Fortified Edge 3.0: A Lightweight Machine Learning based Approach for Security in Collaborative Edge Computing

Seema G. Aarella*, Saraju P. Mohanty* and Elias Kougianos†
*Department of Computer Science and Engineering, University of North Texas, USA
†Department of Electrical Engineering, University of North Texas, USA
Email: seema.aarella@unt.edu, saraju.mohanty@unt.edu,
elias.kougianos@unt.edu

***Abstract*—Machine Learning (ML) at the edge has the advantages of lower costs, reduced bandwidth needs, easy access, and high security. Machine Learning-based security protocols are hence suitable for providing added security at the edge, especially in distributed environments like collaborative edge computing (CEC). Security models are developed using the different ML methods, supervised, unsupervised, and reinforced learning. This research explores the ML methods suitable for security at the edge, particularly the supervised ML methods. The purpose of this research is to test the various supervised ML methods and compare their metrics to propose an efficient method for providing security at the edge. ML at the Edge can help tackle security challenges like external attacks, anomaly detection, data security privacy, and trust management. The edge environment considered is the Edge Data Centers (EDCs) in CEC, the security feature is the Physical Unclonable Function Certificate Authority (PUF CA) based authentication system, which is designed for authenticating EDCs during dynamic load balancing and the ML model is used to monitor the authentication process for intrusion and anomaly detection.**

***Index Terms*—Collaborative Edge Computing, Edge Data Centers, Secure Load Balancing, Secure Authentication, Device Monitoring, Cybersecurity, Machine Learning**

I. INTRODUCTION

Edge computing provides “Edge Intelligence" for utilization by Artificial Intelligence (AI) and ML algorithms. The development of distributed edge computing in the current Internet-of-Things (IoT) environment has opened its own challenges in the process of utilizing edge computing advantages. Edge computing helps leverage edge intelligence and properly manage the data, instead of moving data around the network, it helps processing and decision-making closer to the user. Edge computing as we know reduces the need for bandwidth, computation costs, processing times, response times, and so on, with the service access mostly at the edge, which makes possible high service availability. CEC environment enables task sharing across the various layers of the IoT architecture, The EDCs in the environment play a vital role in processing the shared tasks [1]. Security and privacy of edge computing is a vital area that enables to development of secure edge computing against malicious attacks, many security solutions are proposed which are based on software, and hardware, and with the use of ML and AI for optimizing the security solutions. One of the areas of research is the secure authentication of EDCs in a CEC for the smart village proposed in the research [2], The need for secure authentication of EDCs during load balancing is addressed in the research which proposes PUF-based algorithms, to further optimize the secure authentication system [3], ML-based monitoring and authentication is proposed in the research which helps to monitor authentication requests and detect anomalies and unauthorized request attempts. ML at the edge improves security and helps to optimize security systems. In case of any compromised EDC tries to attempt access/authorization, it can be detected and eliminated from the load-balancing pool of EDCs.

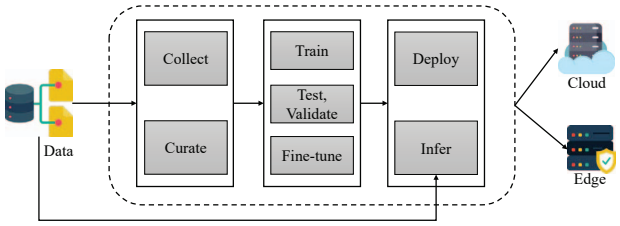

Figure 1: Lifecycle of Edge Machine Learning Systems.

There are many methods proposed for the authentication of EDCs during load balancing, some of which are PUF-based, Token-based authentication, Certificate-based authentication, SDN (Software Defined Network), blockchain, and so on. The purpose of this is to identify, verify, and authenticate devices and servers in the distributed environment. However, the security system should also be resilient against external attacks. In the IoT system, external attacks can happen at every layer, like the sensing layer, network layer, and middleware layer. gateway and application layer, technologies like blockchain and ML are widely studied to detect and prevent attacks at these layers. One of the challenges of using ML for attack detection and mitigation is to choose an efficient ML technique to apply for security systems at the edge [4] that adds additional layers of security to the authentication system.

979-8-3503-5823-0/23/$31.00 ©2023 IEEE

The lifecycle of ML at the edge is shown in Figure 1. However, ML comes with challenges of its own, like Data collection, Data quality, data inefficiency, investment, and challenges in implementing computationally extensive algorithms at the edge. ML though is a powerful tool that can be optimized through proper data mining and model selection techniques to get around the impediments [5]. The Authentication of EDC during load balancing should be a multi-faceted approach that combines various techniques to protect against different types of attacks and threats and maintain system integrity and stability. A combination of hardware and software, for secure authentication and monitoring, therefore, is a must [6].

II. RELATED PRIOR RESEARCH

Security at the edge involves security across the various layers of the IoT architecture. When it comes to EDCs and monitoring some of the aspects that need to be considered are lightweight and efficient ML models that can run on low resources, data analytics to extract useful information from the edge intelligence, choosing specific ML algorithms to suit the application needs, power, and memory efficient ML models. security at the edge using different ML techniques is discussed in the research, it explores the intrusion detection systems and corresponding ML algorithms that can overcome the security and privacy concerns. A study of ML algorithms for use cases such as network traffic data analysis, real-time responses, and time-series data analysis has shown that ML algorithms like SVM and K-means perform best for network traffic analysis, KD tree and Decision Tree/Randon Forest both have better performance in real-time responses, and DNN is adequate for time-series analysis, for implementation of ML-based security at the edge [7].

Considering the resource-constrained infrastructure at the edge, TinyML is considered an efficient tool for implementing the ML models for security applications, a study of various ML models using TinyML shows that the model can predict with high accuracy while utilizing low memory and power and the predictions times are faster as well [8]. Unsupervised ML models have been studied for intrusion detection on modern data, using the dataset of multiple attack types, the studies show that one-class SVM prevailed as a reliable model for intrusion detection, succeeding at detecting unseen attack types [9]. A comparative study of SVM classifier and KNN algorithms on the known dataset for intrusion detection shows the prediction accuracy of each algorithm to be best depending on the feature selection and classifier types. The study also shows that these algorithms are suitable for large datasets with less memory [10]. Some of the prominent research based on ML for cybersecurity are studied and listed in Table I.

III. NOVEL CONTRIBUTIONS OF CURRENT RESEARCH

Considering the infrastructure and constraints of the collaborative edge environment is important for developing ML-based security solutions. On these lines, the lightweight, mutual authentication system for EDCs during load balancing was proposed, it uses SRAM PUF-based CA (Certificate Authority) for mutual authentication, and an ML-based EDC authentication and monitoring model has been implemented using an SVM algorithm which showed efficient intrusion detection capabilities. The novel approaches in this research are listed below:

- A study of various lightweight ML models for implementation at the edge
- Implementing suitable lightweight ML models for intrusion detection and anomaly detection
- comparative study of different ML models for their prediction accuracy
- Comparative analysis of efficient supervised and unsupervised ML models for EDC Monitoring

IV. NEED FOR ROBUST AUTHENTICATION OF EDGE DATA CENTER

In a networked environment like CEC, the EDCs are susceptible to various types of attacks during the load-balancing process. failing to prevent these attacks will lead to disruption in the services and compromise the integrity of the system. some of the common types of attacks on EDC during load balancing are Cryptographic Attacks, Advanced Persistent Threats (APTs), Insider Threats, API and Application Layer Attacks, DNS Attacks, Zero-Day Exploits, and Vulnerability Attacks, these types of attacks are possible if the network security is weak. Some of the prominent security threats are discussed here:

A. Distributed Denial of Service(DDoS)

In a DDoS attack the edge servers are flooded with excessive traffic of incoming services from compromised edge devices, overwhelming the computational and communication capacity of the edge servers. The DDoS Problem can be approached in two ways, first, to detect and eliminate the malicious incoming services, and second, to have high-capacity servers and an effective load balancing scheme to process all requests without the genuine services getting hindered. [17].

B. Brute Force and Stuffing Attacks

The attackers attempting to gain access to a system by repeatedly trying different access credentials is called a Brute Force attack. A stuffing attack is like a brute force attack, but the attacker uses the previously obtained access credentials from past attacks. This type of attack is classified as credential cracking and credential stuffing attacks [18]. These attacks can be mitigated by implementing trusted and secure authentication protocols.

C. Man-in-the-Middle Attacks(MitM)

The attacker intercepts the communication between the edge nodes and server, or any two communication nodes in the system, and tries to alter or capture sensitive information. The attacker can inject malicious content by being connected to the network. This type of attack can be prevented through the use of end-to-end network and server security systems to monitor and detect the attackers [19].

Table I: Comparative Table for State-of-the-Art Literature.

Research	Year	ML Model	Environment	Application
Pacheco et. al. [11]	2020	ANN	Fog Nodes	Intrusion Detection for IoT
Hussain et. al. [12]	2019	DNN	Mobile Edge Computing	Anomaly Detection
Abeshu et. al. [13]	2018	DL	Fog Nodes	Fog-to-Things Distributed Attack Detection
Diro et al. [14]	2018	LSTM	Distributed Fog-to-Things Communication	Intrusion Detection
Sadaf et. al. [15]	2020	Isolation Forest	Fog Computing	Intrusion Detection
Nie et. al. [16]	2022	GAN	Collaborative Edge Computing	Intrusion Detection
Fortified-Edge 1.0 [2]	2022	NA	Collaborative Edge Computing	PUF Based Secure Authentication
Fortified-Edge 2.0 [6]	2023	SVM	Collaborative Edge Computing	Secure Authentication and Monitoring
Fortified-Edge 3.0 (Current Paper)	2023	Various ML	Collaborative Edge Computing	Intrusion Detection

D. Side-Channel Attacks (SCA)

SCA exploits the information leaked through operating characteristics of a system such as power consumption, electromagnetic emanations, or timing measurements. some of the examples of SCA on EDC are power analysis attacks, electromagnetic attacks, timing attacks, cache timing attacks, fault injection attacks, acoustic attacks, and so on. These attacks can potentially compromise the security of EDCs. Therefore, the EDCs need a continuous end-to-end security system to provide physical security, hardware security, noise mitigation, monitoring, data, and cryptographic protection.

Protecting the system against these attacks requires a multilayered security strategy, that should include robust authentication, network monitoring, and intrusion detection systems, along with regular updates, for continuous monitoring against evolving threats. some of the machine learning models used for anomaly detection and intrusion detection are listed in Figure 2. One of the prominent aspects of cybersecurity for EDCs is secure authentication, no matter how robust the authentication system is, it is still vulnerable to authentication and authorization attacks, the major threats to authentication are listed below [20]:

- Dictionary Attacks
- Authentication Protocol Attacks
- Authorization Protocol Attacks
- Overprivilege Attacks

An authentication system demands complex controls and multilevel authentication mechanism to overcome the issues related to it, such as identity management and key exchange for multiple distributed entities, resource-efficient mechanisms, and maintaining the integrity of the authentication session [21]. This research focuses on securing the authentication system against such adversaries, where a secure and trusted authentication system is proposed which also uses ML to provide continuous monitoring and attack detection system.

V. LIGHTWEIGHT ML ALGORITHMS FOR EDC MONITORING

EDC monitoring is essential for ensuring the reliability, security, and efficiency of the infrastructure components and operations. With respect to secure authentication and authorization of EDCs during load balancing, monitoring systems can provide the added security to detect intrusion and malicious requests, hence the need for real-time monitoring is crucial to prevent various types of attacks that are targeted towards the edge servers. However, the monitoring system should be implemented without consuming excessive resources at the edge. some of the lightweight ML algorithms suitable for EDC monitoring are discussed in the current Section.

A. Log-Based Anomaly Detection

1) Isolation Forest: Is a tree-based anomaly detection algorithm that works well with high-dimensional logs. It is lightweight and efficient for detecting unusual patterns in log files generated by EDCs.

2) One-Class SVM: One-Class SVMs are useful for identifying anomalies if using a predominantly normal dataset.

B. Time Series Anomaly Detection

1) Seasonal Hybrid ESD(S-H-ESD): S-H-ESD is an extension of the Extreme Studentized Deviate test specifically designed for time series data. It is lightweight and can efficiently detect anomalies in time series data generated by EDCs.

2) Moving Average and Exponential Smoothing: Is a simple statistical method that can be used for lightweight time series forecasting and anomaly detection.

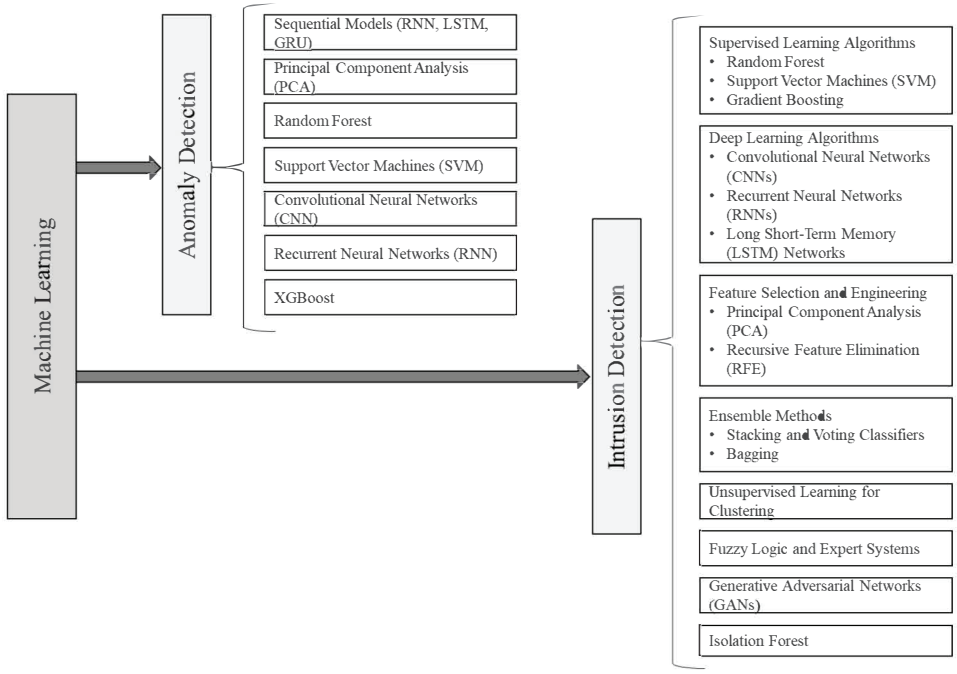

Figure 2: Machine Learning algorithms for Anomaly Detection and Intrusion Detection.

C. Deep Learning Models

1) LSTM(Long Short-Term Memory) Networks: LSTM networks are suitable for analyzing time series data and can be used as a lightweight anomaly detection model when more complex patterns need to be captured. The algorithm can be quantized to make it more lightweight for applications at the EDC.

2) TinyML Models: Tiny Machine Learning (TinyML) models are designed to run on resource-constrained edge devices. They include various network architectures like CNN, and they are optimized for low memory and power consumption.

D. Rule-Based Models

1) Expert Systems: Expert systems use predefined rules and knowledge bases to identify anomalies, these systems can be specifically tailored to suit the requirements of an EDC monitoring system.

E. Ensemble Models

1) Random Forest: Random Forest is an ensemble learning model that can be used for EDC monitoring by combining multiple decision trees. It is relatively lightweight, and it can handle both structured and unstructured data.

2) XGBoost and LightGBM: XGBoost and LightGBM are gradient-boosting algorithms that are lightweight and can be used for anomaly detection and prediction tasks, especially when dealing with tabular data.

When designing and deploying an ML model for EDC monitoring, it is essential to consider the CPU restraints, power consumption, model size, memory available, and the capacity of the model to adapt to changing conditions, for example when new threats or anomalies are detected.

VI. IMPLEMENTATION

Table II: Features Considered for ML model Training.

Features	Variables
Site_ID	S_{id}
EDC_ID	E_{id}
EDC_ID_Requestor	E_{idr}
Latitude_EDC	L_a
Longitude_EDC	L_o
Latitude_EDC_Requestor	L_{ar}
Longitude_EDC_Requestor	L_{or}
Distance	d_r
Certificate_Validity	c_r
Authntication_Time	t_r

The ML-based EDC monitoring for anomaly and intrusion detection system is implemented for the SRAM PUF CA-based authentication system [6], the system design involves EDCs mutual authentication protocols where the EDCs authenticate each other during load balancing by verifying the digital signature and the CA certificate issued by the authentication and authorization server. SVM-based ML algorithm is used for monitoring the authentication system and detecting any anomalies or intrusions, like unauthorized authentication requests. In this research, we are exploring the lightweight ML algorithms that can efficiently detect malicious requests. The ML algorithms studied in this research are One Class SVM (Support Vector Machine), Isolation Forest, Decision Tree, and Random Forest.

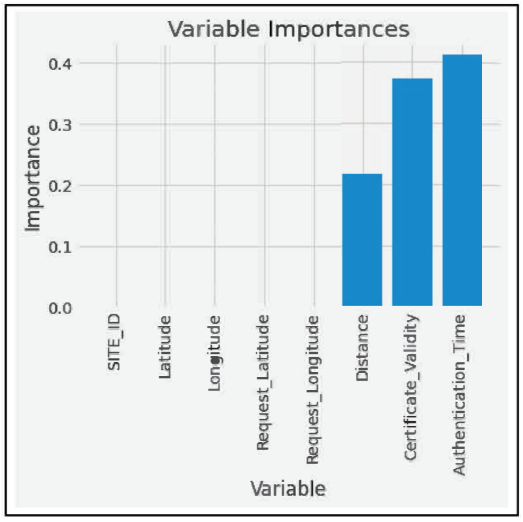

Figure 3: Attributes Seleted from Dataset For Training.

The dataset used is from the EDC authentication data generated, along with the location data of the EDCs from the implementation in research [6]. The dataset has 1000 rows of data, and the attributes of the data are listed in Table II, among the dataset, 800 rows of data are used for training and 200 are used for testing.

VII. Results and Analysis

The SVM classifier is a supervised ML model implemented in the preceding research that had shown the prediction accuracy of 100%, in detecting the malicious authentication requests, the classifier was able to perform better with proper feature selection for analysis [6].

One-Class SVM is an unsupervised model, implemented for anomaly detection, by first detecting the outliers the model is trained to detect anomalies. Three informative features were selected as predictors and the classification was made, 99% of the data was used for the majority class and 1% data was in the minority class. The One-Class classifier labels the normal data as '1' and anomalies as '-1'. The model showed a recall value of 2% for anomaly points in data, the recall value increased with the increase in the threshold number. A prediction accuracy of 98% was obtained for the One-Class SVM model.

The Isolation Forest is widely used for anomaly detection applications. It is an unsupervised model that identifies the anomalies by isolating the outliers in the data. The outliers identified by the algorithm for one of the features are shown in Figure 4. For the data of 1176 rows, the algorithm identified 168 outliers and was able to predict with an accuracy of 100%.

The Random Forest implementation here is considered a supervised, regression ML problem, we use both the data features and the targets to be predicted. The visual representation of the decision tree created by the algorithm is shown in Figure 5. The accuracy of the prediction after the regression is found to be 100%.

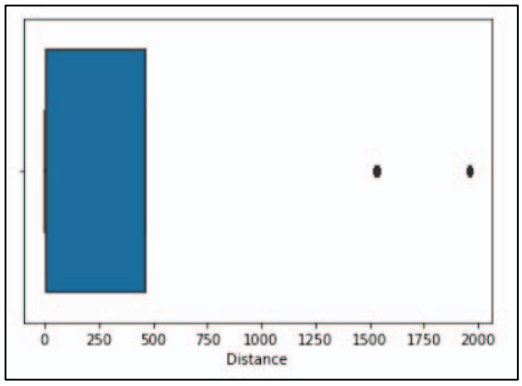

Figure 4: Outliers Identified by the Isolation Forest.

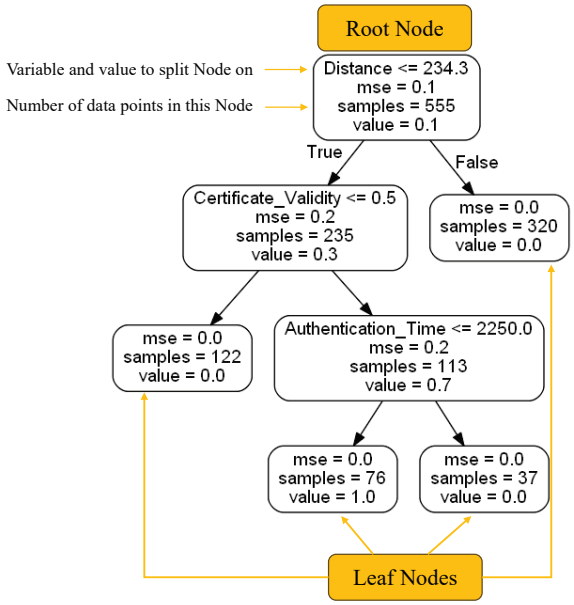

Figure 5: Visualization of the Random forest Tree.

In the Decision Tree implemented each node represents a "test" on an attribute and each branch represents the outcome of the test. The decision is taken after computing all the attributes. The confusion matrix of the DT algorithm is shown in Figure 6, the algorithm is efficient in predicting the labels (outcomes) correctly.

VIII. Conclusions

One of the challenges of implementing an efficient ML model for security at the edge is data. During the research, it was found that proper data acquisition and data preprocessing are important for effective use. The performance of the ML algorithms is enhanced with suitable feature selection and modeling according to the ML algorithms used. The ML models are tested with smaller datasets and show good prediction accuracy only after proper preprocessing of the data. However, the model's performance is not degraded after an increase in the sample sizes of proper data. From the comparative table of results in Table III, it is seen that the lightweight ML

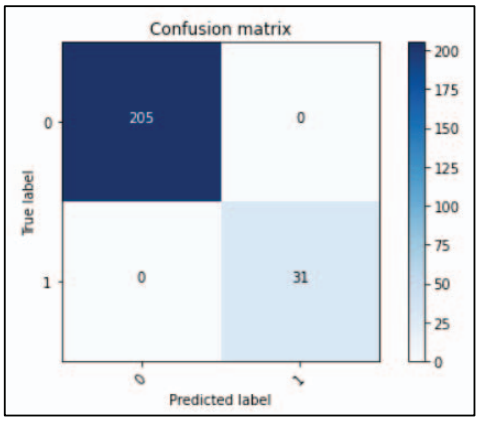

Figure 6: Confusion matrix of the Decision Tree.

Table III: ML Models used and Metrics Calculated.

ML Algorithms	Use Case	Accuracy	Precision	Recall	F1 Score
SVM [6]	Intrusion detection	1.00	1.00	1.00	1.00
One-Class SVM	Intrusion detection	0.97	0.99	0.98	0.98
Isolation Forest	Anomaly Detection	1.00	NA	NA	NA
Random Forest	Anomaly Detection	1.00	NA	NA	NA
Decision Tree	Intrusion Detection	1.00	1.00	1.00	1.00

algorithms perform exceptionally for anomaly detection and Intrusion detection at the edge, provided proper labeling of the anomalies is performed in the data preprocessing stage. Continuing the implemenation of a ML based monitoring and authentication system in Fortified Edge 2.0 [6], which aims at security at the edge level. Fortified Edge 3.0 is a novel study that explores both supervised and unsupervised algorithms capable of anamoly and intrusion detection, and which is suitable for implementation at the edge nodes.Use of these ML algorithms for security applications will be helpful in detecting attacks and external threats, however, the algorithms should be made capable of learning from newer threats that might occur, which is a crucial factor for designing security applications for the CEC.

References

[1] D. Puthal, S. P. Mohanty, S. Wilson, and U. Choppali, "Collaborative Edge Computing for Smart Villages [Energy and Security]," *IEEE Consumer Electronics Magazine*, vol. 10, no. 3, pp. 68–71, 2021.

[2] S. G. Aarella, S. P. Mohanty, E. Kougianos, and D. Puthal, "PUF-based Authentication Scheme for Edge Data Centers in Collaborative Edge Computing," in *IEEE International Symposium on Smart Electronic Systems (iSES)*, 2022, pp. 433–438.

[3] S. G. Aarella, S. P. Mohanty, E. Kougianos, and D. Puthal, "Fortified-Edge: Secure PUF Certificate Authentication Mechanism for Edge Data Centers in Collaborative Edge Computing," in *Proceedings of the Great Lakes Symposium on VLSI (GLSVLSI)*, 2023, p. 249–254.

[4] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures," *IEEE Access*, vol. 7, pp. 82 721–82 743, 2019.

[5] K. Hoffpauir, J. Simmons, N. Schmidt, R. Pittala, I. Briggs, S. Makani, and Y. Jararweh, "A Survey on Edge Intelligence and Lightweight Machine Learning Support for Future Applications and Services," *J. Data and Information Quality*, vol. 15, no. 2, jun 2023. [Online]. Available: https://doi.org/10.1145/3581759

[6] S. G. Aarella, S. P. Mohanty, E. Kougianos, and D. Puthal, "Fortified-Edge 2.0: Machine Learning based Monitoring and Authentication of PUF-Integrated Secure Edge Data Center," in *proc. IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2023, pp. 1–6.

[7] H. Wang, L. Barriga, A. Vahidi, and S. Raza, "Machine Learning for Security at the IoT Edge - A Feasibility Study," in *IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW)*, 2019, pp. 7–12.

[8] S. A. R. Zaidi, A. M. Hayajneh, M. Hafeez, and Q. Z. Ahmed, "Unlocking Edge Intelligence Through Tiny Machine Learning (TinyML)," *IEEE Access*, vol. 10, pp. 100 867–100 877, 2022.

[9] M. Verkerken, L. D'hooge, T. Wauters, B. Volckaert, and F. De Turck, "Unsupervised Machine Learning Techniques for Network Intrusion Detection on Modern Data," in *4th Cyber Security in Networking Conference (CSNet)*, 2020, pp. 1–8.

[10] K. Shanthi and R. Maruthi, "Machine Learning Approach for Anomaly-Based Intrusion Detection Systems Using Isolation Forest Model and Support Vector Machine," in *5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2023, pp. 136–139.

[11] J. Pacheco, V. H. Benitez, L. C. Félix-Herrán, and P. Satam, "Artificial Neural Networks-Based Intrusion Detection System for Internet of Things Fog Nodes," *IEEE Access*, vol. 8, pp. 73 907–73 918, 2020.

[12] B. Hussain, Q. Du, S. Zhang, A. Imran, and M. A. Imran, "Mobile Edge Computing-Based Data-Driven Deep Learning Framework for Anomaly Detection," *IEEE Access*, vol. 7, pp. 137 656–137 667, 2019.

[13] A. Abeshu and N. Chilamkurti, "Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 169–175, 2018.

[14] A. Diro and N. Chilamkurti, "Leveraging LSTM Networks for Attack Detection in Fog-to-Things Communications," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 124–130, 2018.

[15] K. Sadaf and J. Sultana, "Intrusion Detection Based on Autoencoder and Isolation Forest in Fog Computing," *IEEE Access*, vol. 8, pp. 167 059–167 068, 2020.

[16] L. Nie, Y. Wu, X. Wang, L. Guo, G. Wang, X. Gao, and S. Li, "Intrusion Detection for Secure Social Internet of Things Based on Collaborative Edge Computing: A Generative Adversarial Network-Based Approach," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 1, pp. 134–145, 2022.

[17] T. Li, "Optimal cloud assistance policy of end-edge-cloud ecosystem for mitigating edge distributed denial of service attacks," *Journal of Cloud Computing*, 2021. [Online]. Available: https://doi.org/10.1186/s13677-021-00257-3

[18] M. H. N. Ba, J. Bennett, M. Gallagher, and S. Bhunia, "A Case Study of Credential Stuffing Attack: Canva Data Breach," *International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 735–740, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:249928534

[19] S. Z. DİCLE, "Man-In-The-Middle Attack," *European Journal of Science and Technology*, Oct. 2022. [Online]. Available: https://doi.org/10.31590/ejosat.1187984

[20] M. S. Ansari, S. H. Alsamhi, Y. Qiao, Y. Ye, and B. Lee, *Security of Distributed Intelligence in Edge Computing: Threats and Countermeasures*. Cham: Springer International Publishing, 2020, pp. 95–122. [Online]. Available: https://doi.org/10.1007/978-3-030-41110-7_6

[21] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, "A Survey on Security and Privacy Issues in Edge-Computing-Assisted Internet of Things," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4004–4022, 2021.