A Novel Approach for Anomaly Detection using Snort Integrated with Machine Learning

T. Preethi
Department of CSE - Cyber Security
Data Science and Artificial Intelligence
& Data Science
VNR Vignana Jyothi Institute of
Engineering and Technology
Hyderabad, India
preethi_t@vnrvjiet.in

Ponnuru Rakshitha Reddy
Department of CSE - Cyber Security,
Data Science and Artificial Intelligence
& Data Science
VNR Vignana Jyothi Institute of
Engineering and Technology
Hyderabad, India
ponnururakshithareddy@gmail.com

Lekkala Likhitha
Department of CSE - Cyber Security,
Data Science and Artificial Intelligence
& Data Science
VNR Vignana Jyothi Institute of
Engineering and Technology
Hyderabad, India
likhithalekkala@gmail.com

Pendyala Pavan Kumar
Department of CSE - Cyber Security
Data Science and Artificial Intelligence & Data Science
VNR Vignana Jyothi Institute of Engineering and Technology
Hyderabad, India
pendyalapavan0@gmail.com

Abhinav Kamani
Department of CSE - Cyber Security
Data Science and Artificial Intelligence & Data Science
VNR Vignana Jyothi Institute of Engineering and Technology
Hyderabad, India
sidhuabhi753@gmail.com

***Abstract*— In today's digital world, it is crucial to keep a company's information safe from cyber threats. With new and more sophisticated network attacks emerging all the time, better security measures and ways to monitor our networks are needed. Intrusion Detection Systems (IDS) can act as digital guards to help protect our networks from these malicious threats. Snort is a widely used open-source IDS, but it can struggle with the ever-changing nature of these threats. Our research focuses on enhancing Snort's capabilities by integrating it with ML algorithms such as Random Forest, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), and Naive Bayes. We also conducted subset testing using the NSL-KDD dataset. Our results indicate that combining these ML algorithms with Snort can improve our ability to detect and stop malicious network activities. This research can be helpful for anyone interested in developing smarter and more effective intrusion detection systems.**

Keywords*— Intrusion Detection System, Snort, Machine Learning, Random Forest, Decision Tree,Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Naive Bayes, NSL-KDD Dataset.

I. Introduction

In today's constantly changing cybersecurity environment, intrusion detection tools are essential to ensure network infrastructure security and integrity. One of the most popular intrusion detection systems is Snort [1]. This paper will explore how Snort has greatly impacted network security by detecting and responding to various cyber-attacks. Snort analyzes patterns and behaviors of network traffic to identify suspicious activities and produce alerts [8] when incidents occur. This contributes significantly to the early identification and minimization of security breaches. Snort is widely used in cybersecurity because it provides a robust defense against a wide range of threats such as Denial of Service (DoS), Distributed Denial of Service (DDoS) [10][17], SQL Injection, Buffer Overflow, and Cross-Site Scripting (XSS).

Snort's rule-based detection engine is one of its key features, backed by a vast database that enhances its efficiency in identifying known attack patterns. This allows Snort to detect and mitigate both known and new threats. Snort's flexibility makes it an essential tool for safeguarding networks from evolving cyber threats[2]. The proposed integration of machine learning (ML) into Snort[3][4] can further improve its intrusion detection capabilities. The algorithmic learning methods will integrate classical detection techniques with ML for better performance in Snort. This approach enables Snort to identify common and sophisticated attacks[5][21], learn from past incidents, and grow its detection capabilities over time in response to the changing cyber threat landscape. Additionally, the proposed integration will simplify the identification of malicious traffic across devices on the same network [6][9][11][13], resolving the complexities of modern network designs. This makes Snort a must-have tool for comprehensive network security.

II. Related Work

Various authors have conducted studies on the detection and prevention of intrusions and network security, which have significantly enhanced our knowledge and understanding of these critical cybersecurity aspects. A thorough and organized analysis of the literature was conducted, beginning with Ozkan-Okay et al. [1], who provided a comprehensive overview of different intrusion detection systems. This work was primarily a survey and did not delve into specific

978-93-80544-51-9/24/©BVICAM, New Delhi, India

procedures or implementations. The importance of integrating machine learning (ML) approaches into open-source intrusion detection systems was emphasized by Vadhil et al. [2]. However, the fundamental disadvantage of these approaches is the lack of robust evidence to support the improvement claims. Furthermore, the comprehensive study conducted by Alhasani et al. [3] highlighted the key role played by machine learning (ML) in intrusion detection systems, although comprehensive case studies could have further strengthened the findings by demonstrating how they could be applied in real-life circumstances. Given expanding threats, Majdah ALshammari and Mohammad A. Mezher [4] examined the effectiveness of machine learning in detecting intrusions. The study found that decision tree-based machine learning was effective, as demonstrated by examining single, hybrid, and ensemble classifiers. Zhang, et al. [5] proposed an innovative structure that utilizes machine learning strategies to recognize attacks on networks. However, due to the lack of comparative analysis with alternative methods, evaluating its relative effectiveness was challenging.

Xu et al.'s [6] focus was on enhancing the performance of autoencoder-based network anomaly detection, particularly using the NSL-KDD dataset. However, due to the small dataset evaluation, additional study is necessary to determine the study's wider applicability and resilience across various scenarios. wider applicability and resilience across various scenarios. Łukasz Saganowski, et al. [7] developed an anomaly detection preprocessor for Snort that makes use of probabilistic and signal processing algorithms to increase detection accuracy by examining 25 network communication features. A. I. Al Suwailem, et al. [8] analyzed Snort and highlighted how the system is dependent on predetermined rules, making it challenging to detect complex attacks. Despite Abubakar, et al. [9] providing a helpful approach for preventing DDoS attacks in real time, the study could have benefited from an extensive examination of a range of DDoS attack instances. Gyamfi and Jurcut [10] reviewed the literature with a focus on intrusion detection in IoT systems and highlighted the significance of using machine learning and multi-access edge computing, but they did not provide any recommendations or case studies. Together, these efforts advance our understanding of network security and intrusion detection.

Ever, et al. [11] explored the impact of the NSL-KDD dataset on intrusion detection and found that altering the training ratio for attack instances did not directly influence system performance. Verma and Ranga [12] examined classification algorithms against Denial of Service (DoS) malware, revealing that traditional datasets provided only moderate success compared to more advanced defense systems. Maseer, et al. [13] conducted a survey benchmarking machine learning for anomaly-based IDS, emphasizing the superior capabilities of NB-AIDS, K-NN-AIDS, and DT-AIDS in detecting web attacks. Amar, et al.'s [14] study compared power/node scenarios for detecting DoS attacks in mobile IoT, showcasing varying detection rates. The issue of false alarms in IDS was examined by G.C. Tjhai, et al. [15], discussed reducing false alarms through process tuning while recognizing the conflict between preventing false alarms and maintaining safety. Thakur et al. [16] introduced a novel IDS using a Generic-Specific autoencoder architecture, emphasizing ongoing efforts to enhance intrusion detection methodologies while altering risks associated with the internet. To identify DDOS attacks in the network, Muyideen AbdulRaheem, et al. [17] used machine learning integrated with Snort and Zeek. By incorporating new DDoS detection features, the model enhances real-time experiment performance over present methods by lowering false positives as well as execution times.

Lastly, A. Chaudhary, et al. [18] examined the shortcomings of conventional security measures and how reliance on mobile ad hoc networks creates vulnerabilities. The study evaluates the effectiveness of fuzzy logic-based intrusion detection systems, examines their analysis, and suggests future directions for improving intrusion detection. Network system security is of utmost importance, and detecting network intrusions is a crucial aspect of it. In the KDD99 dataset that represents real-time networks, a combination of Discrete Wavelet Transforms and Artificial Neural Networks has been used by Y. Hamid, et al. [19] to provide better accuracy with a more complex detection cost. A novel classification algorithm has been implemented in the original intrusion detection system, which outperforms traditional methods on all performance metrics, as mentioned in the study published by D. Ashok Kumar, et al.[20]. Furthermore, this algorithm exhibits efficient parallel computing, which is a noteworthy progression in network security. Despite difficulties with network security, the probabilistic secure algorithm by M. Azhagiri, et al. [21] improves alert detection at the cluster level. It is more effective than existing systems and has demonstrated its strength thus far, with superior Entropy and Accuracy scores over the majority of other approaches. This will undoubtedly lead to a significantly higher Intrusion Detection Efficiency. Y. Hamid, et al. [22] proposed a classifier that employs SVM and t-SNE dimension reduction for use in network security applications. This classifier can accurately distinguish between malicious and legitimate data packets and provides more precise detection against multiple attack groups.

III. Contributions

In our study, we have made significant contributions towards improving intrusion detection capabilities by combining Snort IDS with machine learning algorithms. We have developed techniques for pre-processing and feature engineering, which have greatly improved the accuracy of the model in identifying cyber threats. Our results have been presented using visual tools, such as pie charts and bar graphs, that help in understanding the performance of our model across different types of attacks.

Our evaluation methodology goes beyond simple classifications and includes scenarios with multiple classes. This allows us to better understand how effective our model is in handling different classification challenges. By focusing on these aspects, our research brings significant improvements to the field of intrusion detection, leading to more adaptive and intelligent security systems.

IV. Methodology

A widely used system for intrusion detection is Snort, which is available for free. It constantly monitors network information in real time and compares it to predefined rules to detect any security threats. However, since the rules only cover known attacks, its accuracy is limited. To improve its performance, the Snort IDS system is integrated with machine learning algorithms. In our work, we are comparing different machine learning models to determine which one can detect threats with higher accuracy. We are using the NSL-KDD dataset to train our models. The models we are considering are Random Forest, Decision Tree, Support Vector Machine, K-Nearest Neighbour, and Naive Bayes. The suggested system design is displayed in Fig.1.

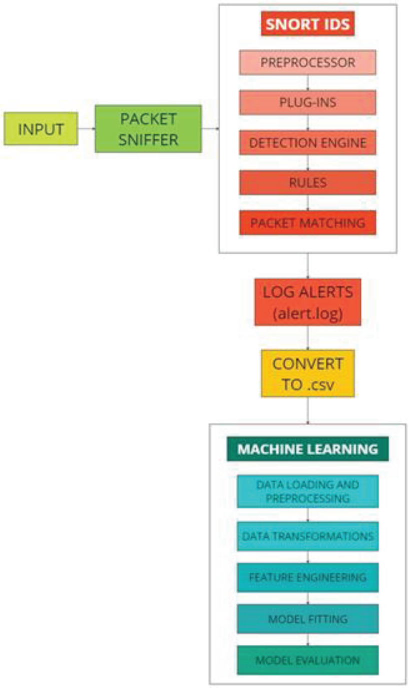

Fig. 1. The Suggested Methodology's System Architecture

A. System Setup

We have set up an Ubuntu 22.04 system to install and configure Snort IDS for monitoring network traffic. We used a virtual machine for Kali and Metasploitable, where Kali is the attacker machine and Metasploitable is the victim machine. Ubuntu and Metasploitable are on the same network, while Kali is on a different network. Snort IDS is installed on the Ubuntu system to protect the victim machine from threats by the attacker machine in the same network.

B. Input

The input for the Snort IDS is the network packets that travel through the network interface of the victim machine. These packets consist of headers that contain source/destination addresses, protocol information, etc., and payloads that carry the actual data.

C. Packet Sniffer

The packet sniffer of the Snort IDS captures the network packets in the network interface. The Promiscuous Mode is turned on, allowing it to capture all the network traffic in the victim's network. Snort utilizes a library called "Libpcap" that provides a common interface for packet capture across various platforms.

D. Snort IDS

Snort IDS is a packet sniffer that captures network packets from the network interface. It uses Promiscuous Mode to capture all network traffic in the victim's network. To capture packets across various platforms, Snort uses a library called "Libpcap".

1) *Preprocessor:* The preprocessor is responsible for preparing captured network packets for analysis. This includes tasks like packet reassembly, protocol decoding, and traffic normalization.

2) *Plug-ins:* Plug-ins are dynamic features that extend Snort's functionality. They enhance Snort's ability to identify and respond to specific threats or vulnerabilities.

3) *Detection Engine:* The detection engine is the core part of the IDS that analyzes network packets against a set of rules and signatures to detect potential security threats.

4) *Rules:* Snort rules, which define patterns or criteria for detecting specific threats or attack patterns in network traffic, are a fundamental part of Snort IDS.

There are two types of rules in Snort:

a) *Community Rules:* These are openly accessible rules developed by the Snort community. Security professionals and organizations contribute to these rules to enhance Snort's detection capabilities by identifying known threats and vulnerabilities.

b) *Local Rules:* These are customizable rules that can be tailored to meet an organization's unique network requirements and security needs. They are customized to address unique security needs, network configurations, or specific threats relevant to an organization's environment.

A Snort rule has the following syntax:

snort_action protocol source_ip_address source_port_number -> destination_ip_address destination_port_number (options)

Rule options include:

a) *msg:* A custom message to include in the alert.

b) *content:* The content (pattern or string) to match within the packet payload.

c) *sid:* A unique Snort ID assigned to the rule.

d) *classtype:* The classification of the rule (e.g., "attempted-recon").

e) *flow:* The direction of the flow (e.g., to_server, to_client).

f) detection_filter: Additional packet filtering criteria.

g) threshold: Threshold settings for triggering the rule based on occurrence count.

h) metadata: Metadata providing information about the rule.

5) Packet Matching: Packet matching involves comparing captured network packets with known attack patterns or any suspicious activity. When a captured packet matches with the signatures, the Snort IDS generates an alert.

E. Log Alerts

The alerts generated by the Snort IDS are logged in a file (alert.log). These alerts contain important details about the detected suspicious activity in the network, including Rule ID, Description, Priority, Timestamp, Source IP address: Port number, Event Specific Details, Protocol Details, and Destination IP address: Port number.

F. Converting to CSV format

To integrate Snort IDS with machine learning, we need to convert the log file generated by Snort into a .csv file format. This is because machine learning models can only read .csv files.

G. Machine Learning

1) Data Loading and Preprocessing: We loaded the training and testing datasets and used the NSL-KDD dataset to train our model. This dataset contains network traffic records categorized into normal and various types of attacks, with features such as protocol type, IP addresses, and attack classes. We defined the columns for the dataset to further analyze the data and then added binary labels to differentiate the network traffic to normal and attack.

2) Data Transformations: We classified the attacks into four primary categories: Denial of Service, Probe, Privilege Escalation, and Remote Access as shown in Fig. 2. These categories were mapped to numerical values (1, 2, 3, and 4 respectively) to distinguish the attacks.

3) Feature Engineering: We used one-hot encoding to convert categorical features like 'protocol_type,' 'service,' and 'flag' into a binary format suitable for machine learning models. Additionally, missing columns in the test dataset were added and filled with zeroes.

4) Model Fitting: We utilized two classifiers for model fitting: a Binary Classifier and a Multi-Class Classifier. The Binary Classifier is responsible for identifying two classes: Normal and Attack. Meanwhile, the Multi-Class Classifier identifies five classes: Denial of Service, Probe, Privilege Escalation, Remote Access, and Normal. We trained the model by fitting the training data for both classifiers and subsequently making predictions for both the training and testing data.

5) Model Evaluation: We have evaluated our model using various machine learning algorithms including Random Forest, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), and Naive Bayes. For each algorithm, we calculated accuracy scores and classification reports for both binary and multi-class classifications. Based on these results, we identified the best machine-learning model that can be integrated with Snort to improve its detection capabilities.

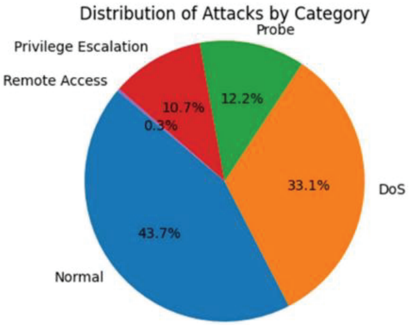

Fig. 2. Distribution of attacks

V. Results and Analysis

The following is a summary of the results of our project's classification models. Table I outlines the binary classification results for our five models, while Table II provides the multi-class classification results. We evaluated the algorithms based on four metrics: accuracy, recall, F1-score, and precision.

We compared the different algorithms for binary and multi-class classification and presented our findings in Fig. 3 and Fig. 4, respectively. Our results indicate that the Random Forest model outperformed the other models, with an accuracy of 99.30% in binary classification and 97.64% in multi-class classification. Therefore, we believe that our project demonstrates the effectiveness of integrating the Random Forest model with Snort IDS for Anomaly Detection.

TABLE I. THE RESULTS OF BINARY CLASSIFICATION

Algorithm	**Binary Classification**			
	Accuracy	***Recall***	***F1-score***	***Precision***
Random Forest	0.9930	0.9942	0.9926	0.9910
Decision Tree	0.9926	0.9942	0.9921	0.9900
SVM	0.9328	0.8977	0.9257	0.9555
KNN	0.9906	0.9885	0.9899	0.9913
Naive Bayes	0.5353	0.0177	0.0344	0.5546

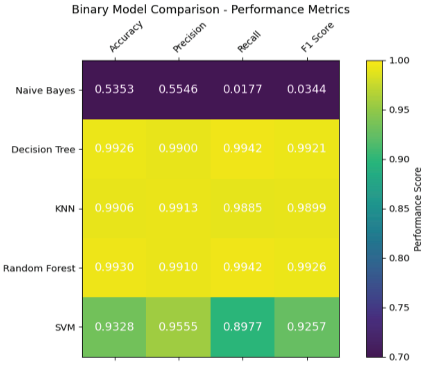

Fig. 3. The comparison of different algorithms for binary classification

TABLE II. THE RESULTS OF MULTI-CLASS CLASSIFICATION

Algorithm	Binary Classification			
	Accuracy	***Recall***	***F1-score***	***Precision***
Random Forest	0.9764	0.8698	0.8932	0.9215
Decision Tree	0.9761	0.8717	0.8577	0.8561
SVM	0.9184	0.5356	0.5388	0.7427
KNN	0.9739	0.8095	0.8298	0.8551
Naive Bayes	0.3806	0.2728	0.1537	0.3288

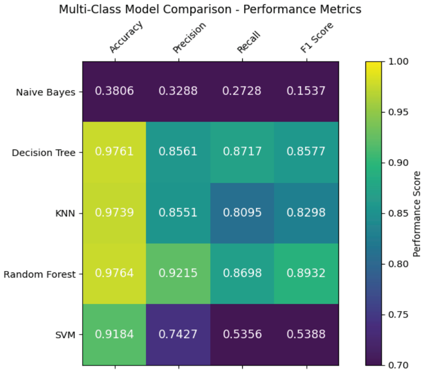

Fig. 4. The comparison of different algorithms for multi-class classification

VI. CONCLUSION AND FUTURE WORK

In this project, we evaluated the precision and efficacy of different machine learning models in detecting network intrusions. We used the NSL-KDD dataset to train five machine-learning models, which were then evaluated using a testing set. The results demonstrated that all four machine learning models performed well on the testing set, with the Random Forest classifier outperforming others by attaining the best accuracy of 99.34%. The multi-class classifier also achieved high accuracy, with an accuracy of 97.65%. This implies that machine learning models can successfully detect network intrusions.

To refine and extend the current findings, further research could focus on several avenues. First, machine learning ensembles, which combine multiple machine learning models, could be investigated to create a more robust and accurate hybrid system. Techniques like stacking or boosting could be explored. Additionally, advanced learning models like neural networks could be used to investigate the identification of increasingly sophisticated and complex attack patterns.

Furthermore, it is essential to implement the integrated Snort and machine learning model in real-world scenarios. Conducting extensive field trials and deployment in diverse network environments would provide invaluable insights into its performance, usability, and scalability. Refining the rules and signature sets of Snort to adapt dynamically to the evolving threat landscape is also a significant area for further exploration.

Lastly, developing a user-friendly interface or dashboard to interpret and visualize the alerts generated by the integrated system would significantly aid network administrators in efficiently monitoring and responding to security threats.

REFERENCES

[1] M. Ozkan-Okay, R. Samet, Ö. Aslan, and D. Gupta, "A comprehensive systematic literature review on intrusion detection systems," IEEE Access, vol. 9, pp. 157727-157760, 2021.

[2] F. A. Vadhil, M. F. Nanne, and M. L. Salihi, "Importance of machine learning techniques to improve the open source intrusion detection systems," Indonesian Journal of Electrical Engineering and Informatics (IJEEI), vol. 9, no. 3, pp. 774-783, 2021.

[3] A. Alhasani, T. Alzahrani, R. alFahhad, M. Alotaibi, et al., "Role of machine learning in intrusion detection system: A systematic review," International Journal of Computer Science & Network Security, vol. 22, no. 3, pp. 155-162, 2022.

[4] M. ALshammari and M. A. Mezher, "A review analysis investigating the efficacy of machine learning in intrusion detection," in Artificial Intelligence for Sustainable Finance and Sustainable Technology: Proceedings of ICGER Springer 2021, vol. 1, pp. 266-274.

[5] C. Zhang, Y. Chen, Y. Meng, F. Ruan, R. Chen, Y. Li, and Y. Yang, "A novel framework design of network intrusion detection based on machine learning techniques," Security and Communication Networks, vol. 2021, pp. 1-15, 2021.

[6] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Improving performance of autoencoder-based network anomaly detection on NSL-KDD dataset," IEEE Access, vol. 9, pp. 140136-140146, 2021.

[7] Ł. Saganowski, M. Goncerzewicz, and T. Andrysiak, "Anomaly detection preprocessor for SNORT IDS system," in Image Processing and Communications Challenges 4, pp. 225-232, Springer, 2013.

[8] A. I. Al Suwailem, M. Al-Akhras, and K. K. A. Ghany, "Evaluating Snort alerts as a classification features set," in Applications of Artificial Intelligence in Engineering: Proceedings of First Global Conference on Artificial Intelligence and Applications (GCAIA 2020), pp. 801-812, Springer, 2021.

[9] A. R. Muhammad, P. Sukarno, and A. A. Wardana, "Integrated security information and event management (SIEM) with intrusion detection system (IDS) for live analysis based on machine learning," Procedia Computer Science, vol. 217, pp. 1406-1415, 2023.

[10] R. Abubakar, A. Aldegheishem, M. F. Majeed, A. Mehmood, H. Maryam, N. A. Alrajeh, C. Maple, and M. Jawad, "An effective mechanism to mitigate real-time DDoS attack," IEEE Access, vol. 8, pp. 126215-126227, 2020.

[11] E. Gyamfi and A. Jurcut, "Intrusion detection in internet of things systems: a review on design approaches leveraging multi-access edge computing, machine learning, and datasets," Sensors, vol. 22, no. 10, p. 3744, 2022.

[12] Y. K. Ever, B. Sekeroglu, and K. Dimililer, "Classification analysis of intrusion detection on NSL-KDD using machine learning algorithms," in International Conference on Mobile Web and Intelligent Information Systems, pp. 111-122, Springer, 2019.

[13] A. Verma and V. Ranga, "Machine learning based intrusion detection systems for IoT applications," Wireless Personal Communications, vol. 111, no. 4, pp. 2287-2310, 2020.

[14] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly-based intrusion detection systems in the CICIDS2017 dataset," IEEE Access, vol. 9, pp. 22351-22370, 2021.

[15] G. C. Tjhai, M. Papadaki, S. M. Furnell, and N. L. Clarke, "Investigating the problem of IDS false alarms: An experimental study using Snort," in IFIP Advances in Information and Communication Technology, vol. 268, pp. 253-267, Springer, 2008.

[16] S. Thakur, A. Chakraborty, R. De, N. Kumar, and R. Sarkar, "Intrusion detection in cyber-physical systems using a generic and domain-specific deep autoencoder model," Computers & Electrical Engineering, vol. 91, p. 107044, 2021.

[17] M. AbdulRaheem, I. D. Oladipo, A. L. Imoize, J. B. Awotunde, C.-C. Lee, G. B. Balogun, and J. O. Adeoti, "Machine learning assisted Snort and Zeek in detecting DDoS attacks in software-defined networking," International Journal of Information Technology, pp. 1-17, 2023.

[18] A. Chaudhary, V. N. Tiwari, and A. Kumar, "Analysis of fuzzy logic based intrusion detection systems in mobile ad hoc networks," BVICA M's International Journal of Information Technology, vol. 6, no. 1, pp. 69-73, 2014.

[19] Y. Hamid, F. A. Shah, and M. Sugumaran, "Wavelet neural network model for network intrusion detection system," International Journal of Information Technology, vol. 11, no. 1, pp. 251-263, 2019.

[20] D. Ashok Kumar and S. R. Venugopalan, "A design of a parallel network anomaly detection algorithm based on classification," International Journal of Information Technology, vol. 14, no. 4, pp. 2079-2092, 2022.

[21] M. Azhagiri and A. Rajesh, "A novel approach to measure the quality of cluster and finding intrusions using intrusion unearthing and probability clomp algorithm," International Journal of Information Technology, vol. 10, no. 2, pp. 329-337, 2018.

[22] Y. Hamid and M. Sugumaran, "A t-SNE based non-linear dimension reduction for network intrusion detection," International Journal of Information Technology, vol. 12, no. 2, pp. 125-134, 2020.