Network intrusion detection system by applying ensemble model for smart home

Malothu Amru[1], Raju Jagadeesh Kannan[2], Enthrakandi Narasimhan Ganesh[3], Surulivelu Muthumarilakshmi[4], Kuppan Padmanaban[5], Jeyaprakash Jeyapriya[6], Subbiah Murugan[7]

[1]Electronics Communication Engineering, CMR Engineering College, Hyderabad, India
[2]Dean Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Tiruchirappalli, India
[3]Department of Electronics and Communication Engineering, St. Peters Institute of Higher Education and Research, Chennai, India
[4]Department of Computer Science and Engineering, S.A. Engineering College, Chennai, India
[5]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vijayawada, India
[6]ML Engineer, Apcomart Private Limited, Bangalore, India
[7]Department of Biomedical Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, India

Article Info

Article history:

Received Oct 2, 2023
Revised Jan 11, 2024
Accepted Jan 12, 2024

Keywords:

Gradient boosting
Intrusion detection
Mirai attacks
Multi-class prediction
Smart home systems

ABSTRACT

The exponential advancements in recent technologies for surveillance become an important part of life. Though the internet of things (IoT) has gained more attention to develop smart infrastructure, it also provides a large attack surface for intruders. Therefore, it requires identifying the attacks as soon as possible to provide a secure environment. In this work, the network intrusion detection system, by applying the ensemble model (NIDSE) for Smart Homes is designed to identify the attacks in the smart home devices. The problem of classifying attacks is considered a classification predictive modeling using eXtreme gradient boosting (XGBoosting). It is an ensemble approach where the models are added sequentially to correct the errors until no further improvements or high performance can be made. The performance of the NIDSE is tested on the IoT network intrusion (IoT-NI) dataset. It has various types of network attacks, including host discovery, synchronized sequence number (SYN), acknowledgment (ACK), and hypertext transfer protocol (HTTP) flooding. Results from the cross-validation approach show that the XGBoosting classifier classifies the nine attacks with micro average precision of 94% and macro average precision of 85%.

Corresponding Author:

Malothu Amru
Electronics Communication Engineering, CMR Engineering College
Hyderabad, India
Email: malothuamru@gmail.com

1. INTRODUCTION

Smart home systems are most sought after these days for securing homes conveniently and automatedly. They are also used for efficient resource management as well. It is particularly useful for wellness-assisted living, monitoring the health condition of older adults who prefer to stay in their homes rather than be in hospitals. Despite so many benefits, smart home systems have security and privacy concerns, especially with baby monitors being hacked. Mirai's malware attack is on closed-circuit television (CCTV) systems, which has questioned the security and privacy of smart home systems.

Technically, these smart home devices have firmware with limited hardware and memory capacity, so neither an antivirus nor a patch can be applied. The end users of smart home devices often need more technical knowledge to handle these devices in case of a breach or a hack. Also, the heterogeneous nature of the multitude of devices available in the market needs a common protocol and standard, which lacks security by design. To counter these issues, all the smart home devices used in a home should be monitored and tracked by a gateway/central hub to provide end-point security. When an intrusion or an abnormal event occurs, proper alert messages should be generated, and the end users must be notified. The proposed system in this work introduces a smart hub device that passively monitors the traffic of the smart home devices connected in a home.

A network intrusion detection (NID) system using a random forest (RF) algorithm is discussed in [1]. It clusters the data after preprocessing to classify the attacks effectively. Asymmetrical uncertainty-based feature sub-set selection is employed to select the sub-set of features. A comprehensive review of the intrusion detection system (IDS) is discussed in [2]. Different signature-based and network-based intrusion detection systems are evaluated by the use of a support vector machine (SVM), genetic algorithms, hidden Markov model, decision tree (DT), k-nearest neighbors (KNN), and random forest (RF). Deep learning (DL) based NID system is discussed in [3]. It uses one one-dimensional convolution layer and a pooling layer to extract the deep features and then a dense layer that utilizes a neural network for intrusion detection. A systematic study of NID systems is discussed in [4]. The strengths and limitations of different machine learning (ML) algorithms such as DT, SVM, KNN, and artificial neural network (ANN), and DL algorithms such as autoencoder, recurrent neural networks (RNN), convolution neural network (CNN), deep neural network (DNN), and deep belief neural network (DBNN) are described.

A weighted RF feature selection approach is discussed in [5] for NID based on Gini impurity. It is a binary classification system. In the preprocessing step, categorical feature encoding is employed first, followed by feature selection and scaling. After scaling, classifiers such as DT, AdaBoost, and gradient boosting trees are employed for the classification. A flow-based NID is described in [6]. It uses energy-based flow classifiers instead of using conventional ML classifiers. It is based on the inverse statistics of labeled benign samples to infer the statistical model. A DBNN-based NID system is discussed in [7]. The backpropagation training algorithm in the CNN is replaced by an extreme learning machine (ELM). Also, a gray wolf optimizer is used to optimize the ELM's parameters. A hybrid classification system for NID is discussed in [8]. It provides real-time detection by classifying the packets when it is received. It uses DT and AdaBoost DT classifiers for detecting attacks. A combination of naive Bayes (NB) and SVM is discussed for NID in [9]. These classifiers are deployed in different layers for the classification, and principal component analysis is used to analyze the common properties of different attacks. Genetic algorithm (GA) based feature selection for NID systems is discussed in [10]. It employs DT as a fitness function and classifies the attacks as normal or abnormal. It uses RF, NB, and DT as classifiers for detection. Different feature selection approaches include Cuckoo search [11], butterfly optimization algorithm [12], particle swarm optimization (PSO) [13], and GA [14], [15]. A multipath delay commutator has been proposed to enhance the throughput and speed [16], [17]. Though, several experts need help to trust ML [18]. A baby monitor is hacked, and the hacker controls the camera and reportedly talks with the kid [19]. This and a few other similar incidents put the security and privacy of smart homes under concern. The vulnerabilities of the D-Link internet protocol camera (DCS930L) make weak authentication, which causes a type of replay attack [20], [21]. Experiments on smart home devices like Nest smoke alarms, Philips light bulbs, and WeMo switches showed that messages are not encrypted and transferred in plain text. Hackers could use this information to detect the presence and to launch back door or replay attacks. Figure 1 shows the attacker sniffing wireless packets to detect the activities inside the home.

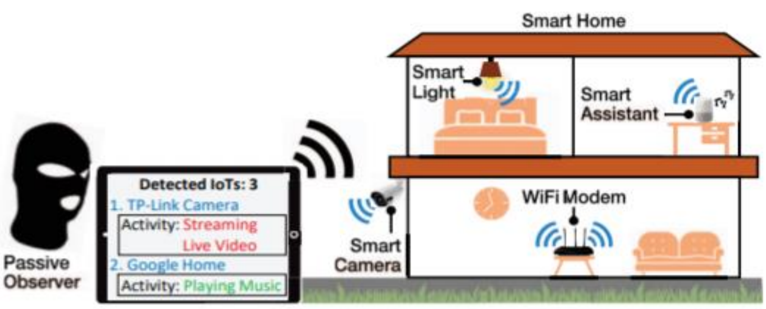

Figure 1. Attacker sniffing wireless packets to detect the activities inside the home

From Figure 1, any passive hacker within 10 m of the target home can passively sniff encrypted packets of the interactions with smart home devices [22]. Mirai, a distributed denial of service (DDOS) attack, affected many IoT systems like IP cameras and home routers, which made the internet inaccessible to users. A Botnet is a collection of devices in a network that an adversary outside of the network controls. There are nearly 13 variants of Mirai reported by early 2020. The attack initiated from a group of "IoT" makes it difficult to trace the malware as it bypasses normal DDOS tracking mechanisms. Table 1 describes the malware attacks on the network level [23].

Table 1. Malware attacks on network level

S.No	Attack name	Description
1	Transmission control protocol (TCP) SYN	The three-way TCP Handshake is exploited. The attacker sends many TCP synchronized sequence number (SYN) packets to the victim with the spoofed address. The acknowledgment (ACK) packets will be sent to the spoofed address, making the victim wait indefinitely.
2	Push and ACK	TCP packets with the PUSH flag and ACK flag set are sent from the agents to the victim. These flags instruct the victim machine to unload all data in the incoming TCP buffer
3	Internet control message protocol (ICMP) flood	A large volume of ICMP ECHO REQUEST packets or packet internet or inter-network Groper (PING) Packets are sent by the agents to the victim, saturating the Bandwidth
4	User datagram protocol (UDP) flood	Several UDP packets are sent to random ports of the victim, exhausting the bandwidth.
5	Smurf Attack	A large ICMP REQUEST Echo messages are sent to an amplifying machine with destination addresses spoofed with the victim's internet protocol (IP) address. When the amplifier machine sends "Ping" messages in the network, all the devices in the network send end ICMP packets to the victim device while shutting down the victim device.
6.	Fraggle attack	Similar to the Smurf attack, where UDP Echo packets are sent, creating an infinite attack loop
7	Domain name system (DNS) flood attack	A lot of spoofed DNS packets are sent. This attack is difficult to identify as spoofed packets look similar to legitimate packets.
8	HTTP flood attack	This is like a replay attack where spoofed HTTP requests are sent to the victim.

Table 1 shows malware has categorized the DDOS attacks into eight categories based on the architecture. They are the agent-handler, reflector, internet relay chat (IRC)-based, web-based, and peer-to-peer (P2P)-based models. At the protocol level, DDOS attacks are categorized into host-based attacks where the device's firmware is physically hampered with robot (BOT) software, network-based, and application-based attacks. They also give a brief description of malware attacks at the network level. In this work, an efficient NIDSE is designed with the help of XGBoosting to classify attacks in smart home devices.

The rest of the paper follows: section 2 discusses the typical steps in NID systems and the proposed system using the XGBoosting classifier. Section 3 discusses the classification performances of the proposed NIDSE on the IoT network intrusion (IoT-NI) dataset, which has nine categories of network attacks, including normal data. The final section concludes the proposed NIDSE for smart home devices.

2. METHODS AND MATERIALS

The IDSs are classified into host-based IDS (HIDS) and network-based intrusion detection systems (NIDS). HIDS is the monitoring activities performed on host/endpoint devices. This typically included routine system checks, call checks, and file system scans. A device driver software or agent program can perform the checks. NIDS monitors the network device using the NIDS sensor installed in the device, which can be a router, gateway, or a dedicated device for passively copying and scanning network traffic.

The typical NIDS comprises four steps: a data source, data preprocessing, a decision-making method, and a defense response [24]. In this work, the input data is obtained from smart home devices such as smart cameras, smart lights, smart assistance devices, and wireless modems. The obtained information is cleaned in the preprocessing stages by removing the duplicates, replacing the missing values, and removing noisy details. In the next module, ML is widely used for anomaly detection to automate the process and deal with the huge number of data that must be processed without writing specific code separately. NIDS should be trained with ML algorithms to build a model to classify DDOS attacks at the edge device. The power of ML tools lies in detecting and analyzing network attacks without having to describe them as previously defined accurately. ML can aid in solving the most common tasks, including regression, prediction, and classification in the era of extremely large amounts of data. This study uses XGBoosting as an ML approach for anomaly detection in smart home devices.

2.1. XGBoosting classifier

XGBoosting is an effective implementation of a stochastic gradient boosting algorithm. It can even handle class imbalance problems by fine-tuning the parameters to pay more attention to minority classes in a skewed distribution. It involves three elements: optimization of the loss function, predictions by a weak learner, and minimizing the loss function by an additive model with weak learners. The optimization function should be chosen based on the type of problem being solved. However, any generic and differential loss function can be used in the boosting framework. Figure 2 shows the XGBoosting procedure.

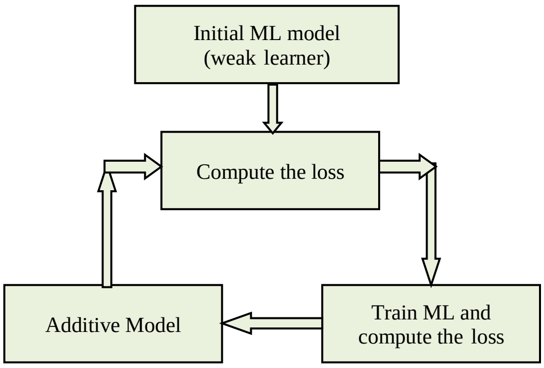

Figure 2. Procedure of XGBoosting

In gradient boosting, DT is used as the weak learner. The best splits are chosen based on the purity scores, and the trees are constructed greedily. To ensure the learners remain weak, large trees are generally constructed with many levels. A gradient descent procedure is employed when adding a tree to minimize the loss in the additive model. All the string values are converted to numerical and normalized before training. A soft probability function is employed as the NIDSE is designed for multi-classification. It is similar to the softmax function defined in (1).

$$soft_{prob(z_i)} = \frac{e^{z_i}}{\sum_{i=1}^{k} e^{z_i}} \tag{1}$$

where z_i is the output from the i^{th} class, and K is the total number of classes. It is a standard exponential function, and the class with high probability is the predicted class once the loss function is minimized. The cross-entropy loss is defined in (2). Where z_i is the true class label. The resulting predictions from each tree have less correlation as they are learned differently.

$$Cross\ Entry\ Loss = \sum_{j=1}^{k} e^{z_i} \log\left(soft\_prob(z_i)\right) \tag{2}$$

3. RESULT AND DISCUSSION

The proposed system is evaluated using the IoT-NI database. It has various network attacks, including host discovery, synchronized sequence number (SYN), acknowledgment (ACK), and hypertext transfer protocol (HTTP) flooding. The IoT-NI database consists of 42 packet files collected from different time points. All are captured directly from the wireless network adaptor under monitor mode, and the headers are removed using Air cracking. The description of network attacks in the IoT-NI database is shown in Table 2.

Table 2. Descriptions of network attacks in IoT-NI database

Category	Sub-Category
Scanning	Host discovery, Port scanning, operating system (OS)/version detection
Mirai botnet	Host discovery, Telnet Brute force, HTTP Flooding, UDP flooding, ACK flooding
Denial of service (DoS)	SYN Flooding
Man in the middle (MITM)	Address resolution protocol (ARP) spoofing
Normal	Normal

Table 2 contains the category and sub-category of attacks in IoT. Scanning, Mirai botnet, Denial of service, Man in the middle, and normal are categories of attacks [25]. Since the proposed model is a passive NIDS, this benchmark dataset is suitable for analyzing the performance of the proposed system. Mirai botnet attacks are simulated and injected from a laptop disguised as if packets originated from the IoT devices. All other attack types are simulated using Nmap. The proposed system is designed to classify nine attacks in the IoT-NI database. The number of instances available in the databases is shown in Table 3.

Table 3. Number of instances or packets available in IoT-NI database

Attacks	#instances
Normal (A1)	137396
Mirai-UDP Flooding (A2)	949284
Mirai- Brute force (A3)	1924
Dos-SYN Flooding (A4)	64646
Mirai-HTTP Flooding (A5)	10464
Mirai-ACK Flooding(A6)	75632
Scan Port –OS (A7)	1817
MITM-ARP Spoofing (A8)	101885
Scan Hot Port (A9)	20939

Table 3 has nine attacks: normal, Mirai-UDP flooding, Mirai-Brute force, Dos-SYN flooding, Mirai-HTTP flooding, Mirai-ACK flooding, scan port–operating system, MITM-ARP spoofing, and scan hot port. Every attack has different instances available in the IoT-NI database [26]. The attack classes and their distribution in the dataset are shown in Figure 3.

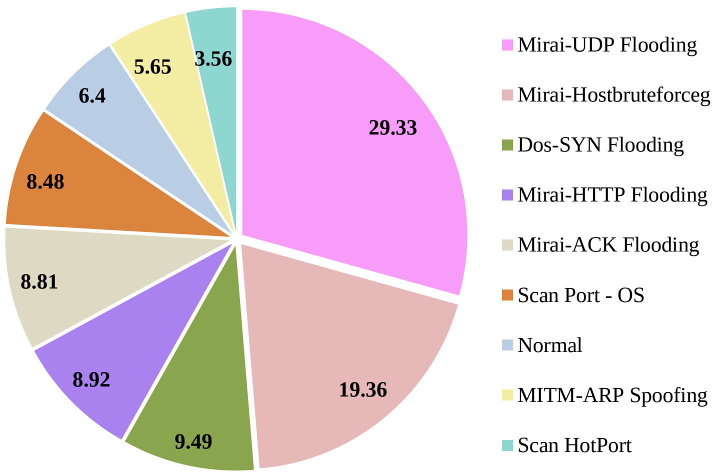

Figure 3. Distributions of network attacks in IoT-NI database

Figure 3 explains the distributions of network attacks in a pie chart, with nine attacks. Every attack mentions different colors for variation from others. From this figure, the Mirai-UDP Flooding attack has 29.33%, and the Mirai the brute force attack contains 19.36%. In addition, the scan HotPort contains 3.56%. The performance of the system is analyzed in terms of precision, recall, F1-score, precision micro average, and precision macro average. The performance metrics are described in Table 4.

Table 4. Descriptions of the performance measures

Measure	Descriptions
Precision (P)	$\frac{TP}{TP+FP}$
Recall (R)	$\frac{TP}{TP+FN}$
F1-Score	$\frac{2*P*R}{P+R}$
Precision micro average	$\sum_{i=1}^{n}\frac{TP_i}{TP_i+FP_i}$
Precision macro average	$\frac{\sum_{i=1}^{n}P_i}{n}$

From Table 4, *n* represents the number of attacks (classes). The number of correct predictions of a particular attack is termed TP, and the misclassification of that attack is termed *FN*. Also, the misclassification of other attacks is termed *FP*. To evaluate the success rate of NIDSE, k-fold cross-validation is used [27]. The dataset is divided into k-folds with an equal number of instances per attack from A1 to A9 in each fold. Then, 1st fold is tested by the NIDSE while the remaining folds are used to fit the NIDSE. This process is repeated for each fold until the kth fold reaches. Finally, the outputs from each fold are combined to evaluate the system's success rate, which is shown in Figure 4. The performances of the proposed system using RF and balanced RF classifiers are shown in Figures 5 and 6. Figure 7 shows the system's performance using the XGBoosting algorithm [28].

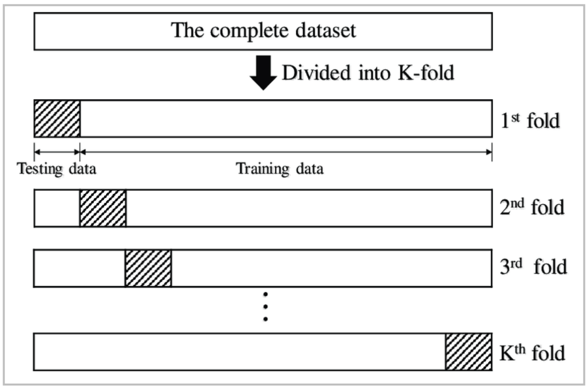

Figure 4. k-fold cross-validation to evaluate the success rate of a system

	precision	recall	f1-score	support
Mirai-Ackflooding	1.00	1.00	1.00	15107
DoS-Synflooding	0.21	0.98	0.35	8707
Scan Port OS	0.29	0.99	0.45	13656
Mirai-Hostbruteforceg	0.30	0.99	0.46	13994
Mirai-UDP Flooding	0.48	0.97	0.64	30268
Mirai-HTTP Flooding	0.61	0.94	0.74	45766
Normal	0.43	0.98	0.60	9937
Scan Hostport	0.14	0.97	0.25	5615
MITM ARP Spoofing	0.34	0.97	0.50	13396
micro avg	0.40	0.97	0.57	156446
macro avg	0.42	0.98	0.55	156446
weighted avg	0.49	0.97	0.63	156446
samples avg	0.59	0.97	0.67	156446

Figure 5. Performance of the proposed system by RF classifier

	precision	recall	f1-score	support
Mirai-Ackflooding	1.00	1.00	1.00	15107
DoS-Synflooding	0.67	0.43	0.53	8707
Scan Port OS	0.09	0.02	0.03	13656
Mirai-Hostbruteforceg	0.14	0.03	0.05	13994
Mirai-UDP Flooding	0.80	0.52	0.63	30268
Mirai-HTTP Flooding	0.92	0.71	0.80	45766
Normal	0.97	0.86	0.91	9937
Scan Hostport	0.50	0.07	0.13	5615
MITM ARP Spoofing	0.48	0.17	0.25	13396
micro avg	0.82	0.50	0.62	156446
macro avg	0.62	0.42	0.48	156446
weighted avg	0.70	0.50	0.57	156446
samples avg	0.50	0.50	0.50	156446

Figure 6. Performance of the proposed system by balanced RF classifier

XGBoost with OVR classifier

	precision	recall	f1-score	support
Mirai-Ackflooding	1.00	1.00	1.00	15107
DoS-Synflooding	0.78	0.37	0.50	8707
Scan Port OS	0.75	0.01	0.02	13656
rai-Hostbruteforceg	0.87	0.03	0.05	13994
Mirai-UDP Flooding	0.84	0.46	0.60	30268
Mirai-HTTP Flooding	1.00	0.70	0.83	45766
Normal	0.99	0.86	0.92	9937
Scan Hostport	0.80	0.05	0.10	5615
MITM ARP Spoofing	0.62	0.08	0.15	13396
micro avg	0.94	0.48	0.64	156446
macro avg	0.85	0.40	0.46	156446
weighted avg	0.88	0.48	0.56	156446
samples avg	0.48	0.48	0.48	156446

Figure 7. Performance of the proposed system by XGBoosting

Figure 4 explains that the complete data set is divided into k-fold cross-validation. Here, the header describes the testing data and the payload explains the training data. It can be seen from Figures 5 to 7 that the proposed NIDSE gives 94% micro average precision of 94% and macro average precision of 85%, which is higher than the RF and balanced RF classifier. Also, the performance of balanced RF is better than that of a conventional RF classifier as it balances at each bootstrap by the random under-samples. The RF classifier classifies the normal packets with micro and macro precision of less than 50%, whereas the balanced RF provides 82% micro and 62% macro precision. The maximum weighted average accuracy of the NIDSE with XGBoosting is 88%. Figure 8 explains the detection ratio of SVM, DT, RF, and NIDSE based on attacker count. From Figure 8, the proposed system NIDSE has a higher detection ratio than SVM, DT, and RF algorithms. The NIDSE system uses the XGBoosting algorithm to improve the attack detection ratio. Furthermore, NIDSE-based attacker detection improves accuracy, precision, and recall.

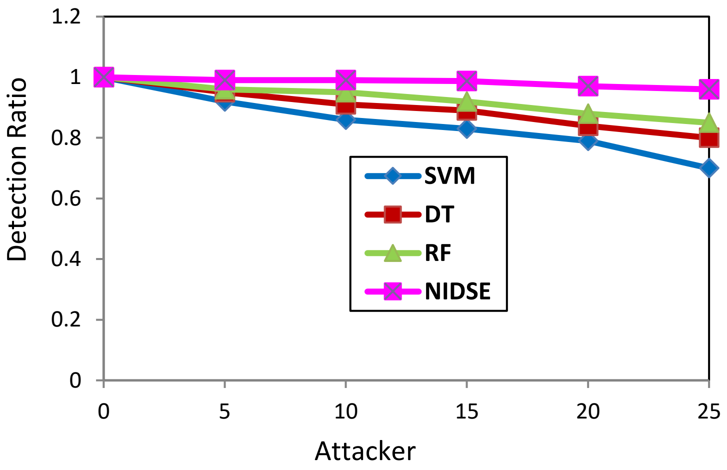

Figure 8. Detection ratio of SVM, DT, RF, and NIDSE against DoS attacker

4. CONCLUSION

The proposed NIDSE focuses on anomaly detection in smart home device data using ML algorithms that provide end-point security for smart home users. It uses an ensemble algorithm to detect the anomalies from nine attacks on smart home devices. One of the dominating algorithms, XGBoosting, is applied in the prediction model. It is designed using gradient-boosted DT. The performance of NIDSE is tested on the IoT-NI database using 10-fold cross-validation. Different ML techniques are used to analyze the traffic data to predict the multiple DDOS Attacks in Smart Home systems. Results show that the XGBoosting algorithm can predict the different attacks of multiple classes with up to 94% accuracy. Compared to the SVM, DT, and RF classifiers, the proposed system improved the network performance. Endpoint security is provided for all smart home devices as the proposed system is executed on edge device software. This could greatly reduce latency and bandwidth issues. However, this mechanism can't detect clone attacks in the network.

REFERENCES

[1] N. Farnaaz and M. A. Jabbar, “Random forest modeling for network intrusion detection system,” *Procedia Computer Science*, vol. 89, pp. 213–217, 2016, doi: 10.1016/j.procs.2016.06.047.

[2] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: techniques, datasets and challenges,” *Cybersecurity*, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[3] L. Ashiku and C. Dagli, “Network intrusion detection system using deep learning,” *Procedia Computer Science*, vol. 185, pp. 239–247, 2021, doi: 10.1016/j.procs.2021.05.025.

[4] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, “Network intrusion detection system: A systematic study of machine learning and deep learning approaches,” *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.

[5] R. A. Disha and S. Waheed, “Performance analysis of machine learning models for intrusion detection system using Gini impurity-based weighted random forest (GIWRF) feature selection technique,” *Cybersecurity*, vol. 5, no. 1, Dec. 2022, doi: 10.1186/s42400-021-00103-8.

[6] C. F. T. Pontes, M. M. C. de Souza, J. J. C. Gondim, M. Bishop, and M. A. Marotta, “A new method for flow-based network intrusion detection using the inverse potts model,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1125–1136, Jun. 2021, doi: 10.1109/TNSM.2021.3075503.

[7] Z. Wang, Y. Zeng, Y. Liu, and D. Li, “Deep belief network integrating improved Kernel-based extreme learning machine for network intrusion detection,” *IEEE Access*, vol. 9, pp. 16062–16091, 2021, doi: 10.1109/ACCESS.2021.3051074.

[8] T. Kim and W. Pak, “Hybrid classification for high-speed and high-accuracy network intrusion detection system,” *IEEE Access*, vol. 9, pp. 83806–83817, 2021, doi: 10.1109/ACCESS.2021.3087201.

[9] T. Wisanwanichthan and M. Thammawichai, “A double-layered hybrid approach for network intrusion detection system using combined naive Bayes and SVM,” *IEEE Access*, vol. 9, pp. 138432–138450, 2021, doi: 10.1109/ACCESS.2021.3118573.

[10] S. M. Kasongo, “An advanced intrusion detection system for IIoT based on GA and Tree based algorithms,” *IEEE Access*, vol. 9, pp. 113199–113212, 2021, doi: 10.1109/ACCESS.2021.3104113.

[11] I. Syarif, R. F. Afandi, and F. Astika Saputra, “Feature selection algorithm for intrusion detection using Cuckoo search algorithm,” in *2020 International Electronics Symposium (IES)*, Sep. 2020, pp. 430–435, doi: 10.1109/IES50839.2020.9231840.

[12] A. S. Mahboob and M. R. O. Moghaddam, “An anomaly-based intrusion detection system using butterfly optimization algorithm,” in *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, Dec. 2020, pp. 1–6, doi: 10.1109/ICSPIS51611.2020.9349537.

[13] L. Guo, “Research on anomaly detection in massive multimedia data transmission network based on improved PSO algorithm,” *IEEE Access*, vol. 8, pp. 95368–95377, 2020, doi: 10.1109/ACCESS.2020.2994578.

[14] N. Sampath, M. A. Jerlin, K. L. B, and A. A, “Intrusion detection in software defined networking using genetic algorithm,” in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Feb. 2020, pp. 1–5, doi: 10.1109/ic-ETITE47903.2020.464.

[15] K. Alrawashdeh and S. Goldsmith, “Optimizing deep learning based intrusion detection systems defense against white-box and backdoor adversarial attacks through a genetic algorithm,” in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, Oct. 2020, pp. 1–8, doi: 10.1109/AIPR50011.2020.9425293.

[16] R. Raman, V. Sujatha, C. Bhupeshbhai Thacker, K. Bikram, M. B Sahaai, and S. Murugan, “Intelligent parking management systems using IoT and machine learning techniques for real-time space availability estimation,” in *2023 International Conference on Sustainable Communication Networks and Application (ICSCNA)*, Nov. 2023, pp. 286–291, doi: 10.1109/ICSCNA58489.2023.10370636.

[17] K. Karthika, S. Dhanalakshmi, S. M. Murthy, N. Mishra, S. Sasikala, and S. Murugan, “Raspberry Pi-enabled wearable sensors for personal health tracking and analysis,” in *2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)*, Oct. 2023, pp. 1254–1259, doi: 10.1109/ICSSAS57918.2023.10331909.

[18] A. Hussaindeen, S. Iqbal, and T. D. Ambegoda, “Multi-label prototype based interpretable machine learning for melanoma detection,” *International Journal of Advances in Signal and Image Sciences*, vol. 8, no. 1, pp. 40–53, Jan. 2022, doi: 10.29284/IJASIS.8.1.2022.40-53.

[19] M. Schiefer, “Smart home definition and security Threats,” in *2015 Ninth International Conference on IT Security Incident Management & IT Forensics*, May 2015, pp. 114–118, doi: 10.1109/IMF.2015.17.

[20] W. Campbell, “Security of internet protocol cameras – a case example,” *SRI Security Research Institute, Edith Cowan University, Perth, Western Australia*, 2013.

[21] S. Notra, M. Siddiqi, H. Habibi Gharakheili, V. Sivaraman, and R. Boreli, “An experimental study of security and privacy risks with emerging household appliances,” in *2014 IEEE Conference on Communications and Network Security*, Oct. 2014, pp. 79–84, doi: 10.1109/CNS.2014.6997469.

[22] K. N. Choi *et al.*, “Poster abstract: passive activity classification of smart homes through wireless packet sniffing,” in *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2020, pp. 347–348, doi: 10.1109/IPSN48710.2020.00-13.

[23] T. Meenakshi, R. Ramani, A. Karthikeyan, N. S. Vanitha, and S. Murugan, “Power quality monitoring of a photovoltaic system through IoT,” in *2023 International Conference on Sustainable Communication Networks and Application (ICSCNA)*, Nov. 2023, pp. 413–418, doi: 10.1109/ICSCNA58489.2023.10370494.

[24] N. Moustafa, B. Turnbull, and K.-K. R. Choo, “An Ensemble Intrusion Detection Technique Based on Proposed Statistical Flow Features for Protecting Network Traffic of Internet of Things,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, Jun. 2019, doi: 10.1109/JIOT.2018.2871719.

[25] A. A. Alahmadi *et al.*, “DDoS attack detection in IoT-based networks using machine learning models: a survey and research directions,” *Electronics*, vol. 12, no. 14, p. 3103, Jul. 2023, doi: 10.3390/electronics12143103.

[26] M. De Donno, N. Dragoni, A. Giaretta, and A. Spognardi, “DDoS-capable IoT malwares: comparative analysis and Mirai investigation,” *Security and Communication Networks*, vol. 2018, pp. 1–30, 2018, doi: 10.1155/2018/7178164.

[27] R. S. Kadurka and H. Kanakalla, “Automated bird detection in audio recordings by a signal processing perspective,” *International Journal of Advances in Signal and Image Sciences*, vol. 7, no. 2, pp. 11–20, Dec. 2021, doi: 10.29284/IJASIS.7.2.2021.11-20.

[28] B. Zhou, Z. Li, S. Zhang, X. Zhang, X. Liu, and Q. Ma, “Analysis of factors affecting hit-and-run and non-hit-and-run in vehicle-bicycle crashes: a non-parametric approach incorporating data imbalance treatment,” *Sustainability*, vol. 11, no. 5, Art. no. 1327, Mar. 2019, doi: 10.3390/su11051327.

BIOGRAPHIES OF AUTHORS

Malothu Amru working as a professor in the electronics and communication engineering in CMR Engineering College, Hyderabad. He has over 17 years of teaching and research experience. His research areas include wireless sensor networks, and ad hoc sensor networks. He has guided 10 MTech dissertations and 20 B.Tech projects. He has published his research findings in 17 journals and conferences both in national and international level. He is lifetime member of IAENG. He has been guiding student community to develop excellent path-breaking projects that are useful to mankind. Email: malothuamru@gmail.com.

Raju Jagadeesh Kannan is a senior IEEE member, working as aSr. professor in Department of Computer Science and Engineering and Dean, Engineering and Technology at SRM Institute of Science and Technology Tiruchirappalli. He received Bachelor of Engineering in instrumentation and control engineering from Madurai Kamaraj University, Madurai, Tamilnadu, India. He secured Master of Engineering in computer science and engineering at Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India. He was awarded Ph.D. in the field of computer science and engineering at Anna University, Chennai, Tamil Nadu India. He carries both industry and academic experience for more than 20 years. He has presented 170 papers in National and International Journals, Conference and Symposiums. He chaired session track in conferences of national and international repute and served as reviewer for peer-reviewed journals. His major area of interest includes cyber physical systems, computational intelligence, and imaging and computer vision. He can be contacted at email jagadeeshkannan.r@vit.ac.in.

Enthrakandi Narasimhan Ganesh having 28 years of Teaching Experience out of which 8 years as Principal and 6 years as Dean at present. He is a M.Tech Graduate from IIT Madras in microelectronics and VLSI design, and PhD from JNTU Hyderabad in Nanotechnology with Gold Medal. He has Executed 10 Funded Projects with 5 patents granted. DST STTP Sponsored Funded one Crore project is in progress and about to be submitted. He Guiding 13 PhD students out of which 8 were awarded degree. He having 60 Scopus indexed International Journals and 22 SCI Journals (Annexure I) with total 320 Publications to my credits. He has executed 12 Consultancy Projects for the cost of 60 Lakhs. He has more than 50 International Conference Publications. Twenty best Conference paper awards with Distinguished Faculty, Researcher and Excellence in teaching awards, reviewer and editor for 8 International Journals with six books Published. He has Academic, Research and administrative experience. Also, three times NAAC and NBA Committee External member (accessor) and also interview panel member for DRDO and ISRO. He can be contacted at email: enganesh50@gmail.com.

Surulivelu Muthumarilakshmi is an associate professor in computer science and engineering at S.A. Engineering College. With over 13 years of teaching experience, my primary focus revolves around computer networks. He particularly interested in investigating network protocols, security measures, and ways to optimize network performance. Her passion lies in researching and publishing articles that delve into these areas, aiming to enhance our understanding of robust network systems and contribute valuable insights to the academic community. She can be contacted at email: smlakshmi74@gmail.com.

Kuppan Padmanaban joined K L University in 2019 upon completing his Ph.D. in Computer Science and Engineering. Currently, he serves as an Associate Professor in the Department of Computer Science and Engineering (Honors) at the School of Computing, Koneru Lakshmaiah Education Foundation, K L University, Andhra Pradesh, India. Specializing in full-stack development, he possesses expertise in various technologies, including MERN, Python, Spring Micro services and the .NET framework. He holds bachelor's and master's degrees in computer science and engineering from Anna University in Tamil Nadu, India. His research focuses on wireless sensor networks, IoT, machine learning, and data analytics. He has published numerous research articles in reputable journals and conferences. He can be contacted at email: padmanaban.k@yahoo.com.

Jeyaprakash Jeyapriya is a ISTE member, working as a ML Engineer, Apcomart Private Limited, Bangalore. She received Bachelor of Engineering in computer science and engineering from Anna University, Chennai, Tamilnadu, India. She secured Master of Engineering in computer science and engineering at Anna University, Chennai, Tamilnadu, India. She was awarded Ph.D. in the field of Computer Science and Engineering at VIT University, Vellore, Tamilnadu India. She carries academic experience for more than 8 years. She has presented 10 papers in National and International Journals, Conference and Symposiums. Her major area of interest includes computational intelligence, and imaging and computer vision. She can be contact at email: priyacseme@gmail.com.

Subbiah Murugan is an adjunct professor, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu, India. He published his research articles in many international and national conferences and journals. His research areas include network security and machine learning. He can be contacted at smuresjur@gmail.com.