Enhanced Intrusion Detection in IoT with a Novel PRBF Kernel and Cloud Integration

Bhargavi Mopuru
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, India
bhargaviphd83@gmail.com (corresponding author)

Yellamma Pachipala
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, India
pachipala.yamuna@gmail.com

Received: 7 May 2024 | Revised: 18 May 2024 | Accepted: 19 May 2024

ABSTRACT

The proliferation of Internet of Things (IoT) devices in various sectors has increased the need for robust security solutions capable of addressing complex network vulnerabilities and sophisticated cyber threats. This study introduces a novel architecture that integrates cloud computing with advanced machine learning techniques to provide efficient and scalable security in IoT systems. A unique Polynomial Radial Basis Function (PRBF) kernel is proposed to enhance the classification accuracy of Support Vector Machine (SVM) beyond traditional Gaussian and polynomial kernels. This study compares the proposed PRBF-SVM with Logistic Regression, SVM, and XGBoost models, optimized through rigorous hyperparameter tuning, to demonstrate significant improvements in detection rates. Furthermore, the integration of cloud services facilitates the offloading of computationally intensive tasks, ensuring scalability and real-time response capabilities. The results highlight the superior performance of the proposed model in accuracy, efficiency, and computation time, making a compelling case for its application in safeguarding IoT environments against evolving threats.

Keywords-cloud assisted IoT security as a service; Intrusion Detection System (IDS); machine learning; SVMs; kernel functions

I. INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices has transformed various sectors, from smart homes and healthcare to critical infrastructure and industrial automation. IoT devices enable unprecedented connectivity and data exchange, facilitating smart and efficient operations. However, this growing network of interconnected devices also presents significant security challenges. As malicious entities exploit inherent vulnerabilities in IoT devices and networks, there is a pressing need for robust security mechanisms. Traditional security measures, such as firewalls and signature-based Intrusion Detection Systems (IDS), are often insufficient due to the unique architecture and resource constraints of IoT devices. [1] These constraints make IoT systems susceptible to various attack vectors, including Distributed Denial of Service (DDoS), Man-in-the-Middle (MitM), and malicious firmware updates. Additionally, the dynamic nature of IoT networks requires real-time adaptability to emerging threats [2].

Machine Learning (ML)-based IDS offers a promising alternative, providing real-time detection by analyzing complex traffic patterns, characteristic of IoT environments. Unlike static security mechanisms, [3] ML models can learn and adapt to new attack patterns over time, significantly improving detection accuracy and reducing false positives. However, several challenges persist:

- Diverse Attack Vectors: IoT networks face numerous types of attacks, ranging from basic DoS to sophisticated zero-day exploits. Detecting this variety requires models that can generalize across different data types and patterns.
- Resource Constraints: Many IoT devices have limited computational capabilities and power resources, necessitating lightweight security solutions.
- Scalability: As IoT networks scale to thousands or millions of devices, IDSs must maintain high throughput and performance.

This study investigates a comprehensive cloud-integrated ML framework for intrusion detection, using Logistic Regression (LR), Support Vector Machines (SVMs), and XGBoost models to analyze traffic patterns and detect anomalies at the device and network levels [4]. This study introduces a novel Polynomial Radial Basis Function (PRBF) kernel for SVMs, to improve classification accuracy compared to traditional Gaussian or polynomial kernels. The proposed framework leverages scalable cloud resources to handle the computationally intensive tasks of training and maintaining the ML models. Key contributions of this work include:

- Novel kernel design: Development and implementation of a PRBF kernel to enhance SVM classification accuracy.
- Hyperparameter tuning: Comprehensive hyperparameter tuning to ensure optimal model performance.
- Cloud integration: A cloud-based architecture enables scalable training and real-time monitoring, offloading computationally intensive tasks from resource-constrained IoT devices.
- Benchmarking against traditional models: The comparative analysis of the proposed framework shows a significant improvement in detection rates.

By addressing the unique challenges of IoT security through innovative ML techniques and cloud integration [5], this paper contributes to the development of a future-proof IDS solution. The proposed framework can significantly enhance the security posture of IoT networks, providing organizations with a scalable and accurate defense against evolving threats.

II. LITERATURE REVIEW

The primary objective is to develop a robust classification model capable of categorizing data in the network and fitting it into an IDS. In [6], the OCTAVE Allegro technique was used to assess the security hazards of smart homes. The OCTAVE Allegro approach focuses on information assets and examines a variety of different types of information containers, including databases, physical items, and people. Authors in [7] aimed to identify the numerous security flaws in IoT-based smart homes, explain the associated risks to homeowners, and offer solutions for mitigating the discovered hazards. The results of this study can be utilized to enhance the security of smart homes powered by the IoT. According to [8], users should be able to have confidence in complicated interconnected items, since their user-centric approach gives both scalability and interoperability. As an example, ARCA-IoT offers a user-centric model that is resilient enough to survive attacks from dishonest actors who want to influence the trustworthiness of the system. ARCA-IoT develops a cloud-based platform for scalability and interoperability. ARCA-IoT is trained using an intuitive Naive Bayes approach that predicts the probability of entities being trustworthy and then recognizes different types of attacks using three proposed algorithms. Simulation findings showed that ARCA-IoT was efficient in terms of performance parameters such as accuracy, sensitivity, specificity, and precision and outperformed previous similar systems in terms of qualitative analysis, as measured by a variety of parametric metrics, including interoperability, scalability, context responsiveness, and humanlike decision-making.

In [9], it was discussed how Artificial Intelligence (AI) will enable many smarter IoT devices that are not just data collectors and forwarders in the future but will also have built-in data wrangling and analysis capabilities to make autonomous field decisions using lightweight ML algorithms. In [10], the Identity-Based Encryption with an Approved Equivalence Test (IBE-AET) system was proposed, that could be used to simultaneously encrypt and search for outsourced information in cloud-assisted IoT. IBE-AET enables an authorized cloud server to check if two encrypted communications with the same identity and messages encrypted with different identities are the same. Additionally, the IBE-AET authorization technique is flexible, allowing a user to fine-grain test capabilities on the cloud server [11]. The IBE-AET was clearly shown to be comparable to the random oracle model's Bilinear Diffie–Hellman (BDH) assumption. Both theoretic analysis and experimental simulation demonstrated the effectiveness of the proposed scheme.

In [12], four techniques for reducing ensemble complexity by adaptive autoencoder deactivation were compared. These systems varied in how to choose which autoencoders to disable (criterion-based or random) and how to disable them (post-training or pre-training). Extensive tests on two real-world IoT intrusion detection datasets showed that the proposed techniques were capable of achieving acceptable detection performance at reduced training, retraining, and inference costs. In [13], a multiclass hybrid ensemble-based SVM classification framework was proposed to optimize accuracy. These approaches can enable the implementation of scalable and efficient IDSs and services on-device or at the edge.

A. Logistic Regression (LR) Model

The coefficients of the LR algorithm are calculated using the training data and a maximum likelihood estimate. Although maximum likelihood estimation includes assumptions about the distribution of data, it is a popular learning approach used by a variety of ML algorithms [14]. The concept underlying the maximum-likelihood LR is that a search strategy looks for coefficients (beta values) that minimize the gap between the model's prediction and the observed probabilities [15].

B. XGBoost Model

XGBoost uses gradient-boosted decision trees, which enable the boosting of machines or the application of boosting to machines, and has been adopted in many studies. After applying decision-tree techniques to a known dataset, XGBoost classifies the data. Model parameters must be determined based on the data. Typically, q denotes the parameters, which may be many, depending on the dataset. In the case of regression, classification, or ranking, the prediction value s_i helps to classify the present scenario. The basic objective is to extract relevant parameters from the training dataset. Initially, an objective function is created to describe the model's performance. It should be noted that each model might change, depending on whatever parameter is used. Depending on the parameters chosen, multiple models might be created on the same dataset. The objective function is divided into two components: training loss and regularization.

$$obj(\theta) = TL(\theta) + R(\theta) \qquad (1)$$

where R is the regularization term, and TL denotes the training loss. The model's ability for prediction is measured by the TL.

III. THE PROPOSED FRAMEWORK

The proposed IoT, adaptive with cloud integration and IDS consists of two layers: (i) the cloud layer and (ii) the IoT device layer. The cloud layer is responsible for managing ML-based IDS models and the device layer is combined into adaptable groups that all run the same IDS model.

A. Cloud Service Layer

To build and train models, the Cloud Model as a Service (MaaS) uses Software as a Service (SaaS) [16-17]. The service collects all potential attacks from nodes or devices and performs the classification process [18]. Because thousands of edge devices connected to the IoT are designed and function differently, a significant amount of cloud computing resources may be required to handle the massive amount of data generated by such a service. There are two options for assigning the cloud MaaS role: to a single cloud node or to a group of cloud nodes. This is determined by the node workload for the allocated device sets and the number of devices in each set.

B. IoT Device Layer

This layer categorizes edge devices into sets, each of them performing the same device function and running on the same operating system. If an anomaly or attack is detected, the edge device informs the service of the updated parameters of the proposed model from the cloud service. Due to the random introduction of noise into the features and the significantly reduced accuracy of the outer bag, the significance value of the expression may be used as a measurement. This indicates that this feature has a significant impact on the sample classification outcomes, in other words, it is more relevant. The purpose of feature selection is to increase model prediction accuracy and develop a faster, less energy-consuming, and more explanatory model. This technique identifies the characteristic factors that are strongly associated with the target variable and chooses the characteristic factors that have fewer associations.

C. ML-based IDS (ML-IDS)

Figure 1 shows the proposed ML-IDS, which was built using Python. The XGBoost package was downloaded, along with the Anaconda software, which comes pre-installed with other packages. The LR, XGBoost, and PRBF-SVM models were executed, and the confusion matrix, accuracy, precision, recall, and F1-score were extracted from the results.

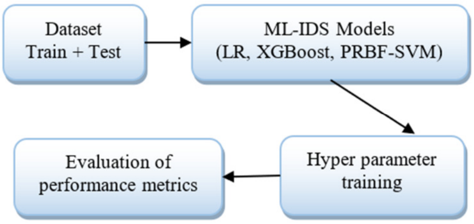

Fig. 1. The ML-based IDS (ML-IDS).

D. Dataset

The BoT-IoT dataset was developed in the UNSW Canberra Cyber Range Lab to create a realistic network environment. Regular and botnet traffic coexisted in the network environment. To aid in labeling, the data were divided into attack types and subgroups. The pcap files total 69.3 GB and include around 72,000 records. In CSV format, the extracted flow traffic is 16.7 GB in size. The dataset includes different DoS, service-scan, key-logging, and data exfiltration attacks, further classified according to the protocol used. To simplify the dataset management, MySQL was used to extract 5% of the original dataset. The extracted data were made up of four files with a total size of roughly 1.07 GB and nearly 3 million records.

E. Selection of ML Model

The need for incremental model learning necessitates the use of supporting ML models. As a result, XGBoost, LR, and Lib-SVM were used.

F. The Proposed Polynomial Radial Basis Function-Based Support Vector Machine (PRBF-SVM)

Having a suitable kernel feature, SVM is a flexible model that learns how to handle nonlinear data for regression and classification tasks. In SVM algorithms, the two main hyperparameters are *C* and γ. Hyperparameters must be adjusted before training the model. The general parameters were used to train the SVM based on the training data, and then the model was verified [18]. Finally, the model was tested on the reference dataset. One-versus-one (ovo), kernel-based polynomials, and RBF were employed for the SVM kernel and the decision function. When adjusting various parameters, such as *C*, *γ*, and kernel, the comparison analysis showed that SVM might not be the best choice for this challenge, as the results showed unsatisfactory precision.

After extensive research, three multiclass problem-solving systems were proposed: one-versus-residue (ovr), *M*(*M*-1)/2, and SVM formula expansion. The modified classifier is written as follows:

$$f_{tar}(x) = \sum_{k=1}^{M} \tau^k f_{src}^k(x) + \Delta f(x) \qquad (3)$$

where $\Delta f(x)$ is the perturbation function learned on a small set of labeled target-domain data in D_{tar}^l, and $\tau^k \in (0,1)$ is the base classifier weight of each $f_{src}^k(x)$. As shown in [19], it has the following form:

$$\Delta f(x) = w^T \varphi(x) = \sum_{i=1}^{N} a_i y_i K(x_i, x) \qquad (4)$$

where *i* is the target-domain case *i*th-labeled function coefficient and $w = \sum_{i=1}^{N} a_i y_i \varphi(x_i)$ is the model parameter under which the labeled example D_{tar}^l and a_i is to be assessed. The kernel function was set to $K = (.,.) \equiv \varphi(.)^T \varphi(.)$ due to non-linear feature mapping. The adapted classifier $f_{tar}(x)$ aims to reduce the classification error in the target-domain instances as well as the distance from the basic classifiers $f_{src}^k(x)$, resulting in a higher bias variance.

The weight controls $\{\tau^k\}_{k=1}^{M}$, based on the weight output of the limited range of target-domain instances in the basic classifiers $f_{src}^k(x)$, are automatically learned using enhanced multiclassifier adjusting. The adaptive classifier is renamed to:

$$y_i \sum_{k=1}^{M} \tau^k f_{src}^k(x) + y_i w^T \varphi(x_i) \geq 1 - \xi_i \qquad (5)$$

where (½)$(\tau)^T \tau$ of the total input of the base classifier steps are added to the regularized loss minimization process. This objective function aims to prevent dependency and overcomplexity $\Delta f(.)$on the basic category. The parameter B balances the two targets. Modifying the objective function as a Lagrange (primal) function minimization problem and setting its terms w, τ, and ξ to zero gives:

$$w = \sum_{i=1}^{N} a_i y_i \varphi(x_i)\tau^{\kappa} = \frac{1}{B}\sum_{i=1}^{N} a_i y_i f_{src}^{k}(x_i) \qquad (6)$$

where τ^k is a weighted sum and $y_i f_{src}^{k}(x_i)$ is the performance of the target domain which is classified. Consequently, well classifying the labeled destination domain info, allocates more massive base classifiers. The new decision function, using (3), (4), and (6), gives:

$$f(x) = \sum_{i=1}^{N} a_i y_i \left(K(x_i, x) + \frac{1}{B}\sum_{i=1}^{N} f_{src}^{k}(x_i) f_{src}^{k}(x) \right) (7)$$

Comparing this multiclassification adaptation model with a regular SVM model $f(x) = \sum_{i=1}^{N} a_i y_i K(x_i, x)$ can be interpreted as applying additional features to the projected labels of the basic classifiers in the target domain. According to this analysis, the scalar B combines the influence of the initial characteristics with more features. To perform nonlinear model operations, it is important to apply the polynomial mapping function.

$$K(l, l') = (\langle l, l' \rangle)^d \qquad (8)$$

$$K(l, l') = (\langle l, l' \rangle + 1)^d \qquad (9)$$

The Gaussian form of this function is:

$$K(l, l') = \exp(-\|l - l'\|^2/(2\sigma^2) \qquad (10)$$

and the Radial Basis can be written as:

$$K(l, l') = exp\big(-\|l - l'\|/(2\sigma)\big) \qquad (11)$$

In the presence of singularities, this solution is a piecewise linear result. Kernel functions are recommended when dealing with non-separable data. Input data are converted into a multidimensional space. As a result, an appropriate kernel must be selected to linearly separate non-separable data. Here, a novel kernel is proposed that can perform well with any dataset, especially those with a diversity of dimensions (datasets with many attributes). Except for high-dimensional datasets, the polynomial function works well with almost all datasets.

$$POLY = (1 + < l_1, l_2 >)^d \qquad (12)$$

where *d* is the degree of the polynomial. The same performance may be attained by using a GRBF, written as:

$$RBF = \exp(-sum(l_1, l_2)^2)\,/(PD) \qquad (13)$$

where P signifies the kernel and D is the input vector dimension. A new kernel function was formulated, the PRBF, which is more complicated and capable of handling high-dimensional datasets. This kernel combines Gauss and polynomial functions to improve performance across all datasets.

$$PRBF = 1 + \exp(-sum(l_1, l_2)^2)\,/(PD)^d \qquad (14)$$

The proposed PRBF kernel function exhibits a higher degree of convexity than the standard polynomial and Gaussian functions. Additionally, it is continuous and diminishes as l decreases. The limits of PRBF/RBF are obtained by:

$$T = \sum(l_{1i} - l_{2i}), \quad \text{and} \quad V = PD \qquad (15)$$

$$PRBF = \left(1 + \frac{e^{-T}}{V}\right)^d \quad \text{and} \quad RBF = \frac{e^{-T}}{V} \qquad (16)$$

As a result, the proposed function has a faster convergence rate compared to GRBF. Here, it is crucial to normalize the data, since the polynomial function diverges at wide intervals [-1, 1], and the proposed function has a faster convergence in this region.

IV. RESULTS

This study employed a personal desktop computer with a Core i5 CPU, 8 GB DDR4-RAM, and an Nvidia GT 1030 GPU. Python was used on Windows 10 to run the models. To make a single dataset that could be used for training and testing, datasets had to be checked for missing values before concatenation. Each row was binary labeled as normal or an anomaly in the 17th column. This feature was the target value to classify data and create the confusion matrix. The dataset was transformed to an all-numeric data type so that calculations could be completed quickly and accurately. The results were obtained by running the function on the combined (converted) dataset.

A. Accuracy

The evaluation employed the four most frequently used criteria for estimating accuracy.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \qquad (17)$$

$$Precision = \frac{TP}{TP+FP} \qquad (18)$$

$$Recall = \frac{TP}{TP+FN} \qquad (19)$$

$$F1\text{-}score = 2\frac{Precision \times Recall}{Precision + Recall} \qquad (20)$$

where the four essential values are True Negatives (TN), True Positives (TP), False Negatives (FN), and False Positives (FP).

B. Time Complexity

Time complexity is the time taken to complete a task. The time complexity of an algorithm is determined by how many basic operations it performs. The non-parametric classification algorithm PRBF-SVM has O(*n*) complexity, while XGBoost and LR have O(log*n* + *n*) and O(*n* log*n*), respectively. After the same preprocessing, all three models were run on the same dataset. The category column was the target column and had four different values: DoS, DDoS, Normal, and Reconnaissance. Figure 2 shows the features and their importance and Figure 3 shows the count versus target encoding.

SVM was used to choose the features, identifying ten key features for the analysis. The focus of the analysis was on operational factors, including CPU processing time without GPU involvement, and model size in memory. The results in Table I and Figure 4 show that the proposed PRBF-SVM model achieved better performance compared to the other two

ML models. Table II and Figure 5 show that the proposed PRBF-SVM model took less time compared to the two other ML approaches.

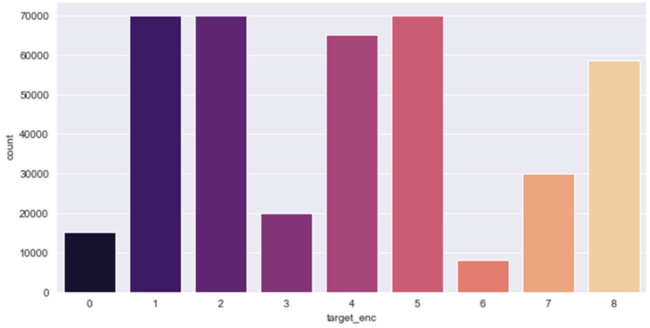

Fig. 2. Features and their importance.

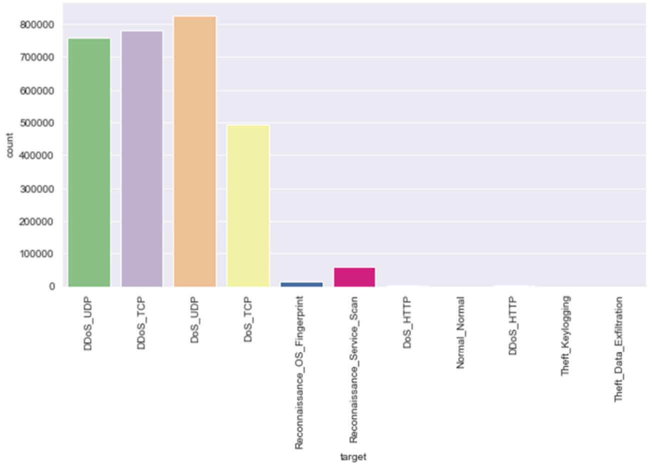

Fig. 3. Count versus target encoding.

TABLE I. PERFORMANCE METRICS COMPARISON OF THE THREE 3 ML MODELS

Model	Accuracy	Precision	Recall	F1-score
XGBoost	91.81	98.12	90.14	93.88
LR	97.24	98.15	98.76	97.54
PRBF-SVM	99.11	99.13	99.22	98.66

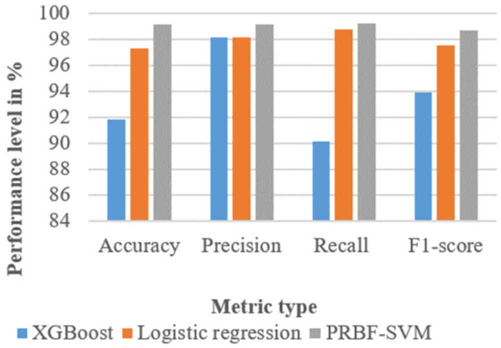

Fig. 4. Performance metrics comparison of the 3 ML models.

TABLE II. COMPUTATIONAL TIME

Model	Base Model	Hyper Parameter Tuning
XGBoost	33.3	403
LR	25.95	225
PRBF-SVM	23.62	118.5

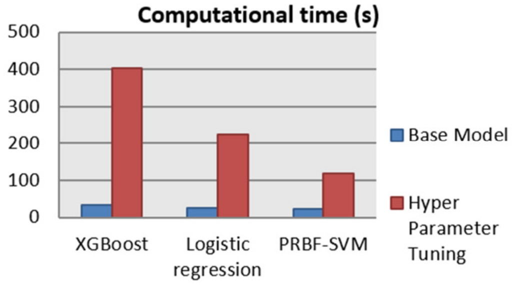

Fig. 5. Computation time of the 3 ML models.

V. CONCLUSION AND OUTLOOK

This study introduced a novel ML-based IDS designed specifically for resource-constrained IoT devices, leveraging cloud-assisted technology to enhance security measures. By integrating ML with cloud resources, this approach addresses significant gaps in existing security solutions, which often struggle with the unique challenges posed by the IoT ecosystem, such as limited device capabilities and complex attack vectors.

A. *Knowledge Gap and Novel Contribution*

Previous models did not fully exploit the synergy between cloud computing and ML for IoT security. This study bridges this gap by developing a PRBF kernel that significantly outperforms traditional kernels (Gaussian and polynomial) in diverse datasets. This kernel not only improves classification accuracy but also improves computational efficiency, making it highly suitable for the dynamic and expansive nature of IoT networks.

B. *Significance*

The significance of this work lies in its potential to drastically reduce the vulnerability of IoT devices to a wide range of security threats. By offloading intensive ML tasks to the cloud, the system not only optimizes performance in real-time but also ensures scalability and adaptability to evolving threats, thus fortifying the security landscape of IoT devices.

C. *Comparative Analysis*

Compared to traditional methods such as Gaussian SVM and LR, the proposed PRBF-SVM model demonstrated superior performance. For instance, the PRBF-SVM model achieved a classification accuracy of 99.11%, which is significantly higher than that of XGBoost (91.81%) and LR (97.24%). In addition, the computational efficiency of the proposed model, as evidenced by shorter training times, further underscores its practicality for real-world applications.

D. *Future Directions*

Looking ahead, the authors aim to explore the integration of other emerging ML techniques, such as deep learning, to further refine intrusion detection capabilities. In addition, extensive field tests are planned across various IoT platforms to validate and potentially enhance the robustness and reliability of the proposed PRBF-SVM model.

REFERENCES

[1] G. S. Mahmood, D. J. Huang, and B. A. Jaleel, "Achieving an Effective, Confidentiality and Integrity of Data in Cloud Computing," *International Journal of Network Security*, vol. 21, no. 2, pp. 326–332, Mar. 2019, https://doi.org/10.6633/IJNS.201903_21(2).17.

[2] O. Saeed and R. Shaikh, "A user-based trust model for cloud computing environment," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 337–346, 2018, https://doi.org/10.14569/IJACSA.2018.090347.

[3] F. Anindra, A. N. Hidayanto, and H. Prabowo, "Critical Components of Security Framework for Cloud Computing Community: A Systematic Literature Review," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 18, pp. 3345–3358, 2018.

[4] G. Thamilarasu and S. Chawla, "Towards Deep-Learning-Driven Intrusion Detection for the Internet of Things," *Sensors*, vol. 19, no. 9, Jan. 2019, Art. no. 1977, https://doi.org/10.3390/s19091977.

[5] S. Rajendran and R. Mary Lourde, "Security Threats of Embedded Systems in IoT Environment," in *Inventive Communication and Computational Technologies*, 2020, pp. 745–754, https://doi.org/10.1007/978-981-15-0146-3_70.

[6] B. Ali and A. I. Awad, "Cyber and Physical Security Vulnerability Assessment for IoT-Based Smart Homes," *Sensors*, vol. 18, no. 3, Mar. 2018, Art. no. 817, https://doi.org/10.3390/s18030817.

[7] A. Verma and V. Ranga, "Machine Learning Based Intrusion Detection Systems for IoT Applications," *Wireless Personal Communications*, vol. 111, no. 4, pp. 2287–2310, Apr. 2020, https://doi.org/10.1007/s11277-019-06986-8.

[8] S. Javaid, H. Afzal, M. Babar, F. Arif, Z. Tan, and M. Ahmad Jan, "ARCA-IoT: An Attack-Resilient Cloud-Assisted IoT System," *IEEE Access*, vol. 7, pp. 19616–19630, 2019, https://doi.org/10.1109/ACCESS.2019.2897095.

[9] G. Cornetta and A. Touhafi, "Design and Evaluation of a New Machine Learning Framework for IoT and Embedded Devices," *Electronics*, vol. 10, no. 5, Jan. 2021, Art. no. 600, https://doi.org/10.3390/electronics10050600.

[10] R. Elhabob, Y. Zhao, N. Eltayieb, A. M. S. Abdelgader, and H. Xiong, "Identity-based encryption with authorized equivalence test for cloud-assisted IoT," *Cluster Computing*, vol. 23, no. 2, pp. 1085–1101, Jun. 2020, https://doi.org/10.1007/s10586-019-02979-1.

[11] M. Alsharif and D. B. Rawat, "Study of Machine Learning for Cloud Assisted IoT Security as a Service," *Sensors*, vol. 21, no. 4, Jan. 2021, Art. no. 1034, https://doi.org/10.3390/s21041034.

[12] A. J. Siddiqui and A. Boukerche, "Adaptive ensembles of autoencoders for unsupervised IoT network intrusion detection," *Computing*, vol. 103, no. 6, pp. 1209–1232, Jun. 2021, https://doi.org/10.1007/s00607-021-00912-2.

[13] C. SaiTeja and J. B. Seventline, "A hybrid learning framework for multi-modal facial prediction and recognition using improvised non-linear SVM classifier," *AIP Advances*, vol. 13, no. 2, Feb. 2023, Art. no. 025316, https://doi.org/10.1063/5.0136623.

[14] G. P. Gupta, "Security Issues and Its Countermeasures in Examining the Cloud-Assisted IoT," in *Examining Cloud Computing Technologies Through the Internet of Things*, IGI Global, 2018, pp. 91–115.

[15] R. Gorle and A. Guttavelli, "A novel dynamic image watermarking technique with features inspired by quantum computing principles," *AIP Advances*, vol. 14, no. 4, Apr. 2024, Art. no. 045024, https://doi.org/10.1063/5.0209417.

[16] D. Roman, A. J. Berre, and J. Langlois, "Model as a Service (MaaS)," presented at the AGILE workshop: Grid technologies for geospatial applications, Hanover, Germany, Jun. 2009.

[17] A. Iacovelli, C. Souveyet, and C. Rolland, "Method as a Service (MaaS)," in *2008 Second International Conference on Research Challenges in Information Science*, Marrakech, Morocco, Jun. 2008, pp. 371–380, https://doi.org/10.1109/RCIS.2008.4632127.

[18] N. A. Alsharif, S. Mishra, and M. Alshehri, "IDS in IoT using Machine Learning and Blockchain," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11197–11203, Aug. 2023, https://doi.org/10.48084/etasr.5992.

[19] P. Mullangi *et al.*, "Assessing Real-Time Health Impacts of outdoor Air Pollution through IoT Integration," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13796–13803, Apr. 2024, https://doi.org/10.48084/etasr.6981.