ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Харківський національний університет радіоелектроніки

Кафедра ЕОМ

Кваліфікаційна робота

Методи виявлення атак на комп'ютерну систему з використанням машинного навчання

Виконав:
ст. гр. СПм-23-5
Ященко О.М.

Керівник:
ас., к.т.н., Кравченко П.О.

Об'єкт дослідження та мета роботи

Метою кваліфікаційної роботи є розробка, обґрунтування та практична реалізація методу виявлення атак на комп'ютерну систему з використанням алгоритмів машинного навчання, що забезпечує підвищення точності, адаптивності та швидкості реагування систем інформаційної безпеки в умовах високої складності та динамічності сучасних мережевих середовищ.
Об'єктом дослідження є процеси моніторингу, аналізу та класифікації мережевого трафіку в корпоративних комп'ютерних системах для виявлення потенційно шкідливої активності та кіберзагроз.
Завдання:

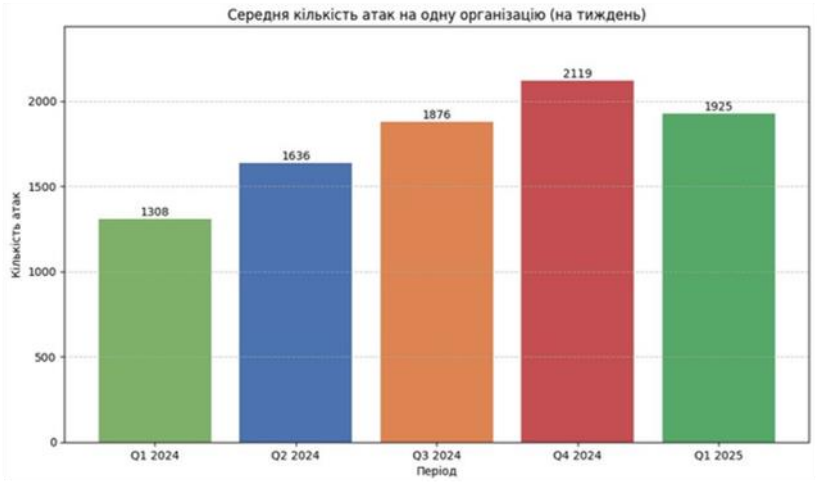
- здійснити аналіз сучасних методів виявлення атак, зокрема з урахуванням застосування інтелектуальних технологій;
- дослідити публічні датасети та визначити критерії для побудови навчальних і тестових вибірок;
- обґрунтувати вибір алгоритму машинного навчання, придатного для задач виявлення аномалій у мережевому трафіку;
- реалізувати запропонований метод в середовищі Google Colab із використанням інструментів Python;
- провести порівняльний аналіз ефективності розробленого методу відносно традиційних моделей за ключовими метриками (точність, повнота, F1, AUC);

Методи обробки та аналізу даних в корпоративних мережах

3

Кількість атак на одну організацію за 2024/2025(1 квартал)

4



Методи виявлення кіберзагроз

5

Метод	Опис		
Сигнатурний	Виявлення загроз за відомими шаблонами атак (сигнатурами). Ефективний для відомих загроз.		
Евристичний	Аналіз підозрілих ознак без повної відповідності шаблонам. Підходить для нових варіацій атак.		
Поведінковий	Виявлення відхилень від нормальної поведінки користувача або системи.		
Машинне навчання	Автоматичне навчання на великих обсягах даних для побудови моделей виявлення аномалій.		
Штучний інтелект	Використання гібридних, самонавчальних моделей, що можуть прогнозувати загрози.		
Інтегровані системи (IDPS)	Комбінування кількох методів (сигнатурний, поведінковий, ML) у комплексній системі.		
Метод	**Типові приклади/алгоритми**	**Переваги**	**Недоліки**
Сигнатурний	Антивірусні бази, IDS типу Snort	Висока точність для відомих атак, швидкість	Не виявляє невідомі атаки
Евристичний	Аналіз дій програм, Sandbox-інструменти	Можливість виявити нові загрози	Хибнопозитивні спрацювання
Поведінковий	UEBA, аналіз профілю активності користувача	Висока ефективність проти цільових атак	Потребує багато історичних даних
Машинне навчання	Decision Trees, Random Forest, SVM, K-Means	Адаптивність, робота з big data	Складність налаштування, ресурсомісткість
Штучний інтелект	Deep Learning, Reinforcement Learning, AutoML	Прогнозування, автоматизація	Потреба у великих обсягах даних, ресурси
Інтегровані системи (IDPS)	Suricata, Zeek, Snort + SIEM інтеграція	Збалансованість, гнучкість	Складність інтеграції, вартість

Порівняльний аналіз систем виявлення атак

6

Система	Функціонал
Snort	Мережевий аналіз трафіку, виявлення атак за сигнатурами. Підтримує реальний час, логування, обробку пакетів
Suricata	Високошвидкісна IDS/IPS, багатопоточна обробка, підтримка сигнатур Snort, DPI, TLS-аналіз, багатоформатне логування
Bro (Zeek)	Мережева IDS з аналітичним ухилом: обробка логіки сесій, сценарний аналіз, поведінкова аналітика
OSSEC	HIDS - виявлення вторгнень на основі аналізу логів, файлів, реєстру, rootkit'ів, підтримка централізованого моніторингу
Prelude	Гібридна система IDS/IPS з підтримкою агентів, логуванням, кореляцією подій, інтеграцією з іншими системами
Система	**Переваги**
Snort	Простота конфігурації, широке поширення, активна спільнота
Suricata	Висока продуктивність, розширені можливості DPI, багатопоточність
Bro (Zeek)	Глибока аналітика, сценарне виявлення, висока гнучкість
OSSEC	Комплексний підхід до моніторингу хостів, відкритий код, підтримка rootkit-аналізу
Prelude	Потужна архітектура, сумісність з іншими IDS, розширена кореляція
Система	**Застосування**
Snort	Корпоративні мережі, захист серверів, моніторинг периметра мережі
Suricata	Потужні мережеві середовища, провайдери, дата-центри, критичні інфраструктури
Bro (Zeek)	Аналітичні центри безпеки, дослідницькі установи, поведінкова аналітика трафіку
OSSEC	Сервери, робочі станції, середовища з потребою в HIDS, відповідність PCI-DSS
Prelude	Організації з потребою централізованого аналізу, інтеграція з зовнішніми джерелами
Система	**Недоліки**
Snort	Лише однониткова обробка, складність з великою кількістю трафіку
Suricata	Більша складність у налаштуванні, вища вимога до ресурсів
Bro (Zeek)	Високий поріг входження, потреба в налаштуванні сценаріїв
OSSEC	Лише HIDS, обмеженість у мережевому аналізі
Prelude	Складність конфігурації, потреба в глибокому розумінні архітектури

7

Розроблений метод виявлення атак

8

Програмна реалізація

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, f1_score, roc_curve, auc
from collections import Counter
from imblearn.over_sampling import SMOTE
```

```
data = pd.read_csv('/content/Wednesday-workingHours.pcap_ISCX.csv')
print(data.shape)
data.head()
```

```
(145755, 79)
```

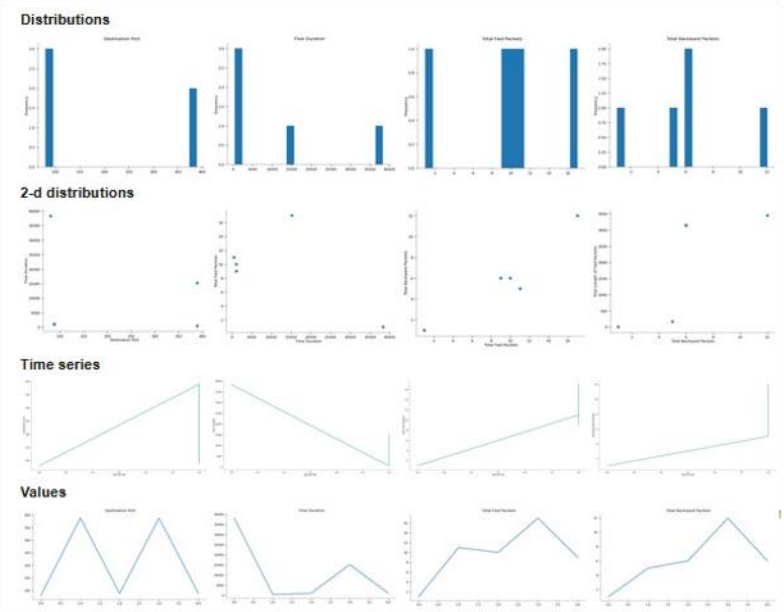
index	Destination Port	Flow Duration	Total Fwd Packets	Total Backward Packets	Total Length of Fwd Packets	Total
0	80	38308	1	1	6	
1	389	479	11	5	172	
2	88	1095	10	6	3150	
3	389	15206	17	12	3452	
4	88	1092	9	6	3150	

Блоки файлу .ipynb

- ❖ імпорт бібліотек;
- ❖ очищення та підготовка даних;
- ❖ масштабування ознак;
- ❖ балансування вибірки;
- ❖ розділення вибірки;
- ❖ навчання моделі;
- ❖ прогнозування та тестування;
- ❖ візуалізація результатів.

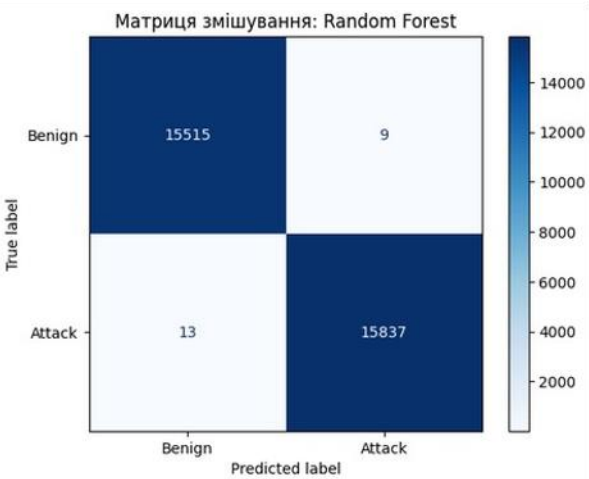
9

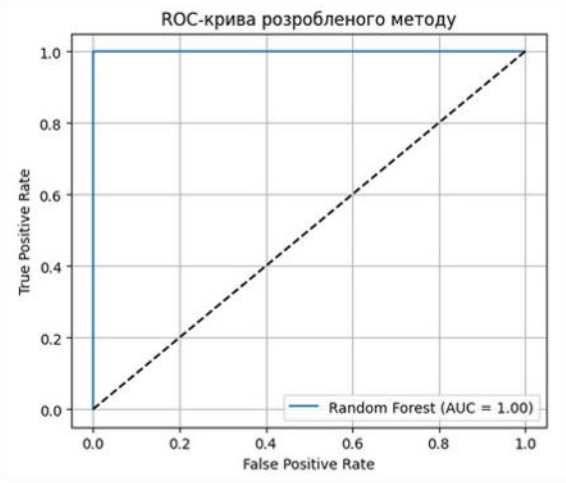
Візуалізація обраних ознак мережевого трафіку



10

Матриця плутанини та ROC-крива



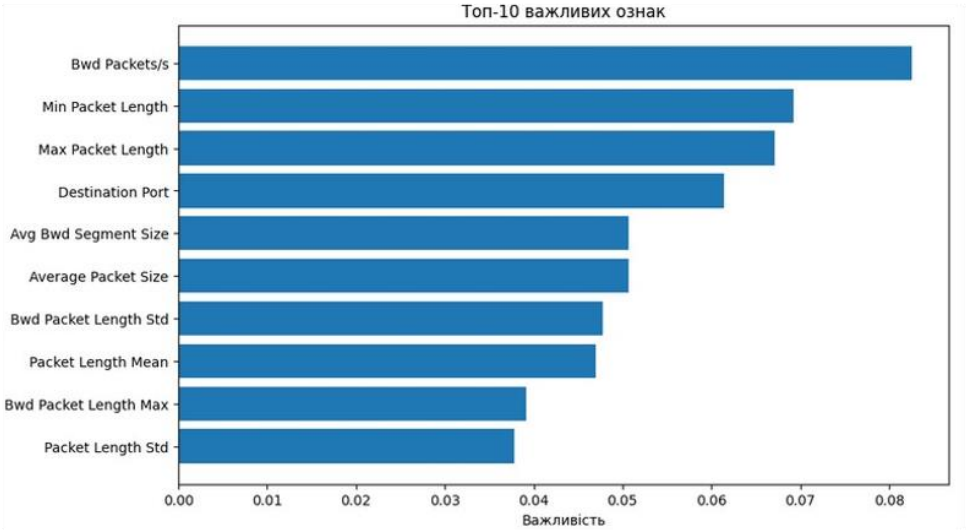


11

Обрані важливі ознаки

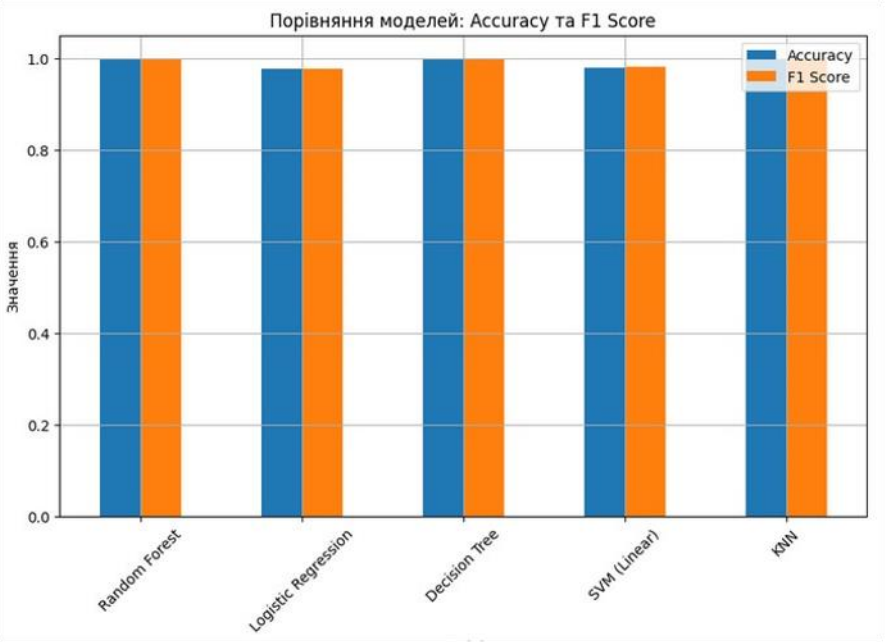
```
importances = model.feature_importances_
indices = np.argsort(importances)[-10:]  # топ-10 ознак

plt.figure(figsize=(10,6))
plt.barh(range(len(indices)), importances[indices], align='center')
plt.yticks(range(len(indices)), [X.columns[i] for i in indices])
plt.title('Топ-10 важливих ознак')
plt.xlabel('Важливість')
plt.show()
```



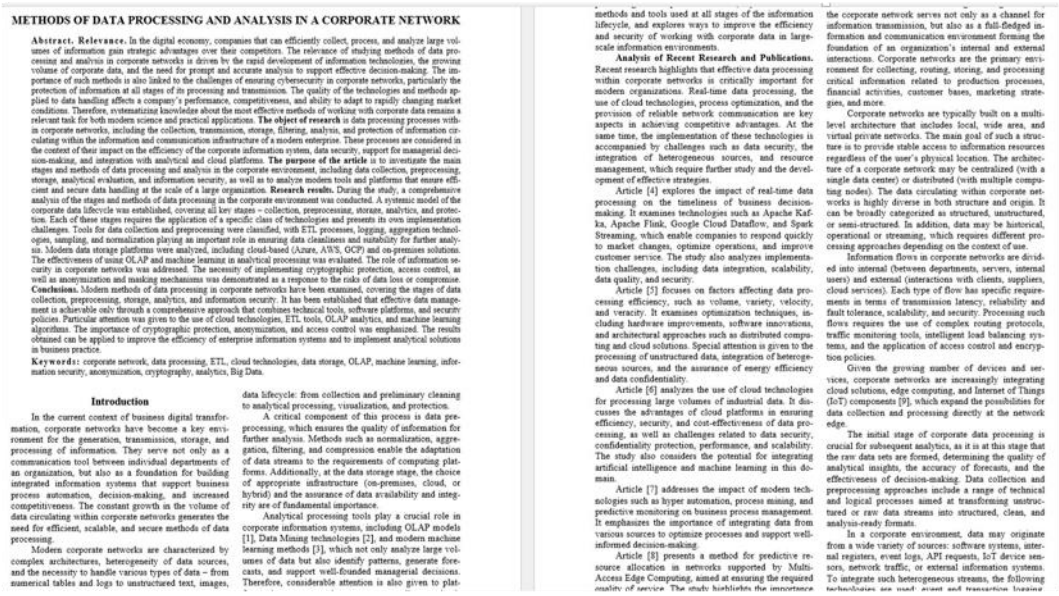
Порівняння існуючих моделей з розробленим методом

12



16

Апробація



METHODS OF DATA PROCESSING AND ANALYSIS IN A CORPORATE NETWORK

Abstract. Relevance. In the digital economy, companies that can efficiently collect, process, and analyze large volumes of information gain strategic advantages over their competitors. The relevance of studying methods of data processing and analysis in corporate networks is driven by the rapid development of information technologies, the growing volume of corporate data, and the need for prompt and accurate analysis to support effective decision-making. The importance of such methods is also linked to the challenges of ensuring cybersecurity in corporate networks, particularly the protection of information at all stages of its processing and transmission. The quality of the technologies and methods applied to data handling affects a company's performance, competitiveness, and ability to adapt to rapidly changing market conditions. Therefore, systematizing knowledge about the most effective methods of working with corporate data remains a relevant task for both modern science and practical applications. **The object of research** is data processing processes within corporate networks, including the collection, transmission, storage, filtering, analysis, and protection of information circulating within the information and communication infrastructure of a modern enterprise. These processes are considered in the context of their impact on the efficiency of the corporate information system, data security, support for managerial decision-making, and integration with analytical and cloud platforms. **The purpose of the article** is to investigate the main stages and methods of data processing and analysis in the corporate environment, including data collection, preprocessing, storage, analytical evaluation, and information security, as well as to analyze modern tools and platforms that ensure efficient and secure data handling at the scale of a large organization. **Research results.** During the study, a comprehensive analysis of the stages and methods of data processing in the corporate environment was conducted. A systemic model of the corporate data lifecycle was established, covering all key stages – collection, preprocessing, storage, analytics, and protection. Each of these stages requires the application of a specific class of technologies and presents its own implementation challenges. Tools for data collection and preprocessing were classified, with ETL processes, logging, aggregation technologies, sampling, and normalization playing an important role in ensuring data cleanliness and suitability for further analysis. Modern data storage platforms were analyzed, including cloud-based (Azure, AWS, GCP) and on-premises solutions. The effectiveness of using OLAP and machine learning in analytical processing was evaluated. The role of information security in corporate networks was addressed. The necessity of implementing cryptographic protection, access control, as well as anonymization and masking mechanisms was demonstrated as a response to the risks of data loss or compromise. **Conclusions.** Modern methods of data processing in corporate networks have been examined, covering the stages of data collection, preprocessing, storage, analytics, and information security. It has been established that effective data management is achievable only through a comprehensive approach that combines technical tools, software platforms, and security policies. Particular attention was given to the use of cloud technologies, ETL tools, OLAP analytics, and machine learning algorithms. The importance of cryptographic protection, anonymization, and access control was emphasized. The results obtained can be applied to improve the efficiency of enterprise information systems and to implement analytical solutions in business practice.

Keywords: corporate network; data processing, ETL, cloud technologies, data storage, OLAP, machine learning, information security, anonymization, cryptography, analytics, Big Data.

Introduction

In the current context of business digital transformation, corporate networks have become a key environment for the generation, transmission, storage, and processing of information. They serve not only as a communication tool between individual departments of an organization, but also as a foundation for building integrated information systems that support business process automation, decision-making, and increased competitiveness. The constant growth in the volume of data circulating within corporate networks generates the need for efficient, scalable, and secure methods of data processing.

Modern corporate networks are characterized by complex architectures, heterogeneity of data sources, and the necessity to handle various types of data – from numerical tables and logs to unstructured text, images, data lifecycle: from collection and preliminary cleaning to analytical processing, visualization, and protection.

A critical component of this process is data preprocessing, which ensures the quality of information for further analysis. Methods such as normalization, aggregation, filtering, and compression enable the adaptation of data streams to the requirements of computing platforms. Additionally, at the data storage stage, the choice of appropriate infrastructure (on-premises, cloud, or hybrid) and the assurance of data availability and integrity are of fundamental importance.

Analytical processing tools play a crucial role in corporate information systems, including OLAP models [1], Data Mining technologies [2], and modern machine learning methods [3], which not only analyze large volumes of data but also identify patterns, generate forecasts, and support well-founded managerial decisions. Therefore, considerable attention is also given to platmethods and tools used at all stages of the information lifecycle, and explores ways to improve the efficiency and security of working with corporate data in large-scale information environments.

Analysis of Recent Research and Publications. Recent research highlights that effective data processing within corporate networks is critically important for modern organizations. Real-time data processing, the use of cloud technologies, process optimization, and the provision of reliable network communication are key aspects in achieving competitive advantages. At the same time, the implementation of these technologies is accompanied by challenges such as data security, the integration of heterogeneous sources, and resource management, which require further study and the development of effective strategies.

Article [4] explores the impact of real-time data processing on the timeliness of business decision-making. It examines technologies such as Apache Kafka, Apache Flink, Google Cloud Dataflow, and Spark Streaming, which enable companies to respond quickly to market changes, optimize operations, and improve customer service. The study also analyzes implementation challenges, including data integration, scalability, data quality, and security.

Article [5] focuses on factors affecting data processing efficiency, such as volume, variety, velocity, and veracity. It examines optimization techniques, including hardware improvements, software innovations, and architectural approaches such as distributed computing and cloud solutions. Special attention is given to the processing of unstructured data, integration of heterogeneous sources, and the assurance of energy efficiency and data confidentiality.

Article [6] analyzes the use of cloud technologies for processing large volumes of industrial data. It discusses the advantages of cloud platforms in ensuring efficiency, security, and cost-effectiveness of data processing, as well as challenges related to data security, confidentiality protection, performance, and scalability. The study also considers the potential for integrating artificial intelligence and machine learning in this domain.

Article [7] addresses the impact of modern technologies such as hyper automation, process mining, and predictive monitoring on business process management. It emphasizes the importance of integrating data from various sources to optimize processes and support well-informed decision-making.

Article [8] presents a method for predictive resource allocation in networks supported by Multi-Access Edge Computing, aimed at ensuring the required quality of service. The study highlights the importance the corporate network serves not only as a channel for information transmission, but also as a full-fledged information and communication environment forming the foundation of an organization's internal and external interactions. Corporate networks are the primary environment for collecting, routing, storing, and processing critical information related to production processes, financial activities, customer bases, marketing strategies, and more.

Corporate networks are typically built on a multi-level architecture that includes local, wide area, and virtual private networks. The main goal of such a structure is to provide stable access to information resources regardless of the user's physical location. The architecture of a corporate network may be centralized (with a single data center) or distributed (with multiple computing nodes). The data circulating within corporate networks is highly diverse in both structure and origin. It can be broadly categorized as structured, unstructured, or semi-structured. In addition, data may be historical, operational or streaming, which requires different processing approaches depending on the context of use.

Information flows in corporate networks are divided into internal (between departments, servers, internal users) and external (interactions with clients, suppliers, cloud services). Each type of flow has specific requirements in terms of transmission latency, reliability and fault tolerance, scalability, and security. Processing such flows requires the use of complex routing protocols, traffic monitoring tools, intelligent load balancing systems, and the application of access control and encryption policies.

Given the growing number of devices and services, corporate networks are increasingly integrating cloud solutions, edge computing, and Internet of Things (IoT) components [9], which expand the possibilities for data collection and processing directly at the network edge.

The initial stage of corporate data processing is crucial for subsequent analytics, as it is at this stage that the raw data sets are formed, determining the quality of analytical insights, the accuracy of forecasts, and the effectiveness of decision-making. Data collection and preprocessing approaches include a range of technical and logical processes aimed at transforming unstructured or raw data streams into structured, clean, and analysis-ready formats.

In a corporate environment, data may originate from a wide variety of sources: software systems, internal registers, event logs, API requests, IoT device sensors, network traffic, or external information systems. To integrate such heterogeneous streams, the following

Rossikhin V., Tarapata Y., Iashchenko O. Methods of data processing and analysis in acorporate network. Системи управління, навігації та зв'язку, вип.3. Полтава, 2025. С. 171-176.

Висновки

14

У процесі виконання кваліфікаційної роботи було здійснено комплексне дослідження підходів до виявлення атак у комп'ютерних системах з використанням сучасних технологій машинного навчання. Теоретичний аналіз показав, що класичні інструменти систем виявлення вторгнень, такі як Snort, Suricata, Bro та інші, потребують доповнення інтелектуальними алгоритмами, здатними адаптивно реагувати на нові типи загроз і виявляти нетипову поведінку в реальному часі.

У межах роботи було обґрунтовано вибір методу машинного навчання на основі алгоритму Random Forest як основного для побудови моделі класифікації мережевого трафіку. Проведено збір, очищення, аналіз і попередню обробку даних на основі реального набору CICIDS2017, із використанням сегменту Wednesday-workingHours.pcap_ISCX.csv. Особливу увагу було приділено балансуванню вибірки за допомогою SMOTE та масштабуванню ознак для покращення узгодженості моделей.

Розроблений метод реалізовано в середовищі Google Colab із використанням Python-бібліотек scikit-learn, pandas, matplotlib, seaborn, а також інструментів для автоматизованого візуального аналізу. Проведене експериментальне тестування продемонструвало виняткову ефективність моделі: показник точності та F1-міра перевищили 0.99, а площа під ROC-кривою досягла 1.00. Порівняльний аналіз з іншими популярними класифікаторами (Logistic Regression, Decision Tree, SVM, KNN) підтвердив перевагу розробленого методу за всіма ключовими метриками.

Візуалізація результатів, включаючи матрицю змішування, ROC-криві, графіки важливості ознак та порівняння моделей, дозволила глибше інтерпретувати поведінку моделі та її здатність узагальнювати залежності у вхідних даних. Дослідження підтвердило, що метод на основі Random Forest є не лише ефективним, але й придатним для практичного впровадження в автоматизовані системи кіберзахисту з можливістю подальшого масштабування та інтеграції з хмарною інфраструктурою.

ДОДАТОК Б

Програмний код

Б.1 Встановлення бібліотек та завантаження датасету

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report,
confusion_matrix
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, f1_score, roc_curve,
auc
from collections import Counter
from imblearn.over_sampling import SMOTE

data = pd.read_csv('/content/Wednesday-
workingHours.pcap_ISCX.csv')
print(data.shape)
data.head()
```

Б.2 Очищення даних та кодування міток класів

```
# Видаляємо нульові або пусті значення
data.replace([np.inf, -np.inf], np.nan, inplace=True)
data.dropna(inplace=True)

# Видалимо непотрібні текстові або ідентифікаційні поля
cols_to_drop = ['Flow ID', 'Source IP', 'Destination IP',
'Timestamp']
data.drop(columns=cols_to_drop, inplace=True, errors='ignore')

data['Label'] = data['Label'].apply(lambda x: 0 if x == 'BENIGN'
else 1)
```

Б.3 Реалізація методу та аналіз результатів

```
data.columns = data.columns.str.strip()
data.drop(columns=['Flow ID', 'Source IP', 'Destination IP',
'Timestamp'], inplace=True, errors='ignore')
data.replace([np.inf, -np.inf], np.nan, inplace=True)
data.dropna(inplace=True)

# Переконуємось у правильній назві стовпця міток
data['Label'] = data['Label'].apply(lambda x: 0 if x == 'BENIGN'
else 1)

X = data.drop('Label', axis=1)
y = data['Label']
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)
class_counts = Counter(y)
print("Кількість зразків у кожному класі:", class_counts)

minority_class_count = min(class_counts.values())
if minority_class_count > 1:
    k_neighbors = min(5, minority_class_count - 1)
    smote = SMOTE(k_neighbors=k_neighbors, random_state=42)
    X_bal, y_bal = smote.fit_resample(X_scaled, y)
    print("SMOTE застосовано:", Counter(y_bal))
else:
    print("SMOTE не застосовано — замало зразків меншості.")
    X_bal, y_bal = X_scaled, y
X_train, X_test, y_train, y_test = train_test_split(X_bal,
y_bal, test_size=0.2, random_state=42)
models = {
    'Random Forest': RandomForestClassifier(n_estimators=100,
random_state=42),
    'Logistic Regression': LogisticRegression(max_iter=1000),
    'Decision Tree': DecisionTreeClassifier(),
    'SVM (Linear)': SVC(kernel='linear'),
    'KNN': KNeighborsClassifier()
}

results = []
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    acc = accuracy_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
    results.append({'Model': name, 'Accuracy': acc, 'F1 Score':
f1})

results_df = pd.DataFrame(results)
results_df.set_index('Model').plot(kind='bar', figsize=(10,6))
plt.title('Порівняння моделей: Accuracy та F1 Score')
plt.ylabel('Значення')
```

```
plt.grid(True)
plt.xticks(rotation=45)
plt.show()
rf_model = RandomForestClassifier(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)
y_proba = rf_model.predict_proba(X_test)[:,1]

fpr, tpr, _ = roc_curve(y_test, y_proba)
roc_auc = auc(fpr, tpr)

plt.figure(figsize=(6, 5))
plt.plot(fpr, tpr, label=f'ROC-крива (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC-крива Random Forest')
plt.legend(loc='lower right')
plt.grid(True)
plt.show()
from sklearn.metrics import ConfusionMatrixDisplay

rf_model = RandomForestClassifier(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)
y_pred_rf = rf_model.predict(X_test)

ConfusionMatrixDisplay.from_estimator(rf_model, X_test, y_test,
display_labels=['Benign', 'Attack'], cmap='Blues')
plt.title("Матриця змішування: Random Forest")
plt.grid(False)
plt.show()

y_proba_rf = rf_model.predict_proba(X_test)[:, 1]
fpr_rf, tpr_rf, _ = roc_curve(y_test, y_proba_rf)
roc_auc_rf = auc(fpr_rf, tpr_rf)

plt.figure(figsize=(6,5))
plt.plot(fpr_rf, tpr_rf, label=f'Random Forest (AUC =
{roc_auc_rf:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC-крива розробленого методу')
plt.legend(loc='lower right')
plt.grid(True)
plt.show()
```