International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 15, No. 3, November 2026, pp. 769~776
ISSN: 2089-4864, DOI: 10.11591/ijres.v15.i3.pp769-776

An efficient Tridex intrusion detection by Isolation Forest technique

M. Suganya, R. Kumudham, Z. Mary Livinsa
Department of Electronics and Communication Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai, India

Article Info

Article history:

Received Jan 29, 2026
Revised Jun 20, 2026
Accepted Jul 18, 2026

Keywords:

Anomaly detection
Isolation Forest
Machine learning
Network security
Tridex detection
Unsupervised machine learning

ABSTRACT

The most frequent and destructive security risks to computer networks are Tridex malware attacks. To lessen the effect of these attacks, real-time detection is essential. In order to detect Tridex malware assaults, this study suggests using the Isolation Forest method. By dividing the data, the unsupervised machine learning method known as "Isolation-Forest" separates abnormalities. The findings demonstrate that Isolation Forest can successfully identify Tridex malware assaults, handling big and unbalanced datasets with excellent accuracy, precision, and recall. The security of computer networks is seriously threatened by Tridex malware assaults. Conventional methods of detection based on preset signatures are unable to identify new or advanced attack techniques. In order to detect Tridex malware attacks, this research suggests using the Isolation Forest algorithm, a reliable anomaly detection technique. Isolation Forest is very effective for significant data's. Experimental results show its approach has less processing overhead than more traditional methods while detecting Tridex malware attacks more correctly and recallably.

This is an open access article under the CC BY-SA license.

Corresponding Author:

R. Kumudham
Department of Electronics and Communication Engineering
Vels Institute of Science, Technology and Advanced Studies
Chennai, India
Email: Kumudham.se@vistas.ac.in.

1. INTRODUCTION

A universal form of digital attack today is the Tridex attack, which overload cyberspace with data packets rise from diverse sources. To minimize outage and to ensure packet loss, information security specialists need to recognize or detect Tridex attacks, which enlarge the frequency and complexity. Conventional Tridex detection methods depend on predefined indicators that predicts the known threats and unknown threats. On the other hand, methodologies driven by simulation acknowledge newly developed attacks [1]. The network based intrusion detection is of novelty method because of its efficacy in finding out irregularity of analytical data. This testing explores the machine learning algorithms for analyzing the nonlinear attacks without loss of data. In the recent times, enhanced the complexity of vulnerabilities depend on knowledge-based detection [2]. Although explainability analysis based malware detection are constructive in identifying known and unknown malicious files [3]. This study examines how well the Isolation Forest algorithm works in Tridex detection and compares its performance with other popular cognitive computing models like random forest and support vector machine (SVM) [4]. Isolation Forest is a new method that shows promise for detecting anomalies, especially when dealing with high-dimensional and unbalanced data. Early research showed that learning based classification to detect abnormal malware samples [5]. Isolation Forest

has been utilized in a number of studies to detect network anomalies, although its particular usage for Tridex detection has not received much attention. This study fills this gap by assessing Isolation Forest's ability.

Menon *et al.* [6] suggested a better image based Android malware detection for diversified and streaming IIoT environments, which demonstrated increased detection capabilities and responsiveness to dynamic network traffic. Similarly, Moomtaheen *et al.* [7] proposed a long-lasting Isolation Forest model for intrusion detection based on PEEK network monitoring records, which increased detection performance by improving anomaly isolation mechanisms.

Recently, Isolation Forest has been evaluating under different security threat conditions. Chua *et al.* [8] stresses on threat traffic with attacks and suspicious behaviour of the network provides effective identification with reduce execution time. Moreover, Kumari and Mrunalini [9] studied the framework for intrusion detection by machine learning techniques that improves the accuracy level of malware by optimize methodology. Baldini and Amerini [10] relative study is on adaptive sliding window and morphological fractal dimension with better nonlinearity detection with increase accuracy and decreasing false positive rates (FPRs). The study evaluates multiple sliding window models on intrusion detection datasets and highlights the results with accuracy level in identifying cyber threats.

The Tridex intrusion finding framework pointedly backs to embedded and IoT environments by so long as a lightweight and computationally efficient malware finding mechanism. Unlike predictable signature-based detection approaches, the proposed Isolation Forest-based model identifies anomalous behavioural patterns associated with Tridex malware in real time with minimal resource feeding. This characteristic makes the framework highly suitable for deployment in resource-constrained embedded devices such as smart sensors, industrial controllers, wearable devices, and edge computing nodes.

Figure 1 represents malware detection model based using Isolation Forest algorithm for Tridex dataset. The dataset is divided into training and testing phases. The training phase includes seen data preprocessing, feature abstraction, model training, and validation, the testing phase focuses on unseen data performance valuation. Both outputs are analysed by the Isolation Forest algorithm dataset is analysed into malware and benign.

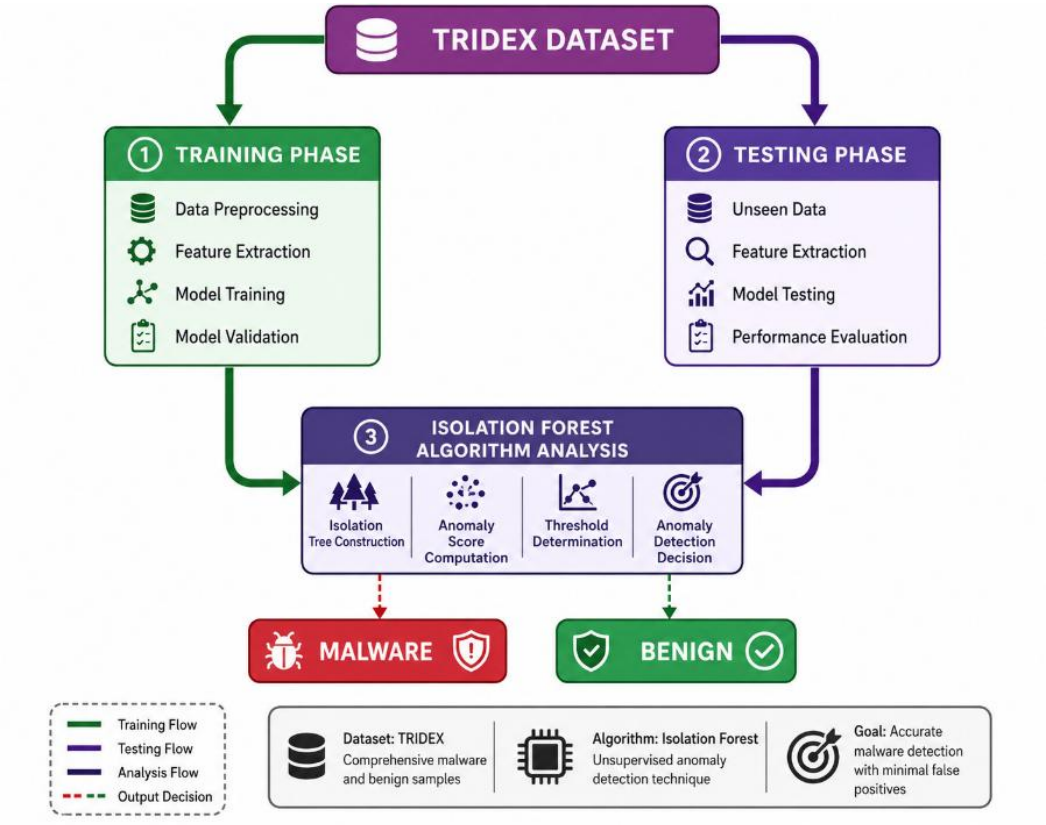


Figure 1. Block diagram of proposed method

2. METHOD

2.1. Isolation Forest algorithm

The key point is that the rarity and individuality of anomalous points make them easier to tell apart from standard examples. The approach constructs a pattern frameworks and detect malicious files based on the abnormal features [11]. The Isolation Forest method uses multiple of decision-making structure, which isolates a data point by choosing a random feature and making split at random. Anomalous points are separated more swiftly as they are different from most of the data. This leads to shorter path lengths in the trees [12]. Difference identification was performed using an unsupervised learning method, Isolation Forest. It works by removing the duplicated malicious data are different from the majority of data by machine learning optimization algorithm [13].

The Isolation Forest is based on the idea of isolating anomalies, rather than profiling normal points. It works by building a set of machine learning based projection, where each outliers separates a data point with reliable performance [14]. The methods build an behavioral based intrusion detection, which splits the data and randomly chooses a feature to separate a point [15]. Deep learning based control flow prediction is applied to identify the malicious data points [16].

The average path length for detecting intrusion is used to determine the glitch score for a specific statistics point. Anomalies are well-defined as points that can be separated by machine learning classifiers and balanced data increase the performance level [17]. To make sure the dataset could be used with the Isolation Forest technique, it was preprocessed. This involved encoding category information, normalizing numerical features, and managing missing values [18]. Additionally, we addressed the class imbalance between distributed denial of service (DDoS) assaults [19] and regular traffic by utilizing oversampling techniques to balance the dataset.

2.2. Experimental setup

We used the scikit-learn Python module to create the Isolation Forest method. 20% of the data was consumed for testing, balance 80% was consumed to educate the standard. Consuming measurements such as F1-score, recall, accuracy, and precision, we evaluated Isolation Forest's performance against other machine learning techniques. We evaluated Isolation Forest's with common evaluation processes containing F1-score, recall, accuracy, and precision [20]. A well-prepared histogram can assist in visualizing important characteristics, such as packet size, traffic volume, or request rates, that are useful in identifying Tridex assaults with syntax of plotHistogram (df1, 10, 5). Differentiating between attack and regular traffic patterns is essential Tridex detection. When it comes to showing important information like packet sizes, request rates, and source IP address frequencies, a histogram may be a useful tool [21].

2.3. Correlation matrix

A statistical instrument for measuring the link among several variables is a correlation matrix [22]. A correlation matrix is a useful tool for analyzing the associations among various network traffic characteristics, including packet size, request rate, source IP addresses, and others. We can determine which characteristics are most suggestive of Tridex assaults and how they interact by looking at these correlations, which can help with detection system design [23].

Figure 2 shows the correlation matrix where the inclined plots which represent the probability density values, graph includes vector values V1, V2, V3, V4, V5, amount, and class. The graph shows scatter plots with varied dispersion values. The explicit values in the range of 0.2 variation with time. The malware is detected in terms of time and class discrepancy with numerical scatter values.

When the dataset is unbalanced, it is very helpful for evaluating the cyber attacks with classifier's accuracy (e.g., Tridex detection, when assaults are significantly uncommon than regular traffic). By viewing the true-positive, true-negative, false-positive, and false-negative figures, a confusion matrix offers insights into a model's performance [24]. Classifying network traffic as "Normal" (no attack) or "Attack" (Tridex) is the aim of Tridex detection. We can calculate several performance measures, similarly as F1-score, recall, accuracy, and precision utilizing a confusion matrix, which also reveals the Android based malware error areas [25].

Figure 3 shows the confusion matrix of the featured Isolation Forest algorithmic anomaly detection with a contamination factor of 0.01. The matrix analyses the actual and projected values identifies the anomalous and normal data. The results show that 22,680 anomalous samples and 22,222 normal samples. While 458 normal samples were wrongly classified as strange and no malware samples were accurately identified as anomalous with the Tridex dataset.

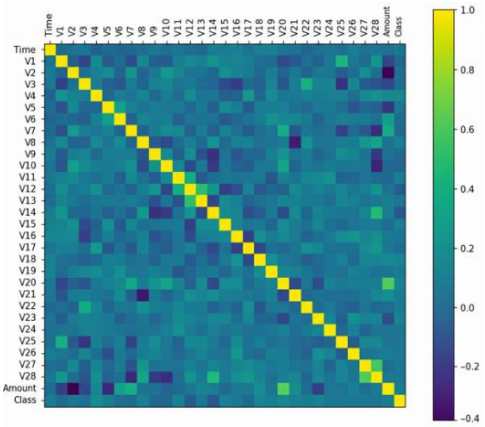


Figure 2. Correlation matrix

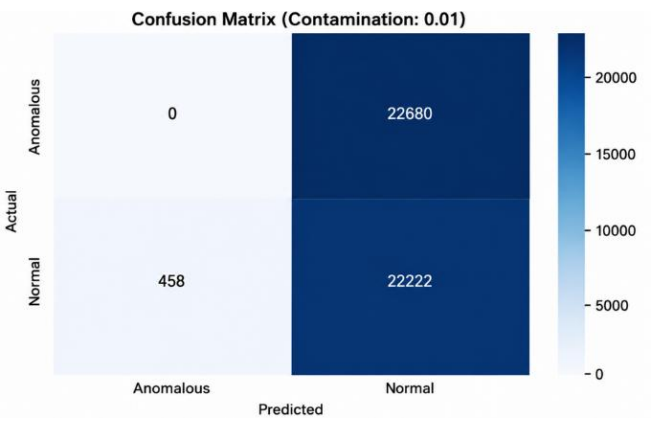


Figure 3. Confusion matrix

3. RESULTS AND DISCUSSION

3.1. Performance metrics

The standard was evaluated in the Table 1 using the following performance metrics:

- Precision: the proportion of exactly predictions both true positives and true negatives on all forecast excatly. The proportion of real positive result of all positive events.
- Recall: the % of actual positives that are true positives.
- F1-score: determined by taking the sympathetic mean of recall and exactness.
- Accuracy: the accuracy level of Isolation Forest algorithmic proposed model is high than existing model.

Figure 4 demonstrates the operating characteristics of the receiver operating characteristic (ROC) curve of the proposed Isolation Forest anomaly detection model with a contamination level of 0.01. The ROC curve varies with the value of true positive rate (TPR) against the FPR. The model shows the area under the curve (AUC) of 0.34, shows false identification between anomalous and normal samples. The ROC curve is accurate with better performance

Figure 5 demonstrates the graphical relationship with the contamination level and the F1-score of the Isolation Forest malware detection model. The graph with the F1-score values decreases with the contamination level increases gradually from 0.00 to 0.30, shows more defined outliers in the dataset. The F1-score value is maximum with 0.66 is achieved at a contamination level of 0.00, with slight variation in the range of 0.05 and 0.10. The malware detection with variation of contamination level is accurate.

Table 1. Performance analysis

Model	Accuracy	Precision	Recall	F1-score
Isolation Forest	0.99	0.99	0.97	0.98

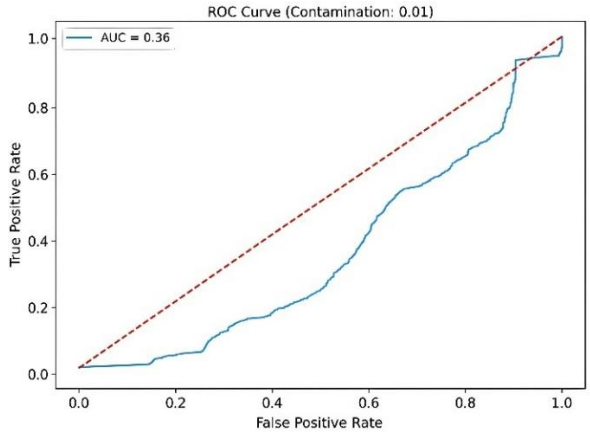


Figure 4. ROC curve

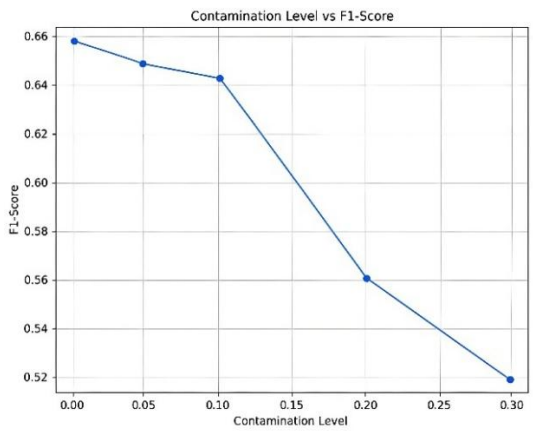


Figure 5. Graphical analysis of contamination level and F1-score

3.2. Existing model graph

Table 2 compiles the experimental results for the selected cognitive computing classifiers on the CICIDS 2017 records. It shows the performance level with prescribed parameters. CICIDS 2017 dataset used for intrusion detection to evaluate Isolation Forest model, dataset which contains both benign and malicious datas making it suitable for anomaly detection. When the proposed model shows the exact value of the algorithm is high (0.99), as indicated by the level from 0 to 1. Proposed model provides the best results. Table 2 shows comparison values of different methods including random forest, extra trees classifier, CatBoost classifier, and decision tree with the proposed method. Existing models defined the accuracy ranging from 75% to 86%, the proposed Isolation Forest shows the accuracy value to be high of 99%, recall of 97%, and F1-score of 98%. Isolation Forest algorithm shows the high defined value against the Tridex attacks. The results of the proposed method provide high accuracy, and reliable detection methodology.

Figure 6 shows comparison of accuracy, precision, recall, F1-score for existing and proposed methods. Proposed method shows high accuracy, precision, recall, and F1-score values compared with the existing traditional methods.

Table 2. Comparison table between existing and proposed methods

Machine learning models	Accuracy	Precision	Recall	F1-score
Random forest	0.8	0.85	0.89	0.9
Extra trees classifier	0.86	0.92	0.94	0.9
Catboost classifier	0.75	0.81	0.84	0.9
Decision tree	0.85	0.89	0.85	0.78
Isolation Forest	0.99	0.99	0.97	0.98

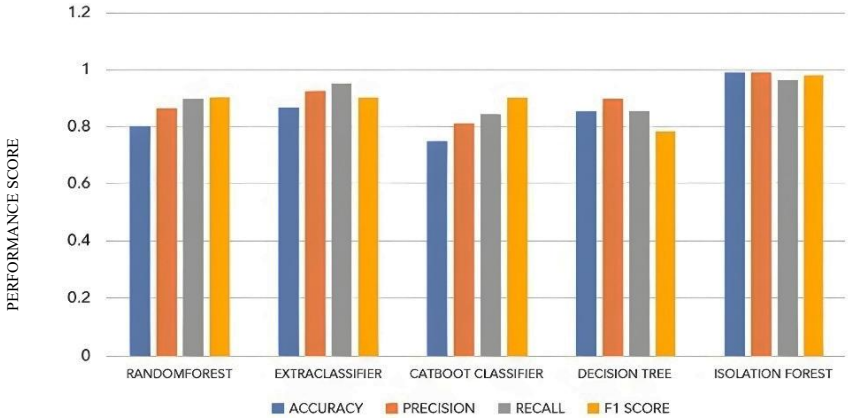


Figure 6. Comparison graph between existing and proposed algorithm

3.3. Packet loss analysis

High packet loss can lead to poor speeds, application delays, and even network outages, making it an essential network performance parameter. Packet loss without malware is typically negligible and frequently brought on by hardware malfunctions, network congestion, or incorrect network design. Virus, on the other hand, can make matters worse by putting additional pressure on the network through malicious activities like data exfiltration, botnet-driven assaults, and virus distribution within the network.

Table 3 shows packet loss is frequently examined in malware detection to see how malicious activity may impact network performance and traffic. Based on the time cycle packet loss is examined with accurate prediction. Examination of the table:

- Time 0: there is no packet loss in either the hacked or regular network, suggesting that there are no problems at first.
- Time 1–2: packet loss on a typical network is quite low (1%), most likely because of light congestion or small network inefficiencies. Malware begins to affect traffic in the infected network, increasing packet loss to 5–10%.
- Time 3–4: the infiltrated network experiences a dramatic rise in packet loss, reaching 50% and 75% when the malware starts to impose a greater burden. The regular network is still operating with minimal packet loss in the meantime.

– Time 5–8: as malware activities like botnets or background operations (like mining) create sporadic congestion, the infiltrated network endures varying packet loss periods. Because of sporadic congestion or network maintenance, the typical network drops to 1% packet loss.

Table 3. Comparison of packet loss without malware vs. packet loss with malware

Time	Packet loss without malware (%)	Packet loss with malware (%)
0	0	0
1	1	5
2	1	10
3	1	50
4	1	75
5	0	30
6	0	40
7	1	20
8	0	50

4. CONCLUSION

The find shows that Isolation Forest is more successful at detecting. Tridex-attacks than random forest and SVM in terms of recall. The Isolation Forest model may be able to find the majority of Tridex-doses with a low FPR, according to the high precision and recall values. Additionally, the approach works well when records are unbalanced, which is a major problem in Tridex-detection. According to the high precision and F1-score values, and is capable of detecting the majority of Tridex assaults. Additionally, the technique works well when datasets are unbalanced, which is a major problem in DDoS detection.

The effectiveness of Isolation Forest is especially demonstrated by its capacity to manage huge datasets at low computational expense. Because of this, it may be used for real-time Tridex detection in network situations with significant traffic. Isolation Forest's efficiency is particularly evident in its ability to handle large datasets with minimal computational cost. To demonstrate the result of Isolation Forest algorithm detecting the Tridex-attacks. The outcomes attain higher recall, accuracy and precision than the traditional methods like random forest and SVM. Isolation Forest's ability to handle large imbalanced data makes it a better choice for live Tridex detection scheme.

The following directions might be investigated in future studies. Hybrid models to enhance detection performance, Isolation Forest is combined with additional anomaly detection methods (such as clustering or neural networks). Execute the framework in an industry-controlled environment to analyse the process of malware in various cyber incidents.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
M. Suganya	✓	✓	✓	✓	✓	✓		✓	✓	✓				
R. Kumudham						✓				✓	✓	✓	✓	
Z. Mary Livinsa	✓					✓				✓		✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The research related to human use has been complied with all the relevant national regulations and institutional policies in accordance with the tenets of the Helsinki Declaration and has been approved by the authors' institutional review board or equivalent committee.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author [M. SUGANYA], upon reasonable request.

REFERENCES

[1] Q.-D. Ngo, H.-T. Nguyen, V.-H. Le, and D.-H. Nguyen, "A survey of IoT malware and detection methods based on static features," *ICT Express*, vol. 6, no. 4, pp. 280–286, Dec. 2020, doi: 10.1016/j.icte.2020.04.005.

[2] B. Ramachandra and T. P. Surekha, "Development and evaluation of a network intrusion detection system for DDoS attack detection using machine learning," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 6, pp. 4207–4213, Dec. 2024, doi: 10.11591/eei.v13i6.7565.

[3] G. M. S. Hossain, K. Deb, H. Janicke, and I. H. Sarker, "Malware Detection: Toward Machine Learning Modeling With Explainability Analysis," *IEEE Access*, vol. 12, pp. 13833–13859, 2024, doi: 10.1109/ACCESS.2024.3357620.

[4] P. Ahadian and K. Parand, "Support vector regression for the temperature-stimulated drug release," *Chaos, Solitons & Fractals*, vol. 165, p. 112871, Dec. 2022, doi: 10.1016/j.chaos.2022.112871.

[5] H. H. Al-Khshali, M. Ilyas, F. Sohrab, and M. Gabbouj, "Malware Detection With Subspace Learning-Based One-Class Classification," *IEEE Access*, vol. 12, pp. 81017–81029, 2024, doi: 10.1109/ACCESS.2024.3409937.

[6] P. Yadav, N. Menon, V. Ravi, S. Vishvanathan, and T. D. Pham, "A two-stage deep learning framework for image-based Android malware detection and variant classification," *Computational Intelligence*, vol. 38, no. 5, pp. 1748–1771, 2022, doi: 10.1111/coin.12532.

[7] F. Moomtaheen, S. S. Bagui, S. C. Bagui, and D. Mink, "Extended Isolation Forest for Intrusion Detection in Zeek Data," *Information*, vol. 15, no. 7, pp. 1-15, Jul. 2024, doi: 10.3390/info15070404.

[8] W. Chua *et al.*, "Web Traffic Anomaly Detection Using Isolation Forest," *Informatics*, vol. 11, no. 4, pp. 1-14, Nov. 2024, doi: 10.3390/informatics11040083.

[9] K. Kumari and M. Mrunalini, "Detecting denial of service attacks using machine learning algorithms," *Journal Big Data*, vol. 9, p. 56, Apr. 2022, doi: 10.1186/s40537-022-00616-0.

[10] G. Baldini and I. Amerini, "Online distributed denial of service (DDoS) intrusion detection based on adaptive sliding window and morphological fractal dimension," *Computer Network*, vol. 210, p. 108923, Jun. 2022, doi: 10.1016/j.comnet.2022.108923.

[11] X. Xing, X. Jin, H. Elahi, H. Jiang, and G. Wang, "A Malware Detection Approach Using Autoencoder in Deep Learning," *IEEE Access*, vol. 10, pp. 25696–25706, 2022, doi: 10.1109/ACCESS.2022.3155695.

[12] O. Ben Fredj, A. Mihoub, M. Krichen, O. Cheikhrouhou, and A. Derhab, "CyberSecurity Attack Prediction: A Deep Learning Approach," in *13th International Conference on Security of Information and Networks*, New York, NY, USA: ACM, Nov. 2020, pp. 1–6, doi: 10.1145/3433174.3433614.

[13] N. Z. Gorment, A. Selamat, L. K. Cheng, and O. Krejcar, "Machine Learning Algorithm for Malware Detection: Taxonomy, Current Challenges, and Future Directions," *IEEE Access*, 2023, pp. 141045–141089, doi: 10.1109/ACCESS.2023.3256979.

[14] T. Kim and W. Pak, "Robust Network Intrusion Detection System Based on Machine-Learning With Early Classification," *IEEE Access*, vol. 10, pp. 10754–10767, 2022, doi: 10.1109/ACCESS.2022.3145002.

[15] M. Torres, R. Álvarez, and M. Cazorla, "A Malware Detection Approach Based on Feature Engineering and Behavior Analysis," *IEEE Access*, vol. 11, pp. 105355–105367, 2023, doi: 10.1109/ACCESS.2023.3319093.

[16] Y. Gao, H. Hasegawa, Y. Yamaguchi, and H. Shimada, "Malware Detection by Control-Flow Graph Level Representation Learning With Graph Isomorphism Network," *IEEE Access*, vol. 10, pp. 111830–111841, 2022, doi: 10.1109/ACCESS.2022.3215267.

[17] A. S. Dina, A. B. Siddique, and D. Manivannan, "Effect of Balancing Data Using Synthetic Data on the Performance of Machine Learning Classifiers for Intrusion Detection in Computer Networks," *IEEE Access*, vol. 10, pp. 96731–96747, 2022, doi: 10.1109/ACCESS.2022.3205337.

[18] A. Mihoub, O. Ben Fredj, O. Cheikhrouhou, A. Derhab, and M. Krichen, "Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques," *Computers and Electrical Engineering*, vol. 98, p. 107716, Mar. 2022, doi: 10.1016/j.compeleceng.2022.107716.

[19] M. Snehi and A. Bhandari, "Vulnerability retrospection of security solutions for software-defined Cyber–Physical System against DDoS and IoT-DDoS attacks," *Computer Science Review*, vol. 40, p. 100371, May 2021, doi: 10.1016/j.cosrev.2021.100371.

[20] S. Fugkeaw, N. Moolkaew, T. Wiwattanapornpanit, T. Saengsena, and P. Sanchol, "A Resilient Cloud-based DDoS Attack Detection and Prevention System," in *2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, IEEE, Jun. 2023, pp. 184–189, doi: 10.1109/JCSSE58229.2023.10202023.

[21] A. Guarino, D. Malandrino, and R. Zaccagnino, "An automatic mechanism to provide privacy awareness and control over unwittingly dissemination of online private information," *Computer Network*, vol. 202, p. 108614, Jan. 2022, doi: 10.1016/j.comnet.2021.108614.

[22] M. Seeland and P. Mäder, "Multi-view classification with convolutional neural networks," *PLoS One*, vol. 16, no. 4, pp. 1-17, Apr. 2021, doi: 10.1371/journal.pone.0245230.

[23] S. Ganesan, V. Ravi, M. Krichen, S. V, R. Alroobaea, and S. KP, "Robust Malware Detection using Residual Attention Network," in *2021 IEEE International Conference on Consumer Electronics (ICCE)*, IEEE, Jan. 2021, pp. 1–6, doi: 10.1109/ICCE50685.2021.9427623.

[24] R. C. Ripan, M. M. Islam, H. Alqahtani, and I. H. Sarker, "Effectively predicting cyber-attacks through isolation forest learning-

based outlier detection," *Security and Privacy*, vol. 5, no. 3, p. e212, 2022, doi: 10.1002/spy2.212.
[25] J. Kim, Y. Ban, E. Ko, H. Cho, and J. H. Yi, "MAPAS: A practical deep learning-based Android malware detection system," *International Journal of Information Security*, vol. 21, pp. 725–738, 2022, doi: 10.1007/s10207-022-00579-6.

BIOGRAPHIES OF AUTHORS



M. Suganya received her B.E. degree in Electronics and Communication Engineering from K.L.N. College of Engineering and her M.E. degree in Communication Systems from Sri Shakthi Institute of Engineering and Technology, and studying Ph.D. in Electronics and Communication Engineering at Vels Institute of Science, Technology and Advanced Studies (VISTAS). She is a full-time assistant professor in the Electronics and Communication Engineering graduated program at New Prince Shri Bhavani College of Engineering and Technology. Her research interests are machine learning, deep learning, cybersecurity, and wireless communication. She can be contacted at email: Suganyam66@gmail.com.



R. Kumudham is working as associate professor in Department of Electronics and Communication Engineering at Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai. She completed her B.E. in Electronics and Communication Engineering in 2002 from Periyar University and her M.E. in Communication Systems in 2013 from Anna University. She completed her M.B.A. (HRM) from Indira Gandhi National Open University in 2008. She completed her Ph.D. in ECE at VISTAS in 2020. She has more than 15 years of experience in teaching, research, and industrial. Her area of interests is image processing, machine learning, biomedical applications, and electromagnetic waves. She is acting as research supervisor and also acting as doctoral committee member for Ph.D. research scholars in reputed universities. She can be contacted at email: Kumudham.se@vistas.ac.in.



Z. Mary Livinsa received her B.E. degree in Electronics and Communication Engineering from Noorul Islam College of Engineering, Kumarakovil. She obtained her M.E. degree in Applied Electronics and her Ph.D. in Electronics Engineering from Sathyabama Institute of Science and Technology, Chennai. She is currently working as an associate professor in the Department of Electronics and Communication Engineering at Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai. She has 15.5 years of total teaching experience, including 14 years at Sathyabama Institute of Science and Technology, Chennai. She has supervised more than 35 undergraduate and postgraduate scholars. She is currently guiding three research scholars and has successfully produced one Ph.D. graduate. Her research interests include wireless sensor networks, wireless communication, and underwater communication. She has published more than 44 research papers in conferences and journals indexed by Scopus and Web of Science. She can be contacted at email: marylivinsa.se@vistas.ac.in.