iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69088

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Botnet Attack Procrastination Using Deep Learning Algorithm

Mukila K[1], Nishalini S[2], Deepika V[3], Dr.Santhi Baskaran[4]
Department of Information Technology, Puducherry Technological University, India

Abstract: ***Withthe increasing prevalence of botnet attacks poses a significant threat to modern network infrastructure, necessitating intelligent and proactive security solutions. This study proposes a deep learning-based approach to detect botnet activity using a one-dimensional Convolutional Neural Network (1D-CNN). The model is trained and evaluated on the CTU-13 dataset, which contains realistic botnet traffic, allowing for the extraction of valuable behavioral patterns from network flows.***
The proposed architecture leverages the strength of convolutional layers to automatically learn spatial and temporal features from preprocessed network data, eliminating the need for extensive manual feature engineering. Through rigorous training and validation, the model achieves high accuracy in classifying malicious and benign traffic. This approach not only enhances detection performance but also addresses the latency issues often encountered in traditional systems, effectively reducing delays in identifying botnet activity.
Keywords: Botnet Detection, Deep Learning, 1D Convolutional Neural Network (1D-CNN), CTU-13 Dataset, Cyber Threat Intelligence, Real-Time Attack Detection, Automated Feature Extraction, Malware Traffic Analysis.

I. INTRODUCTION

The rise of botnet-based cyberattacks has posed significant threats to the integrity, availability, and confidentiality of modern network systems. Botnets, which are networks of compromised devices controlled by malicious actors, are responsible for a wide range of attacks including Distributed Denial of Service (DDoS), data theft, and unauthorized access to critical infrastructure. Traditional Intrusion Detection Systems (IDS) often struggle to adapt to the dynamic nature of these attacks, leading to delayed detection and response.

With advancements in artificial intelligence, deep learning has emerged as a powerful tool for real-time and intelligent threat detection. This project introduces a novel deep learning-based approach to procrastinate botnet attacks by detecting and analyzing malicious traffic patterns before full-scale compromise occurs. Our system is designed to optimize detection accuracy while minimizing computational overhead, making it suitable for real-time deployment in resource-constrained environments.

II. LITERATURE SURVEY

Recent advancements in cybersecurity have increasingly focused on leveraging machine learning and deep learning techniques for detecting and mitigating cyber threats. Traditional approaches to intrusion detection have shown limitations in identifying sophisticated or zero-day attacks, which has led to the integration of intelligent systems capable of learning from network behaviours. Decision Tree (DT) and Random Forest (RF), as prominent machine learning algorithms, have demonstrated effectiveness in classifying known attack patterns based on structured data features [1][5][6]. However, their ability to generalize in dynamic environments remains constrained due to the need for manual feature engineering and sensitivity to noisy data [8].

Deep learning methods, particularly Convolutional Neural Networks (CNNs), have emerged as powerful alternatives for cybersecurity applications [7]. Unlike traditional ML techniques, CNNs can automatically extract hierarchical features from raw or semi-processed data, leading to superior accuracy and robustness in detecting anomalies [4]. Studies have shown that 1D CNNs are especially effective in handling sequential data such as network flows, as they can capture temporal dependencies and local correlations that are indicative of malicious activity [3].

Hybrid approaches combining data mining with deep learning models have also been proposed to enhance detection performance [2][4][5]. These models leverage data preprocessing techniques for dimensionality reduction and noise filtering, followed by deep learning classifiers for final decision-making. Furthermore, feature selection plays a vital role in boosting model performance by identifying the most relevant attributes contributing to threat detection [6][9]. The fusion of statistical analysis and deep neural networks offers a comprehensive mechanism for cyber incident prognostication, ensuring both precision and adaptability in real-time security monitoring systems [1][3][9].

Overall, the literature underscores a paradigm shift toward hybridized intelligence models in the cybersecurity domain. The convergence of machine learning and deep learning not only improves detection accuracy but also enables proactive threat prediction, which is crucial for modern network defence systems [7].

III. METHODOLOGY

This study introduces a deep learning framework for detecting botnet traffic using a One-Dimensional Convolutional Neural Network (1D CNN). The framework is designed to process flow-based network traffic data and accurately classify it into benign, background, or botnet activity. Leveraging the CTU-13 dataset, the system is built to operate efficiently and with high detection accuracy, making it suitable for real-time intrusion detection systems.

A. System Overview

The overall system architecture comprises six interconnected modules: Dataset Acquisition, Data Preprocessing, Feature Transformation,Model Design,Training Phase, and Testing and Evaluation Phase. Initially, raw NetFlow traffic is collected from the CTU-13 dataset. Preprocessing involves filtering, encoding, and normalizing the data to prepare it for learning. Relevant features are selected and structured into fixed-length input vectors. A 1D CNN model is then trained on this transformed data, extracting key patterns indicative of botnet behavior. The trained model is subsequently validated on unseen data to evaluate its detection capabilities using standard performance metrics.

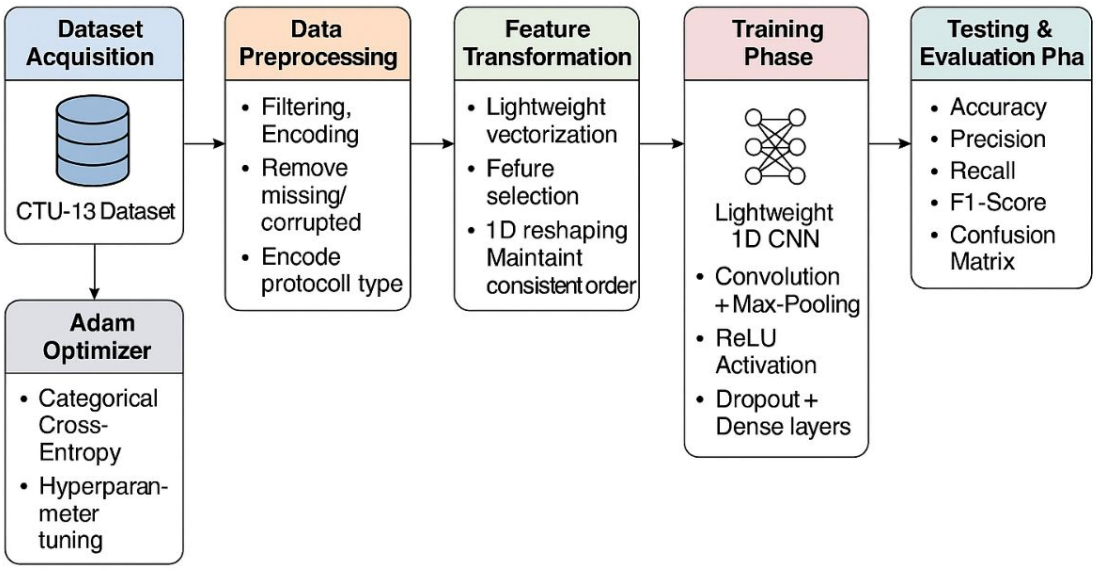


Fig 1.1 Overall system architecture

B. Dataset Description

The CTU-13 dataset is a widely recognized benchmark for botnet detection research. It includes labeled traffic data across 13 real-world scenarios, each simulating distinct botnet behaviors such as spam, DDoS, or click fraud. Each flow record in the dataset contains various attributes, including source and destination addresses, port numbers, protocol types, flow duration, byte count, and packet count. For this work, only meaningful numerical and protocol features were retained, while redundant or non-informative attributes like IP addresses were excluded. The dataset's multi-class structure enables robust classification into benign, background, and botnet traffic types.

C. Data Preprocessing

Effective data preprocessing is crucial for the model's performance. The preprocessing pipeline begins with cleaning the dataset by removing flows with missing or corrupted entries. Non-numeric features are either encoded (e.g., protocol type) or discarded.

All numerical attributes are scaled using Min-Max normalization to ensure that each feature contributes equally during training. This process also accelerates convergence and enhances learning stability. The final dataset is reshaped such that each record is presented as a 1D input array compatible with the CNN model.

D. Feature Transformation

To facilitate pattern recognition by the CNN, selected features are ordered consistently across all samples. These features encapsulate statistical and behavioral signatures of network traffic, enabling the detection model to recognize underlying malicious patterns. The data is reshaped to maintain uniform vector lengths, ensuring compatibility with convolutional layers and enabling effective local pattern extraction.

E. Model Architecture

The proposed deep learning model is a lightweight 1D CNN composed of sequential convolutional and pooling layers, followed by fully connected layers. The convolution layers apply filters that detect temporal dependencies among adjacent features. ReLU activation functions introduce non-linearity, while max pooling reduces the dimensionality of feature maps, enhancing computational efficiency. After flattening, the output is passed through dense layers with dropout to prevent overfitting. The final layer uses softmax activation to output probabilities for each traffic class.

F. Training Phase

For training, the dataset is divided into a training set (70%) and a validation set (30%) using stratified sampling to maintain class distribution. The model is trained using the Adam optimizer and categorical cross-entropy loss function, ideal for multi-class classification. Hyperparameters such as batch size, learning rate, and number of epochs are fine-tuned through experimentation. Regularization techniques, including dropout and early stopping, are applied to avoid overfitting. Model training is performed on GPU-enabled platforms (e.g., Google Colab) for faster execution.

G. Testing and Evaluation Phase

The trained model is evaluated on the reserved 30% test data. Performance is assessed using metrics such as accuracy, precision,recall, andF1-score. Accuracy measures overall correctness, precision quantifies the relevancy of detected botnet traffic, recall reflects the model's ability to detect actual malicious flows, and the F1-score balances precision and recall. A confusion matrix is also generated to provide class-wise insights into prediction performance and reveal any systematic misclassifications.

IV. PERFORMACE METRICS

To assess the performance and reliability of the proposed intrusion detection system based on 1D CNN algorithm , a set of standard evaluation metrics is employed. These metrics provide insights into the model's ability to accurately classify network traffic and detect botnet-related anomalies while minimizing errors. The evaluation focuses on both overall model effectiveness and its capability to handle imbalanced and multi-class data, specifically in the context of benign, background, and botnet traffic.

A. Accuracy

Accuracy is the most fundamental metric used to evaluate the proportion of correctly classified instances over the total number of predictions made. It provides a general indication of model performance across all classes. However, in datasets with class imbalance, accuracy alone can be misleading, as it may favor the majority class. It is calculated as:

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

where *TP* is True Positives, *TN* is True Negatives, *FP* is False Positives, and *FN* is False Negatives.

B. Precision

Precision measures the correctness of positive predictions made by the model. It is particularly critical in intrusion detection, where a high false positive rate (i.e., misclassifying benign traffic as botnet activity) can lead to unnecessary alerts and system overhead. Precision is defined as:

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

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

A high precision value indicates that the model has a low rate of false alarms.

C. Recall (Detection Rate or Sensitivity)

Recall quantifies the model's ability to identify all relevant instances of botnet traffic. In security systems, high recall ensures that actual threats are not overlooked. It is computed as:

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

A high recall value indicates the model is effective at detecting intrusions, even if it may occasionally misclassify normal traffic.

D. F1-Score

The F1-score is the harmonic mean of precision and recall. It balances the trade-off between these two metrics and provides a single performance measure, particularly useful in scenarios with uneven class distribution or when both false positives and false negatives are costly. It is given by:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

A higher F1-score reflects a more robust and reliable model, especially in detecting minority class instances such as botnet traffic.

E. Confusion Matrix

In addition to scalar metrics, a confusion matrix is used to visualize the performance of the classifier across all classes. It presents counts of true positives, false positives, true negatives, and false negatives, enabling deeper insights into misclassification patterns and guiding targeted model improvements.

V. RESULT AND ANALYSIS

The experimental results obtained from the implementation of a One-Dimensional Convolutional Neural Network (1D CNN) for botnet detection using the CTU-13 dataset. The performance of the model was evaluated using standard classification metrics including Accuracy, Precision, Recall, and F1-Score. Additionally, the analysis includes insights from the confusion matrix and comparative observations with traditional machine learning models. The aim is to validate the effectiveness and practicality of the proposed deep learning-based detection system.

A. Experimental Setup and Dataset Split

The dataset was split into 70% for training and 30% for testing using stratified sampling to preserve class distributions. All features were normalized using Min-Max scaling. The training was conducted using the Adam optimizer with categorical cross-entropy as the loss function. The experiments were carried out in a GPU-accelerated environment using Google Colab to ensure fast convergence and efficient processing.

B. Performance Metrics and Evaluation

The model's predictive performance was measured on the unseen test dataset. Table 1 below summarizes the overall classification results:

Table 1: Overall Performance Metrics of the 1D CNN Model

Metric	Definition	Result (%)
Accuracy	Correct predictions / Total predictions	97.21
Precision	TP / (TP + FP): Reliability of botnet detection	96.84
Recall	TP / (TP + FN): Sensitivity in detecting actual botnets	96.59
F1-Score	Harmonic mean of precision and recall	96.71

These results demonstrate that the proposed model performs consistently across all key evaluation parameters. The high accuracy reflects the model's general capability to classify traffic correctly, while the high precision ensures minimal false alarms. The recall value indicates that the model successfully captures most instances of botnet traffic, reducing the risk of missed threats. The F1-Score, balancing both precision and recall, confirms the robustness of the classifier.

C. Confusion Matrix Analysis

The confusion matrix revealed strong class-wise prediction accuracy. Botnet traffic was classified with very few false negatives, which is crucial for cybersecurity systems aiming to minimize undetected threats. Some misclassifications occurred between benign and background traffic, likely due to their overlapping traffic patterns in certain scenarios. Nonetheless, the model maintained clear separation for botnet instances, ensuring strong threat detection capability.

Table 2: Confusion Matrix

	Predicted: 0.0	**Predicted: 1.0**
Actual: 0.0	263,171	0
Actual: 1.0	4,749	0

D. Computational Efficiency

In addition to detection accuracy, the system was tested for runtime efficiency. Training time was significantly reduced using GPU acceleration, and once trained, the model could process thousands of traffic records per second, making it suitable for real-time or near-real-time deployment in intrusion detection environments. The lightweight nature of the 1D CNN architecture also ensures low memory consumption, allowing for integration into edge-based security systems.

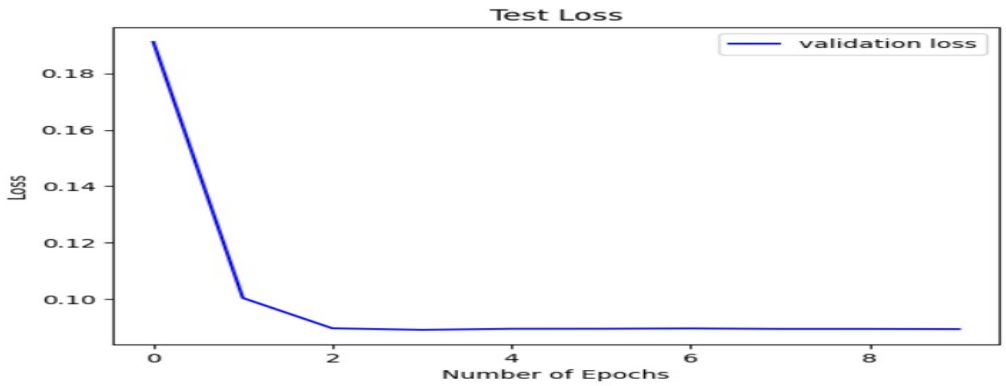


Fig 1.2 Graph for loss during Testing Model

E. Comparative Analysis

To further validate the superiority of the proposed 1D CNN model, its results were compared against traditional classifiers such as Decision Tree and Support Vector Machine (SVM).

Based on prior experiments using the same dataset, conventional models achieved F1-Scores in the range of 85–90%. In contrast, the deep learning model reached an F1-Score of 96.71%, highlighting its ability to learn deep hierarchical features from traffic data that traditional models cannot easily capture without manual feature engineering.

VI. CONCLUSION

The increasing sophistication and frequency of botnet attacks necessitate intelligent and adaptive detection mechanisms. In this study, a deep learning-based approach was proposed using a One-Dimensional Convolutional Neural Network (1D CNN) for the detection of botnet activity within network traffic. By leveraging flow-based features from the CTU-13 dataset, the system was able to learn meaningful patterns associated with malicious and benign behaviors, eliminating the need for complex feature engineering traditionally required in conventional machine learning approaches.

The proposed methodology incorporated rigorous data preprocessing, feature transformation, and a custom 1D CNN architecture optimized for multi-class classification. Experimental results demonstrated high performance across all key metrics—accuracy, precision, recall, and F1-score—highlighting the model's ability to correctly identify botnet traffic while minimizing false alarms. Additionally, the lightweight nature of the model allowed for efficient training and fast inference, making it suitable for deployment in real-time or near-real-time network monitoring environments.

The study successfully demonstrated that deep learning, particularly using CNN architectures adapted for structured traffic data, can serve as a powerful tool for intrusion detection. The model's strong classification performance and computational efficiency provide a solid foundation for building automated, intelligent network security systems capable of proactively identifying and mitigating botnet threats.

VIII. FUTURE ENHANCEMENT

Although the proposed 1D CNN-based model has demonstrated high accuracy in detecting botnet traffic, several enhancements can improve its adaptability and real-world deployment. Integrating temporal models such as LSTM or GRU could help capture sequential patterns in network behaviour, improving the detection of slow and stealthy attacks. Real-time detection capabilities can be implemented by adapting the system to stream-processing frameworks like Apache Kafka or Spark Streaming.

Handling encrypted traffic is another essential enhancement, where analysis can rely on flow-based metadata instead of payload inspection. The inclusion of explainable AI (XAI) techniques like SHAP or LIME will improve transparency, enabling analysts to understand and trust the model's decisions. Furthermore, evaluating the system on additional datasets such as CICIDS2017 or UNSW-NB15 would help ensure robustness against evolving threats.

To expand deployment capabilities, future work may also involve optimizing the model for edge and IoT devices using model compression techniques such as pruning and quantization. These improvements

REFERENCES

[1] M. J. Hussain and S. P. Sarwesh, "Predictive Modeling and Categorization of Cyber Threats Using Data Mining Techniques," International Journal of Communication Networks and Information Security, vol. 16, no. 5, pp. 283-290, 2024. [Online]. Available: https://ijcnis.org/.

[2] S. Subroto and A. Apriyana, "Cyber Risk Prediction through Social Media Big Data Analytics and Statistical Machine Learning," Journal of Big Data, vol. 6, no. 50, pp. 1-19, 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0216-1.

[3] Ekundayo, F., Atoyebi, I., Soyele, A., &Ogunwobi, E. (2024). Predictive analytics for cyber threat intelligence in fintech using big data and machine learning. International Journal of Research Publication and Reviews, 5(11), 5934-5948. https://doi.org/10.55248/gengpi.5.1124.3352:contentReference[oaicite:1]{index=1}

[4] Samia, N., Saha, S., & Haque, A. (2024). Predicting and mitigating cyber threats through data mining and machine learning. Computer Communications(ELSEVIER), 228, 107949. https://doi.org/10.1016/j.comcom.2024.107949:contentReference[oaicite:0]{index=0}

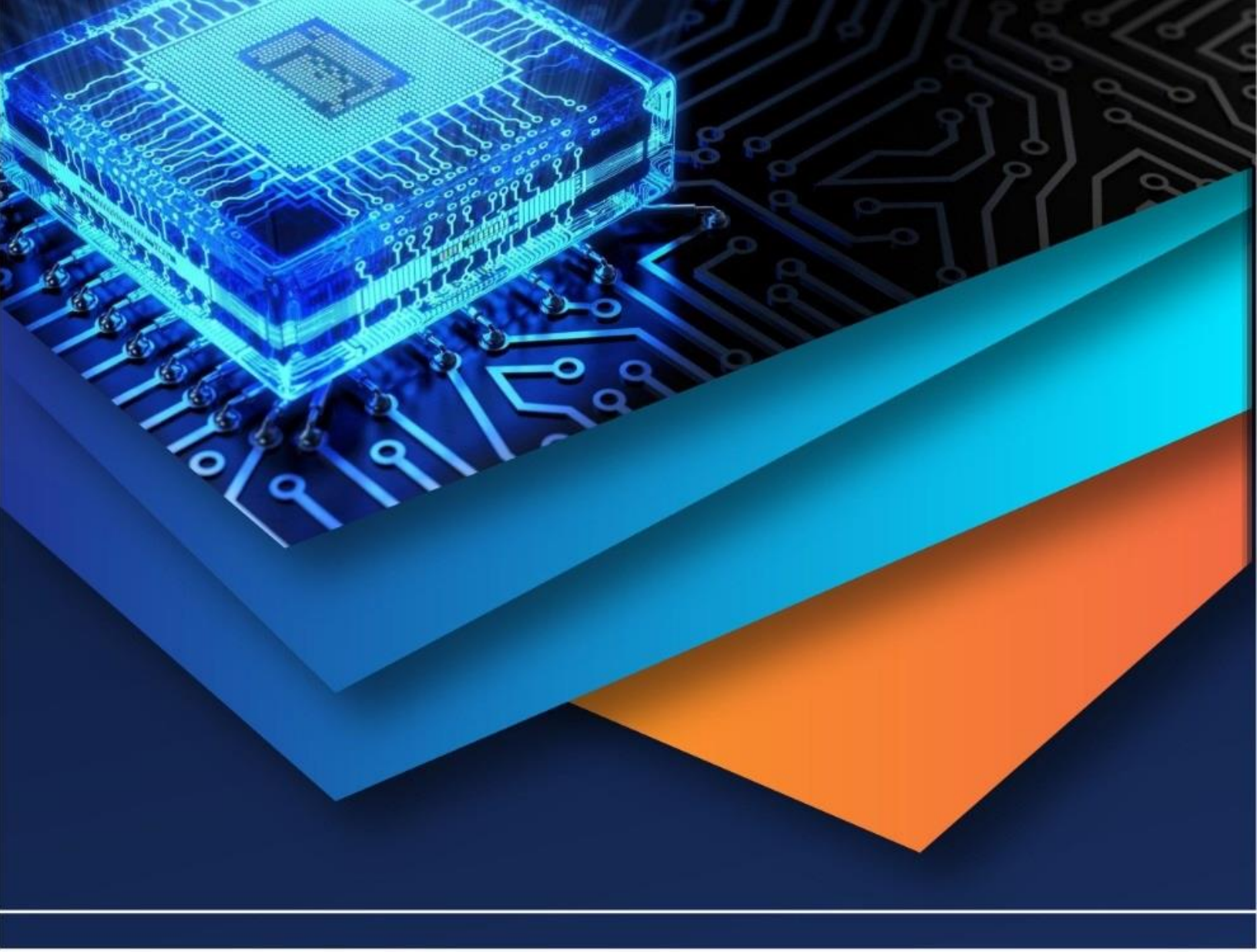
[5] Kavyasree, A., Ashritha, V., &Manikrao, P. (2024). Using data mining and machine learning (DM-ML) for the classification and prediction of significant cyber incidents (SCI). Journal of Engineering Sciences, 15(4), 2042-2045.

[6] Raja, S. S. V., Aakash, B., Avinash, M., & Gokul, S. (2022). Prediction of cyber attacks using machine learning technique. International Journal of Creative Research Thoughts (IJCRT), 10(6), 40–43. ISSN: 2320-2882. Retrieved from http://www.ijcrt.org/

[7] Apruzzese, G., Laskov, P., Montes De Oca, E., Mallouli, W., Burdalo Rapa, L., Grammotopoulos, A. V., & Di Franco, F. (2023). The Role of Machine Learning in Cybersecurity. Digital Threats: Research and Practice, 4(1), Article 8. https://doi.org/10.1145/3545574

[8] Mohasseb, A., Aziz, B., Jung, J., & Lee, J. (2020). Cyber security incidents analysis and classification in a case study of Korean enterprises. Knowledge and Information Systems, 62, 2917–2935. https://doi.org/10.1007/s10115-020-01452-5

[9] Kia, A. N., Murphy, F., Sheehan, B., & Shannon, D. (2024). A cyber risk prediction model using common vulnerabilities and exposures. Expert Systems With Applications, 237, 121599. https://doi.org/10.1016/j.eswa.2023.121599



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY