування ознак

Для кожного запису журналу формуються ознаки, що використовуються моделлю машинного навчання: час виникнення події; рівень важливості повідомлення; тип події; код помилки; частота повторення події; інтервал між подіями.

У результаті формується матриця ознак, яка подається на вхід алгоритму.

Етап 4. Навчання моделі

На цьому етапі створюється модель Isolation Forest. Основними параметрами моделі є: кількість дерев; максимальна глибина дерева; очікувана частка аномалій у наборі даних.

Модель виконує побудову ансамблю випадкових дерев для ізоляції аномальних записів.

Етап 5. Виявлення аномалій

Після навчання модель аналізує всі записи журналу подій та присвоює кожному з них оцінку аномальності.

За результатами аналізу кожен запис класифікується як:

1 нормальна подія;

-1 аномальна подія.

Етап 6. Формування результатів

Результати роботи алгоритму відображаються користувачу у вигляді: списку аномальних подій; статистики виявлених аномалій; графічної візуалізації результатів; звітів у форматах CSV або Excel.

Блок-схема алгоритму

Послідовність роботи алгоритму можна подати у вигляді такої блок-схеми поданої на рисунку 2.2.

Рисунок 2.2 – Блок схема алгоритму роботи системи

Вибір алгоритму Isolation Forest

Для реалізації програмного забезпечення алгоритм Isolation Forest обрано з таких причин:

- не потребує розмічених даних;
- ефективно працює з великими наборами журналів подій;
- має низьку обчислювальну складність;
- забезпечує високу швидкодію;
- демонструє хороші результати під час виявлення рідкісних подій.

Запропонований алгоритм дозволяє автоматизувати процес аналізу журналів подій та підвищити ефективність моніторингу інформаційних систем шляхом своєчасного виявлення аномальної активності.

2.4 Реалізація програмного забезпечення та опис основних модулів системи

Розроблена система виявлення аномалій у журналах подій інформаційних систем побудована за модульним принципом і складається з таких основних компонентів:

- модуль збору та завантаження логів;
- модуль парсингу та структуризації даних;
- модуль попередньої обробки та формування ознак;
- модуль виявлення аномалій;
- модуль візуалізації та формування звітів.

Загальна архітектура відповідає типовій схемі pipeline-процесингу даних у системах SIEM та Data Mining.

Модуль завантаження та збору логів

Модуль відповідає за імпорт журналів подій із різних джерел: локальні CSV-файли; системні журнали ОС; журнали додатків; мережеві логи.

Реалізація рисунок 2.3

```
import pandas as pd
def load_logs(path: str):
    df = pd.read_csv(path)
    return df
```

Рисунок 2.3 – Реалізація модуля завантаження та збору логів

Особливості: підтримка великих файлів; можливість потокового завантаження (future extension); уніфікація формату даних.

Модуль парсингу та структуризації логів

Призначений для перетворення неструктурованих або напівструктурованих логів у табличний формат.Даний модуль забезпечує такі операції:

- виділення timestamp;
- визначення рівня події (INFO, ERROR, WARNING);
- ідентифікація користувача та IP-адреси;
- класифікація типу події.

Реалізація подана на рисунку 2.4

```
import re
def parse_event(row):
    pattern = r"(?P<timestamp>[\d\-:\s]+),
(?P<level>\w+),(?P<user>\w+),(?P<ip>[\d\.]+),(?P<event>.+)"
    match = re.match(pattern, row)
    return match.groupdict() if match else None
```

Рисунок 2.4 – Реалізація модуля парсингу та структуризації логів

Модуль попередньої обробки даних

Даний модуль призначений для підготовки даних для алгоритмів машинного навчання.

Основні операції які він здійснює:кодування категоріальних змінних; нормалізація числових значень; створення часових ознак; генерація поведінкових метрик.

Реалізація даного модуля наведена на рисунку 2.5

```
def preprocess(df):
    df["is_error"] = (df["level"] == "ERROR").astype(int)
    df["user_code"] = df["user"].astype("category").cat.codes
    df["timestamp"] = pd.to_datetime(df["timestamp"]).astype("int64") //
10**9
    return df
```

Рисунок 2.5 – Реалізація модуля попередньої обробки даних

Модуль виявлення аномалій

Призначення модкля полягає у аналітичному аналізі, який здійснює ідентифікацію аномальних подій.

Використаний метод при цьому Isolation Forest, який є ефективним для роботи з високовимірними даними без необхідності розміченого набору.

Математична ідея полягає у тому що алгоритм ізолює рідкісні спостереження за допомогою випадкових дерев розбиття простору.

Реалізація подана на рисунку 2.6

```
from sklearn.ensemble import IsolationForest
def detect_anomalies(df):
    features = ["is_error", "user_code", "timestamp"]
    model = IsolationForest(
        n_estimators=100,
        contamination=0.1,
        random_state=42 )
    df["anomaly"] = model.fit_predict(df[features])
    return df
```

Рисунок 2.6 – Реалізація модуля виявлення аномалій

Модуль аналізу результатів

Призначення моддуля для інтерпретації результатів роботи моделі та виділення аномальних подій.

Логіка роботи як була описана вище а саме:

значення -1 → аномалія;

значення 1 → нормальна поведінка.

Реалізація подана на рисунку 2.7

```
def get_anomalies(df):
    return df[df["anomaly"] == -1]
```

Рисунок 2.7 – Реалізація модуля аналізу результатів

Модуль візуалізації

Призначення модуля для графічного представлення результатів аналізу для інтерпретації.

Використані методи це часові графіки кількості аномалій; гістограми подій; scatter plot для кластеризації.

Реалізація подана на рисунку 2.8

```
import matplotlib.pyplot as plt
def plot_anomalies(df):
    plt.figure()
    plt.plot(df["timestamp"], df["anomaly"], marker="o")
    plt.title("Anomaly Detection Results")
    plt.show()
```

Рисунок 2.8 – Реалізація модуля візуалізації

Загальний сценарій роботи системи

Повний pipeline роботи системи виглядає наступним чином:

1. Завантаження логів;
2. Парсинг і структуризація;
3. Попередня обробка;
4. Формування ознак;
5. Застосування моделі Isolation Forest;
6. Визначення аномалій;
7. Візуалізація результатів.

Розроблена система дозволяє автоматизувати процес аналізу журналів подій та виявлення аномалій без використання розмічених даних. Використання алгоритму Isolation Forest забезпечує високу ефективність при роботі з великими обсягами логів та дозволяє виявляти як явні, так і приховані аномальні патерни поведінки.

2.5 Тестування програми та аналіз результатів виявлення аномалій

Для перевірки працездатності розробленого програмного забезпечення було проведено тестування на наборі даних журналів подій інформаційної системи. Метою тестування була оцінка здатності програми автоматично виявляти аномальні записи та визначати потенційно небезпечні події серед великої кількості звичайних системних повідомлень.

Для експерименту використовувався тестовий журнал подій у форматі CSV, який містив інформацію про час події, рівень повідомлення, ім'я користувача, IP-адресу та тип події. До набору даних були включені як типові записи про успішну роботу системи, так і спеціально додані аномальні події, що імітували помилки та підозрілу активність користувачів.

Фрагмент тестового журналу подій наведено нижче (приклад, також наведено у додатку А):

```
timestamp,level,user,ip,event
2026-05-20 10:00:01,INFO,admin,192.168.1.10,login_success
```

2026-05-20 10:00:05,INFO,admin,192.168.1.10,view_dashboard

2026-05-20 10:00:10,INFO,user1,192.168.1.20,login_success

2026-05-20 10:00:25,ERROR,user3,192.168.1.22,db_timeout

2026-05-20 10:01:10,CRITICAL,user7,10.0.0.5,multiple_failed_logins

2026-05-20 10:01:30,ERROR,user8,172.16.1.50,unauthorized_access

Під час тестування програма виконувала попередню обробку даних, що включала перетворення текстових ознак у числовий формат за допомогою кодування категорій та формування набору ознак для алгоритму машинного навчання. Після цього здійснювалося навчання моделі Isolation Forest та визначення аномальних записів.

У результаті роботи програми кожному запису було присвоєно оцінку належності до нормальної або аномальної поведінки. Для позначення результатів використовувалися значення:

1 – нормальний запис;

-1 – аномальний запис.

Фрагмент результатів роботи програми наведено в таблиці 2.1.

Таблиця 2.1 - Результати роботи прогрми

Подія	**Результат**
login_success	Нормальна
view_dashboard	Нормальна
query_data	Нормальна
db_timeout	Аномалія
multiple_failed_logins	Аномалія
unauthorized_access	Аномалія

Аналіз результатів показав, що алгоритм успішно виявив усі події, які були навмисно внесені до журналу як аномальні. Зокрема, були зафіксовані

спроби несанкціонованого доступу, критичні помилки бази даних та серія невдалих спроб автентифікації користувача.

Для наочного представлення результатів програма будувала графік розподілу записів журналу подій, де нормальні записи відображалися окремо від аномальних. Це дозволило візуально оцінити розташування аномалій у наборі даних та підтвердити коректність роботи алгоритму.

Під час експерименту було встановлено, що алгоритм Isolation Forest демонструє високу швидкість роботи навіть при збільшенні кількості записів журналу до декількох тисяч подій. Середній час обробки тестового набору даних становив менше однієї секунди на персональному комп'ютері із середніми технічними характеристиками.

Серед переваг розробленого програмного забезпечення можна виділити:

- автоматичне виявлення аномалій без необхідності попереднього маркування даних;
- можливість роботи з великими обсягами журналів подій;
- простоту інтеграції в існуючі системи моніторингу;
- швидку обробку та аналіз даних;
- можливість подальшого розширення функціоналу.

Водночас під час тестування були виявлені певні обмеження. Алгоритм може помилково класифікувати окремі рідкісні, але легітимні події як аномалії. Крім того, точність виявлення залежить від якості підготовки даних та коректності вибору параметра contamination, який визначає очікувану частку аномалій у наборі даних.

Отримані результати свідчать про ефективність використання алгоритму Isolation Forest для аналізу журналів подій інформаційних систем. Розроблена програма успішно виявляє нетипові події, потенційні порушення безпеки та критичні системні помилки, що підтверджує можливість її практичного застосування в системах моніторингу та кібербезпеки.

Таким чином, проведене тестування підтвердило правильність реалізації програмного забезпечення та доцільність використання методів машинного

навчання для автоматизованого виявлення аномалій у журналах подій інформаційних систем. Отримані результати можуть бути використані як основа для подальшого вдосконалення програмного продукту та розширення його функціональних можливостей.

Сформований файл виявлених аномалій подано на рисунку 2.9

	timestamp	level	user	ip	event	hour	severity	is_error	event_enc	user_enc	ip_enc
1	2026-05-20 10:00:20	INFO	user3	192.168.1.22	login_success	10	1	0	8	5	5
2	2026-05-20 10:00:25	ERROR	user3	192.168.1.22	db_timeout	10	3	1	2	5	5
3	2026-05-20 10:01:10	ERROR	guest	10.0.0.5	account_locked	10	3	1	1	1	0

Рисунок 2.9 – Приклад роботи програми файл виявлених аномалій

У файлі міститься 3 виявлені аномалії зокрема подамо ці аномалії таблицею 2.2

Таблиця 2.2 - Виявлені 3 аномалії

Час	**Користувач**	**Подія**	**Severity**
2026-05-20 10:00:20	user3	login_success	1
2026-05-20 10:00:25	user3	db_timeout	3
2026-05-20 10:01:10	guest	account_locked	3

Виявлення аномалій у журналі подій подано графічно на рисунку 2.10

На графіку представлено результати виявлення аномалій у журналі подій інформаційної системи. Більшість записів віднесено до нормальної поведінки системи та формують основне скупчення даних. Окремі точки, позначені як аномальні, суттєво відрізняються від загальної вибірки за своїми характеристиками та можуть свідчити про нетипову активність, помилки або потенційні інциденти безпеки. Отримані результати підтверджують ефективність алгоритму Isolation Forest для автоматичного виявлення аномалій у журналах подій.

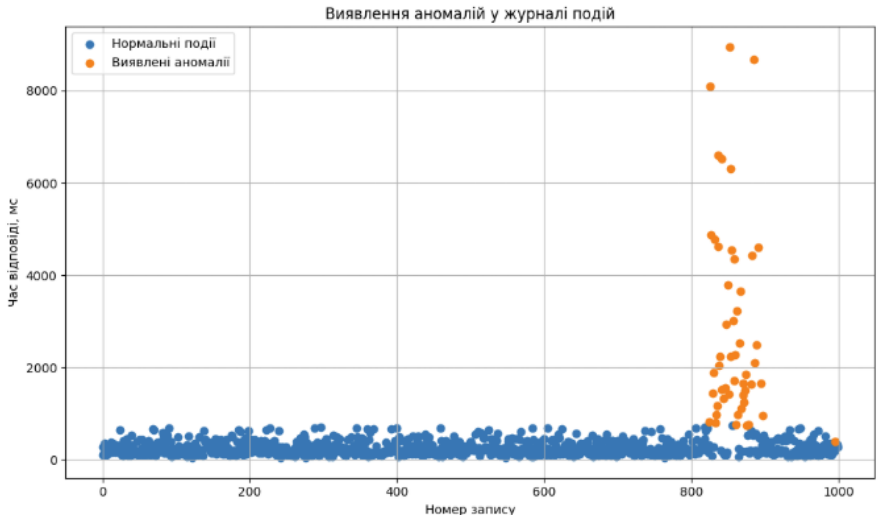

Рисунок 2.10 – Приклад виявлення аномалій в журналі подій

Розподіл подій у журналі подано на рисунку 2.11

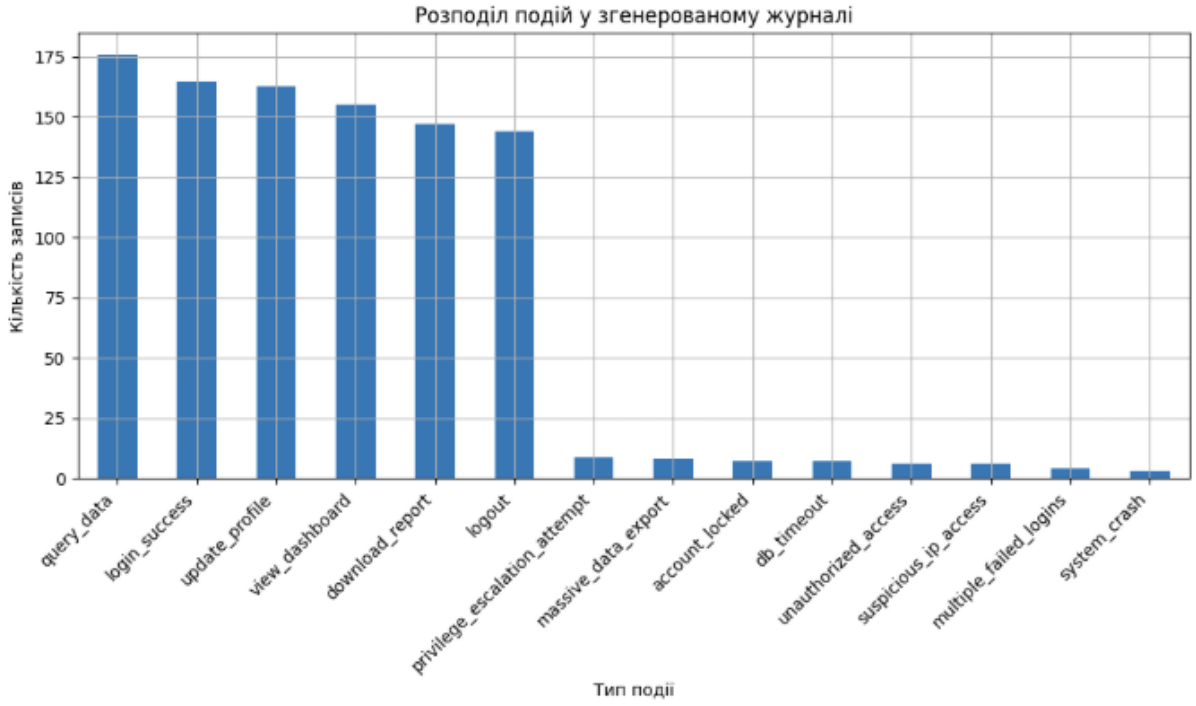

Рисунок 2.11 – Приклад розподілу подій у журналі

На графіку відображено розподіл подій у журналі інформаційної системи за їх типами або категоріями. Аналіз результатів показує, що основна частина подій належить до нормальної службової активності системи, тоді як події з ознаками помилок або попереджень зустрічаються значно рідше. Отриманий розподіл

дозволяє оцінити характер роботи системи, виявити найбільш поширені типи подій та визначити потенційно проблемні ділянки, які потребують додаткового аналізу.

Типи подій, які модель визначила як аномальні подано на рисунку 2.12

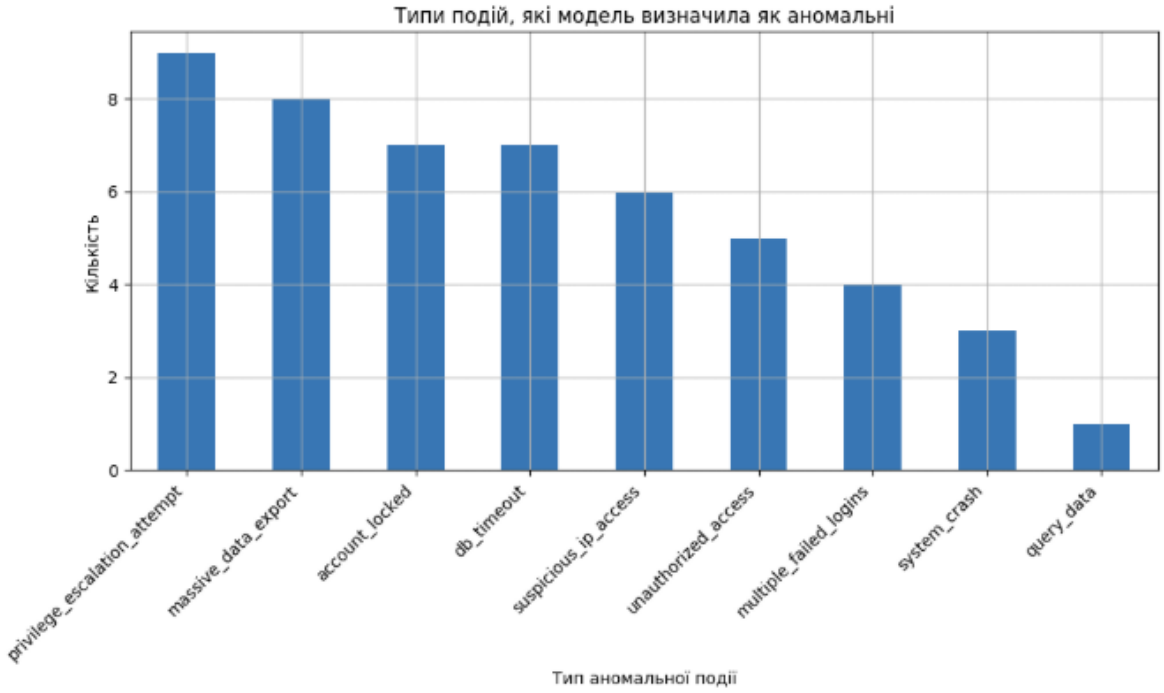

Рисунок 2.12 – Приклад подій, які модель визначила як аномальні

На графіку представлено типи подій, які алгоритм Isolation Forest класифікував як аномальні. Найчастіше до аномалій належать події, пов'язані з помилками, невдалими спробами автентифікації, нетиповою активністю користувачів або іншими відхиленнями від нормальної роботи системи. Отримані результати дозволяють визначити потенційні джерела загроз і зосередити увагу на подіях, що потребують додаткового аналізу та перевірки.

Оцінка аномальності записів Isolation Forest подана на рисунку 2.13

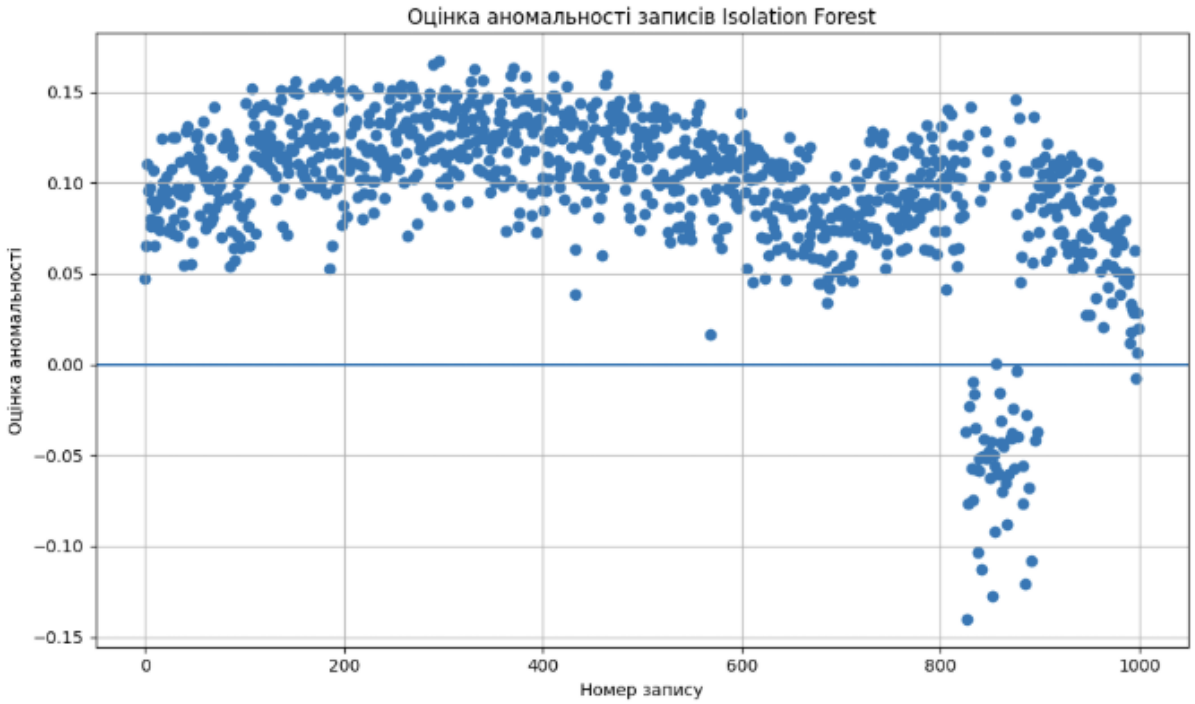

Рисунок 2.13 – Приклад оцінки аномальності записів за допомогою Isolation Forest

На рисунку наведено оцінки аномальності записів, отримані за допомогою алгоритму Isolation Forest. Чим нижче значення оцінки, тим вищою є ймовірність того, що запис є аномальним. Більшість подій мають оцінки, характерні для нормальної роботи системи, тоді як окремі записи суттєво відрізняються від загальної вибірки та були класифіковані як аномалії. Отримані результати демонструють здатність алгоритму ефективно виявляти нетипову активність у журналах подій.

Таким чином було протестовано 1000 подій, з них 50 аномальних. Програма виявила 50 аномалій, з них 49 правильно, 1 пропущена, 1 хибне спрацювання.

2.6 Висновок до другого розділу

У другому розділі виконано проєктування та розробку програмного забезпечення для виявлення аномалій у журналах подій інформаційних систем. Сформульовано функціональні та нефункціональні вимоги до програмної

системи, розроблено її архітектуру та визначено структуру обробки журналів подій. Запропоновано алгоритм аналізу даних, який включає етапи завантаження журналів, попередньої обробки, формування ознак, навчання моделі та виявлення аномальних записів. Для реалізації системи обрано алгоритм Isolation Forest, що дозволяє ефективно знаходити аномалії без попереднього маркування даних. Також описано реалізацію основних програмних модулів мовою Python із використанням сучасних бібліотек аналізу даних та машинного навчання. Розроблена програмна система створює основу для автоматизації моніторингу інформаційних систем і підвищення ефективності виявлення потенційних загроз та збоїв у їх роботі.

РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

3.4 Безпека життєдіяльності. Мета та завдання

Вся сукупність видів людської активності утворює поняття діяльності. Якраз діяльність і вирізняє людину від інших живих істот, вона є специфічно людською формою активності, необхідною умовою існування людського суспільства. Форми діяльності розмаїті. Вони охоплюють практичні, інтелектуальні і духовні процеси, які протікають в побуті, громадській, культурній, виробничій, науковій та інших сферах життя.

Діяльністю займаються всі – діти, дорослі, люди похилого віку, тому безпека діяльності має відношення до всіх людей. Небезпеки підстерігають людей не тільки на виробництві, тому вивчення лише виробничого травматизму в системі загальної безпеки життєдіяльності не висвітлює проблеми.

Безпека – це стан діяльності, при якому з певному ймовірністю виключається прояв небезпек. Безпека – це мета, а безпека життєдіяльності – засоби, шляхи, методи її досягнення.

Актуальність дисципліни ще більше зростає у зв’язку з існуванням аксіоми про потенційну небезпеку діяльності: в жодному виді діяльності неможливо досягнути абсолютної безпеки, будь-яка діяльність потенційно небезпечна.

Завдання БЖД є розробка методів прогнозування, вивчення та ідентифікації шкідливих факторів, їх впливу на людину і довкілля.

Курс БЖД призначений:

– сприяти усвідомленню, що в центрі уваги повинна бути людина, як головна цінність суспільства, та виховати в людині гуманне, свідоме

– ставлення до питань особистої безпеки та безпеки оточуючих в усіх сферах відносин;

– виробити навики ідентифікації небезпечних та шкідливих факторів і створення сприятливих умов життєдіяльності людей на певній території;

– тримати на контролі проектування нової техніки і технологічних процесів згідно з сучасними вимогами екології і з урахуванням стійкості функціонування господарських об’єктів та технічних систем;

– прогнозувати можливу обстановку і приймати грамотні рішення в умовах надзвичайних ситуацій щодо захисту населення та персоналу об’єктів від можливих негативних наслідків;

– забезпечити якісне засвоєння нового стереотипу поведінки людини з метою виживання в нових природних та антропогенних умовах.

Безпека життєдіяльності базується на досягненнях таких дисциплін, як інженерна психологія, фізіологія людини, охорона праці, екологія, ергономіка, економіка тощо. Вона була і є в центрі уваги людей. З древніх часів до наших днів людина прагнула забезпечити свою безпеку. З розвитком промисловості це потребує спеціальних знань. БЖД особливо актуальна зараз, в добу науково-технічного прогресу. Вона покликана відіграти важливу роль в стабілізації людського суспільства.

Завдання курсу “Безпека життєдіяльності” (БЖД) полягає у чіткому розумінні небезпечних чинників у ситуаціях, що виникають як у середовищі проживання людини, так і у середовищі навчання і праці.

БЖД – це ступінь захисту людини від надзвичайної небезпеки, де під терміном “небезпека” мається на увазі вплив на людину факторів, які можуть викликати відхилення стану її здоров'я від нормального. Природа цих факторів може бути пов’язана як з причинами природного або соціально-економічного характеру (екологічними катастрофами, низьким економічним рівнем життя та ін.), так і з причинами техногенного характеру (з рівнем забруднення навколишнього середовища як наслідком виробничої діяльності людини, аваріями, катастрофами на підприємстві, транспорті, війнами та ін.).

Викладання дисципліни має на меті:

– вивчення структури, змісту і взаємозв’язку життєдіяльності людини із середовищем праці й проживання;

– визначення чинників, причин і параметрів, що сприяють виникненню надзвичайних ситуацій;

– визначення принципів і способів захисту людей в умовах повсякденного життя, а також в умовах надзвичайних ситуацій.

3.4 Інформаційне забезпечення БЖД

Відповідно до ДСТУ 2938-94 "Системи обробки інформації. Основні поняття. Терміни та визначення" комп'ютер — це функційний пристрій, що складається з одного чи кількох взаємопов'язаних центральних процесорів і периферійних пристроїв і може виконувати обчислення без участі людини

Основними функціями комп'ютера є введення та виведення інформації, її зберігання та обробка. В якості пристроїв введення часто використовуються клавіатура та сканер, який забезпечує більшу швидкодію. Інформацію також можна вводити шляхом її зчитування з магнітних, оптичних та оптико-магнітних носіїв. Комп'ютер може отримувати інформацію і з комп'ютерної мережі.

Введена в системний блок інформація впорядковується або опрацьовується відповідно до програми („програмне забезпечення”), яке визначає логічні кроки процесу опрацювання. Цей процес повністю автоматизований і здійснюється без зовнішнього впливу.

Пристроями виведення можуть бути дисплеї, друкувальні (принтер) та графопобудовуючі (плотер) пристрої. Інформація може також виводитись на магнітні або оптико-магнітні носії, які потім, в свою чергу, можуть використовуватись для введення інформації.

Для зберігання програм та інформації застосовують, головним чином магнітні, оптико-магнітні та оптичні диски, які дають можливість довільного доступу до даних і забезпечують високу швидкодію.

У зв'язку з бурхливим розвитком комп'ютерної техніки щороку зростає спектр їх різновидів. За призначенням комп'ютери можна умовно поділити на:

– побутові комп'ютери — власне ПК, що призначені для індивідуальної роботи в домашніх умовах;

– навчальні комп'ютери — призначені для використання в системі освіти, як вищої так і середньої. Основні вимоги таких комп'ютерів — надійність, достатня потужність та невисока ціна (можливість придбання більшої кількості однотипних комп’ютерів закладами освіти). Користувачі — учні, студенти, викладачі та ін.;

– професійні комп'ютери — робочі станції для роботи на виробництві, в офісах установ, які, як правило, об'єднані в локальну комп'ютерну мережу. Від „побутових” відрізняються більш високими показниками за всіма параметрами — продуктивністю, функціональними можливостями, якісткю зображення на дисплеї та ін. Користувачі — службовці, оператори ВДТ;

– сервери — потужні комп'ютери, призначені для локальних та глобальних мереж. Вони виконують функції керування робочими станціями, зберігання значних масивів інформації та ін. Користувачі — менеджери, і адміністратори локальних комп'ютерних мереж, системні та прикладні програмісти для підтримки програмного забезпечення тощо;

– графічні станції — використовуються для роботи з графічними зображеннями, відео та анімацією. Володіють надзвичайно високими ресурсами за всіма основними параметрами.

– На сьогодні найбільш розповсюдженими є персональні комп'ютери (рис 7.1). В мінімальний базовий комплект ПК входять наступні блоки чи компоненти:

– системний блок, в якому зосереджені життєво важливі елементи комп'ютера;

– дисплей (монітор), який призначений для виведення (відображення) інформації;

– клавіатура, яка призначена для введення інформації в комп'ютер;

– графічний маніпулятор „миша”, який слугує для керування роботою програм шляхом вибору різних пунктів меню, виділення та „перетягування” об'єктів.

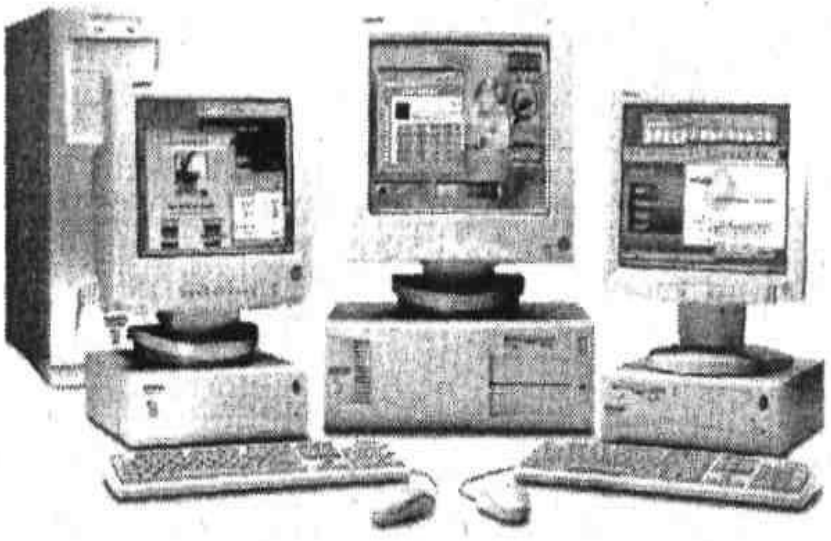

Рисунок 3.1 – Загальний вигляд ПК

При такому апаратному комплектуванні та при наявності відповідного програмного забезпечення вже можна повноцінно працювати за комп'ютером. Саме в такому комплектуванні комп'ютери, як правило, надходять у продаж.

3.4 План ліквідації аварій на виробничому об’єкті

Коротка характеристика аварій і катастроф.

Великі аварії і катастрофи на об'єктах можуть виникати в результаті стихійного лиха, а також порушення технології виробництва, правил експлуатації різних машин, обладнання і встановлених заходів безпеки. Їх дії подібні до стихійних лих.

Під аварією розуміють раптову зупинку роботи або порушення процесу виробництва на промисловому підприємстві, транспорті, інших об'єктах, що приводять до пошкодження або знищення матеріальних цінностей.

Під катастрофою розуміють раптове лихо; подію, що спричиняє за собою трагічні наслідки. Катастрофи супроводжуються руйнуванням будівель різних споруд, знищенням матеріальних цінностей і загибеллю людей.

Найбільш небезпечним наслідком великих аварій і катастроф є пожежі і вибухи. У ряді випадків, особливо на підприємствах нафтової, хімічної і газової промисловості, аварії викликають загазованість атмосфери, розлив нафтопродуктів, агресивних рідин і сильнодіючих отруйних речовин. Аварії і катастрофи можуть бути на залізничному, повітряному і водному транспорті, а також в результаті обвалення при будівництві і монтажі споруд і конструкцій різних об'єктів.

Основи використання формувань при стихійних лихах, великих аваріях і катастрофах. Для ліквідації наслідків, викликаних стихійними лихами, можуть притягуватися як формування загального призначення, так і формування служб ДО. В окремих випадках крім вказаних формувань можуть притягуватися військові частини ДО.

Основне завдання формувань при ліквідації наслідків стихійних лих, великих аварій і катастроф — спасіння людей і матеріальних цінностей. Характер і порядок дій формувань при виконанні цього завдання залежать від виду стихійного лиха, аварії або катастрофи, обстановки, що склалася, кількості і підготовленості сил цивільної оборони, що залучаються, пори року і доби, погодних умов і інших чинників.

Успіх дій формувань багато в чому залежить від своєчасної організації і проведення розвідки і обліку конкретних умов обстановки.

У районах стихійних лих розвідка визначає: межі осередку лиха і напряму його поширення, об'єкти і населені пункти, яким загрожує безпосередня небезпека, місця скупчення людей, шляхи підходу техніки до місць робіт, стан пошкоджених будівель і споруд, а також наявність в них уражених людей, місця аварій на комунально-енергетичних мережах, об'єм рятувальних і невідкладних аварійно-відновних робіт.

При крупних аваріях і катастрофах розвідка уточнює ступінь і об'єм руйнувань і можливість проведення робіт без засобів індивідуального захисту, можливість обвалення будівель і споруд, які можуть спричинити за собою збільшення розміру аварії або катастрофи, місця скупчення людей і ступінь загрози для їх життя, а також стан комунально-енергетичних мереж і транспортних комунікацій.

Розвідку ведуть розвідувальні групи і ланки. До складу розвідувальних формувань рекомендується включати фахівців, що знають розташування об'єкту і специфіку виробництва. Якщо в районі майбутніх дій можуть бути сильнодіючі отруйні речовини, то до складу розвідувальних формувань необхідно включати фахівців-хіміків і медичних працівників.

У зв'язку з раптовістю виникнення стихійних лих, великих аварій і катастроф сповіщення особового складу формувань, їх укомплектовування, створення угрупування проводяться в короткі терміни.

У перший ешелон угрупування сил зазвичай включаються формування об'єктів, де відбулися лиха, а в другій — формування сусідніх об'єктів (районів). Висунення формувань із районів збору в район дій здійснюється на максимально можливих швидкостях.

У районах стихійних лих і місцях великих аварій рятувальні роботи в першу чергу проводять з метою попередження виникнення катастрофічних наслідків, лих (аварій), запобігання виникненню вторинних причин, які можуть викликати загибель людей і матеріальних цінностей.

Командири формувань повинні постійно знати обстановку в районі робіт і відповідно до її зміни уточнювати або ставити нові завдання підрозділам.

Після виконання поставлених завдань формування виводяться в район постійного розквартировування.

Рятувальні і невідкладні аварійно-відновні роботи при ліквідації наслідків великих аварій і катастроф.

При землетрусах для проведення рятувальних і невідкладних аварійно-відновних робіт притягуються рятувальні, зведені загони (команди), загони

(команди) механізації робіт, аварійно-технічні команди, інші формування, які мають на оснащенні бульдозери, екскаватори, крани, механізований інструмент і засоби малої механізації.

При проведенні рятувальних робіт в осередку землетрусу перш за все витягують з-під завалів, із напівзруйнованих будівель людей, яким надають першу медичну допомогу, що горять; влаштовують в завалах проїзди; локалізують і усувають аварії на інженерних мережах, які загрожують життю людей або перешкоджають проведенню рятувальних робіт; обрушують або укріплюють конструкції будівель і споруд, що знаходяться в аварійному стані; обладнали пункти збору, що постраждали і медичні пункти; організовують водопостачання.

Послідовність і терміни виконання робіт встановлює начальник цивільної оборони об'єкту, що опинився в зоні землетрусу.

При повенях для проведення рятувальних робіт залучають рятувальні загони, команди і групи, а також відомчі спеціалізовані команди і підрозділи, оснащені плавзасобами, санітарні дружини і пости, гідрометеорологічні пости, розвідувальні групи і ланки, зведені загони (команди) механізації робіт, формування будівельних, ремонтно-будівельних організацій, охорони громадського порядку.

При великих аваріях і катастрофах організація робіт по ліквідації наслідків проводиться з урахуванням обстановки, що склалася після аварії або катастрофи, ступеня руйнування і пошкодження будівель і споруд, технологічного обладнання, агрегатів, характеру аварій на комунально-енергетичних мережах і пожеж, особливостей забудови території об'єкту і інших умов.

Роботи по організації ліквідації наслідків аварій і катастроф проводяться в стислі терміни: необхідно швидко врятувати людей, що знаходяться під уламками будівель, в завалених підвалах, і надати їм екстрену медичну допомогу, а також запобігти іншим катастрофічним наслідкам, пов'язаним із загибеллю людей і втратою великої кількості матеріальних цінностей.

Із виникненням аварії або катастрофи начальник цивільної оборони на підставі даних розвідки і особистого спостереження ухвалює рішення на ліквідацію наслідків і ставить завдання формуванням.

Начальники ділянок керують рятувальними і невідкладними аварійно-відновними роботами. Вони вказують командирам формувань найбільш доцільні прийоми і способи виконання робіт, визначають матеріально-технічне забезпечення, терміни закінчення робіт і представляють донесення про об'єм виконаних робіт, організовують живлення, зміну і відпочинок особового складу формувань.

Заходи щодо попередження великих аварій і катастроф. Великі виробничі аварії і катастрофи завдають великого збитку народному господарству, тому забезпечення безаварійної роботи має виключно велике державне значення. Сучасне промислове підприємство є складним інженерно-технічним комплексом. Успіх його роботи багато в чому залежить від стану інших підприємств галузі, об'єктів суміжних галузей, що забезпечують постачання по кооперації, а також від стану енергопостачання, транспортних комунікацій, зв'язку і т.п. Заходи щодо попередження аварій і катастроф є найбільш складними і трудомісткими. Вони представляють комплекс організаційних і інженерно-технічних заходів, направлених на виявлення і усунення причин аварій і катастроф, максимальне зниження можливих руйнувань і втрат у випадку, якщо ці причини повністю не вдається усунути, а також на створення сприятливих умов для організації і проведення рятувальних і невідкладних аварійно-відновних робіт.

Найбільш ефективним заходом є закладка в проекти новостворюваних об'єктів планувальних, технічних і технологічних рішень, які повинні максимально зменшити ймовірність виникнення аварій або значно понизити матеріальний збиток у випадку, якщо аварія відбудеться. Так, для зниження пожежної небезпеки передбачається зменшення питомої ваги матеріалів, що згорають. При проектуванні нових і реконструкції існуючих систем водопостачання враховується потреба у воді не тільки для виробничих цілей, але

і для випадку виникнення пожежі. Подібні рішення розробляються і по інших елементах виробництва. Враховуються вимоги охорони праці, техніка безпеки, правила експлуатації енергетичних установок, обладнання підйомного крану, місткостей під високим тиском і т.д. Таким чином, ці заходи розробляються і впроваджуються комплексно, із обхватом всіх питань, від яких залежить безаварійна робота об'єктів, з урахуванням їх виробничих і територіальних особливостей, із залученням всіх ланок керування виробничою діяльністю.

3.4 Висновок до третього розділу

У третьому розділі кваліфікаційної роботи висвітлено ключові питання безпеки життєдіяльності, сформульовано її мету та завдання. Окрему увагу приділено розробленню плану ліквідації аварій на виробничому об’єкті, аналізу причин і наслідків аварій та катастроф, а також особливостям виконання рятувальних і невідкладних аварійно-відновлювальних заходів при усуненні наслідків надзвичайних ситуацій техногенного характеру.

ВИСНОВКИ

В даній кваліфікаційній роботі розглянуто методи аналізу виявлення аномалій у журналах подій та була розроблена програми для виявлення аномалій у журналах подій інформаційних систем. Описано основні моменти щодо журналів подій в інформаціних системах їх роль у забезпеченні безпеки, розглянуті причини винекнення аномалій, проведений аналіз методів їх виявлення.

При створенні програми було опрацьовано наступні задачі:

1. Проведено аналіз літературних джерел, у результаті якого досліджено існуючі методи виявлення аномалій та визначено їх переваги, недоліки й особливості застосування. Обґрунтовано доцільність використання алгоритму Isolation Forest для поставленої задачі.

2.Розроблено архітектуру програмного забезпечення, яка включає модулі завантаження, обробки, аналізу даних та візуалізації результатів. Запропонована структура забезпечує гнучкість і можливість подальшого розширення системи.

3. Створено програму для виявлення аномалій у журналах подій інформаційних систем на основі алгоритму Isolation Forest. Програмний засіб автоматично виконує аналіз даних та визначає нетипові події.

4. Проведено тестування та оцінювання ефективності програми, які підтвердили її працездатність і здатність виявляти аномальні записи. Отримані результати були візуалізовані та проаналізовані для оцінки якості роботи алгоритму.

У розділі «Основи безпеки життєдіяльності та охорони праці» розглянуто основні аспекти забезпечення безпеки людини в процесі професійної діяльності та в надзвичайних ситуаціях. Особливу увагу приділено призначенню, завданням і принципам безпеки життєдіяльності, а також питанням охорони праці. Крім того, проаналізовано можливі аварійні ситуації на об'єктах та наведено основні заходи щодо їх попередження і ліквідації. Отримані результати спрямовані на

підвищення рівня безпеки працівників і мінімізацію ризиків виникнення небезпечних ситуацій.

ПЕРЕЛІК ДЖЕРЕЛ

1. Aggarwal C. C. Outlier Analysis / C. C. Aggarwal. — 2nd ed. — Cham : Springer, 2017. — 446 p.
2. Chandola V. Anomaly Detection: A Survey / V. Chandola, A. Banerjee, V. Kumar // ACM Computing Surveys. — 2009. — Vol. 41, № 3. — P. 1–58.
3. Liu F. T. Isolation Forest / F. T. Liu, K. M. Ting, Z.-H. Zhou // Proceedings of the IEEE International Conference on Data Mining. — Pisa, Italy, 2008. — P. 413–422.
4. Liu F. T. Isolation-Based Anomaly Detection / F. T. Liu, K. M. Ting, Z.-H. Zhou // ACM Transactions on Knowledge Discovery from Data. — 2012. — Vol. 6, № 1. — P. 1–39.
5. Bishop C. M. Pattern Recognition and Machine Learning / C. M. Bishop. — New York : Springer, 2006. — 738 p.
6. Goodfellow I. Deep Learning / I. Goodfellow, Y. Bengio, A. Courville. — Cambridge : MIT Press, 2016. — 775 p.
7. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow / A. Géron. — 3rd ed. — Sebastopol : O’Reilly Media, 2022. — 851 p.
8. Han J. Data Mining: Concepts and Techniques / J. Han, M. Kamber, J. Pei. — 4th ed. — Burlington : Morgan Kaufmann, 2022. — 740 p.
9. Scikit-learn Developers. Scikit-learn: Machine Learning in Python [Electronic resource]. — Access mode: https://scikit-learn.org
10. McKinney W. Python for Data Analysis / W. McKinney. — 3rd ed. — Sebastopol : O’Reilly Media, 2022. — 579 p.
11. VanderPlas J. Python Data Science Handbook / J. VanderPlas. — Sebastopol : O’Reilly Media, 2017. — 548 p.
12. Turnbull J. The Logstash Book / J. Turnbull. — New York : James Turnbull, 2014. — 420 p.

13. Chuvakin A. Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management / A. Chuvakin, K. Schmidt, C. Phillips. — Waltham : Syngress, 2012. — 432 p.
14. Bejtlich R. The Practice of Network Security Monitoring / R. Bejtlich. — San Francisco : No Starch Press, 2013. — 376 p.
15. Easttom C. Network Defense and Countermeasures / C. Easttom. — Cham : Springer, 2020. — 428 p.
16. Stallings W. Network Security Essentials: Applications and Standards / W. Stallings. — 7th ed. — London : Pearson Education, 2020. — 816 p.
17. Kim D. Fundamentals of Information Systems Security / D. Kim, M. Solomon. — 4th ed. — Burlington : Jones & Bartlett Learning, 2021. — 720 p.
18. Van der Aalst W. Process Mining: Data Science in Action / W. Van der Aalst. — 2nd ed. — Berlin : Springer, 2016. — 467 p.
19. Splunk Inc. Splunk Documentation [Electronic resource]. — Access mode: https://docs.splunk.com
20. Elastic N.V. Elastic Stack Documentation [Electronic resource]. — Access mode: https://www.elastic.co/guide
21. Литвиненко Я.В., Лупенко С.А., Щербак Л.М. Моделювання та обробка циклічних сигналів серця на ЕОМ. // Вимірювальна та обчислювальна техніка в технологічних процесах.- Хмельницький: Вид.-во “Навчальна книга”. – 2000. №3, -132-139с.
22. Литвиненко Я., Щербак Л. Система комп’ютерних прогам для автоматизованої обробки та моделювання кардіосигналів // Тези доповідей п’ятої наук.-техн. конф. ТДТУ ”Прогресивні матеріали, технології та обладнання в машино- і приладобудуванні”. Тернопіль. – 2001. – 16.
23. Осухівська Г.М. Обгрунтування вибору фільтру для статистичного аналізу тональних сигналів. Вісник Тернопільського державного технічного університету. 1997. Т. 2, № 2. С. 57-62.
24. Литвиненко Я., Лупенко С., Щербак Л. Статистичний метод визначення зонної структури електрокардіосигналу в автоматизованих діагностичних

системах. Вісник Тернопільського державного технічного університету. Тернопіль, 2005. Т. 10, № 3. С. 165-175.

25. Литвиненко Я., Щербак Л. Система комп'ютерних прогам для автоматизованої обробки та моделювання кардіосигналів. Тези доповідей п'ятої наук. конф. ТДТУ. Тернопіль. 2001. С. 16.
26. Лупенко С. А., Литвиненко Я. В., Сверстюк А. С. Статистичний сумісний аналіз кардіосигналів на основі вектора циклічних ритмічно пов'язаних випадкових процесів. Електроніка та системи управління. Національний авіаційний університет. Київ, 2008. № 4 (18). С. 22-29.
27. Лупенко С., Литвиненко Я., Сверстюк А. Сумісна статистична обробка синхронно зареєстрованих кардіосигналів на базі їх моделі у вигляді циклічних ритмічно пов'язаних випадкових процесів. Матеріали дванадцятої наукової конференції Тернопільського державного технічного університету імені Івана Пулюя, м. Тернопіль, 14-15 травня 2008 р. Тернопіль, 2008. С. 111.
28. Литвиненко Я.В. Моделювання та методи визначення зонної часової структури електрокардіосигналу в автоматизованих діагностичних системах: автореф. дис. … канд. техн. наук: 01.05.02. Тернопільський державний технічний університет імені Івана Пулюя. Тернопіль, 2006. 20 с.
29. Імітаційне моделювання взаємопов'язаних економічних циклічних процесів на основі вектора циклічних ритмічно пов'язаних випадкових процесів / А. Б. Горкуненко, С. А. Лупенко, Н. Р. Дем'янчук, Я. В. Литвиненко // Електроніка та системи управління. К: НАУ, 2011. № 2. С. 133–141.
30. Інформаційна технологія моделювання, аналізу та прогнозування циклічних економічних процесів / А. Б. Горкуненко, С. А. Лупенко, Г. М. Осухівська, Н. Б. Стадник // Вимірювальна та обчислювальна техніка в технологічних процесах. Хмельницький: ХНУ, 2012. № 2. С. 167−176.
31. Інформаційна технологія прогнозування циклічних економічних процесів / А. Горкуненко, Р. Козак, Я. Литвиненко [та ін.] // Вісник Тернопільського національного технічного університету ім. І. Пулюя. Тернопіль: ТНТУ, 2012. № 1. С. 143−154.

32.I.V. Lytvynenko. Method of segmentation of determined cyclic signals for the problems related to their processing and modeling/ I.V. Lytvynenko / Scientific Journal of the ternopil national technical university. 2017, Vol. 88, No. 4, pp. 153-169.

33.I.V. Lytvynenko. The method of segmentation of stochastic cyclic signals for the problems of their processing and modeling/ I.V. Lytvynenko / Journal of Hydrocarbon Power Engineering, Oil and Gas Measurement and Testing. 2017, Vol. 4, No. 2, pp. 93-103.

34.I. Lytvynenko. Segmentation and Statistical Processing of Geometric and Spatial Data on Self-Organized Surface Relief of Statically Deformed Aluminum Alloy. // Iaroslav Lytvynenko, Pavlo Maruschak, Sergiy Lupenko, Sergey Panin // Applied Mechanics and Materials, 2015, Vol. 770, pp. 288-293.

35.I.V. Lytvynenko. Software for segmentation, statistical analysis and modeling of surface ordered structures // I.V. Lytvynenko, P.O. Maruschak, S.A. Lupenko, Yu. I. Hats, A. Menou, S.V. Panin // MECHANICS, RESOURCE AND DIAGNOSTICS OF MATERIALS AND STRUCTURES (MRDMS-2016): Proceedings of the 10th International Conference on Mechanics, Resource and Diagnostics of Materials and Structures. AIP Publishing, 2016, Vol. 1785, No.1, pp. 030012-1-030012-7.

36.I.V. Lytvynenko. Method of segmentation of determined cyclic signals for the problems related to their processing and modeling/ I.V. Lytvynenko / Scientific Journal of the ternopil national technical university. 2017, Vol. 88, No. 4, pp. 153-169.

37.I. Lytvynenko. Segmentation and Statistical Processing of Geometric and Spatial Data on Self-Organized Surface Relief of Statically Deformed Aluminum Alloy. // Iaroslav Lytvynenko, Pavlo Maruschak, Sergiy Lupenko, Sergey Panin // Applied Mechanics and Materials, 2015, Vol. 770, pp. 288-293.

38.I.V. Lytvynenko. Method of the quadratic interpolation of the discrete rhythm function of the cyclical signal with a defined segment structure / I.V. Lytvynenko /

Scientific Journal of the ternopil national technical university. 2016, Vol. 84, No. 4, pp. 131-138.

39. S. Lupenko, A. Lupenko, I. Lytvynenko, V. Martsenyuk. Methods for Estimating the Discrete Rhythmic Structure of Cyclic Random Processes Using Adaptive Interpolation Conference on Computer Science and Information Technologies CSIT 2020: Advances in Intelligent Systems and Computing V pp 614-627 Conference paper. First Online: 23 December 2020. Part of the Advances in Intelligent Systems and Computing book series (AISC, volume 1293).
40. Method of Evaluation of Discrete Rhythm Structure of Cyclic Signals with the Help of Adaptive Interpolation Lytvynenko, I., Lupenko, S., Onyskiv, P. 2020 IEEE 15th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2020 - Proceedings, 2020, 1, pp. 155–158, 9321878
41. I.V. Lytvynenko. Software for segmentation, statistical analysis and modeling of surface ordered structures // I.V. Lytvynenko, P.O. Maruschak, S.A. Lupenko, Yu. I. Hats, A. Menou, S.V. Panin // MECHANICS, RESOURCE AND DIAGNOSTICS OF MATERIALS AND STRUCTURES (MRDMS-2016): Proceedings of the 10th International Conference on Mechanics, Resource and Diagnostics of Materials and Structures. AIP Publishing, 2016, Vol. 1785, No.1, pp. 030012-1-030012-7.
42. Software for statistical processing and modeling of a set of synchronously registered cardio signals of different physical nature Lupenko, S., Lytvynenko, I., Sverstiuk, A., Horkunenko, A., Shelestovskyi, B. CEUR Workshop Proceedings, 2021, 2864, pp. 194–205
43. Modeling and Methods of Statistical Processing of a Vector Rhytmocardiosignal I Lytvynenko, S Lupenko, P Onyskiv, A Zozulia The Open Bioinformatics Journal 14 (1) 73-86
44. I.V. Lytvynenko, P.O. Marushak, S.A. Lupenko, Yu.I. Hats, A.Menou. Software tools for the analysis of the self-organizing material surface after deformation for the problems of its sementation and statistical processing // Proc. of International

Symposium Aircraft materials. ACMA 2016. (May 11-13). - 2016. - Morocco, Agadir. – P. 138-139.

45. Lupenko, S., Lytvynenko, I., Stadnyk, N. Method of Statistical Processing of Discrete Cycle Random Processes, by their Reduction to Isomorphic Periodic Random Sequences 2020 10th International Conference on Advanced Computer Information Technologies, ACIT 2020 - Proceedings, 2020, pp. 209-212, 9209004

ДОДАТКИ

Додаток А

Приклад вхідного файлу для роботи з журналами подій який містить аномалії

Фрагмент вхідного файлу logs.csv

```
timestamp,level,user,ip,event
2026-05-20 10:00:01,INFO,admin,192.168.1.10,login_success
2026-05-20 10:00:05,INFO,admin,192.168.1.10,view_dashboard
2026-05-20 10:00:10,INFO,user1,192.168.1.20,login_success
2026-05-20 10:00:15,INFO,user2,192.168.1.21,login_success
2026-05-20 10:00:20,INFO,user3,192.168.1.22,login_success
2026-05-20 10:00:25,ERROR,user3,192.168.1.22,db_timeout
2026-05-20 10:00:30,INFO,user1,192.168.1.20,query_data
2026-05-20 10:00:35,INFO,user2,192.168.1.21,download_report
2026-05-20 10:00:40,INFO,user1,192.168.1.20,logout

2026-05-20 10:01:01,INFO,guest,10.0.0.5,login_attempt
2026-05-20 10:01:02,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:03,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:04,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:05,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:06,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:07,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:08,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:09,ERROR,guest,10.0.0.5,login_failed
2026-05-20 10:01:10,ERROR,guest,10.0.0.5,account_locked

2026-05-20 10:02:00,INFO,admin,192.168.1.10,login_success
2026-05-20 10:02:05,INFO,admin,192.168.1.10,access_sensitive_file
2026-05-20 10:02:10,INFO,admin,192.168.1.10,export_data
2026-05-20 10:02:15,INFO,admin,192.168.1.10,delete_log_entry
2026-05-20 10:02:20,ERROR,system,127.0.0.1,permission_denied
2026-05-20 10:02:25,ERROR,system,127.0.0.1,permission_denied

2026-05-20 10:03:00,INFO,user4,192.168.1.30,login_success
2026-05-20 10:03:05,INFO,user4,192.168.1.30,view_profile
2026-05-20 10:03:10,INFO,user4,192.168.1.30,update_settings
2026-05-20 10:03:15,INFO,user4,192.168.1.30,logout

2026-05-20 10:04:01,INFO,unknown,185.199.108.50,login_attempt
2026-05-20 10:04:02,ERROR,unknown,185.199.108.50,login_failed
```

```
2026-05-20 10:04:03,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:04,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:05,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:06,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:07,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:08,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:09,ERROR,unknown,185.199.108.50,login_failed
2026-05-20 10:04:10,ERROR,unknown,185.199.108.50,account_blocked

2026-05-20 10:05:00,INFO,user2,192.168.1.21,login_success
2026-05-20 10:05:05,INFO,user2,192.168.1.21,query_data
2026-05-20 10:05:10,INFO,user2,192.168.1.21,query_data
2026-05-20 10:05:15,ERROR,user2,192.168.1.21,db_timeout
2026-05-20 10:05:20,ERROR,user2,192.168.1.21,db_timeout

2026-05-20 10:06:00,INFO,admin,192.168.1.10,login_success
2026-05-20 10:06:05,INFO,admin,192.168.1.10,bulk_export
2026-05-20 10:06:10,INFO,admin,192.168.1.10,bulk_export
2026-05-20 10:06:15,INFO,admin,192.168.1.10,bulk_export
2026-05-20 10:06:20,ERROR,admin,192.168.1.10,rate_limit_exceeded

2026-05-20 10:07:00,INFO,user5,192.168.1.40,login_success
2026-05-20 10:07:05,INFO,user5,192.168.1.40,view_dashboard
2026-05-20 10:07:10,INFO,user5,192.168.1.40,logout

2026-05-20 10:08:00,INFO,bot,203.0.113.99,scan_port
2026-05-20 10:08:01,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:02,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:03,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:04,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:05,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:06,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:07,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:08,ERROR,bot,203.0.113.99,login_failed
2026-05-20 10:08:09,ERROR,bot,203.0.113.99,blocked_ip
```

Додаток Б

Приклад коду програми для виявлення вномалій в журналах подій

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.ensemble import IsolationForest
from sklearn.preprocessing import LabelEncoder

# ==========================
# 1. Завантаження даних
# ==========================
df = pd.read_csv("logs.csv")

print("Перші записи журналу:")
print(df.head())

# ==========================
# 2. Попередня обробка
# ==========================

# Перетворення часу
df['timestamp'] = pd.to_datetime(df['timestamp'])

# Виділення години доби
df['hour'] = df['timestamp'].dt.hour

# Кодування типу події
encoder = LabelEncoder()
df['event_type_encoded'] = encoder.fit_transform(df['event_type'])

# Формування ознак
features = df[
    [
        'hour',
        'event_type_encoded',
        'severity',
        'response_time'
    ]
]

# ==========================
# 3. Навчання моделі
# ==========================
```

```
model = IsolationForest(
    n_estimators=100,
    contamination=0.05,
    random_state=42
)

model.fit(features)

# ==========================
# 4. Пошук аномалій
# ==========================

df['anomaly'] = model.predict(features)

# -1 = аномалія
#  1 = нормальна подія

anomalies = df[df['anomaly'] == -1]

print("\nВиявлені аномалії:")
print(anomalies)

# ==========================
# 5. Збереження результатів
# ==========================

anomalies.to_csv(
    "anomalies.csv",
    index=False,
    encoding="utf-8"
)

# ==========================
# 6. Візуалізація
# ==========================

plt.figure(figsize=(10, 6))

normal = df[df['anomaly'] == 1]
abnormal = df[df['anomaly'] == -1]

plt.scatter(
    normal.index,
    normal['response_time'],
```

```
    label='Нормальні події'
)

plt.scatter(
    abnormal.index,
    abnormal['response_time'],
    label='Аномалії'
)

plt.xlabel('Номер запису')
plt.ylabel('Час відповіді')
plt.title('Виявлення аномалій у журналах подій')
plt.legend()
plt.grid(True)

plt.show()
```