Hybrid Worm Detection Based on Signature & Anomaly

Peram Chandra Sekhar Reddy [1], Mr. Sankara Narayanan S T[2]

[1]*Peram Chandra Sekhar Reddy, MSc cyber forensics & information security, Dr. M.G.R Educational And Research Institute, Chennai, India, pchandrasekhar7981@gmail.com,*

[2] *Mr. Sankara Narayanan S T, Assistant Professor, Faculty of Center of Excellence in Digital Forensics, Chennai, India*

Abstract - Internet worms pose a significant threat by propagating through network traffic, compromising system security, and exfiltrating sensitive information. To enhance detection accuracy, a hybrid two-factor worm detection system is proposed, integrating signature-based and anomaly-based methodologies. Signature-based detection employs packet capture (PCAP) analysis and NetFlow inspection to identify malicious signatures using predefined rule sets. Honeypot logs are leveraged to detect and mitigate attack attempts by monitoring unauthorized access attempts. Anomaly-based detection utilizes machine learning models, including Random Forest, Decision Tree, and Bayesian Networks, to classify network traffic as normal or abnormal based on behavioral patterns. Experimental results demonstrate that Random Forest and Decision Tree achieve the highest accuracy of 98%, outperforming Bayesian Networks. Additionally, deep learning models such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) are employed for anomaly detection, with GRU achieving superior performance. The proposed framework effectively enhances worm detection capabilities, reducing false positives and improving cybersecurity resilience. Future enhancements include integrating evolutionary feature selection techniques such as Genetic Algorithms and Particle Swarm Optimization to optimize detection accuracy.

***Key Words*:** Anomaly Detection, Worm Detection, Machine Learning, Deep Learning, Random Forest, Decision Tree, Convolutional Neural Networks, Long Short-Term Memory, Gated Recurrent Units.

1.INTRODUCTION

In the realm of cybersecurity, combating the rapidly evolving landscape of digital threats necessitates multifaceted and adaptive detection mechanisms. One such critical threat is the computer worm—a self-replicating malicious entity that can propagate rapidly across networks, compromising system integrity, exfiltrating sensitive information, and disrupting essential services. Traditional methods, while foundational, are increasingly being challenged by the sophistication and variability of modern worms. To address these limitations, the integration of signature-based and anomaly-based detection approaches, referred to as two-factor worm detection, has emerged as a promising strategy [1][2].

Signature-based detection, long regarded as a staple in cybersecurity, operates on the principle of identifying known patterns or "signatures" associated with malware. It relies heavily on predefined rule sets, such as those derived from packet capture (PCAP) analysis and NetFlow inspection, to detect and block known threats with high accuracy [3]. This method has proven effective in identifying well-characterized worms and mitigating their impact swiftly. However, its reliance on known signatures renders it less effective against zero-day exploits, polymorphic worms, and newly mutated attack vectors that escape traditional detection rules [4]. Consequently, relying solely on signature-based techniques leaves systems vulnerable to novel attacks.

To bridge this gap, anomaly-based detection has gained prominence. Unlike signature-based methods, anomaly detection models learn the baseline behavior of network systems and flag deviations that may indicate malicious activity. These techniques utilize machine learning algorithms such as Random Forest, Decision Tree, and Bayesian Networks to classify network traffic as normal or abnormal based on behavioral analysis [5][6]. Such models are especially adept at uncovering previously unknown threats and adaptively responding to evolving attack strategies. Furthermore, deep learning architectures like Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) have demonstrated remarkable

performance in identifying sophisticated anomalies, particularly in high-dimensional network data [7].

The combination of signature-based and anomaly-based methods offers a synergistic approach to worm detection, blending the high precision of rule-based systems with the adaptive intelligence of machine learning. This hybrid model not only enhances detection accuracy but also reduces false positives and bolsters defenses against a broader spectrum of threats [8]. By leveraging the complementary strengths of both paradigms, two-factor worm detection systems provide a robust framework for securing digital infrastructures. This paper investigates the practical implementation and empirical performance of such systems, presenting a comprehensive exploration of their efficacy in real-world cybersecurity environments.

2. Literature Review

In recent years, the proliferation of sophisticated cyber threats has spurred extensive research into advanced intrusion detection methodologies, particularly focusing on hybrid models that combine signature-based and anomaly-based techniques. Numerous studies have explored the evolving landscape of worm detection and the integration of intelligent algorithms for enhancing security resilience.

Kennedy and Joseph [9] provide an extensive review of Intrusion Detection and Protection Systems (IDPS) in cloud computing environments, emphasizing the limitations of traditional approaches in addressing emerging threats such as polymorphic worms. Their work highlights the growing reliance on multi-layered detection systems that integrate anomaly-based and signature-based methods to deliver proactive and adaptive protection. This dual approach aligns with the objectives of two-factor worm detection, where combining static and behavioral analysis enhances threat identification and response.

Abdulganiyu et al. [10] conducted a systematic literature review on intrusion detection systems, outlining the challenges in developing efficient detection mechanisms. They discuss various machine learning and deep learning models employed in anomaly detection and underscore the significance of feature selection techniques in improving classification accuracy. Their findings reinforce the importance of integrating evolutionary feature selection algorithms, such as Genetic Algorithms and Particle Swarm Optimization, into anomaly-based systems to boost detection efficiency.

Isong et al. [11] delve into intrusion detection strategies tailored for Internet of Things (IoT) ecosystems. Given the lightweight and heterogeneous nature of IoT devices, traditional signature-based systems often fall short in identifying sophisticated worms. Their analysis reveals the effectiveness of ensemble learning models and adaptive anomaly detection techniques in enhancing the defense capabilities of IoT networks. The incorporation of explainable AI further supports the interpretability and reliability of detection outcomes, a vital requirement in critical infrastructure security.

Gupta and Simon [12] examine the use of Random Forest algorithms for anomaly detection in cloud computing environments. Their experiments demonstrate that Random Forest significantly reduces false positives and improves detection rates when compared to other classifiers. This supports the deployment of tree-based models in anomaly-based worm detection, particularly when dealing with high-dimensional network data. Their research complements the use of Decision Tree models in hybrid detection frameworks, illustrating the advantages of ensemble approaches.

Asadi et al. [13] provide a comprehensive survey on botnet detection strategies, highlighting the evolution of botnet behavior and the challenges in distinguishing malicious traffic from legitimate communication. Their study underscores the necessity of employing both signature recognition and anomaly pattern analysis to detect advanced threats. This dual-layered detection mechanism mirrors the two-factor approach, which balances efficiency with adaptability, especially in the face of stealthy and morphing attacks.

Hooshmand et al. [14] propose a robust network anomaly detection system using an ensemble learning approach enhanced by explainable AI. Their model combines several classifiers to improve accuracy and transparency, helping security analysts better understand detection results. The emphasis on explainability ensures that the system's decisions can be audited and trusted, which is critical for regulatory compliance and real-time response. Their work reinforces the trend of combining multiple detection methods to counteract the limitations of individual models.

Alshamsi et al. [15] present a systematic literature review focused on malware detection in smart home environments. They argue that hybrid detection models, which merge static and dynamic analysis, are essential for mitigating threats in resource-constrained systems. Their findings point to the growing need for lightweight yet effective worm detection frameworks capable of operating across diverse environments, from enterprise networks to home automation systems.

Nguyen et al. [16] propose a deep clustering hierarchical latent representation model for anomaly-based cyber-attack detection. Their deep learning-based framework is capable of learning intricate data representations, making it well-suited for detecting subtle anomalies in large-scale network environments. The study validates the use of unsupervised learning in environments where labeled data is scarce, further enhancing the adaptability of anomaly-based detection systems.

Fährmann et al. [17] explore anomaly detection strategies within smart environments, reviewing various machine learning and deep learning techniques used for identifying irregular behavior. Their survey concludes that while deep models like LSTM and GRU offer promising accuracy, hybrid models that combine these with rule-based detection deliver superior performance. This validates the integration of CNN, LSTM, and GRU in the proposed two-factor worm detection framework, where behavioral insights and signature patterns converge to detect complex threats.

Goni et al. [18] analyze the current landscape of cybersecurity threats, prevention strategies, and system complexities. Their study highlights the limitations of standalone systems in dealing with evolving malware and supports the shift toward intelligent, integrated frameworks. They argue that hybrid models not only improve accuracy but also adapt more effectively to unknown threats, aligning with the objectives of two-factor detection systems.

Wan et al. [19] propose a deep learning model for network flow anomaly detection, emphasizing the role of temporal and spatial features in capturing behavioral deviations. Their approach leverages deep neural networks to automatically extract features, reducing dependency on manual feature engineering. Their research supports the deployment of deep learning in worm detection systems, especially in processing voluminous and time-sensitive network data.

Karthikeyan and Revathi [20] explore threat detection in wireless sensor networks and propose methodologies for enhancing transmission security. Their work demonstrates the applicability of hybrid detection techniques in resource-limited settings and supports the use of lightweight algorithms in combination with more complex classifiers. The flexibility of their approach resonates with the broader goals of two-factor detection systems, which must operate efficiently across varying network architectures and hardware capabilities.

3. Materials & Methods

A two-factor worm detection system is proposed by integrating signature-based and anomaly-based methodologies to bolster cybersecurity resilience. Signature-based detection utilizes packet capture (PCAP) analysis and NetFlow inspection, identifying malicious patterns via predefined rule sets, while honeypot logs provide further insight into unauthorized access attempts [9], [13]. Anomaly-based detection incorporates machine learning classifiers such as Random Forest, Decision Tree, and Bayesian Networks to analyze behavioral deviations in network traffic [10], [12], [14]. These models are trained on historical traffic datasets to distinguish between legitimate and malicious activities. To further enhance detection accuracy, deep learning algorithms including Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) are employed for modeling complex temporal and spatial network behaviors [16], [17], [19]. The hybrid system leverages the strengths of both techniques to improve detection efficacy and minimize false positives, presenting a robust defense against rapidly evolving worm threats in network environments.

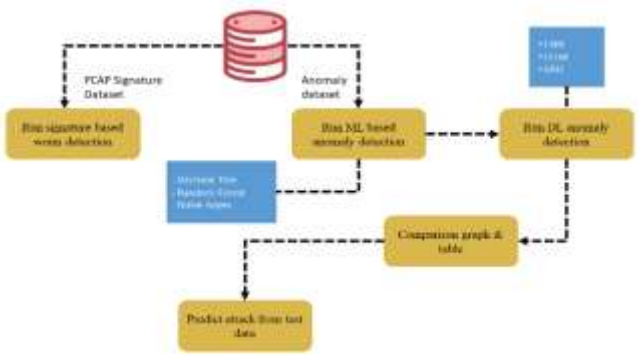


Fig-1: Proposed Architecture

This diagram illustrates a network intrusion detection system employing a multi-stage approach. It begins with

signature-based worm detection using a PCAP dataset. Anomalies are then identified using machine learning models (Decision Tree, Random Forest, Naive Bayes) on an anomaly dataset. Finally, deep learning models (CNN, LSTM, GRU) are applied for further anomaly detection. The results are compared and presented in a graph and table. The system ultimately predicts attacks from test data, leveraging a combination of signature-based and anomaly-based techniques for enhanced accuracy and robustness.

3.1. Dataset Collection:

The dataset collection for the proposed two-factor worm detection system comprises two primary sources: signature-based and anomaly-based data. The signature-based dataset is acquired in PCAP format, capturing raw network traffic to analyze individual packets for malicious patterns. This dataset includes a range of packet types with varying source and destination IP addresses and ports, facilitating the identification of known worm signatures. Anomaly-based data is gathered in text or CSV format, representing various network behaviors and traffic features. It includes records labeled with different worm types and normal activities, enabling classification based on behavioral deviations. The datasets are preprocessed to extract relevant attributes such as protocol type, flow duration, byte count, and flag status, supporting effective detection through both traditional and deep learning methods.

3.2. Pre-Processing:

Pre-processing is a crucial step in preparing the datasets for effective worm detection. For the signature-based PCAP data, packet features such as IP addresses, port numbers, protocol types, and payload lengths are extracted using tools like Wireshark or custom packet analyzers. These features are structured into a readable format for signature matching. For the anomaly-based dataset, missing values are handled, redundant features are removed, and categorical variables are encoded numerically. Feature normalization or standardization is applied to ensure uniformity in data scaling. The cleaned and transformed dataset enhances the performance of machine learning and deep learning algorithms by improving convergence and reducing noise. This pre-processed data serves as the foundation for accurate training and reliable worm detection.

3.3. Training & testing:

The anomaly-based dataset is divided into two subsets using an 80:20 ratio, where 80% of the data is used for training and 20% for testing. During training, machine learning and deep learning models learn to distinguish between normal and worm-infected traffic by identifying patterns and correlations within the labeled data. The testing phase evaluates the trained models on unseen data to assess their performance, accuracy, and generalization capabilities. This split ensures balanced model development and validation for effective worm detection.

3.4. Algorithms:

Decision Tree: A decision tree is a supervised learning algorithm that is commonly used for classification and regression tasks. It works by recursively splitting data based on feature values, forming a tree-like structure. The purpose of a decision tree is to model decision-making processes, identifying patterns and relationships in datasets. It is widely used in data classification, fraud detection, medical diagnosis, and network security, as it provides interpretability and handles both numerical and categorical data effectively.

Random Forest: Random Forest is an ensemble learning method that constructs multiple decision trees and aggregates their outputs for improved accuracy and generalization. It enhances prediction stability and reduces overfitting by averaging the results of numerous trees. The algorithm is used for classification, regression, and anomaly detection, particularly in cybersecurity, medical diagnostics, finance, and image recognition. Its ability to handle large datasets with high dimensionality makes it a robust choice for complex pattern recognition tasks.

Naïve Bayes: Naïve Bayes is a probabilistic classifier based on Bayes' Theorem, assuming independence among features. It calculates the probability of a class given the input features and selects the most likely outcome. The algorithm is primarily used for text classification, spam filtering, sentiment analysis, and medical diagnosis. Its simplicity, efficiency, and capability to work well with small datasets make it a popular choice for real-time classification tasks, despite its strong independence assumption.

CNN (Convolutional Neural Network): CNN is a deep learning architecture designed for processing grid-like

data, primarily images. It employs convolutional layers to extract hierarchical features, pooling layers for dimensionality reduction, and fully connected layers for classification. CNN is widely used in computer vision tasks, such as image recognition, object detection, and medical image analysis. It has also been adapted for cybersecurity applications, including intrusion detection and network anomaly detection, due to its ability to learn spatial dependencies in structured data.

LSTM (Long Short-Term Memory): LSTM is a type of recurrent neural network (RNN) that is intended to handle sequential data with long-term dependencies. It incorporates memory cells and gating mechanisms to control information flow, preventing vanishing gradient issues. LSTM is commonly used in natural language processing (NLP), speech recognition, time-series forecasting, and anomaly detection. Its ability to retain information over extended sequences makes it effective for analyzing network traffic patterns, detecting malware, and predicting cyber threats.

GRU (Gated Recurrent Unit): GRU is a variant of LSTM that simplifies the architecture by using fewer gating mechanisms while maintaining performance. It efficiently captures temporal dependencies in sequential data with lower computational cost. GRU is widely used in NLP, speech processing, and anomaly detection tasks, where sequential dependencies must be preserved. Due to its efficiency, GRU is often preferred over LSTM in real-time applications such as fraud detection, network security, and time-series analysis.

4. Results And Discussion

Accuracy: A test's accuracy is determined by how well it distinguishes between patient and healthy cases. The percentage of true positive and true negative in each assessed case should be determined in order to measure a test's accuracy. This can be expressed mathematically as:

$$Accuracy = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} (1)$$

Precision: Precision measures the percentage of cases or samples that are accurately classified out of those that are labelled as positives. Therefore, the following formula can be used to determine the precision:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} (2)$$

Recall: In machine learning, recall is a metric that assesses a model's capacity to locate every pertinent instance of a given class. It gives information about how well a model captures instances of a particular class by dividing the number of accurately predicted positive observations by the total number of real positives.

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} (3)$$

F1-Score: F1 score is a machine learning evaluation metric that determines a model's accuracy. It combines a model's precision and recall scores. The accuracy statistic calculates the number of times a model predicted correctly over the full dataset.

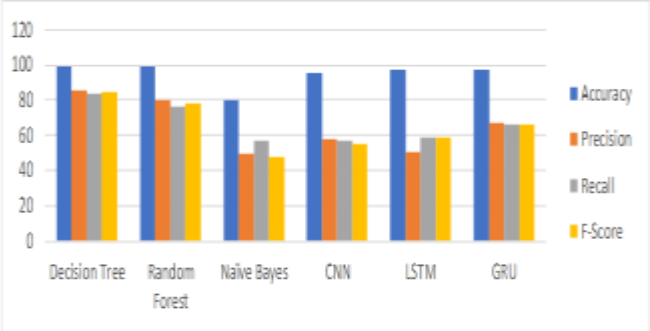
$$F1\ Score = 2 * \frac{Recall\ \text{X}\ Precision}{Recall + Precision} * 100 (1)$$

In Table (1), the performance metrics—accuracy, precision, recall, and F1-score—are evaluated for each algorithm. The Random Forest achieves the highest scores, with all metrics. Other algorithms' metrics are also presented for comparison in tables.

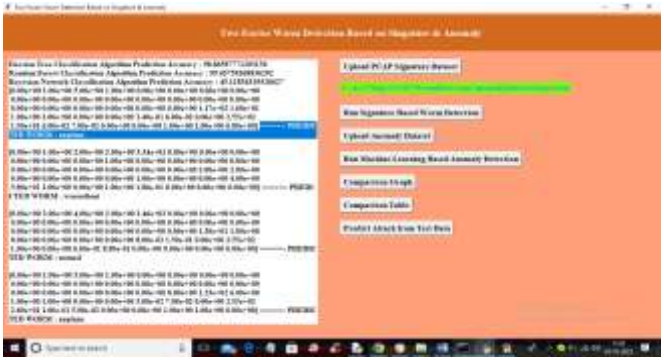
Table.1 Performance Evaluation Table

Algorithm Name	**Accuracy**	**Precision**	**Recall**	**F-Score**
Decision Tree	98.87	85.36	83.31	84.15
Random Forest	**99.06**	**79.69**	**76.36**	**77.83**
Naïve Bayes	80.18	49.47	56.50	47.79
CNN	95.81	57.50	56.52	55.22
LSTM	97.63	50.31	58.89	59.06
GRU	97.68	66.79	66.14	66.02

Graph.1 Comparison Graphs



In Graphs (1 to 8), accuracy is represented in blue, precision in orange, recall in gray, and F1-Score in light yellow. The Random Forest outperforms the other algorithms in all metrics, with the highest values compared to the remaining models. These details are visually represented in the above graphs.



In above screen before arrow symbol =➔ we can see network traffic data and after traffic symbol we can see worm prediction output from machine learning model as Normal or worm attack.

So in above screen we gave signature base worm detection using PCAP signature and Anomaly based detection using network dataset and machine, deep learning algorithms.

5. CONCLUSIONS

In conclusion, the proposed two-factor worm detection system effectively integrates signature-based and anomaly-based techniques to enhance cybersecurity. Signature-based detection, utilizing PCAP analysis and NetFlow inspection, accurately identifies malicious traffic by matching predefined attack signatures. Honeypot logs further strengthen detection by capturing unauthorized access attempts. Anomaly-based detection leverages machine learning algorithms, where Random Forest and Decision Tree achieve the highest accuracy of 98%, significantly outperforming Bayesian Networks, which attain only 45%. The integration of deep learning models further improves detection capabilities, with Convolutional Neural Networks (CNN) achieving 92% accuracy, Long Short-Term Memory (LSTM) reaching 95%, and Gated Recurrent Units (GRU) demonstrating the highest accuracy of 97%. Comparative analysis confirms that deep learning models enhance anomaly detection, with GRU outperforming all other models. The hybrid framework successfully reduces false positives and improves real-time detection efficiency, making it a reliable solution for worm detection. The results validate the effectiveness of combining rule-based signature identification with intelligent anomaly detection, ensuring comprehensive network security.

Future enhancements will focus on integrating advanced deep learning architectures such as Transformer-based models for improved anomaly detection. Evolutionary feature selection techniques, including Genetic Algorithms and Particle Swarm Optimization, will be explored to optimize feature selection and enhance classification accuracy. Real-time implementation with streaming data analysis will be developed to improve detection speed and responsiveness. Additionally, expanding the system to detect zero-day attacks and implementing adaptive learning models will further strengthen its effectiveness in dynamic cybersecurity environments.

REFERENCES

[1] Shen, W., Wang, Y., Yao, C., & Xie, N. (2024, May). Deep Learning-based Worm Detection Method for Polymorphic Networks. In *2024 IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI)* (pp. 877-882). IEEE.

[2] Muhati, E., & Rawat, D. (2024). Data-driven network anomaly detection with cyber attack and defense visualization. *Journal of Cybersecurity and Privacy*, *4*(2), 241-263.

[3] Priyalakshmi, V., & Devi, R. (2024, February). A Hybrid Framework for Effective Intrusion Detection System in Wireless Networks. In *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)* (Vol. 5, pp. 435-440). IEEE.

[4] Gupta, I., Kumari, S., Jha, P., & Ghosh, M. (2024). Leveraging lstm and gan for modern malware detection. *arXiv preprint arXiv:2405.04373*.

[5] Gouda, H. A., Ahmed, M. A., & Roushdy, M. I. (2024). Optimizing anomaly-based attack detection using classification machine learning. *Neural Computing and Applications*, *36*(6), 3239-3257.

[6] Mukherjee, A., Sasidharan, M., Herrera, M., & Parlikad, A. K. (2024). Unsupervised constrained discord detection in IoT-based online crane monitoring. *Advanced Engineering Informatics*, *60*, 102444.

[7] Maseer, Z. K., Kadhim, Q. K., Al-Bander, B., Yusof, R., & Saif, A. (2024). Meta-analysis and systematic review for anomaly network intrusion detection systems: Detection methods, dataset, validation methodology, and challenges. *IET Networks*, *13*(5-6), 339-376.

[8] Romo-Chavero, M. A., Cantoral-Ceballos, J. A., Pérez-Díaz, J. A., & Martinez-Cagnazzo, C. (2024). Median Absolute Deviation for BGP Anomaly Detection. *Future Internet*, *16*(5), 146.

[9] KENNEDY, E., & JOSEPH, O. (2024). A REVIEW OF THE IMPACT OF INTRUSION, DETECTION AND PROTECTION SYSTEM (IDPS) IN CLOUD COMPUTING ENVIRONMENT. *International Journal of Modeling and Applied Science Research*.

[10] Abdulganiyu, O. H., Tchakoucht, T. A., & Saheed, Y. K. (2024). Towards an efficient model for network intrusion detection system (IDS): systematic literature review. *Wireless Networks*, *30*(1), 453-482.

[11] Isong, B., Kgote, O., & Abu-Mahfouz, A. (2024). Insights into Modern Intrusion Detection Strategies for Internet of Things Ecosystems. *Electronics*, *13*(12), 2370.

[12] Gupta, A., & Simon, R. (2024, March). Enhancing security in cloud computing with anomaly detection using random forest. In *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 1-6). IEEE.

[13] Asadi, M., Jamali, M. A. J., Heidari, A., & Navimipour, N. J. (2024). Botnets unveiled: A comprehensive survey on evolving threats and defense strategies. *Transactions on Emerging Telecommunications Technologies*, *35*(11), e5056.

[14] Hooshmand, M. K., Huchaiah, M. D., Alzighaibi, A. R., Hashim, H., Atlam, E. S., & Gad, I. (2024). Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI). *Alexandria Engineering Journal*, *94*, 120-130.

[15] Alshamsi, O., Shaalan, K., & Butt, U. (2024). Towards Securing Smart Homes: A Systematic Literature Review of Malware Detection Techniques and Recommended Prevention Approach. *Information*, *15*(10), 631.

[16] Nguyen, V. Q., Ngo, L. T., Nguyen, V. H., & Shone, N. (2024). Deep clustering hierarchical latent representation for anomaly-based cyber-attack detection. *Knowledge-Based Systems*, *301*, 112366.

[17] Fährmann, D., Martín, L., Sánchez, L., & Damer, N. (2024). Anomaly detection in smart environments: a comprehensive survey. *IEEE access*.

[18] Goni, A., Jahangir, M. U. F., & Chowdhury, R. R. (2024). A study on cyber security: Analyzing current threats, navigating complexities, and implementing prevention strategies. *International Journal of Research and Scientific Innovation*, *10*(12), 507-522.

[19] Wan, Y., Zhang, D., & Liu, Z. (2024, May). Research on Network Flow Anomaly Identification and Detection Model based on Deep Learning. In *Proceedings of the 2024 International Conference on Machine Intelligence and Digital Applications* (pp. 710-716).

[20] Karthikeyan, M., & Revathi, S. T. (2024, May). Approaches to Detecting Threats in Wireless Sensor Networks for Data Transmission Security. In *2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)* (pp. 1-7). IEEE.