applied sciences

MDPI

Article

AI-Based RPA's Work Automation Operation to Respond to Hacking Threats Using Collected Threat Logs

Joosung Kim [1], Soo Hyun Kim [2] and Inwhee Joe [1,*]

[1] Department of Computer Science, Hanyang University, Seoul 04763, Republic of Korea; gronkoutt@gmail.com
[2] Department of Mathematics, Pusan National University, Busan 43241, Republic of Korea; 5hkim@pusan.ac.kr
* Correspondence: iwjoe@hanyang.ac.kr

Abstract: With the rapid acceleration of the Fourth Industrial Revolution, cyber threats have become increasingly frequent and complex. However, most public and private institutions still rely heavily on manual cybersecurity operations, which often lead to delayed responses and human errors, exposing critical vulnerabilities. A particular challenge lies in the inability to efficiently integrate and automate the analysis of threat logs collected from various sources, limiting the effectiveness of threat prediction and mitigation. To address these challenges, this study proposes an AI-based RPA (Robotic Process Automation) system designed to automate the collection, analysis, and dissemination of cyber threat logs. By minimizing human intervention, the proposed system significantly enhances real-time response capabilities and reduces errors. Additionally, standardizing and centralizing diverse log formats lays a foundation for the future development of AI models capable of predicting cyberattack patterns. This system is particularly well-suited for government and public organizations, offering a cost-effective solution that enhances cybersecurity while maintaining compatibility with existing infrastructures. The experimental results demonstrate that the proposed AI-based RPA system outperforms traditional manual systems in terms of log processing speed, prediction accuracy, and error reduction. This study highlights the critical role of automated AI-driven systems in enabling real-time threat response and prevention, presenting a practical and scalable approach for modern cybersecurity environments.

Keywords: fourth industrial revolution era; government hacking response policy; AI; RPA; hacking threat response process

Citation: Kim, J.; Kim, S.H.; Joe, I. AI-Based RPA's Work Automation Operation to Respond to Hacking Threats Using Collected Threat Logs. *Appl. Sci.* **2024**, *14*, 10217. https://doi.org/10.3390/app142210217

Academic Editor: Stefan Fischer

Received: 5 October 2024
Revised: 23 October 2024
Accepted: 5 November 2024
Published: 7 November 2024

1. Introduction

1.1. Theoretical Background

As the adoption of new technologies from the Fourth Industrial Revolution accelerates, the frequency and complexity of cyberattacks are continuously increasing [1]. In addition to traditional attacks such as ransomware and DDoS, new forms of attacks have emerged, including AI-powered automated attacks, exploitation of IoT vulnerabilities, and APT (Advanced Persistent Threat) attacks based on social engineering tactics. These attacks pose significant threats to the critical information systems of both corporations and government institutions [2]. Despite these growing challenges, existing security operations still heavily rely on manual processes, leading to delays in response and human errors, which in turn, become critical vulnerabilities and weak points in the system. These weaknesses provide attackers with optimal entry points, enabling more sophisticated and devastating cyberattacks. Consequently, such vulnerabilities can result in the leakage of key information assets or cause significant damage to the systems of corporations and government agencies. The inability of traditional operational systems to respond effectively to the rapidly changing digital environment has become a major cause of declining trust in overall security systems. Therefore, there is an urgent need for an automated operational framework to address these vulnerabilities.

1.2. Challenges of Manual Cybersecurity Processes

Traditionally, cybersecurity involves several steps, such as log data collection, threat analysis, security alert transmission, false positive elimination, and threat reporting [3]. However, most of these processes are performed manually, which limits both the speed and accuracy of responses. Manual methods also depend heavily on the experience and expertise of analysts, increasing the risk of human errors when handling large volumes of log data and complex security events. Such errors, including misinterpretation of logs, may result in serious security threats being undetected or discovered too late, leading to significant damage. Furthermore, manual systems lack the scalability and flexibility required to adapt to the rapidly evolving digital environment, making it difficult to address the growing complexity of security threats effectively [4].

1.3. Automation's Impact on Cybersecurity

The integration of AI in the field of cybersecurity provides innovative solutions that significantly enhance traditional security systems. AI technology not only detects security threats but also predicts and proactively responds to potential risks, establishing a new paradigm for effectively mitigating cyber threats [5]. Moreover, automating the processing of security threat logs in cyber environments drastically reduces processing times and eliminates delays in threat information sharing caused by human error [6]. However, government institutions aiming to implement such systems often require parliamentary approval [7], making RPA an attractive alternative that meets technical requirements while offering cost-saving benefits.

RPA (Robotic Process Automation) [8] which automates repetitive and rule-based tasks traditionally performed by humans through software robots is highly effective at automating the processing, filtering, and analysis of pre-collected security logs, allowing security teams to concentrate on threat detection. Manual systems require human intervention at multiple stages, whereas RPA automates these processes, enhancing efficiency and reducing costs. Additionally, AI excels at real-time analysis of large volumes of log data, predicting new threats, and detecting abnormal activities. By leveraging machine learning and deep learning algorithms, AI can learn threat patterns and detect cyberattacks early, making it a crucial component of future security systems [9,10]. However, since different government departments and agencies generate security logs in various formats, there is a need to standardize these logs [11]. The collection of threat logs must accompany this effort. Consequently, the adoption of AI and RPA [12] is becoming a fundamental component of modern cybersecurity systems [13]. By doing so, companies and government institutions will be able to respond more swiftly and efficiently to the rapidly changing threat landscape. Future security systems must be built around automated processes centered on AI and RPA to enable real-time responses and prevention of cyberattacks.

1.4. Research Direction and Objectives

This study aims to propose a business process and model that can be immediately applied in practice by actively reflecting the characteristics of government and intelligence agencies, which require easy and low-cost integration without changing existing systems. To achieve this, the study seeks to identify weak points in the current business processes and propose improved ones. Furthermore, it aims to introduce a model that leverages RPA [14] to maintain existing systems while allowing for rapid and accurate responses compared to manual-based security systems, with an emphasis on cost savings. Additionally, to provide scalability, the study intends to standardize unstructured data collected from various systems and accumulate these data for the future development of more robust cybersecurity models.

2. Research Background

2.1. Increasing Cybersecurity Threats

With the advent of the Fourth Industrial Revolution, the world is undergoing rapid digital transformation, and as a result, cybersecurity threats are increasing rapidly [15]. In addition to traditional attack methods such as ransomware, APT (Advanced Persistent Threat), and DDoS, there are now intelligent cyberattacks that exploit advancements in AI and Internet of Things (IoT) technologies, posing serious threats to the information systems of nations, public institutions, and private enterprises. These attacks are becoming more sophisticated and complex, challenging the effectiveness of current security solutions [16]. Today, the frequency and destructive power of cyberattacks have reached levels that existing security systems struggle to handle. Consequently, responses to these attacks now demand not only simple defense mechanisms but also real-time response and prevention. In this changing environment, the limitations of traditional, manual-based security systems are becoming increasingly evident.

2.2. Limitations of Existing Security Systems

Most of the security systems currently in use by many organizations rely heavily on manual processes. The detection of threats and the analysis of security log data are largely dependent on the experience and subjective judgment of security practitioners, which introduces several issues. First, the response speed is exceedingly slow. While cyberattacks can damage systems or exfiltrate data in a very short amount of time, manual responses cannot guarantee immediate action. Second, the likelihood of human error is high. During manual data processing, security threats may go undetected or false alerts may be triggered due to human error, potentially leading to serious security incidents. Third, there are challenges in processing unstructured data. Log data generated by various security solutions often come in different formats, making it difficult to integrate and analyze them consistently. These issues hinder immediate responses to cyber threats and accurate detection of security risks, particularly in the case of complex and large-scale cyberattacks, leading to a degradation in response capabilities. It has become clear that relying solely on manual methods is no longer an effective way to handle cybersecurity.

2.3. Overcoming Cybersecurity Challenges with RPA and AI

To overcome the limitations of existing security systems, it is essential to adopt Robotic Process Automation (RPA) and Artificial Intelligence technologies. Traditional, manual security frameworks often require significant human intervention for repetitive and rule-based log processing tasks, which are both time-consuming and prone to human error. RPA addresses these issues by automating such tasks, minimizing human involvement, and significantly improving the speed of security response. Moreover, RPA can standardize and integrate log data generated by various security systems, thereby enhancing the efficiency and consistency of security analysis.

AI further complements this process by analyzing large-scale log data in real time, detecting emerging threats, and predicting future cyberattacks based on historical patterns. By applying time-series models like Long Short-Term Memory (LSTM), as shown in Figure 1, AI can predict recurring cyberattack patterns based on historical data, enabling proactive responses [17]. LSTM is particularly effective for this purpose because it can remember important past information for long periods, making it well-suited for predicting events based on time-series data. The integration of RPA and AI technologies, therefore, significantly enhances the overall capabilities of security systems by enabling real-time responses and accurate predictions of cyber threats.

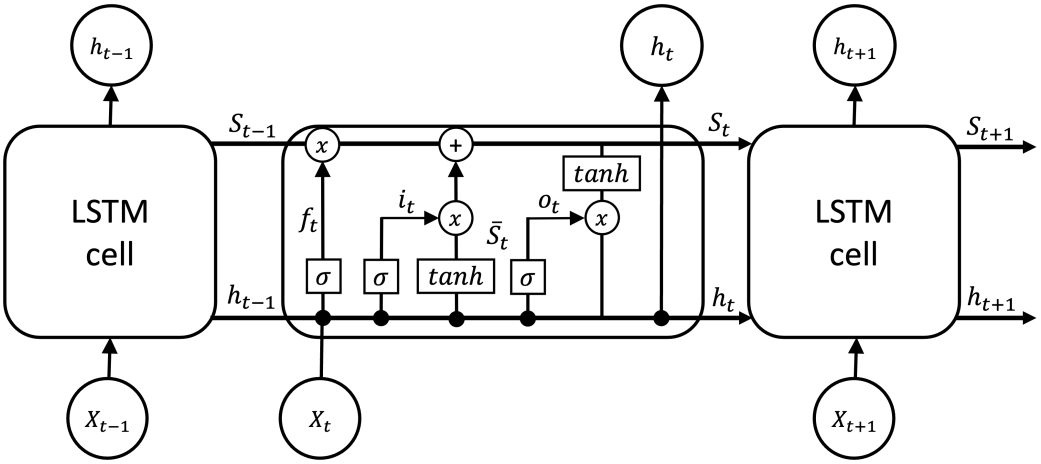

Figure 1. LSTM (Long Short-Term Memory).

The goal of this research is to develop an automated and intelligent cybersecurity response system by integrating RPA and AI, effectively overcoming the limitations of traditional manual security frameworks. The proposed system aims to minimize the involvement of security practitioners by automating repetitive tasks, enabling real-time data processing, and ensuring immediate responses to threats. Additionally, the system standardizes and integrates log data from various security solutions, improving the accuracy of data analysis and supporting the development of AI models capable of predicting future attack patterns. Notably, this solution offers a cost-effective, highly efficient security framework that can be implemented even in resource-constrained environments, such as government agencies, public institutions, and military facilities. The results of this research are expected to enhance the cybersecurity capabilities of both public and private sectors, enabling faster and more accurate responses in an increasingly dynamic digital landscape.

3. Methodology

3.1. Research Design

The objective of this study is to design and implement an automated cyber threat response system that reflects the unique characteristics of national institutions, public agencies, intelligence organizations, and military facilities. The proposed system is designed to be seamlessly integrated into existing infrastructure without modifications, thereby addressing the weaknesses inherent in current hacking threat response processes and standardizing log data to support diverse AI model training. Additionally, the system provides functionality for analyzing security log data and predicting future attack patterns using AI models.

In most national institutions, public agencies, intelligence organizations, and military facilities, the cyber threat response process typically detects security threats within each system as shown in Figure 2 and notifies the security administrator. The administrator then subjectively assesses the situation and relays the information to relevant agencies. While this approach may be effective for handling small-scale data, it becomes highly inefficient considering the volume and complexity of modern cyber threats. Furthermore, the limitations of human intervention in handling these threats may lead to critical consequences, and the data available for AI-based cybersecurity model development is restricted.

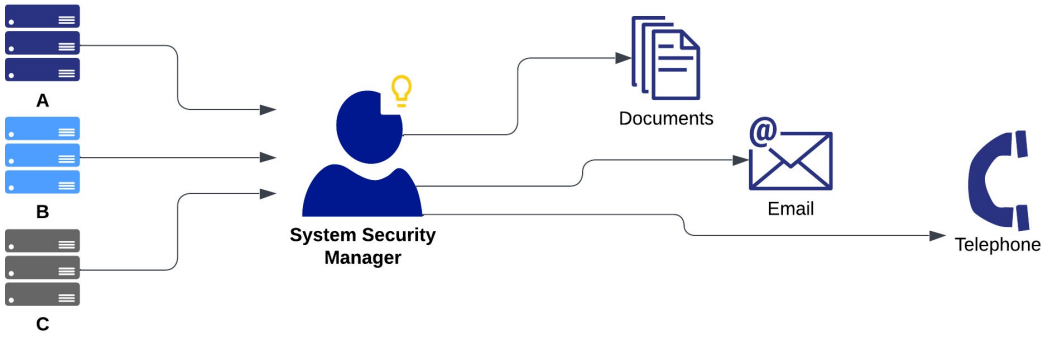

Figure 2. Existing manual system design (A: JSON, B: XML, C: CSV).

To address these issues, we propose an advanced cyber threat response process. As shown in Figure 3, humans play a crucial role in supervising and verifying the system. While the automated system is optimized for performing repetitive and rule-based tasks, human intervention is essential for handling complex cyber threats, responding to unexpected situations, and making ethical decisions. This human involvement complements the reliability and efficiency of the system, allowing it to effectively address exceptional scenarios that may be challenging for an automated system.

Moreover, the adoption of AI-based RPA reduces the need for certain manual tasks, which could potentially lead to workforce displacement in the cybersecurity field. However, this shift also creates opportunities for upskilling and reskilling, enabling cybersecurity professionals to transition into roles that require human expertise and judgment. By automating repetitive tasks, the system allows cybersecurity personnel to focus on more strategic and complex aspects of security operations, ensuring that limited cybersecurity resources are utilized efficiently and assigned to where they are most impactful.

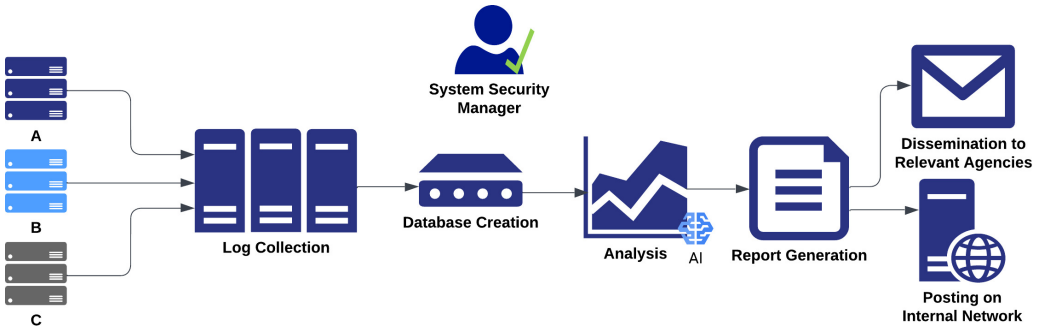

Figure 3. Proposed improved system design (A: JSON, B: XML, C: CSV).

To enhance implementation efficiency, this study limits the data formats to three representative types: JSON, XML, and CSV. This selection aims to reduce the complexity of data generated by various institutions and provide a standardized approach to integrating data, thereby creating an environment optimized for AI model training. This process demonstrates that data integration and standardization are critical factors for improving the performance of AI-based cybersecurity models.

The collected data are integrated into a single database in real time, facilitating the construction of an efficient database. This database is then used to predict current and future attack patterns through the Long Short-Term Memory (LSTM) model, which excels in time-series analysis. The results of these predictions are automatically compiled into reports, which are distributed to relevant institutions. Furthermore, the outcomes are posted on a server in real time, enhancing inter-agency collaboration and response capabilities.

In the past, national institutions, public agencies, intelligence organizations, and military facilities have pursued system integration for efficient management and the adoption of new solutions. However, these efforts often required large-scale projects that impacted the entire system and demanded substantial budget investments, making them challeng-

ing to implement. In contrast, the approach proposed in this study offers a significant advantage by addressing existing problems without necessitating changes to current systems. Additionally, it facilitates the generation of big data on cyber threat logs, which is essential for the upcoming AI era and serves as a valuable foundational resource for the advancement of AI-based cybersecurity technologies.

3.2. Data Collection and Preprocessing

The data used in this study were meticulously curated from logs generated by the CERT security team of government institutions over the past decade. This dataset, sampled for research purposes, focuses specifically on firewall systems, intrusion detection systems, and unauthorized access-blocking systems. To ensure a high-quality training process for the model, a total of 30,000 well-structured cybersecurity threat logs were utilized, providing a solid foundation for AI learning. The data files are available in JSON, XML, and CSV formats, containing critical information such as source IP addresses, destination IP addresses, and attack patterns.

The data preprocessing phase involved multiple steps to optimize the dataset for AI training. Initially, unnecessary information and outliers were removed to enhance model accuracy. Given the heterogeneous nature of the data collected from various security solutions, it was essential to standardize and convert these logs into CSV format to ensure efficiency and consistency in data management. The converted CSV files were subsequently transformed into *.db files using SQLite, thereby improving the efficiency of database management and enabling effective storage and accessibility. The advantages of this conversion process are summarized in Table 1.

Table 1. Comparison of database and CSV file characteristics.

Category	Database	CSV File
Data Integrity and Consistency	Constraints (primary keys, foreign keys, etc.) can be set	Constraints cannot be applied
Efficient Data Search	Fast search through indexing is possible	Requires reading the entire file, which is slow for large datasets
Data Consistency	Atomicity is guaranteed through transaction management	Conflicts may occur during concurrent modifications
Representation of Data Relationships	Various relationships between tables can be defined	Difficult to manage when data structure becomes complex
Handling Large Data Sets	Efficient processing of large datasets is possible	Performance degrades as file size increases
Data Transformation and Analysis	Easily transformed and analyzed through SQL	Additional processing required
Security and Access Control	User-specific access permissions can be set	Weak security management
AI Training Data Preparation	Batch processing and automation are possible	Automation is difficult as the file must be read each time

Particularly, as illustrated in Figure 4, we developed a system that standardizes and automates the real-time collection and preprocessing of log data from multiple security solutions. This system focuses on enhancing data quality by performing a series of data refinement tasks, such as removing duplicate records, deleting unnecessary columns, and handling missing values. By eliminating redundancy in the dataset, the overall data volume was reduced while maintaining the reliability of the information, thus maximizing the efficiency of the model training process. Handling missing values was also crucial to ensuring the completeness of the data, thereby minimizing inaccuracies during the learning phase.

```python
def start_monitoring():
    event_handler = FileHandler()
    observer = Observer()
    observer.schedule(event_handler, path=folder_path, recursive=False)
    observer.start()
    try:
        while True:
            pass
    except KeyboardInterrupt:
        observer.stop()
    observer.join()

if __name__ == "__main__":
    print("Starting folder monitoring...")
    start_monitoring()
```

Figure 4. Monitoring for real-time collection.

The system is designed to collect log data in real-time from different security solutions, ensuring that the latest threat information is always available for analysis. This real-time data collection process is critical for the timely detection and response to cybersecurity threats, allowing the system to immediately capture and process emerging threats.

As shown in Figure 5 the data preprocessing also involved encoding key features to prepare the data for effective AI training. For example, the source and destination IP addresses, as well as attack patterns, were transformed into numerical representations using LabelEncoder. This transformation was instrumental in enhancing the model's ability to learn from the data, as numerical values are better suited for most machine learning algorithms. Moreover, the encoded features were normalized using MinMaxScaler, ensuring that all values remained within a consistent range. This normalization step improved the convergence speed of the learning algorithms and prevented bias towards specific value ranges.

```python
from sklearn.preprocessing import LabelEncoder, MinMaxScaler
from watchdog.observers import Observer
from watchdog.events import FileSystemEventHandler

dfs = []
folder_path = 'C:/Users/gronk/OneDrive/Desktop/CyberAttackLog/DB'
save_path = 'C:/Users/gronk/OneDrive/Desktop/CyberAttackLog/DBC'

def preprocess_and_save(df_cleaned):
    if not df_cleaned.empty:
        df_cleaned = df_cleaned.drop(columns=['UnnecessaryColumn1', 'UnnecessaryColumn2'], errors='ignore')
        df_cleaned = df_cleaned.drop_duplicates()
        df_cleaned = df_cleaned.dropna()
```

Figure 5. Feature encoding and normalization for enhanced model learning.

Through these processes of data refinement, encoding, and normalization, the consistency of the dataset was assured, thereby providing a reliable foundation for the security analysis models. These efforts significantly enhanced the accuracy and reliability of the cybersecurity threat analysis. The automation of data preprocessing and standardization also minimized errors and delays typically associated with manual processes, further improving the performance of AI-based analysis models. Ultimately, this automated system has shortened the data processing time, enabling real-time cybersecurity threat analysis and strengthening overall security response capabilities.

3.3. System Implementation

3.3.1. Automation of the Cyber Threat Response System

For the RPA tool, UiPath was used to collect and convert cyber threat logs into standardized data. These logs are analyzed in real-time, and reports are automatically generated and sent to the relevant departments. To implement this, the entire workflow was automated using UiPath, as shown in the following Figure 6. First, in the log collection automation stage, log data from each security system are automatically aggregated and collected in real time. The collected logs are then converted into CSV files, and a database is constructed. Subsequently, threats are automatically analyzed and disseminated along with the relevant reports.

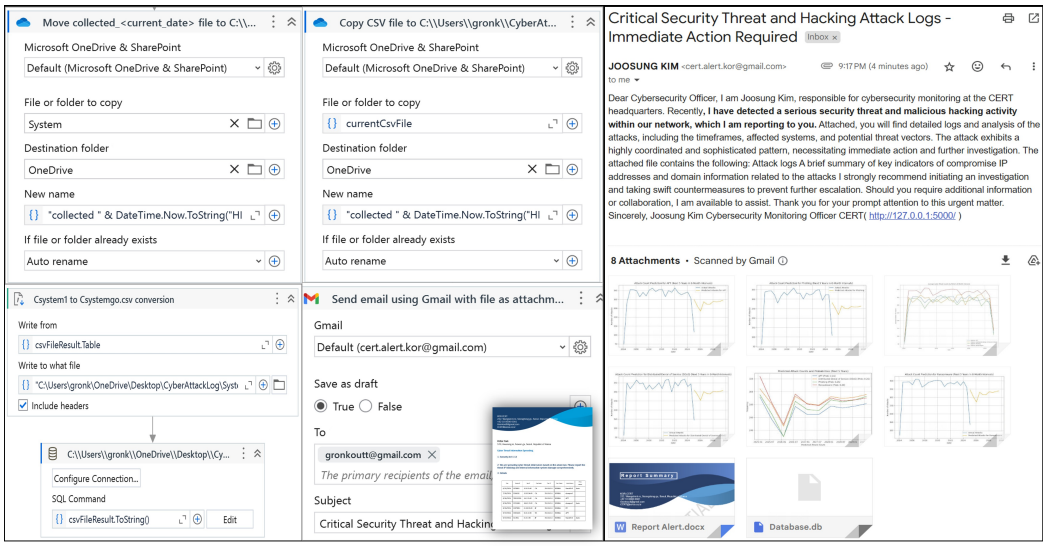

Figure 6. Manager who supervises and verifies the system.

3.3.2. Database Construction

Cybersecurity threat logs are stored in a database, as shown in the following Figure 7. This database supports automated processes such as cyberattack trend analysis, automatic report generation, and dissemination to relevant departments. The threat detection results and corresponding response actions are stored in the database, which can later be utilized for attack origin analysis, security enhancement, and AI learning.

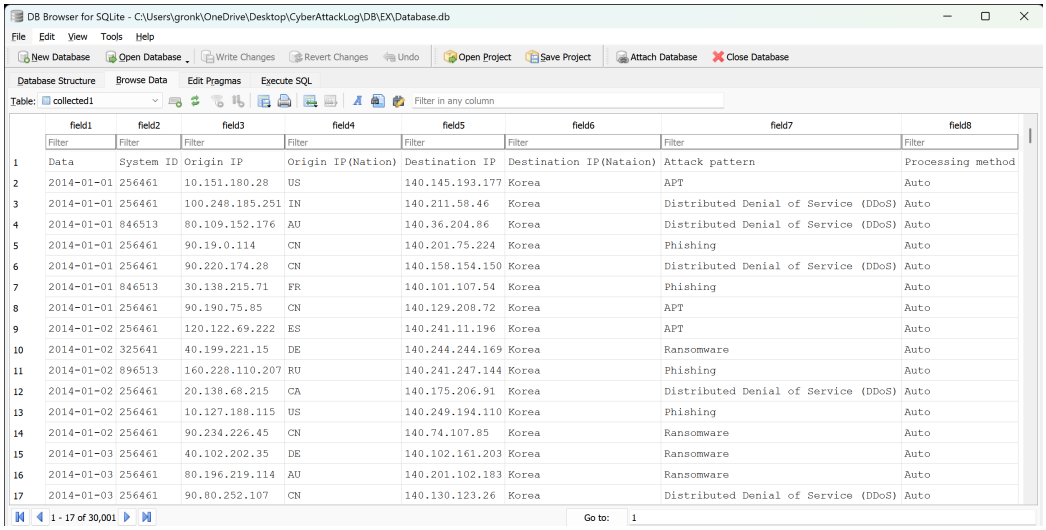

Figure 7. Manager who supervises and verifies the system.

3.3.3. Cybersecurity Threat Status and Prediction Model Development

In this study, considering the time-series characteristics of cyber threat logs and the advantages outlined in Table 2, we utilized the widely known and highly effective LSTM (Long Short-Term Memory) model [18] to predict cyberattack patterns. A process was developed to leverage time-series analysis on the data extracted from the database. Through this, the security response system was improved by predicting potential future attack patterns based on historical data.

Table 2. Advantages of LSTM.

Feature	Description
Handling Long-term Dependency	Remembers past information in time-series data, reflecting the characteristic that past attacks can influence future attacks.
Controlling Information Flow	Controls what information to remember or forget through input, forget, and output gates, capturing significant patterns.
Non-linear Relationship Modeling	Effectively models complex patterns and relationships by handling non-linearity.
Robustness to Data Scarcity	Capability of learning even with small or noisy datasets, making it advantageous in diverse situations.
Processing Versatile Input	Able to integrate various inputs such as log data, network traffic, and user behavior data for learning.

First, as shown in Figure 8, the cyberattack log data were extracted from the database and utilized for analysis. SQLite3 and pandas were used to load the attack log data from the SQLite database. The data retrieved from the database were refined based on the date field and analyzed by grouping it according to attack patterns.

```
db_path = r'C:\Users\gronk\OneDrive\Desktop\CyberAttackLog\DB\EX\LSTM\Database.db'
save_path = r'C:\Users\gronk\OneDrive\Desktop\CyberAttackLog\DB\EX\Result'

if not os.path.exists(save_path):
    os.makedirs(save_path)
conn = sqlite3.connect(db_path)
data_query = "SELECT * FROM collected1;"
data = pd.read_sql(data_query, conn)
data['field1'] = pd.to_datetime(data['field1'], errors='coerce')
data = data.dropna(subset=['field1'])
```

Figure 8. Extracting cyber attack log data from the database.

Next, as shown in Figure 9, each attack pattern was grouped in 6-month intervals and visualized. Through this process, the temporal trends of each pattern were identified, and based on these trends, future attack patterns were predicted. The visualized results are automatically saved, and when sent to relevant organizations via email, the visualized data are included as an attachment.

```
pattern in attack_patterns:
pattern_data = data[data['field7'] == pattern]
pattern_data.set_index('field1', inplace=True)
resampled_data = pattern_data.resample('6M').size().reset_index(name='attack_count')
scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(resampled_data['attack_count'].values.reshape(-1, 1))

sequence_length = 3
X = []
y = []

for i in range(sequence_length, len(scaled_data)):
    X.append(scaled_data[i-sequence_length:i, 0])
    y.append(scaled_data[i, 0])
```

Figure 9. Visualizing cyber attack patterns.

As shown in Figure 10, for research purposes, the data for each attack pattern were grouped into 6-month intervals, and the average number of attacks was predicted. MinMaxScaler was used to normalize the data, and then the LSTM model was trained to predict attack patterns for the next five years. The prediction model was constructed using a Sequential model, with a dropout layer added to prevent overfitting. The predicted results were also automatically saved to a local directory.

Considering the different systems and data environments of each institution, functionality was included to allow for performance evaluation of the model for tuning purposes. To provide metrics for assessing the reliability of the data, as shown in Figure 11, the performance evaluation results were also saved and disseminated to relevant organizations.

```
model = Sequential()
model.add(LSTM(units=100, return_sequences=True, input_shape=(X.shape[1], 1)))
model.add(Dropout(0.2))
model.add(LSTM(units=50))
model.add(Dropout(0.2))
model.add(Dense(1))

optimizer = Adam(learning_rate=0.0005)
model.compile(optimizer=optimizer, loss='mean_squared_error')
model.fit(X, y, epochs=50, batch_size=16, verbose=1)
test_input = scaled_data[-sequence_length:]

predictions = []
for _ in range(10):
    test_input = np.reshape(test_input, (1, sequence_length, 1))
    predicted_value = model.predict(test_input)
    predictions.append(predicted_value[0, 0])
    test_input = np.append(test_input[0, 1:, 0], predicted_value[0, 0])
```

Figure 10. Predicting cyber attack frequency.

```
performance_file_path = os.path.join(save_path, 'performance_evaluation.jpg')
plt.savefig(performance_file_path)
```

Figure 11. Including performance evaluation tools.

Afterward, using UiPath's automation functionality, the results are disseminated, allowing each institution to respond promptly based on the provided information. The results predicted using the database employed in this study are as follows. We conducted performance evaluations optimized for the system and database used in the research, improving the model architecture by adjusting data preprocessing, layer unit numbers, and other parameters. Additionally, we tuned the model by adjusting the learning rate and epochs [19]. As shown in Figure 12, the four most frequent attack patterns from 2014 to 2 October 2024 (APT, Phishing, DDoS, Ransomware) were graphed, illustrating their trends (the sharp drop-off at both ends of the graph is due to the research scope limiting the time frame used for training). Furthermore, we provided predictions for the threats that may occur over the next five years based on each period.

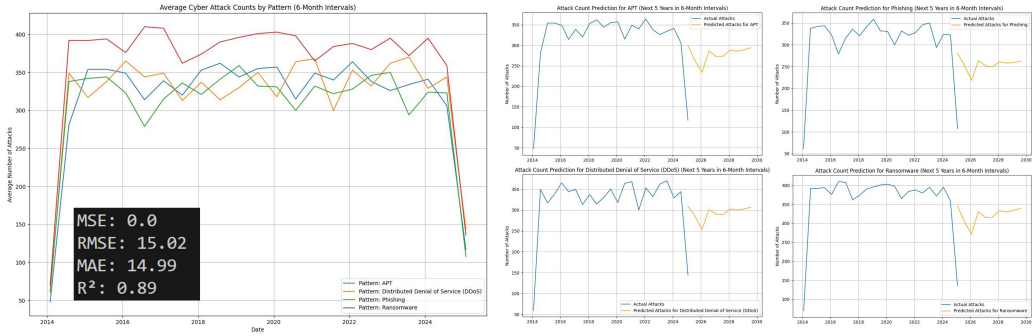

Figure 12. Trends in Major Attack Patterns (2014–2024) and Predictions for the Next 5 Years.

Additionally, considering the characteristics of national and public institutions, intelligence agencies, and military facilities where the internet is not installed, a web-based system was implemented to transmit and analyze cyber threat logs in real time through an internal network, aside from report generation. This system was developed based on the Flask [20] framework and was designed to display real-time image files (e.g., visualizations of log analysis results) from a local directory on a webpage. This system operates in the manner shown in Table 3 and was implemented as illustrated in Figure 13.

Through this, as shown in Figure 14, the server can be run to enable real-time transmission of cyber threat logs to a webpage. Since the Flask server is implemented to serve image files, security personnel from relevant organizations can immediately check the analyzed results and log the status in real-time, as shown in Figure 15.

```python
from flask import Flask, render_template, send_from_directory
import os

app = Flask(__name__)

IMAGE_FOLDER = r'C:\Users\gronk\OneDrive\Desktop\CyberAttackLog\DB\EX\Result'

@app.route('/')

def index():
    images = [file for file in os.listdir(IMAGE_FOLDER) if file.endswith(('png', 'jpg', 'jpeg', 'gif'))]
    return render_template('index.html', images=images)

@app.route('/images/<filename>')
def image(filename):
    return send_from_directory(IMAGE_FOLDER, filename)

if __name__ == '__main__':
    app.run(debug=True)
```

Figure 13. Web-based situation dissemination implementation code.

```
   Running on http://127.0.0.1:5000
Press CTRL+C to quit
   Restarting with stat
   Debugger is active!
   Debugger PIN: 403-531-910
127.0.0.1 - - [03/Oct/2024 14:07:48] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/APT_attack_prediction.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/Distributed%20Denial%20of%20Service%20(DDoS)_attack_prediction.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/Phishing_attack_prediction.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/performance_evaluation.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/average_attack_counts_by_pattern.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:07:49] "GET /images/Ransomware_attack_prediction.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/APT_attack_prediction.jpg HTTP/1.1" 304 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/KG.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/average_attack_counts_by_pattern.jpg HTTP/1.1" 200 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/Distributed%20Denial%20of%20Service%20(DDoS)_attack_prediction.jpg HTTP/1.1" 304 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/performance_evaluation.jpg HTTP/1.1" 304 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/Phishing_attack_prediction.jpg HTTP/1.1" 304 -
127.0.0.1 - - [03/Oct/2024 14:11:46] "GET /images/Ransomware_attack_prediction.jpg HTTP/1.1" 304 -
```

Figure 14. Running the web-based situation dissemination server.

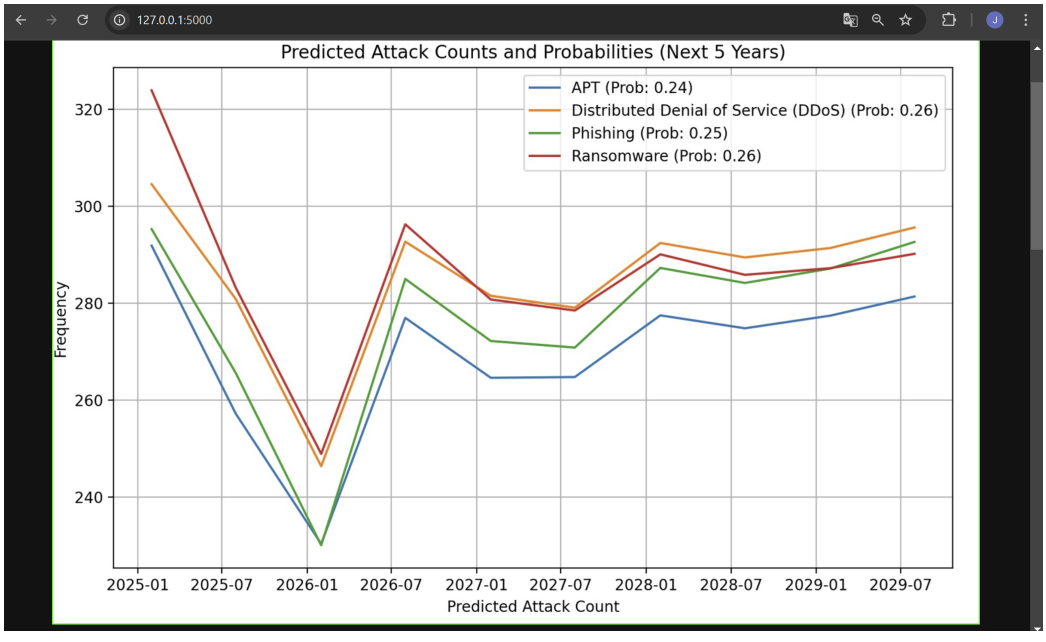

Figure 15. Web-based situation dissemination demonstration.

Table 3. Explanation of web-based situation dissemination system.

Feature	Description
Log Collection & Analysis	Collect and analyze logs from various security solutions in real time and visualize the results.
Image File Storage	Save the analyzed results as image files in a specified local path.
Flask Server Configuration	Use Python 3.10.14 's Flask framework to create a server that sends the saved images to a webpage for real-time visualization.

4. Experiments and Results

4.1. Experiment Setup

For research purposes, original data generated over the past 10 years by the CERT security team of government institutions were sampled, as shown in Figure 16, and experiments were conducted in an environment identical to the actual hacking threat response process. For security reasons, measurements were taken within a single network. The experiment setup involved three different environments: the existing system, which relied entirely on manual processes; the AI-only system, where repetitive tasks typically handled by RPA in the improved system were instead performed manually and the improved system, which used a combination of AI and RPA to automate both advanced threat detection and repetitive tasks. The performance evaluation environment consisted of an H13th Gen Intel(R) Core(TM) i9-13900H 2.60 GHz, 32GB RAM, Windows 11 64-bit OS, VS Code 1.93.1, UiPath 2024.10.5 Community Edition, DB Browser for SQLite Version 3.13.0, and Anaconda Navigator 2.4.0. The labor costs were calculated based on the average annual salary of G7 countries [21], at $55,000 per year (approximately $23.9 per hour, based on an 8-h workday). In human work scenarios, random samples of 60 teams (three people per team) were used to test the average task processing time over one month, rounded to the nearest minute, and metrics such as processing time, error rates, cost-sharing ratios for related institutions, and administrative delay rates were evaluated. Additionally, based on limited-use products solely used in the hacking threat response process, the cost of introducing an AI program was estimated at $6000 per year, with the result rounded accordingly.

	A	B	C	D	E	F	G	H
1	Data	System ID	Origin IP	Origin IP(Nation)	Destination IP	Destinatior	Attack pattern	Processing method
2	1/1/2014	256461	10.151.180.28	US	140.145.193.177	Korea	APT	Auto
3	1/1/2014	256461	100.248.185.251	IN	140.211.58.46	Korea	Distributed Denial of Service (DDoS)	Auto
4	1/1/2014	846513	80.109.152.176	AU	140.36.204.86	Korea	Distributed Denial of Service (DDoS)	Auto
5	1/1/2014	256461	90.19.0.114	CN	140.201.75.224	Korea	Phishing	Auto
6	1/1/2014	256461	90.220.174.28	CN	140.158.154.150	Korea	Distributed Denial of Service (DDoS)	Auto
7	1/1/2014	846513	30.138.215.71	FR	140.101.107.54	Korea	Phishing	Auto
8	1/1/2014	256461	90.190.75.85	CN	140.129.208.72	Korea	APT	Auto
9	1/2/2014	256461	120.122.69.222	ES	140.241.11.196	Korea	APT	Auto
10	1/2/2014	325641	40.199.221.15	DE	140.244.244.169	Korea	Ransomware	Auto
11	1/2/2014	896513	160.228.110.207	RU	140.241.247.144	Korea	Phishing	Auto
12	1/2/2014	256461	20.138.68.215	CA	140.175.206.91	Korea	Distributed Denial of Service (DDoS)	Auto
13	1/2/2014	256461	10.127.188.115	US	140.249.194.110	Korea	Phishing	Auto
14	1/2/2014	256461	90.234.226.45	CN	140.74.107.85	Korea	Ransomware	Auto
15	1/3/2014	256461	40.102.202.35	DE	140.102.161.203	Korea	Ransomware	Auto
16	1/3/2014	256461	80.196.219.114	AU	140.201.102.183	Korea	Ransomware	Auto
17	1/3/2014	256461	90.80.252.107	CN	140.130.123.26	Korea	Distributed Denial of Service (DDoS)	Auto
18	1/3/2014	256461	10.72.189.112	US	140.103.195.102	Korea	Ransomware	Auto
19	1/3/2014	685954	80.195.204.160	AU	140.40.59.225	Korea	Ransomware	Auto
20	1/3/2014	256461	20.106.82.130	CA	140.78.206.188	Korea	Phishing	Auto
21	1/3/2014	256461	120.205.168.150	ES	140.80.174.237	Korea	Distributed Denial of Service (DDoS)	Auto
22	1/3/2014	256461	60.209.68.248	KR	140.23.35.71	Korea	Distributed Denial of Service (DDoS)	Auto
23	1/3/2014	256461	40.18.130.55	DE	140.150.171.1	Korea	Ransomware	Auto
24	1/3/2014	256461	110.196.46.36	GB	140.186.153.238	Korea	APT	Auto
25	1/3/2014	256461	30.223.163.228	FR	140.135.237.114	Korea	Distributed Denial of Service (DDoS)	Auto
26	1/3/2014	256461	160.203.154.70	RU	140.180.213.49	Korea	Distributed Denial of Service (DDoS)	Auto
27	1/3/2014	256461	60.191.97.129	KR	140.41.39.136	Korea	Distributed Denial of Service (DDoS)	Auto

Figure 16. Sample Data of 10 Years of Cyber Threat Logs by CERT.

4.2. Results and Performance Analysis

4.2.1. Performance Comparison Results of the Automated Threat Response System

As illustrated in Table 4, the comparative performance analysis of the Existing System, AI-only System, and Improved System reveals substantial differences across key metrics. The Improved System, which integrates AI and Robotic Process Automation (RPA), consistently demonstrates superior performance in terms of processing time, error rates, cost efficiency, information sharing, and administrative delays.

Table 4. Comparison of hacking threat response processes (existing, improved).

Business Process	Stages	Situation Arises	Action Situation	Report Situation	Follow Up
Processing time	Existing	30 min	30 min	60 min	30 min
	AI-only	10 min	15 min	30 min	15 min
	Improved	0.3 s	0.3 s	0.3 s	0.3 s
Budget required (per hour)	Existing	$35.85	$35.85	$71.70	$35.85
	AI-only	$15.00	$15.00	$30.00	$15.00
	Improved	$1.30	$1.30	$2.60	$1.30
Processing error rate	Existing	25%	35%	5%	30%
	AI-only	10%	15%	2%	12%
	Improved	0%	0%	0%	0%
Sharing with related organizations	Existing				
	AI-only	70%	68%	72%	69%
	Improved	99%	98%	99%	99%
Administrative delay rate	Existing				
	AI-only	20%	25%	18%	22%
	Improved	0%	0%	0%	0%

Processing Time: The most pronounced improvement is observed in processing time. The Existing System, dependent on manual processes, incurs significant delays at each stage (Incident Occurrence, Action Taken, Report Submitted, and Follow-up), requiring between 30 and 60 min per stage. This manual dependency inherently limits the ability to respond to rapidly evolving cyber threats in real time. The AI-only System mitigates this delay somewhat, reducing the processing time to 10–30 min per stage, leveraging AI-driven analytics while still relying on human intervention for repetitive tasks. However, the Improved System achieves near-instantaneous response times, reducing each stage's processing time to just 0.3 s, a reduction by several orders of magnitude.

Mathematically, the reduction factor in processing time compared to the Existing System is calculated as follows:

$$\text{Processing Time Reduction Factor (Improved vs. Existing)} = \frac{30\,\text{min}}{0.3\,\text{s}} = \frac{1800\,\text{s}}{0.3\,\text{s}} = 6000$$

This equates to a 6000-fold improvement, which is critical in cybersecurity environments where even milliseconds can prevent significant system damage. Similarly, when comparing the Improved System to the AI-only System, we observe a processing time reduction factor of approximately 2000, further demonstrating the efficacy of integrating RPA with AI for complete automation.

Cost Efficiency: From a cost perspective, the labor-intensive nature of the Existing System translates into substantial operational expenses, ranging from $35.85 to $71.7 per hour, depending on the stage. The AI-only System achieves significant cost savings, reducing the expense to $15.0 to $30.0 per hour by automating only certain analytic tasks, while still requiring human labor for repetitive processes. However, the Improved System drastically reduces costs to just $1.3 to $2.6 per hour.

To quantify this reduction:

$$\text{Cost Reduction Rate (Improved vs. Existing)} = \frac{71.7 - 2.6}{71.7} \times 100 \approx 96.4\%$$

This 96.4% cost reduction represents a dramatic increase in operational efficiency, making the Improved System especially viable for organizations with budgetary constraints, such as government agencies and public institutions.

Error Rates: Error rates in the Existing System range from 5% to 35%, primarily due to the reliance on manual operations, where human error plays a significant role in security failures. The AI-only System reduces error rates to between 2% and 15%, as AI automates some of the decision-making processes. However, the Improved System eliminates errors entirely, achieving a 0% error rate across all stages. This is a statistically significant improvement, with error variance approaching zero.

Mathematically, this improvement in accuracy can be represented as follows:

$$\sigma_{\text{Error Rate}} \approx 0 \,(\text{Improved System})$$

This result underscores the dramatic enhancement in reliability and accuracy that complete automation provides, ensuring that human errors no longer jeopardize system security.

Information Sharing with Related Institutions: Another critical area of comparison is the ability to share threat information with related organizations. The Existing System lacks this capability, severely limiting inter-agency collaboration. The AI-only System achieves moderate success in this area, with a 68% to 72% information-sharing rate, primarily due to automated reporting. However, the Improved System excels with a 98% to 99% sharing rate, enabling real-time collaboration between organizations and significantly enhancing coordinated responses to cyber threats.

Administrative Delay Rate: The administrative delay rate in the Existing System is high, primarily due to manual processing. Each administrative task introduces delays that hinder timely responses to ongoing cyber threats. The AI-only System improves this metric, reducing delays to between 18% and 25%. However, the Improved System eliminates administrative delays almost entirely, achieving near-zero delay rates.

In conclusion, the comparative performance analysis unequivocally demonstrates that the Improved System, which integrates AI for advanced threat detection and RPA for automating repetitive tasks, provides the most effective solution across all performance metrics. The drastic reductions in processing time and operational costs, combined with the complete elimination of errors and improved inter-agency collaboration, position the Improved System as the optimal solution for modern cybersecurity challenges.

Statistically, the integration of AI and RPA reduces human error variability and enhances system responsiveness. From a mathematical perspective, the improvements observed across all metrics reflect a systemic optimization that offers scalability and adaptability in real-world applications, making this a robust, future-proof solution for the increasing complexities of cyber threats.

4.2.2. Qualitative Performance Evaluation of the Proposed System

The integration of AI-based cyber threat prediction and analysis in our system represents a significant advancement over traditional reactive incident response methods. Unlike conventional approaches that rely on post-incident mitigation, our system facilitates proactive defense, addressing the growing complexity and frequency of cyber threats. By providing real-time prediction, analysis, and dissemination of attack patterns, the system minimizes response times and enhances overall resilience.

As illustrated in Figure 17, the system captures and integrates cyber threat logs from multiple security platforms in real-time, storing them in a unified database. This consolidated approach facilitates efficient analysis and prediction, significantly improving data processing speed, accuracy, and automation. The use of standardized log for-

mats addresses inconsistency issues that have been a major limitation in traditional cybersecurity approaches.

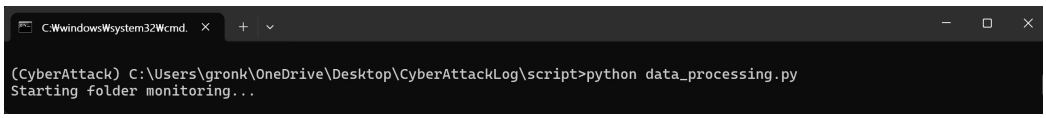

Figure 17. Real-Time Collection and Integration of Cyber Threat Logs.

Previously, inconsistent data formats from multiple security solutions hindered log collection and analysis, causing delays, and inaccuracies, and reduced the effectiveness of real-time responses. Each security solution operated in isolation without standardized data integration, limiting overall efficiency. Our system standardizes and stores log data in a consistent format, creating a robust dataset for efficient analysis and addressing issues like prolonged processing times and high error rates. This standardization provides a reliable foundation for machine learning models to effectively learn from diverse security events.

The system consolidates data into an integrated database, accessible by authorized institutions in real-time via a Flask server. This approach reduces the need for manual file exchanges, which are prone to errors and delays, and enhances data accessibility, allowing seamless collaboration between organizations. As a result, institutions can share critical threat information instantly, enabling rapid and coordinated responses to cyber threats. This ability for cross-institution collaboration represents a significant advancement in coordinated cyber defense, particularly given the increasing sophistication of cyberattacks targeting multiple organizations.

By leveraging AI for cyber threat prediction and analysis, the proposed system enables proactive responses that go beyond traditional reactive methods. Machine learning algorithms for attack pattern recognition and predictive analysis empower the system to detect potential threats at an early stage, allowing for a timely response and mitigation. Real-time data sharing and predictive analysis help institutions anticipate future attack patterns, enhancing the robustness and scalability of defenses. The integrated database supports real-time attack prediction and AI model training, advancing security technologies and fostering a proactive approach to cybersecurity. Real-time data normalization, feature encoding, and preprocessing ensure high-quality, consistent data for AI models, leading to higher prediction accuracy and reduced false positives.

In conclusion, the proposed system represents a substantial evolution in cybersecurity defense, offering a scalable, automated, and proactive solution to address advanced cyber threats. By leveraging real-time data processing, AI-based analysis, and institutional collaboration, the system provides a robust framework that addresses the limitations of legacy systems and prepares organizations for future cybersecurity challenges. The experimental results demonstrate the superior performance of this system compared to traditional manual systems, particularly in terms of log processing speed, prediction accuracy, error reduction, and operational efficiency.

This capability allows for the prompt analysis and prediction of attack patterns. The system's real-time file monitoring function automatically preprocesses newly collected data and stores them in the database, greatly enhancing data processing speed and accuracy while increasing the level of automation.

Previously, data were siloed across individual security solutions, limiting collaboration and reducing data usability across systems. The proposed system overcomes this limitation by standardizing and centralizing data into a shared database, accessible by all institutions in real time via a Flask server. This eliminates the need for manual file exchanges, enabling rapid collaboration between organizations and facilitating swift responses to cyber threats. The system's ability to allow multiple institutions to collaborate seamlessly and act promptly represents a significant advancement in coordinated cyber defense.

This real-time data-sharing infrastructure greatly improves the speed and accuracy of cyber threat response, making data more accessible and usable. By enabling real-time data

sharing and predictive analysis, the system equips institutions with the tools needed to anticipate future attack patterns, marking a substantial improvement over legacy systems. Furthermore, the establishment of a database that supports real-time attack prediction and various AI-based learning models plays a key role in advancing security technologies.

4.3. Limitations

The AI-based RPA automation system proposed in this study demonstrates significant improvements over traditional manual security responses. However, there are still several areas that require further refinement. The system's performance is heavily dependent on the quality and consistency of the data it processes. The diversity of log formats and data sources across different institutions introduces complexities in data integration and analysis, which can, in turn, impact the accuracy of threat prediction. Therefore, it is essential to implement standardization and optimization measures tailored to the specific environments of individual institutions to ensure optimal performance.

While the system is generally designed for efficient real-time processing, resource constraints or network latency could pose challenges under certain conditions. This issue is particularly relevant in environments with limited infrastructure, such as public institutions relying on legacy systems, where customized optimization may be required to maintain performance.

Moreover, AI systems can be sensitive to subtle biases in data processing, uncertainties in interpretation, and sophisticated security threats. Although this study does not directly address ethical concerns or adversarial attacks, these issues remain critical for future research to ensure the system's robustness and reliability. Automated data standardization and verification mitigate some risks, but additional measures are needed to further enhance the overall security and stability of the system.

In conclusion, while the proposed system has shown promising results in smaller-scale environments, further validation in large-scale, real-world settings is necessary. Future research should focus on evaluating the system's performance across a broad range of operational environments to ensure consistent results under diverse conditions. This will lay the groundwork for further optimization and help ensure the system meets the varying needs of different organizations.

Furthermore, the introduction of AI-based RPA in cybersecurity may reduce the need for certain manual tasks, potentially leading to workforce displacement if proper upskilling and reskilling initiatives are not implemented. It is crucial to address these issues proactively to prevent large-scale job reductions or the obsolescence of roles in the cybersecurity sector.

5. Conclusions

In this study, we proposed automating cyber threat response by integrating AI and RPA technologies. The proposed system addresses the weak points, such as human error, that emerged from manual security frameworks, while maintaining existing systems and being immediately implementable at a low cost. This is particularly practical for countries with limited budgets, public institutions, and military facilities, as the system can be utilized instantly without the need for system changes.

Traditional security systems mostly responded passively after a cyberattack occurred, and there were limitations in efficiently analyzing and responding to security threats due to the lack of integration of log data generated from various security solutions. However, the proposed system overcomes these issues by standardizing and integrating log data, allowing public institutions to consistently manage data from different security solutions. This also establishes a foundation for training AI models capable of predicting future attack patterns, thus laying the groundwork for the development of stronger cybersecurity systems in the future.

Additionally, AI-based threat prediction enables proactive responses beyond simple reactive measures, and automation through RPA maximizes operational efficiency while minimizing human error and dramatically reducing processing times. These capabilities

will play a significant role in the field of cybersecurity, where rapid and accurate responses are essential in the fast-changing digital environment.

As a result, the system proposed in this study not only overcomes current technical limitations but also provides a foundation for building future-oriented security systems. Moving forward, standardized log collection and integration across various institutions and systems will further enhance the learning and predictive performance of AI models, continuously improving national cybersecurity capabilities.

Author Contributions: Conceptualization, J.K.; Methodology, J.K.; Software, J.K.; Validation, S.H.K.; Writing—original draft, J.K.; Writing—review & editing, S.H.K. and I.J.; Supervision, I.J.; Project administration, I.J. All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00212300).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the datasets: The datasets presented in this article contain cybersecurity threat information and are part of an ongoing internal research project, which makes them not readily available. For inquiries regarding access to the datasets, please contact the first author via email at gronkoutt@gmail.com.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Naseer, I. Machine Learning Applications in Cyber Threat Intelligence: A Comprehensive Review. *Asian Bull. Big Data Manag.* **2023**, *3*, 190–200. [CrossRef]
2. Aslan, Ö.; Aktuğ, S.S.; Ozkan-Okay, M.; Yilmaz, A.A.; Akin, E. A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. *Electronics* **2023**, *12*, 1333. [CrossRef]
3. Trifonov, R.; Manolov, S.; Tsochev, G.; Pavlova, G. Automation of cyber security incident handling through artificial intelligence methods. *WSEAS Trans. Comput.* **2019**, *18*, 274–280.
4. González-Granadillo; G.; González-Zarzosa, S.; Diaz, R. Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures. *Sensors* **2021**, *21*, 4759. [CrossRef] [PubMed]
5. Sarker, I.H. *AI-Driven Cybersecurity and Threat Intelligence*; Springer: Cham, Switzerland, 2024; Volume 26, pp. 1–26.
6. Trifonov, R.; Yoshinov, R.; Manolov, S.; Tsochev, G.; Pavlova, G. Artificial intelligence methods suitable for incident handling automation. *MATEC Web Conf.* **2019**, *292*, 01044. [CrossRef]
7. The U.S. House Committee on the Budget. Available online: https://budget.house.gov/budgets/process (accessed on 4 October 2024).
8. Dhabliya, D.; Ghule, G.; Khubalkar, D.; Moje, R.K.; Kshirsagar, P.S.; Bendale, S.P. Robotic Process Automation in Cyber Security Operations: Optimizing Workflows with AI-Driven Automation. *J. Electr. Syst.* **2023**, *19*, 96–105. [CrossRef]
9. Sarker, I.H. Machine Learning: Algorithms, Real-World Applications. *Expert Syst. Appl.* **2021**, *21*, 160.
10. Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy. *SN Comput. Sci.* **2021**, *2*, 420. [CrossRef] [PubMed]
11. Karlsen, E.; Luo, X.; Zincir-Heywood, N.; Heywood, M. Benchmarking Large Language Models for Log Analysis, Security, and Interpretation. *J. Netw. Syst. Manag.* **2024**, *32*, 59. [CrossRef]
12. Yarlagadda, R.T. The RPA and AI automation. *Int. J. Creat. Res. Thoughts (IJCRT)* **2018**, *6*, 2320–2882.
13. Ribeiro, J.; Lima, R.; Eckhardt, T.; Paiva, S. Robotic process automation and artificial intelligence in industry 4.0—A literature review. *Procedia Comput. Sci.* **2021**, *181*, 51–58. [CrossRef]
14. Wewerka, J.; Reichert, M. Robotic process automation—A systematic literature review and assessment framework. *arXiv* **2020**, arXiv:2012.11951.
15. Fischer, E.A. *Cybersecurity Issues and Challenges: In Brief*; Congressional Research Service: Washington, DC, USA, 2014; Volume 12, pp. 1–12.
16. He, K.; Kim, D.D.; Asghar, M.R. Adversarial machine learning for network intrusion detection systems: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 538–566. [CrossRef]
17. Sherstinsky, A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
18. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef] [PubMed]

19. Banu, J.F.; Rajeshwari, S.B.; Kallimani, J.S.; Vasanthi, S.; Buttar, A.M.; Sangeetha, M.; Bhargava, S. Modeling of Hyperparameter Tuned Hybrid CNN and LSTM for Prediction Model. *Intell. Autom. Soft Comput.* **2022**, *33*, 1393–1405. [CrossRef]
20. Spencer, R.; Smalley, S.; Loscocco, P.; Hibler, M.; Andersen, D.; Lepreau, J. The Flask security architecture: System support for diverse security policies. In Proceedings of the 8th USENIX Security Symposium (USENIX Security 99), Washington, DC, USA, 23–36 August 1999; Volume 18, pp. 1–18.
21. Charted: Average Wage Growth in G7 Countries. Available online: https://www.visualcapitalist.com/charted-average-wage-growth-in-g7-countries-2000-2022/ (accessed on 4 October 2024).