Optimizing Intrusion Detection Mechanisms for IoT Network Security

Ahmed Aziz[1,*], Sanjar Mirzaliev[1]

[1]Tashkent State Universtiy of Economics, Tashkent, Uzbekistan

Emails: a.mohamed@tsue.uz; sanjar2611@gmail.com

Abstract

The ubiquity of interconnected devices within the Internet of Things (IoT) paradigm has revolutionized modern connectivity, simultaneously amplifying the susceptibility of networks to diverse security threats. This study addresses the pressing necessity for robust intrusion detection mechanisms tailored for IoT networks. Utilizing a simulated dataset reflecting a spectrum of network intrusions within a military environment, the research employs sophisticated methodologies, notably harnessing Decision Tree (DT) algorithms optimized via Grey Wolf Optimization (GWO) for hyperparameter tuning. The investigation meticulously evaluates and refines intrusion detection mechanisms, emphasizing the pivotal role of feature importance analysis in fortifying network security. Results demonstrate the efficacy of the optimized DT algorithm in the precise classification of network traffic, illuminating key attributes instrumental for intrusion detection. These findings underscore the significance of adaptive and interpretable detection strategies in mitigating evolving threats within IoT networks, advocating for resilient approaches to bolster network security.

Keywords: Intrusion Detection; Internet of Things (IoT); Network Security; Cybersecurity Measures; Threat Detection; Security Frameworks; Intrusion Mitigation; IoT Devices; Cyber Threats; Network Vulnerabilities; Security Optimization; Anomaly Detection; Machine Learning.

1. Introduction

The Internet of Things (IoT) has brought about unprecedented convenience and efficiency in various sectors through the proliferation of interconnected devices. However, this rapid growth has also exposed IoT networks to numerous security vulnerabilities that can be exploited by malicious actors [1]. Security breaches in IoT networks are highly risky as they may lead to loss of sensitive data, disruption of operations, and violation of user privacy [2]. Intrusion detection mechanisms in these networks have become crucial safeguards that help identify and mitigate unauthorized access, malicious activities, and anomalies. It is important to improve these mechanisms so as to strengthen the security of IoT networks against emerging cyber threats [3].

The evolving paradigm of IoT technology has revolutionized connectivity, interlinking devices, and systems, thereby fostering unprecedented levels of efficiency and functionality across industries. Nevertheless, this interconnectedness has introduced a complex web of security challenges, as IoT devices become susceptible targets for cyber intrusions and exploitations [4-5]. Intrusion detection mechanisms serve as critical defense mechanisms against these threats, acting as sentinels tasked with identifying and thwarting unauthorized access, anomalous behaviors, and potential breaches within IoT networks. To ensure the integrity, confidentiality, and availability of data transmitted across these networks, optimizing intrusion detection mechanisms becomes an imperative endeavor [3-6].

The exponential growth of interconnected systems has been driven by the widespread integration of IoT devices in various domains such as healthcare, manufacturing, smart homes, and transportation. However, this rapid expansion has also led to a corresponding increase in vulnerabilities that expose these networks to different cyber threats. The

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

interconnection of IoT devices is often characterized by heterogeneous architectures and different security protocols, which makes it difficult to maintain a unified and strong security framework [6-8]. This complex network of devices combined with inherent security gaps further emphasizes the need for strengthening intrusion detection mechanisms that are specifically designed for the unique complexities of IoT environments [9].

The first line of defense against possible cyber threats in IoT ecosystems is intrusion detection mechanisms. These mechanisms include a range of techniques such as anomaly detection, signature-based detection, and machine learning algorithms that can identify patterns that indicate unauthorized access or malicious activities [10-13]. Detecting and mitigating intrusions in real-time is crucial for protecting IoT networks in advance, thus preventing data breaches, service interruptions or compromise of critical infrastructure. However, optimizing these mechanisms is a complex task that requires a balanced approach to efficiency, accuracy, and adaptability when dealing with emerging threats [14-15].

2. System Design

The objective of this section is to provide a complete plan for developing and optimizing intrusion detection systems that are specifically designed for the dynamic and heterogeneous nature of IoT networks.

2.1. System Model

This research proposes a system model that takes a holistic approach to strengthening intrusion detection mechanisms in IoT networks. The model is based on a tiered architecture that includes edge devices, gateways, and a centralized monitoring and analysis unit. Edge devices are the first point of contact for network interaction, which collects and sends data to the gateways that act as intermediaries between devices and the central unit. The central unit coordinates intrusion detection activities using a combination of signature-based detection, anomaly detection algorithms, and machine learning models for real-time analysis. Additionally, the system model incorporates a dynamic database that contains normal network behavior patterns and known attack vectors to enable adaptive responses to emerging threats. By leveraging this hierarchical structure and incorporating scalable, lightweight algorithms specifically designed for resource-constrained IoT devices, the system model aims to achieve higher accuracy and responsiveness in detecting and mitigating intrusions across various IoT environments [16-17].

2.2. Threat Model

Our work's threat model has been delineated by considering a range of potential threats that could target IoT networks. These include but are not limited to distributed denial-of-service (DDoS) attacks, malware propagation, unauthorized access attempts, and insider threats. The threat landscape in IoT networks is complex due to the variety of connected devices and communication protocols. Threat actors can exploit vulnerabilities in device firmware, and unsecured communication channels or use sophisticated techniques to bypass existing security measures [18]. In addition, the dynamic nature of IoT ecosystems poses challenges in monitoring and protecting against evolving threats, thus requiring a threat model that encompasses both known and emergent attack vectors. To address these threats within the proposed system model, continuous threat intelligence gathering, proactive vulnerability assessments and adaptive mechanisms capable of detecting anomalous behaviors amidst the complexity of IoT network traffic are necessary [19].

3. Methodology

The section outlines the systematic approach used to design, implement, and evaluate intrusion detection mechanisms for IoT network security.

Preparation of IoT network data for intrusion detection involves a series of preprocessing steps aimed at refining raw data into a suitable format for analysis and modeling. The first step involved a thorough cleansing of the data to address missing values, outliers, and inconsistencies. Missing data points were handled using imputation techniques to ensure that the dataset remained intact. Afterward, feature engineering techniques were implemented to extract relevant attributes and create new features that encapsulate pertinent information for intrusion detection. This included transforming categorical variables into numerical representations through encoding methods such as one-hot encoding

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

or label encoding. Furthermore, scaling and normalization techniques were applied to standardize feature magnitudes, ensuring uniformity and mitigating the influence of features with disparate scales on the learning process of the intrusion detection models [5]. Dimensionality reduction methods such as Principal Component Analysis (PCA) or feature selection algorithms were also employed to streamline the dataset by retaining informative attributes while reducing computational complexity. These preprocessing steps collectively aimed to enhance the quality, relevance, and suitability of the IoT network data for subsequent model training and evaluation phases [9].

The Decision Tree (DT) algorithm is a non-parametric supervised learning method commonly used in classification tasks, which provides interpretability and flexibility in modeling complex relationships within datasets. In the context of intrusion detection in IoT networks, DTs are effective classifiers that can differentiate between normal network behavior and abnormal or potentially malicious activities. The hierarchical structure of a DT consists of decision nodes that split data based on feature attributes, leading to terminal nodes or leaves representing class labels. In the case of network traffic classification, the DT algorithm recursively divides the dataset based on features such as source and destination IP addresses, port numbers, protocol types, packet sizes, and timings [20]. This hierarchical segmentation enables the creation of decision rules that can distinguish benign from suspicious network traffic patterns. The algorithm's ability to handle categorical and numerical features as well as its capability to handle missing values with little pre-processing makes it suitable for IoT network datasets which are complex and diverse in nature.

The trained DT model in the classification process uses the learned decision rules to traverse the tree structure and classify incoming network traffic into predefined categories such as normal or anomalous behavior. Each decision node evaluates specific features, and as the data progresses through the tree, it ultimately reaches a leaf node representing the classification outcome. DTs offer interpretability, enabling analysts to comprehend the sequence of decisions made by the model, and providing valuable insights into the features most crucial for distinguishing between different classes of network traffic. Hyperparameter optimization is important in fine-tuning machine learning algorithms for optimal performance. In optimizing the DT algorithm for intrusion detection in IoT networks, Grey Wolf Optimizer (GWO) serves as a metaheuristic algorithm that can effectively explore and optimize hyperparameters governing the DT model.

Steps of GWO for Hyperparameter Optimization:

Initialization: GWO commences by initializing a population of grey wolves, mirroring the hierarchical structure of the wolf pack—alpha, beta, delta, and omega wolves represent the top solutions within the search space.

Encoding Hyperparameters: Each wolf corresponds to a potential solution vector, encoding the hyperparameters of the DT model. These hyperparameters include but are not limited to the maximum depth of the tree, the minimum number of samples required to split a node, and criteria for node splitting (e.g., Gini impurity or entropy).

Fitness Evaluation: Evaluate the fitness of each wolf (solution vector) by employing the DT model with the encoded hyperparameters on a subset of the training data using cross-validation techniques. The fitness function typically quantifies the performance of the DT model, considering metrics such as accuracy, precision, recall, or F1-score.

Updating Alpha, Beta, Delta, and Omega: Wolves update their positions within the search space based on fitness scores, mimicking the leadership hierarchy observed in wolf packs. Alpha, beta, delta, and omega wolves represent the best solutions encountered during the optimization process.

Exploration and Exploitation: Wolves utilize search mechanisms inspired by hunting behaviors—encouraging exploration to discover new promising regions while exploiting known favorable areas. This balance between exploration and exploitation aids in efficiently navigating the hyperparameter space to converge toward optimal or near-optimal configurations.

Termination Criteria: The optimization process iterates until a predefined termination criterion is met, such as a maximum number of iterations or convergence to a satisfactory solution.

Applying the GWO algorithm for hyperparameter optimization of the DT model allows for the automatic selection or tuning of hyperparameters, enhancing the performance and robustness of the intrusion detection system within IoT

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

networks. The optimized DT model derived from GWO-assisted hyperparameter tuning exhibits improved generalization and accuracy in classifying network traffic, thereby fortifying the overall effectiveness of the intrusion detection mechanisms [20].

4. Simulation Setup

This section goes into the details of how to simulate different network conditions and cyber threats, including the methodologies, tools, and parameters used. In our experiments, we use a public case study of network attacks that was designed to include a wide range of simulated intrusions in a military network environment. This dataset was intentionally created to mimic a typical US Air Force Local Area Network (LAN) which is an actual operational environment that has been subjected to various types of attacks. The data set collection involved capturing raw TCP/IP dump data encapsulating network activities within the LAN. In this simulated environment, each connection represents a sequence of TCP packets spanning a specific time interval, delineating the exchange of data between source and target IP addresses under predefined protocols. These connections were carefully labeled as either ‘normal’ or belonging to a particular attack type, thus ensuring binary categorization for each connection record. The dataset consists of about 100 bytes per record and contains 41 different quantitative and qualitative features extracted from both normal and attack instances. Within these features, three are qualitative, while the remaining 38 are quantitative in nature. The class variable embedded within the dataset enables classification into two distinct categories: 'Normal' representing benign network behavior, and 'Anomalous' denoting instances indicative of potential intrusions or attacks.

To evaluate the detection performance of our mode, we use the following performance metrics.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall(Sensitivity)} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{3}$$

$$\text{F1} - \text{score} \ \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

We offer an overview of the hardware and software settings for the simulation setup in Table 1 so that you can study the specifications of our experimental design.

Table 1: Implementation setups reviewed.

S/W	OS	Python	SK-Learn
Specifications	Windows 10	3.8.0	0.2
H/W	HDD	RAM	CPU
Specifications	1T	64GB	Intel Core i5-13600K

Moreover, we provide an inclusive overview of hyper-parameters of ML models in Table 2, to emphasize the transparency of simulation setups of our experiments.

Table 2: Summary of models’ hyper-parameters in our experiments.

Hyper-parameter	Setting
USI	ce46
Use GPU	FALSE
Stratify Train-Test	FALSE
Shuffle Train-Test	TRUE
session_id	421
Remove Perfect Collinearity	TRUE
Original Data	41
Ordinal Features	FALSE
Numeric Imputer	mean
Numeric Features	3
Missing Values	TRUE

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

Log Experiment	FALSE
Label Encoded	None
Iterative Imputation Numeric Model	None
Iterative Imputation Iteration	None
Iterative Imputation Categorical Model	None
Imputation Type	simple
High Cardinality Method	None
High Cardinality Features	FALSE
Fold Generator	StratifiedKFold
Fix Imbalance Method	SMOTE
Feature Selection Method	classic
Experiment Name	clf-default-name
CPU Jobs	-1
Categorical Imputer	constant
Categorical Features	5

5. Results and Discussion

The results of the experiment are presented in this section, which examines the quantitative and qualitative evaluations of the intrusion detection system's performance under different scenarios and attack vectors. Table 1 summarizes the results of extensive statistical analyses that were conducted to quantitatively evaluate the performance of proposed intrusion detection mechanisms. This table is a collection of important metrics such as detection accuracy, false positive rates, and response times that were carefully gathered and analyzed across various simulation scenarios. The statistical analysis in Table 1 shows how effective the system is at detecting abnormal activities and possible intrusions into IoT networks with different conditions.

Table 1: Statistical Analysis of Traffic in IoT Networks

	count	mean	std	min	0.25	0.5	0.75	max
duration	25192	305.0541	2686.556	0	0	0	0	42862
src_bytes	25192	24330.63	2410805	0	0	44	279	3.82E+08
dst_bytes	25192	3491.847	88830.72	0	0	0	530.25	5151385
land	25192	0.000079	0.00891	0	0	0	0	1
wrong_fragment	25192	0.023738	0.260221	0	0	0	0	3
urgent	25192	0.00004	0.0063	0	0	0	0	1
hot	25192	0.198039	2.154202	0	0	0	0	77
num_failed_logins	25192	0.001191	0.045418	0	0	0	0	4
logged_in	25192	0.394768	0.488811	0	0	0	1	1
num_compromised	25192	0.22785	10.41735	0	0	0	0	884
root_shell	25192	0.001548	0.039316	0	0	0	0	1
su_attempted	25192	0.00135	0.048785	0	0	0	0	2
num_root	25192	0.249841	11.50084	0	0	0	0	975
num_file_creations	25192	0.014727	0.529602	0	0	0	0	40

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

num_shells	25192	0.000357	0.018898	0	0	0	0	1
num_access_files	25192	0.004327	0.098524	0	0	0	0	8
num_outbound_cmds	25192	0	0	0	0	0	0	0
is_host_login	25192	0	0	0	0	0	0	0
is_guest_login	25192	0.00913	0.095115	0	0	0	0	1
count	25192	84.5911	114.6735	1	2	14	144	511
srv_count	25192	27.69875	72.46824	1	2	8	18	511
serror_rate	25192	0.286338	0.447312	0	0	0	1	1
srv_serror_rate	25192	0.283762	0.447599	0	0	0	1	1
rerror_rate	25192	0.11863	0.318745	0	0	0	0	1
srv_rerror_rate	25192	0.12026	0.322335	0	0	0	0	1
same_srv_rate	25192	0.660559	0.439637	0	0.09	1	1	1
diff_srv_rate	25192	0.062363	0.17855	0	0	0	0.06	1
srv_diff_host_rate	25192	0.095931	0.256583	0	0	0	0	1
dst_host_count	25192	182.5321	98.9939	0	84	255	255	255
dst_host_srv_count	25192	115.063	110.6469	0	10	61	255	255
dst_host_same_srv_rate	25192	0.519791	0.448944	0	0.05	0.51	1	1
dst_host_diff_srv_rate	25192	0.082539	0.187191	0	0	0.03	0.07	1
dst_host_same_src_port_rate	25192	0.147453	0.308367	0	0	0	0.06	1
dst_host_srv_diff_host_rate	25192	0.031844	0.110575	0	0	0	0.02	1
dst_host_serror_rate	25192	0.2858	0.445316	0	0	0	1	1
dst_host_srv_serror_rate	25192	0.279846	0.446075	0	0	0	1	1
dst_host_rerror_rate	25192	0.1178	0.305869	0	0	0	0	1
dst_host_srv_rerror_rate	25192	0.118769	0.317333	0	0	0	0	1

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

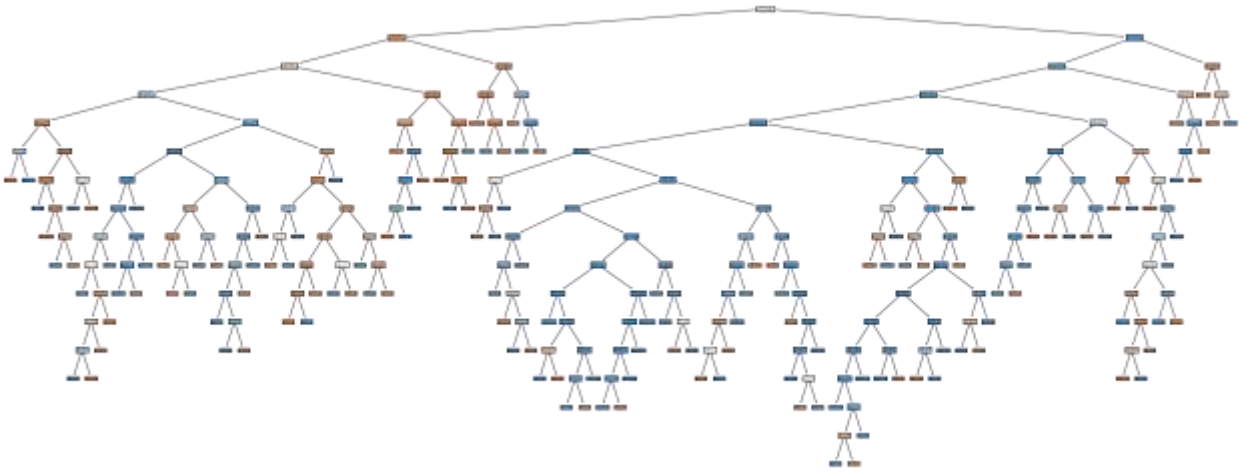

Figure 1: Decision Tree Model for Intrusion Detection in IoT Networks

The decision tree model used in this study is shown in Figure 1. It demonstrates the hierarchical representation of decision nodes and branches derived from the applied machine learning algorithm. This decision tree is a visual representation of the decision-making process that underlies intrusion detection mechanisms in IoT networks. By showing the architecture of the decision tree, this figure provides insights into the interpretability and structure of the algorithm, which helps to understand why intrusion detection classifications were made in this study. The feature importance chart in Figure 2 shows how important each feature is in the applied decision tree algorithm. It shows which features are most important for intrusion detection in IoT networks. The feature importance chart helps to understand the order of features and what is important for making decisions by the intrusion detection system. This visual representation makes the algorithm's feature selection process more interpretable, and it emphasizes the key attributes that are essential for effective intrusion detection as shown in Figure 2.

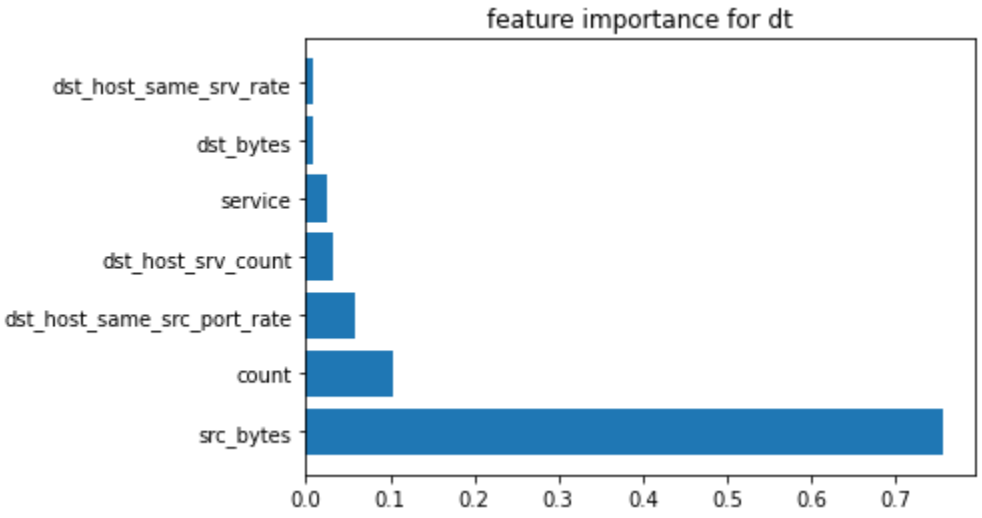

Figure 2: Feature Importance Analysis within the Decision Tree Model for IoT Network Intrusion Detection

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

6. Conclusion

This research shows the importance of strong intrusion detection systems in IoT networks, which are essential for protecting against various threats. The use of advanced techniques such as Decision Tree (DT) algorithms optimized by Grey Wolf Optimization (GWO) for hyperparameter tuning is a promising development in network security. Our study demonstrates the effectiveness of the proposed methodology by using a carefully simulated dataset that represents different types of network intrusions in a military setting. The results show that the optimized DT algorithm can accurately classify network traffic and that feature importance analysis is important for identifying key attributes for intrusion detection. These findings add to the range of methods available to strengthen IoT network security, highlighting the need for adaptive, robust, and interpretable intrusion detection mechanisms in an ever-changing threat landscape.

References

[1] Chaabouni, Nadia, Mohamed Mosbah, Akka Zemmari, Cyrille Sauvignac, and Parvez Faruki. 2019. "Network Intrusion Detection for IoT Security Based on Learning Techniques." IEEE Communications Surveys and Tutorials. https://doi.org/10.1109/COMST.2019.2896380.

[2] Otoum, Yazan, Dandan Liu, and Amiya Nayak. 2022. "DL-IDS: A Deep Learning–Based Intrusion Detection Framework for Securing IoT." Transactions on Emerging Telecommunications Technologies. https://doi.org/10.1002/ett.3803.

[3] Kan, Xiu, Yixuan Fan, Zhijun Fang, Le Cao, Neal N Xiong, Dan Yang, and Xuan Li. 2021. "A Novel IoT Network Intrusion Detection Approach Based on Adaptive Particle Swarm Optimization Convolutional Neural Network." Information Sciences 568: 147–62.

[4] Liu, Jingyu, Dongsheng Yang, Mengjia Lian, and Mingshi Li. 2021. "Research on Intrusion Detection Based on Particle Swarm Optimization in IoT." IEEE Access 9: 38254–68.

[5] Fatani, Abdulaziz, Mohamed Abd Elaziz, Abdelghani Dahou, Mohammed A A Al-Qaness, and Songfeng Lu. 2021. "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization." IEEE Access 9: 123448–64.

[6] Luo, Ke. 2023. "A Distributed SDN-Based Intrusion Detection System for IoT Using Optimized Forests." Plos One 18 (8): e0290694.

[7] Fatani, Abdulaziz, Abdelghani Dahou, Mohammed A A Al-Qaness, Songfeng Lu, and Mohamed Abd Elaziz. 2021. "Advanced Feature Extraction and Selection Approach Using Deep Learning and Aquila Optimizer for IoT Intrusion Detection System." Sensors 22 (1): 140.

[8] Yang, Aimin, Yunxi Zhuansun, Chenshuai Liu, Jie Li, and Chunying Zhang. 2019. "Design of Intrusion Detection System for Internet of Things Based on Improved BP Neural Network." Ieee Access 7: 106043–52.

[9] Ramaiah, Mangayarkarasi, Vanmathi Chandrasekaran, Vinayakumar Ravi, and Neeraj Kumar. 2021. "An Intrusion Detection System Using Optimized Deep Neural Network Architecture." Transactions on Emerging Telecommunications Technologies 32 (4): e4221.

[10] A. Metwaly, A. and Elhenawy, I. (2023) "Protecting IoT Devices from BotNet Threats: A Federated Machine Learning Solution", Sustainable Machine Intelligence Journal, 2. doi: 10.61185/SMIJ.2023.22105.

[11] Liang, Wei, Kuan-Ching Li, Jing Long, Xiaoyan Kui, and Albert Y Zomaya. 2019. "An Industrial Network Intrusion Detection Algorithm Based on Multifeature Data Clustering Optimization Model." IEEE Transactions on Industrial Informatics 16 (3): 2063–71.

[12] Nazir, Anjum, and Rizwan Ahmed Khan. 2021. "A Novel Combinatorial Optimization Based Feature Selection Method for Network Intrusion Detection." Computers \& Security 102: 102164.

[13] Savanović, Nikola, Ana Toskovic, Aleksandar Petrovic, Miodrag Zivkovic, Robertas Damaševičius, Luka Jovanovic, Nebojsa Bacanin, and Bosko Nikolic. 2023. "Intrusion Detection in Healthcare 4.0 Internet of Things Systems via Metaheuristics Optimized Machine Learning." Sustainability 15 (16): 12563.

[14] Aziz, Mohammad R, and Ali Saeed Alfoudi. 2023. "Different Mechanisms of Machine Learning and Optimization Algorithms Utilized in Intrusion Detection Systems." In AIP Conference Proceedings. Vol. 2839.

[15] Roy, Souradip, Juan Li, Bong-Jin Choi, and Yan Bai. 2022. "A Lightweight Supervised Intrusion Detection Mechanism for IoT Networks." Future Generation Computer Systems 127: 276–85.

[16] Phalguna Krishna, Ediga Sathyanarayana, and Thangavelu Arunkumar. 2021. "Hybrid Particle Swarm and Gray Wolf Optimization Algorithm for IoT Intrusion Detection System." International Journal of Intelligent Engineering \& Systems 14 (4).

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023

[17] Kunang, Yesi Novaria, Siti Nurmaini, Deris Stiawan, and Bhakti Yudho Suprapto. 2021. "Attack Classification of an Intrusion Detection System Using Deep Learning and Hyperparameter Optimization." Journal of Information Security and Applications 58: 102804.
[18] Kavitha, S, N Uma Maheswari, and R Venkatesh. 2023. "Intelligent Intrusion Detection System Using Enhanced Arithmetic Optimization Algorithm with Deep Learning Model." Tehnički Vjesnik 30 (4): 1217–24.
[19] Kumar, Ravinder, Amita Malik, and Virender Ranga. 2022. "An Intellectual Intrusion Detection System Using Hybrid Hunger Games Search and Remora Optimization Algorithm for IoT Wireless Networks." Knowledge-Based Systems 256: 109762.
[20] Abdel-Basset, M., Hawash, H., Chakrabortty, R. K., & Ryan, M. J. (2021). Semi-supervised spatiotemporal deep learning for intrusions detection in IoT networks. IEEE Internet of Things Journal, 8(15), 12251-12265.

Doi : https://doi.org/10.54216/JCIM.130106
Received: May 22, 2023 Revised: August 16, 2023 Accepted: December 19, 2023