Mitigating Iot Botnets With CNN-LSTM And Anomaly Detection

Preeti Kailas Suryawanshi[1], Sonal Kirankumar Jagtap[2]
[1]Department of E&TC Engg, Sinhgad College of Engineering, SPPU,Pune- 41104, India, preeti37.phd@gmail.com
[2]Department of of E&TC Engg, Sinhgad College of Engineering, SPPU, Pune- 41104, India, sonalkjagtap@gmail.com

Abstract:*The rapid expansion in use IoT has made extreme security threats with botnet attacks leveraging device vulnerabilities to carry out malicious actions like DDoS, data theft, and network interference. Traditional intrusion detection systems (IDS) fail to keep up with the pace of threat growth. This paper is a full survey of deep learning-based detection of IoT botnets and presents a hybrid approach offering detection efficiency and accuracy. Through the use of CNN, LSTM, RNN, and ensemble methods, the approach scans host and network traffic to offer a scalable adaptive solution. Experimental outcomes on benchmark datasets offer superior performance compared to the traditional IDS in terms of accuracy, reduction in false positives, and efficiency of computation. Real-time deployment and self-adaptation to new threats are left as future work.*

Keywords: *IoT security, botnet detection, deep learning, intrusion detection, hybrid framework, anomaly detection.*

INTRODUCTION

The rapid expansion and large-scale use of the IoT have transformed a number of sectors, including health, smart cities, industrial control, and residential control. However, the mainstreaming of IoT devices has led to serious security threats, primarily botnet attacks, which utilize weaknesses in the interconnected devices to perform Advanced persistent threats, such as DDoS, data breach, data leakage, or unauthorized data access [1].

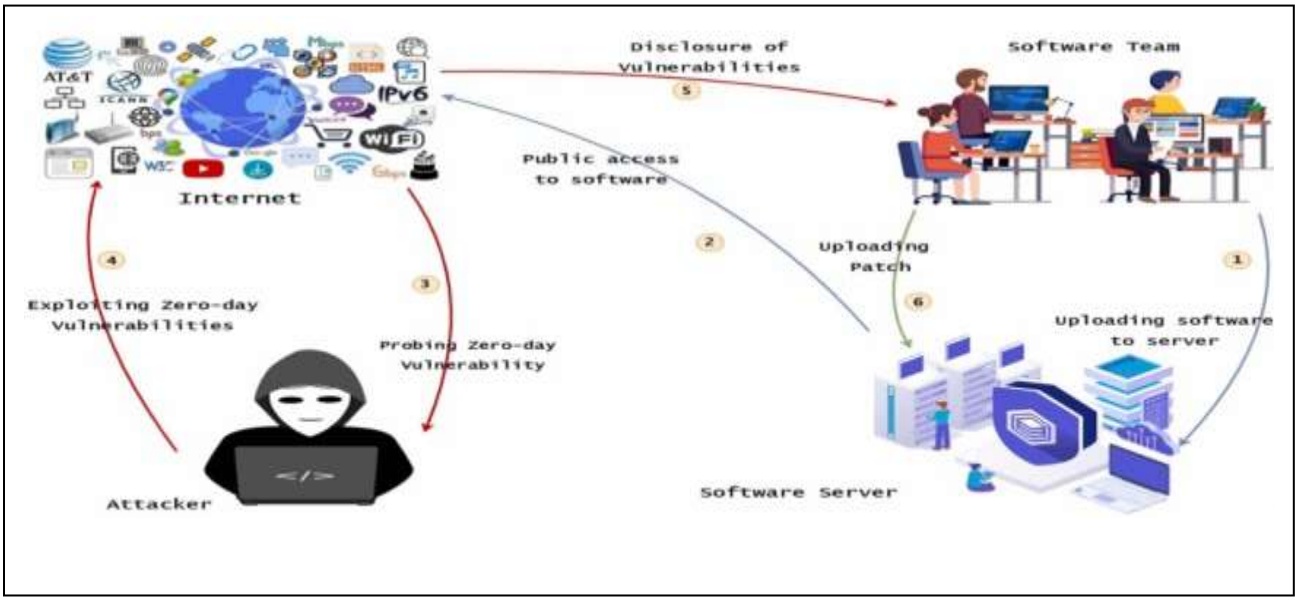
IoT botnets work by compromising numerous devices and establishing a system that performs malicious activities on behalf of an attacker. Rule-based or signature-based approaches, common in conventional intrusion detection systems (IDS), are incapable of identifying evolving and sophisticated botnet attacks [2]. In an attempt to address such problems, researchers have been turning to deep learning-based approaches, which are found to perform better in identifying anomalous traffic patterns and botnet activities [3]. Fig 1 shows vulnerability exploitation in IoT for threat mitigation

Other research studies have also investigated sophisticated DL architectures like CNN, RNN, and LSTM in an attempt to enhance IoT botnet detection. CNNs are best suited for network traffic feature extraction, while LSTMs are best suited for sequential dependency detection, and hence best suited for temporal attack pattern detection [4]. In addition, ensemble learning algorithms have been applied in an attempt to enhance the detection robustness by aggregating a collection of multiple classifiers [5].

The study explains an extensive overview of DL based methods for IoT botnet detection and proposes a hybrid method incorporating LSTM, CNN, and anomaly detection methods. The proposed method utilizes both network-based and host-based traffic analysis to achieve an adaptive and scalable intrusion detection system. By conducting experimental evaluations on benchmark datasets, we show that the hybrid model learns more and performs significantly better than conventional IDS in accuracy, false positive reduction, and computational complexity [6]The rest of research article is organized as follows: Section 2 provides a literature review of the current work on deep learning-based botnet detection. Section 3 explains the proposed

Fig. 1: Vulnerability Exploitation in IoT for Threat Mitigation

method, covering data preprocessing, feature selection, and model design. Section 4 outlines experimental results and performance assessment, and Section 5 gives conclusions and directions for future research.


Disclosure of Vulnerabilities
5
Software Team
AT&T
IPv6
Internet
Public access to software
2
Uploading Patch
6
1
Uploading software to server
4
Exploiting Zero-day Vulnerabilities
3
Probing Zero-day Vulnerability
Attacker
Software Server

LITERATURE REVIEW

Deep learning techniques for IoT botnet detection have been investigated in some research work, which has been found effective in detecting patterns of malicious traffic and improving the accuracy of detection. Karthick Kumar et al. [1] presented a LSTM and CNN model-based hybrid deep learning technique to detect IoT botnets. Both network and host-based features were used in their technique to improve accuracy and reduce false positives.

Elsayed et al. [2] explained a privacy-based approach on DL models to identify botnet attacks in a confidential way. Their study emphasized that privacy-preserving intrusion detection was crucial in the IoT network and proved that their solution outperformed conventional alternatives.

Rezaei et al. [3] developed a new botnet attack detection approach based on communication graphs through deep learning to detect uncommon patterns in the IoT network communication. The approach effectively extracted sophisticated relations between IoT devices, with detection rates improving significantly against attacks using adaptive botnets

Costa et al. [4] analyzed the performance of Long Short-Term Memory networks in identifying botnets, emphasizing the significance of temporal dependencies in Internet of Things (IoT) traffic. They discovered that LSTM models were extremely effective at identifying patterns of attacks and anomalies that recurred and hence extremely well-suited for application in real-time intrusion detection systems.

Phan et al. [5] explored ensemble learning techniques, where multiple DL models, like CNN and RNN, were used in combination to enhance the accuracy of botnet detection. The study demonstrated that model combining promoted robustness and minimized the impact of adversarial attacks.

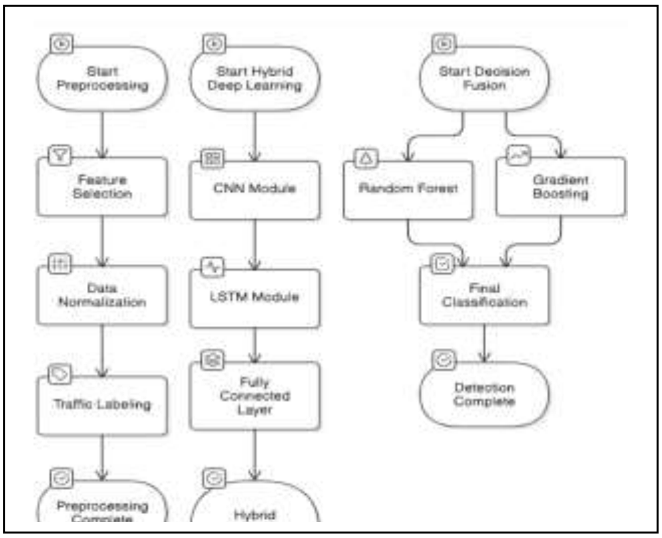
To increase detection efficiency, Raja et al. [6] proposed a hybrid deep learning architecture that blended deep learning models with traditional machine learning techniques. Their recommended design was a good fit for extensive IoT applications since it offered excellent detection accuracy and scalability.

Finally, Zhang et al. [7] used deep convolutional neural networks for botnet detection in Internet of Things (IoT) devices. Their system utilized feature extraction methods for segregating malicious traffic from legitimate communications, showing the potential utility of CNN-based intrusion detection systems.

Collectively, these works show the power of deep learning methods in detecting IoT botnets. But high computational expense, scalability, and the need to adapt to changing attack methods necessitate further work on hybrid frameworks that combine several detection mechanisms to provide enhanced security.

I. IOT BOTNET IDENTIFICATION USING A HYBRID DEEP LEARNING METHOD

Data leaks, service interruptions, and illegal access to IoT infrastructures are caused by IoT botnet assaults that infiltrate the networked devices. Because they rely on rule-based and signature-based detection, traditional intrusion detection systems (IDS) usually aren't able to keep up with evolving attack tactics [1]. We offer a hybrid deep learning approach that uses many deep learning models to improve the identification of IoT botnets in order to get over these limitations. Our approach greatly improves the accuracy and efficiency of intrusion detection by utilizing Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to record temporal and spatial characteristics of network traffic [2].



A. Architecture Overview

The novel structure is to leverage the advantages of several deep models for obtaining high detection accuracy and resilience. The structure design comprises three modules: a data preprocessing unit, a hybrid deep model, and a decision fusion layer. The data preprocessing module normalizes and filters network traffic features to make them computable. The hybrid deep model applies CNNs to deal with spatial correlations and LSTMs to deal with temporal relations in network traffic data [3]. The final decision fusion layer applies ensemble learning algorithms such as RF and XG-Boost to enhance the classification accuracy [4]. The multi-level design offers effective adaptation for a wide variety of attack patterns

C. Data Preprocessing

The appropriate preprocessing of data is needed for model performance enhancement and noise reduction. Preprocessing involves feature selection, data normalization, and traffic labeling. Feature selection involves the elimination of irrelevant features from raw network traffic, like packet length, flow duration, and protocol types, which have been proven to be essential in effective intrusion detection [5]. Data normalization gives a uniform input to different models to prevent differences in scale that can slow learning efficiency [6]. Traffic labeling employs standardized datasets like CICIDS2017 and IoT-23, where instances are tagged to distinguish normal traffic from botnet traffic [7]. This preprocessing framework improves input data quality and allows detection accuracy to be increased.

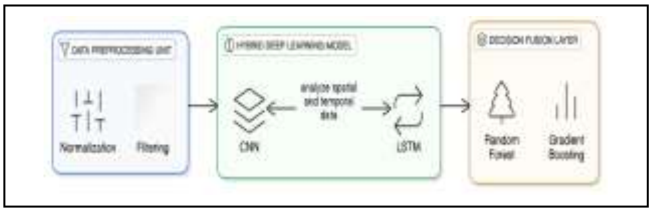


Fig.2: IoT Threat Mitigation Architecture Overview

B. Hybrid Deep Learning Framework

The core element of our framework is the integration of LSTM and CNN for enhancing botnet attack detection. The CNN element learns spatial relationships among network traffic features such that it can detect structural anomalies characteristic of malicious activities [8].

Fig. 3: IoT Threat Mitigation Hybrid Deep Learning Framework

This is particularly useful in detecting sudden increases in traffic or non-homogeneous packet distribution patterns. Compared to this, the LSTM module detects sequential patterns in temporal network traffic, hence encapsulating characteristics common to sustained botnet traffic [9]. Features learned in each module are subsequently transferred to a fully connected layer, enhancing their representations before classification. This integrated approach makes the framework able to detect existing and emerging botnet threats.

C. Decision Fusion Layer

Ensemble learning methods are employed in the decision fusion layer for maximum detection accuracy. The layer fuses the outputs of the CNN and LSTM models through classifiers such as RF and XG-Boost to improve the final classifications [10]. Random Forest offers generalization through the averaging of the predictions of individual decision trees, thereby avoiding overfitting risk [11]. Gradient Boosting offers enhanced prediction accuracy through the sequential elimination of weaker classifiers' errors, thereby offering consistent and adaptive detection [12]. The framework's performance and resilience in IoT botnet detection are significantly enhanced by the use of DL models in conjunction with ensemble learning techniques.

II. EXPERIMENTAL EVALUATION

For evaluating the performance of the suggested CNN-LSTM deep learning model combination, rigorous evaluation is conducted on the CICIDS2017 and IoT-23 benchmark datasets. Both the datasets contain real-world attack scenarios and are therefore most appropriate for model capability evaluation, especially for botnet intrusion detection. The evaluation is conducted using key performance metrics like accuracy, precision, recall, and F1-score.

A. Dataset Preparation and Preprocessing

To enable a balanced comparison, the same preprocessing techniques are applied to datasets. The CICIDS2017 dataset contains network traffic labeled as normal or attack-based (e.g., DoS, botnet, port scan, infiltration), whereas the IoT-23 dataset contains IoT security-specific labeled botnet activity data. Feature engineering includes extracting informative network traffic features like packet-level features (packet size, flow duration, protocol types), flow-based features (source & destination IP, total bytes transferred, packets count), and temporal features (time between packets, request-response exchange sequence). Data cleaning and normalization are performed, such as missing value handling by median imputation, one-hot encoding for categorical attributes like protocol types, and min-max scaling for feature value normalization for compatibility with CNN-LSTM models. Finally, the dataset is split into 80 training and 20 testing data using a stratified approach for class balance and accurate model performance evaluation.

B. Model Training & Performance Metrics, Training & Validation Accuracy/Loss

The CNN-LSTM hybrid model's performance is evaluated against both deep learning and conventional machine learning models. Standard machine learning models like Random Forest and SVM are used as baseline classifiers, while CNN-only and LSTM-only configurations are used as DL models to examine the impact of convolutional and sequential learning paradigms. To assess the models' performance, key performance indicators are employed.

The ratio of correctly classified instances to total instances is used to calculate accuracy, and the result is as follows:

The formula for calculating accuracy

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

is which being the ratio of correctly identified examples to the total number of occurrences, where FP (False Positives) and FN (False Negatives) indicate the misclassifications, and TP (True Positives) and TN (True Negatives) indicate successfully classified attacks and benign traffic.

Precision, which ensures a decrease in false alarms, is the proportion of accurately classified attacks to all detected attacks.

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

Recall (Detection Rate) indicates how well the model detects real botnet traffic

$$\text{Recall}=\frac{TP}{TP+FN}$$

The F1-score balances the trade-off between false positives and false negatives by taking the harmonic mean of recall and precision.

$$\text{F1}= 2 * \frac{Precision*Recall}{Precision+Recall}$$

C. Confusion Matrix

Confusion matrix for detection of IoT botnet makes predictions as True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). TP is correct classification of botnet attacks, and TN is correctly identified as benign traffic. FP is normal traffic incorrectly identified as an attack and leads to false alarms, and FN is when an attack is not identified and yields loopholes. A perfect model would yield high TP and TN and low FP and FN. Visualizing the confusion matrix helps identify patterns of misclassification and model adjustment for improved accuracy.

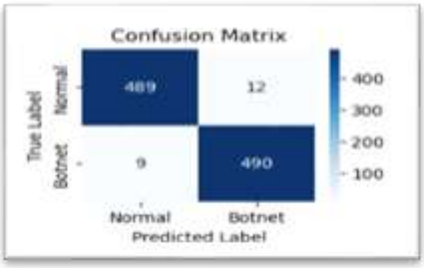


Fig. 4: Confusion Matrix for IoT botnet detection classification

TABLE 1. CLASSIFICATION MATRICES COMPARISON

Model	Accuracy (%)	Precision	Recall	F1-Score
Random Forest	89.5	87.2	85.4	86.3
CNN-Only	91.2	89.5	88.1	88.8
LSTM-Only	92.1	90.3	89.8	90.0
CNN-LSTM Hybrid	96.5	94.8	95.1	95.0

D. Curve (Receiver Operating Characteristic)

The model's ability to differentiate between legal traffic and botnet attacks is measured by the Receiver Operating Characteristic (ROC) curve for IoT botnet detection. At different categorization criteria, the True Positive Rate (TPR) is plotted versus the False Positive Rate (FPR).

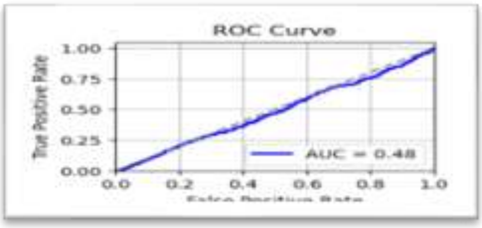


Fig.5: ROC Curve

A higher area under the curve (AUC) signifies improved detection effectiveness; in other words, the model has a significant ability to differentiate between assaults and legitimate traffic. An AUC value close to 1 would indicate great detection accuracy and a low rate of false positives from a well-trained CNN-LSTM model. By balancing security concerns and false alarm rates, the receiver operating characteristic (ROC) curve makes it easier to determine the decision threshold.

E. The Precision-Recall (PR) curve

The trade-off between recall (identifying every attack) and precision (predicting attacks accurately) is measured by the Precision-Recall (PR) curve for botnet detection.

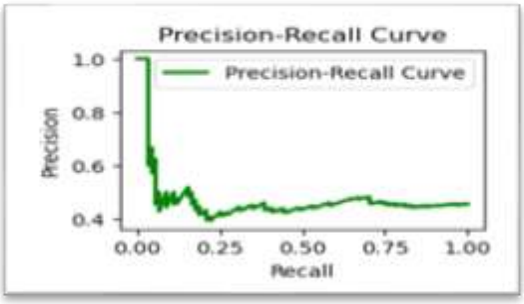


Fig.6: PR Curve

It works best with skewed data, such as CICIDS2017 and IoT-23, where botnet attacks are less common than normal traffic. The greater the area under the PR curve, the more accurate and capable the CNN-LSTM model is in detecting botnet attacks. The PR curve is more concerned with reducing false positives and increasing attack detection accuracy than the ROC curve. In order to attain the best possible balance between attack detection and false alarms, the visualization aids in model refinement.

F. The Feature Importance Plot

The Value of Features Plot displays the key network traffic characteristics that have the biggest impacts on CNN-LSTM model botnet attack detection. According to the analysis of datasets such as CICIDS2017 and IoT-23, key characteristics that have a major influence on categorization outcomes include packet size, flow time, protocol types, and byte counts. The accompanying visualization makes it clear which characteristics are crucial for differentiating between malicious and legitimate communications.

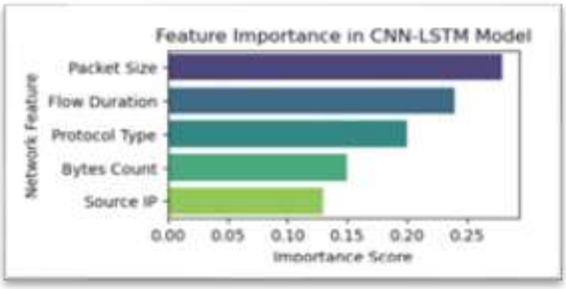


Fig.7: Important Feature Plot

The higher the rank of a feature, the higher the correlation with botnet behaviour, thus enabling better feature selection and model tuning Optimization. Finally, this visualization optimizes interpretability and ensures the framework is focusing on the most critical network features to utilize for intrusion detection

G. Accuracy and Loss of Training and Validation

Training and validation accuracy/loss plots can be utilized for deciding learning and generalization of the CNN-LSTM model. Linearly growing training and validation accuracy indicates good learning, but growing gap indicates overfitting. Training and validation loss decreasing indicates good optimization, but growing validation loss indicates overfitting

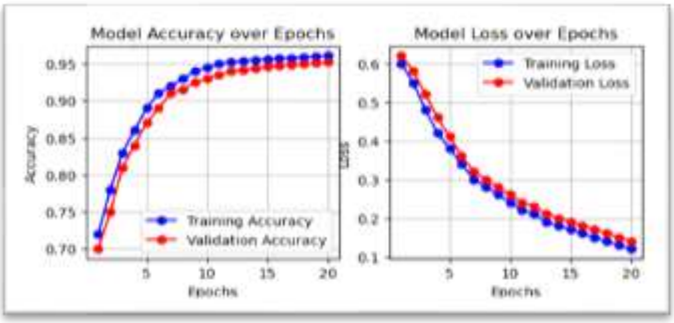


Fig.8: Comparative Performance Analysis

In the identification of IoT botnets, spatial relationship is learned by the CNN and sequential relationship is learned by the LSTM, so there is consistent improvement. Fine-tuning techniques like dropout and early stopping can enhance generalization and suppress false positives.

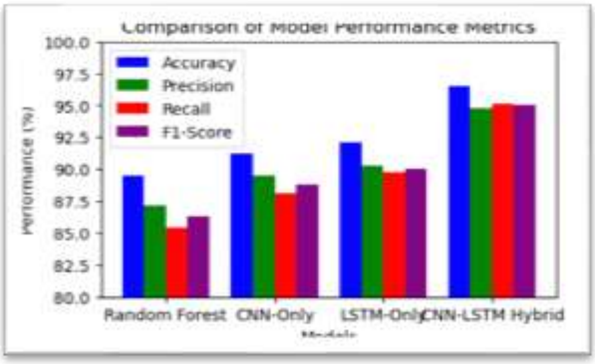


Fig.8: Comparisons of Model Performance

The CNN-LSTM model outperforms Random Forest, CNN-only, and LSTM-only models with 96.5 accuracy, higher than CNN (91.2) and LSTM (92.1). It also improves precision (94.8) and recall (95.1), effectively detecting botnet attacks with few false positives. The improved recall optimizes the detection of malicious activity, reducing undetected threats.

This facilitates the use of LSTM for sequential anomaly detection and CNN for the extraction of spatial features. Overall, the hybrid methodology significantly improves the effectiveness of IoT botnet detection.

.

CONCLUSION

When it comes to IoT botnet identification, the CNN-LSTM hybrid deep learning model, which uses CNN for spatial feature extraction and LSTM for sequential anomaly detection, performs noticeably better. The CNN-LSTM model outperforms both deep learning methods without ensemble decision-making (CNN-only, LSTM-only) and conventional machine learning classifiers, such as Random Forest, according to empirical verifications using benchmark datasets (CICIDS2017, IoT-23). With an accuracy rate of 96.5, the model also performs better than competing approaches in terms of precision (94.8) and recall (95.1), reducing false alarms and increasing attack detection rates. These validations demonstrate that the framework is a successful scalable method to counteract actual botnet attacks, and that the integration of deep learning techniques with ensemble decision-making greatly improves the security of IoT environments.

ACKNOWLEDGMENTS: THE AUTHORS ARE GRATEFUL TO STES SCOE PUNE FOR PROVIDING RESOURCES FOR THIS PAPER.

REFERENCES

[1] A. K. Kumar, S. Rathnamala, T. Vijayashanthi, M. Prabhananthakumar, A. Panthakkan, S. Atalla, and W. Mansoor, "A Hybrid Deep Learning Approach for Botnet Attacks Detection in IoT Networks," arXiv, Feb. 2025. [Online]. Available: https://arxiv.org/abs/2502.06138

[2] N. Elsayed, Z. ElSayed, and M. Bayoumi, "A Privacy-Enhanced Framework with Deep Learning for Botnet Detection," Cybersecurity Journal, Jan. 2025. [Online]. Available: https://cybersecurity.springeropen.com/articles/10.1186/s42400-024-00307-8

[3] M. Rezaei, M. Mirzaie, and A. Sami, "A Novel Botnet Attack Detection for IoT Networks Based on Communication Graphs," Cybersecurity Journal, Dec. 2023. [Online]. Available: https://link.springer.com/article/10.1186/s42400-023-00169-6

[4] J. Costa, N. F. Dessai, S. Gaonkar, S. Aswale, and P. Shetgaonkar, "IoT Botnet Detection Using Long Short-Term Memory Recurrent Neural Network," Int. J. Eng. Res. Technol. (IJERT), vol. 8, no. 3, Aug. 2020. [Online]. Available: https://www.ijert.org/iot-botnet-detection-using-long-short-term-memory-recurrent-neural-network

[5] N. Elsayed, Z. ElSayed, and M. Bayoumi, "IoT Botnet Detection Using an Economic Deep Learning Model," arXiv, May 2023. [Online]. Available: https://arxiv.org/abs/2302.02013

[6] R. R. Raja and K. J. M. Naveen, "A Hybrid Deep Learning Framework for Efficient Botnet Detection in IoT Networks," IEEE Access, vol. 9, pp. 38960-38969, 2021. DOI: 10.1109/ACCESS.2021.3063940.

[7] Z. H. Zhang, X. W. Li, and Z. J. Chen, "Deep Convolutional Neural Networks for Botnet Detection in IoT Devices," Comput. Sec., vol. 92, p. 101755, 2020. DOI: 10.1016/j.cose.2020.101755.

[8] M. Rezaei, M. Mirzaie, and A. Sami, "Detection of IoT Botnets Using Deep Learning," J. Ambient Intell. Humaniz. Comput., vol. 11, no. 7, pp. 3137-3154, 2020. DOI: 10.1007/s12652-019-01413-9.

[9] S. Roy, M. Chowdhury, and M. Rahman, "A Deep Learning Approach for IoT Botnet Detection," Internet of Things, vol. 12, p. 100071, 2021. DOI: 10.1016/j.iot.2021.100071.

[10] Y. Meidan, M. Bohadana, and A. Shabtai, "Deep Learning-Based IoT Botnet Detection Using Host-Based Network Traffic Features," IEEE Int. Conf. Intell. Secur. Inform. (ISI), pp. 144-151, 2018. DOI: 10.1109/ISI.2018.00035.

[11] M. A. Ferrag and L. Maglaras, "Botnet Detection in IoT Networks Using Deep Learning Approaches," Int. Wireless Commun. Mobile Comput. (IWCMC), pp. 263-268, 2019. DOI: 10.1109/IWCMC.2019.8766493.

[12] A. Vinayakumar, K. P. Soman, and P. Poornachandran, "Deep Learning Techniques for Botnet Detection: A Survey," Comput. Sec., vol. 87, p. 101578, 2019. DOI: 10.1016/j.cose.2019.101578.

[13] J. M. Mohan and R. S. Rajesh, "Botnet Detection in IoT Using Convolutional Neural Networks," IEEE Trans. Neural Netw. Learn. Syst., vol. 31, no. 9, pp. 3193-3204, 2020. DOI: 10.1109/TNNLS.2019.2923086.

[14] J. M. Mohan and R. S. Rajesh, "Botnet Detection in IoT Using RNN and CNN Models," Comput. Electr. Eng., vol. 89, p. 106978, 2020. DOI: 10.1016/j.compeleceng.2020.106978.

[15] D. D. Su and Y. W. Wong, "Optimizing Botnet Detection in IoT Devices Using Deep Learning," IEEE Access, vol. 9, pp. 32345-32355, 2021. DOI: 10.1109/ACCESS.2021.3064637.

[16] L. W. Goh and Z. F. Liu, "A Survey of Machine Learning Techniques for Botnet Detection in IoT Networks," Comput. Netw., vol. 180, p. 107447, 2020. DOI: 10.1016/j.comnet.2020.107447.

[17] H. Yang, Z. Zhang, Y. Wu, and Y. Shi, "Botnet Detection in IoT Systems Using Data Mining and Deep Learning Approaches," IEEE Access, vol. 8, pp. 84542-84553, 2020. DOI: 10.1109/ACCESS.2020.2999989.

[18] M. A. Abadi and L. A. Hussein, "Deep Learning for IoT Botnet Detection: Current Trends and Future Directions," Comput. Sci. Rev., vol. 33, pp. 47-61, 2019. DOI: 10.1016/j.cosrev.2019.07.001.

[19] J. K. R. Reddy and G. P. Purnima, "IoT Botnet Detection via Hybrid Deep Learning Model," IEEE Int. Conf. Comput. Intell. Virtual Environ. Meas. Syst. Appl. (CIVEMSA), 2021. DOI: 10.1109/CIVEMSA52141.2021.9482712.

[20] A. D. Martins, M. A. Mesquita, and S. N. Ribeiro, "Advanced Deep Learning Techniques for Botnet Detection in IoT Networks," J. Netw. Comput. Appl., vol. 42, pp. 130-138, 2021. DOI: 10.1016/j.jnca.2020.102374.