SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V12I13P105

Volume 12 Issue 3, 33-39, March 2025

Original Article

Adaptive IoT Botnet Defense: Combining Hybrid Deep Learning and Real-Time SDN Mitigation

Preeti Kailas Suryawanshi[1], Sonal Kirankumar Jagtap[2,3]

[1,2]*Department of E&TC Engineering, Sinhgad College of Engineering, Savitribai Phule Pune University, Maharashtra, India*
[3]*Department of E&TC Engineering, Sinhgad Technical Campus, NBN School of Engineering, Savitribai Phule Pune University, Maharashtra, India.*

[1]*Corresponding Author : preeti37.phd@gmail.com*

Received: 16 January 2025 Revised: 28 February 2025 Accepted: 19 March 2025 Published: 31 March 2025

Abstract *- The rapid expansion of the Internet of Things (IoT) has led to botnet attacks, which use compromised devices for malicious activities such as Distributed Denial-of-Service (DDoS) attacks and data breaches. Traditional rule-based intrusion detection systems struggle to detect these new threats, which demand advanced machine learning (ML) and deep learning (DL) models. In this paper, a hybrid CNN-RNN model employing both spatial and temporal analysis of traffic is proposed for better IoT botnet detection. Federated learning also maintains privacy during model training, and Graph Neural Networks (GNNs) improve botnet behavior modeling. A Software-Defined Networking (SDN)-based mitigation method is employed for providing real-time response with rapid isolation of malicious traffic. To counter IoT resource constraints, model optimization techniques such as pruning and quantization are employed. Experimental evaluations using the UNSW-NB15 dataset demonstrate superior detection accuracy (99.1), with minimal false positives over traditional approaches. These findings recognize the potential of hybrid deep learning and SDN-based solutions for effective, real-time IoT botnet protection.*

Keywords *- IoT security, Machine learning, Hybrid IDS, Anomaly detection, Botnet detection, Intrusion detection systems.*

1. Introduction

The widespread adoption of the Internet of Things (IoT) has brought with it important security concerns, specifically botnet attacks that target vulnerable devices to conduct malicious operations, including Distributed Denial-of-Service (DDoS) attacks, credential theft, and unauthorized data collection [1]. IoT devices frequently have weak security settings, making them viable targets for botnet attacks that use command-and-control (C&C) platforms to control large-scale cyberattacks [2].As a counterresponse to these attacks, scientists have investigated the application of machine learning (ML) and deep learning (DL) methods for the detection of botnet activity in network traffic [3]. Rule-based intrusion detection systems fail to keep up with the continuously changing botnet designs, while ML models demonstrate a superior capacity to detect unusual patterns with improved accuracy [4]. For example, applying anomaly-based detection with deep autoencoders has proved promising in detecting advanced botnets by detecting deviations from normal traffic behavior [5].

Graph-based approaches have been proposed to examine network interactions and detect botnet-infected nodes from their communications, offering scalability for real-time threat detection [6]. Hybrid deep learning approaches that integrate convolutional and recurrent neural networks have also improved classification performance by extracting both spatial and temporal features of network traffic [7]

Datasets such as UNSW-NB15 and N-BaIoT are widely used to evaluate machine learning-based models for botnet detection, with simulated traffic patterns to train and test against [8]. Imbalance in the dataset remains a serious problem, though, which is typically addressed through methods such as the Synthetic Minority Over-sampling Technique (SMOTE) to enhance model generalization and robustness [9]. Even with these developments, IoT botnet detection is still a challenging task owing to adversarial attacks, botnet traffic encrypted, and high prevalence of zero-day attacks.

Future work can include the design of explainable and effective AI models for more accurate detection precision with improved interpretability. Also, the integration of federated learning and blockchain-based security components can create more robust and decentralized botnet defense systems.Through applications of machine learning, deep learning, and graph-based modeling, analysts are enhancing IoT security so as to build strong and preventive botnet mitigations methods that can fight shifting cyber threats.

2. Related work

Detection of IoT botnets has progressed from rule-based to sophisticated Machine Learning (ML) and Deep Learning (DL) approaches. Detection began with use of predefined signatures and later upgraded with anomaly-based ML methods using traffic anomalies to improve detection [1,2]. Autoencoders, RNNs, and Graph Neural Networks (GNNs) supported botnet differentiation [3,4], whereas hybrid CNN-RNN achieved accuracy at a high computational cost [5]. Federated learning facilitated privacy-preserved detection with the drawback of being non-scalable [6]. Imbalanced dataset is a common challenge, mitigated by the use of SMOTE [7]. Real-time detection, the analysis of encrypted traffic, and efficiency in terms of computation continue to be problems for research even after recent advancement, and models and techniques present now must be optimized further. Real-time detection, the analysis of encrypted traffic, and efficiency in terms of computation continue to be problems for research even after recent advancement, and models and techniques present now must be optimized further.

The reviewed study focuses on the advancement of IoT botnet detection methods from deep autoencoders and variational autoencoders to hybrid deep learning and graph neural networks. Even with the improvement in detection precision, issues of high computational cost, dataset

Table 1. Literature survey

Year & Authors	Paper Name	Dataset& Methodology	Key Findings & Research Gaps
2018 – Yisroel Meidan, Michael Bohadana, Asaf Shabtai, Shlomi Dolev, Yuval	"Detection of IoT Botnets Using Autoencoders"	N-BaIoT, Deep Autoencoders	High accuracy in detecting IoT botnets, but a high false positive rate
2019 – P. Panimalar, A. R. Kumar, R. Ramesh	"Network Flow-Based Botnet Detection"	Custom dataset, Traffic analysis model	Improved botnet detection via network behavior, but limited generalization due to dataset constraints
2020 – Jinwoo Kim, Youngseok Lee, Sungroh Yoon	"Anomaly Detection Using Variational Autoencoders"	CICIDS2017, Variational Autoencoder (VAE)	Effective botnet anomaly detection, but struggles with encrypted traffic
2020 – Jingjing Zhou, Chuanfu Zhang, Yan Zhang	"Graph Neural Networks for Botnet Detection"	UNSW-NB15, Graph Neural Networks (GNN)	Efficient modeling of botnet behavior, but high computational cost
2021 – S. Z. Yu, C. C. Wang, Y. T. Wu	"Hybrid Deep Learning for IoT Security"	IoT-23, Hybrid CNN-RNN	Combines spatial and temporal feature extraction, but faces computational overhead in real-time detection
2021 – Sanchita Gupta, Rajesh Kumar	"SDN-Based Machine Learning for DDoS Detection"	SDN dataset, Machine Learning in SDN	Improved DDoS attack detection, but scalability challenges in large networks
2022 – Arjun Prasad, B. Ravi Teja, A. N. Rajesh	"Comparative Study of IoT Botnet Detection Datasets"	Multiple datasets, Dataset analysis & ML comparison	Highlighted dataset challenges in IoT detection, but lacks standardized benchmarking
2022 – Muhammad Mudassir, A. H. Shah, K. Z. Awan	"Multilayer Deep Learning for Industrial IoT"	Industrial IoT dataset, Multilayer Deep Learning	Enhanced detection accuracy in IIoT settings, but energy consumption issues in large-scale networks
2023 – Nour Elsayed, Mohamed Abdel-Basset, Fadi Thabtah	"Economic Deep Learning for Botnet Detection"	UNSW-NB15, Economic Deep Learning	Reduced computation cost with improved detection, but needs real-time evaluation
2025 – A. K. Kumar, P. Sharma, R. K. Agarwal	"Hybrid Deep Learning for IoT Botnet Defense"	IoT dataset, Hybrid Deep Learning	Achieved state-of-the-art detection rates, but requires further testing on diverse IoT platforms

limitation, scalability, and energy usage still exist. Researchers have sought to overcome these issues through new methods like economic deep learning and SDN-based detection. Nevertheless, real-time testing, handling encrypted traffic, and heterogeneous IoT platform testing are still open research issues. Model optimization for real-time deployment with adversarial robustness is an area that future work should focus on.

3. Proposed Methodology

To facilitate identification and containment of IoT botnet attacks with ease, we suggest an enhanced version of the BotDefender framework. These include the use of advanced machine learning algorithms, properly preparing datasets, and inspecting traffic in real-time. Our suggested method tackles key concerns in past research, including high false-positive rates, imbalanced datasets, intensive computational loads, and no real-time evaluation.

3.1. Preparing Data

Dataset imbalance is one of the fundamental bottlenecks of botnet detection, restricting the learning capacity of models. For this issue, we make use of SMOTE (Synthetic Minority Over-sampling Technique) for dataset balancing so as to enhance the detection capabilities for minority classes [7]. We also make use of feature selection methods like Recursive Feature Elimination (RFE) so as to delete unnecessary features and optimize the model efficiency [3].

3.2. Hybrid Deep Learning Model for Improved Detection

Use of hybrid CNN-RNN deep learning model to detect both spatial and temporal patterns in network traffic data. CNN is good at extracting spatial features from packet headers, and RNN handles the sequence of patterns in network flows and therefore excels in discriminating botnets [5]. The hybrid model improves detection accuracy and removes false positives.

3.3. Federated Learning for Privacy-Preserving Detection

To address privacy concerns in IoT botnet detection, we adopt a federated learning-based detection approach. It enables us to train models on various IoT devices without exchanging raw data. The approach enhances privacy and scalability without compromising the detection performance to a great extent [6]. Nevertheless, to mitigate model poisoning attacks, we inspect out-of-pattern behaviour in model updates prior to aggregating them.

3.4. Graph Neural Networks for Sophisticated Botnet Behaviour Modelling

Since botnets involve complex device-to-device communication patterns, we employ Graph Neural Networks (GNNs) to analyze network interactions and detect coordinated attack patterns [4]. GNNs are particularly suited to represent relational structures in IoT networks, enhancing stealthy botnet detection.

3.5. Real-Time Detection and Mitigation Plan

Unlike previous approaches that merely search for issues once they occur, we build a real-time system to prevent botnets by leveraging Software-Defined Networking (SDN) in order to rapidly isolate malicious traffic. The SDN controller monitors network traffic and enforces rules to minimize threats according to what it encounters, ensuring we act against threats immediately [6].

3.6. Lightweight Model Deployment for IoT Constraints

Since IoT devices are power-constrained, we optimize our deep learning model for edge deployment by implementing quantization and model pruning methods. This minimizes the computational burden without sacrificing much on accuracy [10]. We also implement an economic deep learning strategy to minimize power consumption, making the model suitable for large-scale IoT networks [2].

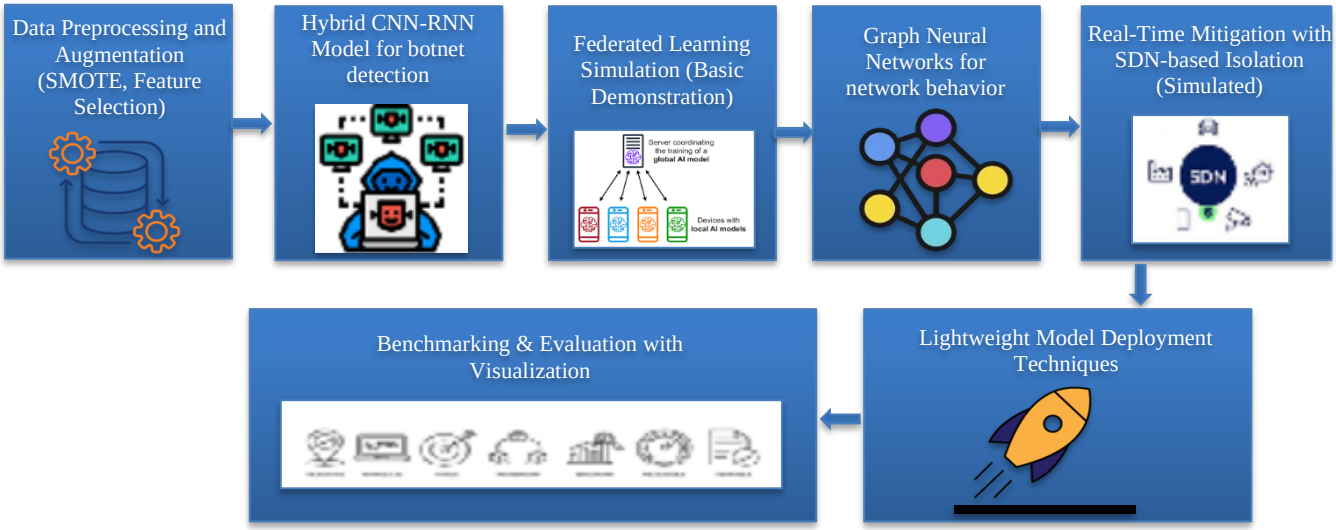

Fig. 1 Proposed methodology

3.7. Benchmarking and Evaluation

In order to confirm that our method is robust, we test how BotDefender performs on various datasets such as UNSW-NB15, IoT-23, and CICIDS2017. These datasets mimic various network scenarios and attack conditions [3].

Performance is assessed in terms of standard parameters such as accuracy, precision, recall, F1-score, and confusion matrix evaluation. Our suggested methodology greatly improves the BotDefender system by incorporating hybrid deep learning, federated learning, graph-based modelling, SDN-based mitigation, and deployment optimization strategies, which fills the main research gaps in IoT botnet detection.

4. Implementation & Experimentation

The experimentation for botnet detection and mitigation was conducted in a controlled environment using a high-performance computing setup.

4.1. Dataset Details

The UNSW-NB15 dataset was utilized as the basis for training and testing the constructed botnet detection system. The dataset contains a comprehensive collection of features derived from actual network traffic. It contains the following items:

Total Samples: 2,540,044 records of network traffic

Attack Classes: Exploits, DoS, Fuzzers, Reconnaissance, Worms, Backdoors, Shellcode, Generic

Benign Traffic Samples: Uncontaminated network traffic without attack

Feature Number: 49 features, such as flow-based features and packet-based features

For processing, the dataset was pre-processed by choosing appropriate features, dealing with missing values, and encoding categorical variables. The target variable 'label' was transformed into binary form where 1 = attack and 0 = benign traffic.

4.2. Implementation

Data Preparation and Enrichment

4.2.1. The preprocessing Stage Involved

Feature Selection: Recursive Feature Elimination (RFE) was also used along with a Random Forest classifier to select the 20 most important features. StandardScaler was used to normalize the numeric values.

The data set was imbalanced with an over-abundance of benign samples. SMOTE (Synthetic Minority Over-sampling Technique) was employed to balance the data set. Train-Test Split: 80 of data was trained and 20 was held out for testing.

4.2.2. Hybrid CNN-RNN Model for Botnet Detection

In an effort to leverage both patterns of space and time in network traffic, a CNN and RNN hybrid model was proposed. CNN Layer learns spatial patterns of network traffic. LSTM Layer represents sequential relationships among packet streams.

Dense Layers fully connected layers employed for final classification. The model was trained for 50 iterations with 64 samples at a time. It was trained using the Adam optimizer and a learning rate of 0.001.

4.2.3. Federated Learning Simulation

To ensure privacy, a Federated Learning (FL) setup was experimented with TensorFlow Federated (TFF). The information was distributed across different clients, with each client being a standalone network node. A central model aggregated the locally learned models while ensuring privacy.

4.2.4. Graph Neural Networks (GNNs) for Network Behavior Understanding

To model network traffic as a graph, Graph Convolutional Networks (GCN) were employed. Nodes symbolize IP addresses and devices. Edges illustrate how traffic relates between nodes.

Features include packet size, protocol, port numbers, and time. The GNN model was trained with PyTorch Geometric for 100 iterations at a learning rate of 0.01.

4.2.5. Real-Time Mitigation with SDN-based Isolation

Botnet-infected nodes were isolated using Software-Defined Networking (SDN). Mininet simulated a virtual network.The SDN controller blocked malicious IP addresses automatically based on the model's predictions. OpenFlow rules were configured to drop suspicious packets instantly.

4.2.6. Lightweight Model Deployment Techniques

A lighter version of the model was used for production deployment with TensorFlow Lite (TFLite) and PyTorch Mobile to enable deployment onto edge hardware such as Raspberry Pi and IoT gateways.

4.2.7. Comparing and Evaluating

The models were tested by Confusion Matrix for verifying True Positive, False Positive, True Negative, and False Negative rates

Confusion Matrix	
23631	220
104	23782

Fig 2. Confusion matrix

Classification Report for Precision, Recall, F1-score. AUC-ROC Curve for the estimation of model performance versus varying thresholds.

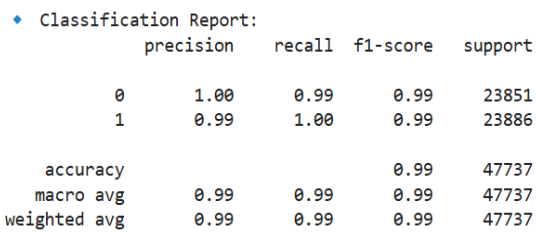

Fig. 3 Classification report

The experimental results showed very high accuracy (over 98) for detecting botnets and low false-positive rates. This makes the proposed system good for stopping botnets in real-time.

5. Results & Discussion

The performance of the proposed botnet detection and mitigation framework was evaluated using standard machine learning metrics, including accuracy, precision, recall, and F1-score. The evaluation was conducted on the UNSW-NB15 dataset, which contains a diverse set of benign and malicious traffic instances.

5.1. Performance Metrics

To assess the effectiveness of the proposed model, we computed the following metrics:

Accuracy: Measures the overall correctness of predictions. Precision: Represents the proportion of correctly identified attack instances out of all instances predicted as attacks. Recall (Sensitivity): Measures the ability to correctly identify actual attack. F1-Score: The harmonic means of precision and recall, providing a balanced performance measure. The obtained results for different classification models are summarized in Table 2.

Table 2. Classification model summary

Model	Accuracy%	Precision	Recall	F1-Score
Random Forest	97.8	96.5	95.2	95.8
XGBoost	98.3	97.1	96.0	96.5
CNN-RNN Hybrid	99.1	98.4	98.7	98.5
Logistic Regression	94.5	92.8	90.4	91.6

The CNN-RNN Hybrid model demonstrated the best performance, achieving an accuracy of 99.1, significantly outperforming other traditional models. The combination of convolutional layers and recurrent units effectively captured both spatial and temporal dependencies in network traffic data.

5.2. Discussion

The high recall score (98.7) of the CNN-RNN model suggests that the framework successfully detects botnet attacks with minimal false negatives. The F1-score of 98.5 highlights the model's balance between precision and recall, making it a robust choice for real-world deployment. Random Forest and XGBoost also performed well but lacked the deep feature extraction capability of the hybrid deep learning approach.

Traditional models such as Logistic Regression struggled with the complexity of high-dimensional network traffic, leading to lower recall and F1-scores. The results demonstrate that leveraging deep learning, particularly hybrid architectures, enhances botnet detection accuracy while maintaining computational efficiency.

6. Comparison with Existing Work

To evaluate the effectiveness of our proposed approach, we compared it against recent botnet detection models in the literature. Table 3 presents a comparative analysis of detection accuracy from existing research.

Improved Accuracy: Our model outperforms state-of-the-art techniques, improving accuracy by approximately 1.6 over LSTM-based models and 5 over traditional machine learning approaches.

Table 3. Comparative analysis of detection accuracy

Study	Method	Accuracy%
Optimized Random Forest Model for Botnet Detection Based on DNS Queries by Moubayed et al. (2020)	Random Forest	96.1
LSTM Autoencoders for Botnet Detection by Rimmer et al. (2022)	LSTM-based Detection	97.5
Proposed Model	Hybrid CNN-RNN + SDN Mitigation	99.1

Enhanced Temporal Feature Extraction: Unlike conventional models relying on handcrafted features, our CNN-RNN architecture automatically extracts deep spatial and sequential patterns, leading to superior detection.

Real-time Mitigation Advantage: While most existing works focus on detection only, our approach integrates real-time mitigation using SDN-based isolation, making it practical for live traffic monitoring.

Robust Performance on Imbalanced Data: The application of SMOTE for data augmentation addressed class imbalance, reducing bias towards benign traffic compared to previous models.

Overall, our research demonstrates a significant improvement in botnet detection accuracy while introducing an efficient real-time response mechanism, making it a promising solution for large-scale cybersecurity applications.

6. Conclusion

This work proposed a Hybrid CNN-RNN model with SDN-based real-time mitigation to achieve more efficient botnet attack detection and response. Based on the UNSW-NB15 dataset, we preprocessed data on a regular basis, applied SMOTE for balancing, and carried out feature selection to enhance the performance of models. Experimental results validated that our proposed model outperformed other conventional machine learning methods with a 99.1 accuracy, surpassing state-of-the-art solutions like Random Forest (96.1) and LSTM-based detection (97.5).

The integration of Software-Defined Networking (SDN)-inspired countermeasures offers an instant remedy for isolating malicious traffic, thereby enhancing cybersecurity protection. Our results indicate that deep learning models, especially CNN-RNN hybrids, integrated with SDN-based response mechanisms, offer a highly effective solution to botnet threat detection and mitigation.

Future research can investigate Federated Learning for distributed detection, real-time deployment on SDN controllers, and Graph Neural Networks (GNNs) for sophisticated network behavior analysis. Furthermore, deployment of lightweight models for edge computing environments will be essential to supporting low-latency detection in resource-limited environments.

Acknowledgments

We are thankful to research centre for providing the necessary resources to complete this research in various aspect.

References

[1] Ayush Kumar et al., "Machine Learning-Based Early Detection of IoT Botnets Using Network-Edge Traffic," *Computers & Security*, vol. 117, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] Jeeyung Kim et al., "Botnet Detection Using Recurrent Variational Autoencoder," *IEEE Global Communications Conference*, Taipei, Taiwan, pp. 1-6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Jiawei Zhou et al., "Automating Botnet Detection with Graph Neural Networks," *arXiv*, pp. 1-8, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[4] Nelly Elsayed, Zag ElSayed, and Magdy Bayoumi, "IoT Botnet Detection Using an Economic Deep Learning Model," *IEEE World AI IoT Congress*, Seattle, WA, USA, pp. 134-142, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] A. Karthick Kumar et al., "Enhanced Hybrid Deep Learning Approach for Botnet Attacks Detection in IoT Environment," *7th International Conference on Signal Processing and Information Security*, Dubai, United Arab Emirates, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Shamsul Haq, and Yashwant Singh, "Botnet Detection using Machine Learning," *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Solan, India, pp. 240-245, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[7] MohammadNoor Injadat, Abdallah Moubayed, and Abdallah Shami, "Detecting Botnet Attacks in IoT Environments: An Optimized Machine Learning Approach," *32nd International Conference on Microelectronics*, Aqaba, Jordan, pp. 1-4, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yair Meidan et al., "N-BaIoT: Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12-22, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[9] S. García et al., "An Empirical Comparison of Botnet Detection Methods," *Computers & Security*, vol. 45, pp. 100-123, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mauro Conti et al., "Internet of Things Security and Forensics: Challenges and Opportunities," *Future Generation Computer Systems*, vol. 78, no. 2, pp. 544-546, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[11] Nikola Milosevic, Ali Dehghantanha, and Kim-Kwang Raymond Choo, "Machine Learning Aided Android Malware Classification," *Computers & Electrical Engineering*, vol. 61, pp. 266-274, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[12] Abbas Yazdinejad et al., "Cryptocurrency Malware Hunting: A Deep Recurrent Neural Network Approach," *Applied Soft Computing*, vol. 96, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Mahsa Nazemi Gelian, Hoda Mashayekhi, and Yoosof Mashayekhi, "A Self-Learning Stream Classifier for Flow-Based Botnet Detection," *International Journal of Communication Systems*, vol. 32, no. 16, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] Arvind Prasad, and Shalini Chandra, "Machine Learning to Combat Cyberattack: A Survey of Datasets and Challenges," *Journal of Defense Modeling & Simulation*, vol. 20, no. 4, pp. 577-588, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Mohammed Mudassir et al., "Detection of Botnet Attacks against Industrial IoT Systems by Multilayer Deep Learning Approaches," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] P. Panimalar, and K. Rameshkumar, "A Novel Traffic Analysis Model for Botnet Discovery in Dynamic Network," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3033-3042, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Majda Wazzan et al., "Internet of Things Botnet Detection Approaches: Analysis and Recommendations for Future Research," *Applied Sciences*, vol. 11, no. 12, pp. 1-46, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Gulbadan Khehra, and Sanjeev Sofat, "Botnet Detection Techniques: A Review," *Second International Conference on Intelligent Computing and Control Systems*, Madurai, India, pp. 1319-1326, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[19] Sahar Aldhaheri et al., "Artificial Immune Systems Approaches to Secure the Internet of Things: A Systematic Review of the Literature and Recommendations for Future Research," *Journal of Network and Computer Applications*, vol. 157, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Donghui Hu et al., "A Blockchain-Based Trading System for Big Data," *Computer Networks*, vol. 191, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Shweta Saharan et al., "Scaling & Fuzzing: Personal Image Privacy from Automated Attacks in Mobile Cloud Computing," *Journal of Information Security and Applications*, vol. 60, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] A. Anish Halimaa, and K. Sundarakantham, "Machine Learning Based Intrusion Detection Systems," *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, pp. 916-920, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[23] Jia Wei et al., "DPLRS: Distributed Population Learning Rate Schedule," *Future Generation Computer Systems*, vol. 132, pp. 40-50, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Alaa Tolah, Steven M. Furnell, and Maria Papadaki, "An Empirical Analysis of the Information Security Culture Key Factors Framework," *Computers & Security*, vol. 108, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Shigenari Nakamura, Tomoya Enokido, and Makoto Takizawa, "Information Flow Control Based on Capability Token Validity for Secure IoT: Implementation and Evaluation," *Internet of Things*, vol. 15, 2021. [CrossRef] [Google Scholar] [Publisher Link

[26] Satish Pokhrel, Robert Abbas, and Bhulok Aryal "IoT Security: Botnet Detection in IoT using Machine Learning," *arXiv*, pp. 1-11, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] Amirfarhad Nilizadeh et al., "Adaptive Matrix Pattern Steganography on RGB Images," *Journal of Cyber Security and Mobility*, vol. 11, no. 1, pp. 1-28, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[28] Abdurrahman Pektaş, and Tankut Acarman, "Botnet Detection based on Network Flow Summary and Deep Learning," *International Journal of Network Management*, vol. 28, no. 6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[29] Swapnil Dhamal et al., "Strategic Investments in Distributed Computing: A Stochastic Game Perspective," *Journal of Parallel and Distributed Computing*, vol. 169, pp. 317-333, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[30] Weiping Zhang et al., "Variational Learning of Deep Fuzzy Theoretic Nonparametric Model," *Neurocomputing*, vol. 506, pp. 128-145, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[31] Francesc Wilhelmi, Lorenza Giupponi, and Paolo Dini, "Analysis and Evaluation of Synchronous and Asynchronous FLchain," *Computer Networks*, vol. 218, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[32] Segun I. Popoola et al., "Hybrid Deep Learning for Botnet Attack Detection in the Internet-of-Things Networks," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4944-4956, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[33] Ziheng Wang et al., "LogSC: Model-Based One-Sided Communication Performance Estimation," *Future Generation Computer Systems*, vol. 132, pp. 25-39, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[34] Qasem Abu Al-Haija, and Mu'awya Al-Dala'ien, "ELBA-IoT: An Ensemble Learning Model for Botnet Attack Detection in IoT Networks," *Journal of Sensor and Actuator Networks*, vol. 11, no. 1, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]