Federated Learning Empowered by Attention-Augmented CNN for Edge-Based Intrusion Detection

B. S. Kiruthika Devi[1], Sivesh Shrutik Muhilvannan[2], Nishanth Babu H[3]
[1]*Assistant Professor School of computing Sathyabama Institute of Science and Technology Chennai, India*
[2,3] *UG Scholar Department of CSE with Cybersecurity Sathyabama Institute of Science and Technology Chennai, India*

***Abstract*— In today's swiftly developing technological environment, where iot and edge computing is becoming vaster and more dominant. The rapid growth introduces a variety of concerns regarding different cyber threats and attacks which are focused on edge computing networks and its devices. The increasing number of interconnected devices in the edge network would proportionally have a large amount of data flowing through it on a daily basis. This exposes itself to the constraints of the conventional Intrusion Detection System (IDS). The data collection process under Traditional and Centralized Intrusion Detection Systems occurs through a solitary server that handles data acquisition from edge devices which leads to major privacy problems and security weaknesses. The centralized models are not best suited for edge environments where computational resources are constrained. Federated learning permits the models to be trained across the decentralized devices without the need to share their raw data to themselves or to the server. The proposed framework aggregates the updates from each edge device and the models trained by them to create a global IDS model to ensure security and efficiency at a global range. The data used for this framework is simulated using the Edge-IIoTset Cyber Security Dataset of IoT & IIoT dataset. This approach is evaluated by detection accuracy, model convergence and impact of federated aggregation on performance. The results validate that validate federated learning as a privacy-preserving IDS solution. The successful implementation of the system could considerably enhance the security of edge computing environment, making it more efficient and safer.**

Keywords*— *Federated Learning, Intrusion Detection Systems, Edge Computing, Privacy-Preserving, IoT Security, Edge-IIoTset Dataset, Detection Accuracy.

I. INTRODUCTION

The rise of IoT devices and edge computing technologies and their collaboration brought significant digital alterations that enable smart devices and industrial sensors to operate more effectively. This connectivity, however, introduces new threats and cybersecurity issues, hence increasing the need for a decentralized solution for threat detection. Conventional IDSs, struggle with scalability and privacy in edge environments due to their systems being dependent on a centralized framework for data preprocessing.

By allowing edge devices to train and develop local models without sharing their local data with other edge devices, federated learning issues an alternative decentralized approach compared to standard centralized frameworks. This lowers privacy issues whilst only requiring less connectivity between the devices and the global server. A periodic aggregation process of the local models allows them to develop a global model that functions efficiently as a secure intrusion detection system (IDS). According to [3] [4], federated learning approaches have the ability to boost the cybersecurity of Edge networks by integrating deep intrusion detection models. The proposed framework adds value through its combination with local training that employs an Attention-Augmented CNN model. Through a design that joins convolutional layers with attention mechanisms the framework lets vital features in the dataset information receive attention. This dynamically emphasizes key patterns and enhances detection accuracy , especially for complex or subtle anomalies, as described in [13].

Combining federated learning with an attention-augmented CNN enhances resilience, adaptability, and security in IoT and edge computing networks. The resultant system is real-time as well as privacy-preserving, thus overcoming significant limitations in modern cybersecurity landscapes.

II. RELATED WORK

The incorporation of federated learning and attention mechanisms has significantly enhanced the

capabilities of Intrusion Detection Systems (IDSs) for IoT and edge environments. Traditional IDS frameworks are resource constrained due to their high computational demands and need for secure connections. This led to the exploration and discovery of decentralized methods of training models keeping in mind the privacy concerns. As demonstrated in studies like [1] and [2], Federated learning permits edge devices to cooperatively train models whilst securing their data. Notably, asynchronous federated learning approaches, in conjunction with deep hybrid detection models, have been promising in enhancing the cybersecurity of IIoT networks [3].

The precision of deep learning architectures grew stronger when attention mechanisms were added to them. The model achieves enhanced network traffic data analysis by using attention mechanisms which help it identify crucial features during detection.

III. PROPOSED METHODOLOGY

A. Proposed Architecture

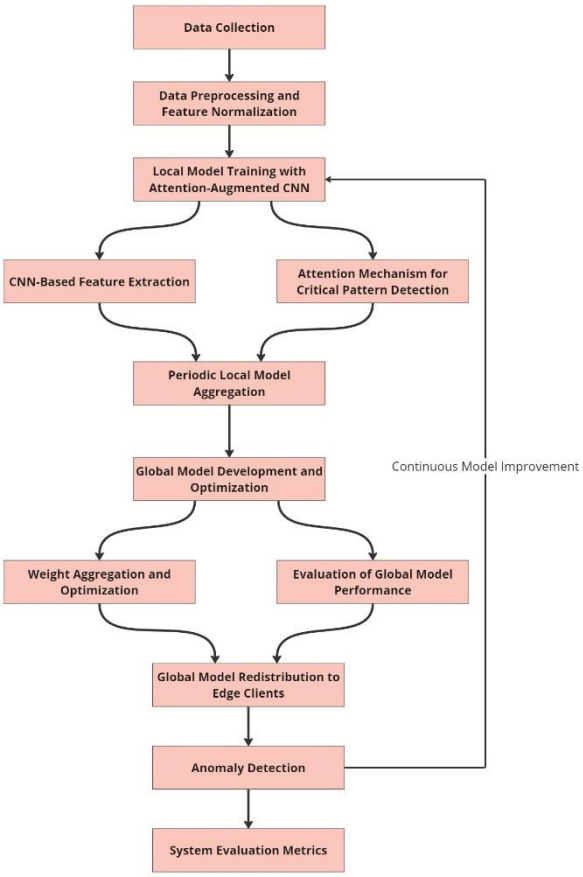

Fig .1. Proposed Architecture

The proposed system will use Federated Learning (FL) and an Attention-Augmented Convolutional Neural Network (CNN) model to detect network intrusions. As indicated in Fig. 1, the architecture provides traffic data from the dataset to multiple Edge IoT Devices, pre-processes the data, and extracts critical features for intrusion detection [13]. Each device initializes and trains using the extracted features. Once the local models are trained, they communicate with a Federated Learning global client, where model aggregation combines the locally trained models into a unified Global Model [9],[16]. This global model is redistributed to edge devices for further local refinement. The Attention Mechanism in the CNN boosts the model's proficiency to focus on the most critical data features, improving its accuracy in detecting intrusions. After the update, the global model undergoes evaluation by computing accuracy, precision, and recall. Then the validated global updated model is fed into Continuous Monitoring to update with evolving threats in the network. This is illustrated in the iterations of Fig. 1 in creating an efficient IDS with adaptive edge detection for distributed devices, where such a mechanism does not impact the accuracy rate of the detector .

B. Data Collection and Preprocessing

The Edge-IIoTset Cybersecurity Dataset is the primary baseline dataset for this project, which contains all labeled network traffic patterns drawn from the IoT and IIoT environments. The preprocessing pipeline loads the dataset and checks for missing values. Missing data is filled with zeros to preserve dataset integrity during training.

The dataset includes both numeric and non-numeric columns. Non-numeric columns include ip.src_host, ip.dst_host, and Attack_type. These columns are chosen as they are irrelevant to training. The column Attack_label, which contains categorical labels for attack types, is encoded using LabelEncoder to convert categorical values into a numeric representation.

The relevant numeric features are selected from the dataset. The columns that are of numerical data types, including float64 and int64, are retained. The features are normalized using StandardScaler to ensure that all feature values are on the same scale. This step prevents the model from becoming biased toward any particular feature due to varying scales. The normalization is performed using the equation (1)

$$X_{scaled} = \frac{X - \mu}{\sigma} \quad (1)$$

This normalization allows that every feature contributes equally to the model and prevents large-value bias.

Finally, the preprocessed data splits into feature variables (X) and target variable (y, which is the Attack_label), The feature matrix X is normalized and, y consists of encoded attack labels. Now the dataset is ready for model training, at which point the data are reshaped to be able to feed it into the CNN model [13].

C. Local Model Training

In this step the edge clients independently execute their network traffic data subsets for training while preserving the data at local levels and avoiding data transmission to a central global client. This tactic ensures privacy by preserving the data at a local level [15] and saves communication overhead, rendering it remarkably well-suited for IoT and edge computing environments. The training process consists of two key components: The Attention-Augmented CNN architecture serves as a core component that detects vital attack patterns [13] before proceeding to the Training Phase where development occurs through advanced methods for use in aggregation processes. Together, these components work collaboratively to enhance intrusion detection, ensuring both precision and efficiency in edge environments.

1) Attention Augmented CNN

The attention-augmented CNN architecture is very pivotal to intrusion detection, as it enhances the model's capability to detect complex patterns in network traffic [11]. The process begins with the convolution layers extracting intricate features from the acquired data. However, simply extracting features is not enough, as many attack signatures may be subtle or buried within noise. Here, the attention mechanism comes into play. The attention mechanism learns to determine significant features through weight distribution applied to different regions of input data. This allows the model to focus on further relevant information, such as unusual traffic spikes or specific packet characteristics, likely to indicate an attack.

After the convolution layers extract these features, the attention mechanism dynamically adjusts the importance of each feature map generated by the convolution operation. It assigns a weight value to each of the spatial feature maps, indicating their impact in recognizing intrusions. Through the training period, the values are learned where the system takes notice of crucial patterns. Again, the attention mechanism acts as a dynamic filter, zooming into the critical regions of the selected data. This process allows the model to adjust to the varying nature of network traffic, which may fluctuate and contain noise in edge environments.

At the multi-scale feature extraction phase, the learned attention weights will be utilized. It is a step in which the CNN extracts finer-grained features of such patterns, capturing anomalies in packets or trends like malicious traffic flows. As it concentrates on crucial patterns at diverse scales, attention enables detection even of subtle threats overlooked by other techniques [11].

Finally, attention-augmented CNN becomes more effective in detecting normal and malicious activities. Through the attention mechanism, the detection of sophisticated threats and complex attack patterns is more enhanced, and the model detects even subtle or uncertain attack patterns. As noted in [11], the attention-augmented CNN would show superior performance in augmenting anomaly detection, especially on edge environments by focusing on most relevant data patterns. This leads to accurate detection efficiency, even towards sophisticated and diverse complex threats across time, giving intrusion detection an advantage.

2) Training Phase

During the local training phase, edge clients utilize Edge IIotset Cybersecurity dataset to train the models [12]. The global model is cloned to each virtual client, ensuring that all local models share similar architecture and weights. The local models are then trained on their respective subsets of the dataset using the Adam optimizer. This iterative process refines the model parameters to improve detection accuracy, focusing on attack signatures and anomalies that are crucial for identifying threats. The Adam optimizer refines the model parameters effectively, and the weight updates are computed according to the following equation (2):

$$\theta_t = \theta_t - 1 - \frac{\eta}{\sqrt{\upsilon_t} + \epsilon} \times m_t \qquad (2)$$

In equation (2), θt is the parameter at time step t, η is the learning rate, υ_t is the bias-corrected second moment estimate, ϵ is the small constant to stop division by zero, and mt is the bias-corrected first moment estimate.

The local model training phase ensures that each edge client's model can refine its understanding of network traffic. This process focuses on feature extraction and prioritization, allowing the models to specialize in detecting patterns specific to their local data. After each client has trained, the optimized weights and biases are sent to the global client for aggregation. This aggregation of parameters from all edge clients builds a global model with higher capacity to generalize across diverse conditions, reflecting the varied datasets and local learning environments. This global model shows higher capacity to generalize across the wide range of conditions as it reflects diverse learning experiences and datasets of devices [14].

D. Periodic Local Model Aggregation

After each edge client has independently trained their local model, with reference to the attention-augmented CNN model, on their unique subset of the dataset. Then, model parameters like weights and biases are shared with the central global client to be aggregated [8]. This global client aggregates the models that are locally trained using an algorithm named FedAvg: it combines the locally trained models by averaging the model parameters (weights and biases) from all participating edge clients. According to McMahan et al. [17], FedAvg is straightforward and efficient at model aggregation, making it useful for non-IID data as well. All model updates from the participating edge clients in the aggregation procedure get equal consideration in the update of the global model, as shown in equation (3):

$$W_{global} = \frac{1}{N}\sum_{i=1}^{N} W_i \quad (3)$$

Where Wglobal is the globally updated model, and Wi denotes the model parameters from the i-th edge client. This averaging technique prevents any dominance of any single device from dominating the model updates [17].

FedAvg fills the gap between data heterogeneity across edge clients. Because each device functions within a unique environment, perhaps with varying network traffic distributions, local models could have different generalization. FedAvg thus averages updates in such a way that balances the differences and offers an even more robust generalized intrusion detection model. Since devices may notice different attack patterns in non-IID data FedAvg is especially effective for it. For instance, certain devices may detect unique patterns that are not visible to other devices [6].

A key challenge in this process is device variability, including differences in resources and datasets, can affect training efficiency. FedAvg assumes uniform contribution from all devices, but in reality, some devices may provide more informative updates than others. This can be handled by employing weighted averaging, which can be used to give more importance to devices with larger or more representative datasets. After aggregation, the global client updates the model with collective knowledge from all edge devices. This step maintains distributed, privacy-preserving learning while improving threat detection. [17].

E. Global Model Redistribution

Once the model updates are aggregated using FedAvg, each client now receives the global model for further training. At this point of the iterative process the model parameters, rather than the raw data, are shared, preserving data privacy.

The overall model becomes more refined, as each client performs further training with the new parameters and its ability to categorize a broader spectrum of cyber threats improves. The model becomes more robust and adaptable to attack patterns through this continuous cycle of aggregation.

By repeatedly refining the global model in this manner, the system remains effective at detecting cyber threats in a real time application. As the model receives new updates with each iteration, it becomes increasingly capable of responding to evolving network conditions and attack tactics, ensuring a highly effective intrusion detection system in dynamic IoT and edge computing environments.

F. Real Time Anomaly Detection

This is done in an iterative cycle of feedback at both the local and global levels. After validation on a validation set using accuracy, recall, F1-score and precision the global model updates itself by implementing findings from detected attack patterns and anomalies. The learning process applies repeated updates to the model for detecting recent cybersecurity threat patterns emerging over time [10]. Over time the anomaly detection model obtains evaluations based on its recall performance as well as precision measures and accuracy scores and F1-score ratings when discovering new attack signatures from Edge-IIoTset Cybersecurity Dataset. The

effectiveness of these updates in distinguishing benign and malicious traffic is highlighted by [5], showcasing the importance of frequent model updates in hardening IDSes against various attacks. Every edge client executes FedAvg through its local model to conduct an update during each iteration according to the model aggregation function. The parameters from multiple edge clients are averaged together, ensuring fairness and generalization. This prevents overfitting towards attack patterns from a single client, making the model more broadly applicable. Once the global model has been updated, it is redistributed back to the edge clients, allowing local models to further train with the new parameters.

As these updates are continuously aggregated and redistributed, the detection network evolves to handle new, sophisticated threats. This iterative process between local model training, global aggregation, and redistribution strengthens the model, improving its detection performance while maintaining adaptability to the changing landscape of cybersecurity threats. This feedback loop is crucial for robust cybersecurity defences in dynamic IoT and edge environments, as noted by [7] and [13], ensuring long-term effectiveness and adaptability.

G. System Evaluation Metrics

The model evaluation criteria are defined by average classification metrics, including accuracy rate, precision, recall, and F1-score, as represented by equations (5), (6), (7), and (8).

$$\text{Accuracy}(\%) = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} X100 \quad (5)$$

$$\text{Precision}(\%) = \frac{\text{TP}}{\text{TP+FP}} X100 \quad (6)$$

$$\text{Recall}(\%) = \frac{\text{TP}}{\text{TP+FN}} X100 \quad (7)$$

$$\text{F1 Score}(\%) = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision+Recall}} X100 \quad (8)$$

Where:

- TP = True Positives
- FP = False Positives
- FN = False Negatives
- TN = True Negatives

IV. RESULTS AND DISCUSSION

A. Outputs

1) Output Metrics

The Output Metrics for Attention Augmented CNN trained using various other federated algorithms are depicted in a comparative tabulated format and it is depicted in Table.1

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Federated Average	97.2	98.1	95	97
Federated Proximal	96	97	95	96
Federated Stochastic Gradient	73	53	73	61

Table.1

The following Table.2 presents various algorithms used with FedAvg.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Attention Augmented CNN	97.2	98.1	95	97
SVM	93.4	88	91.6	87
LSTM	73	71	53	84
Random Forest	73	53	73	61

Table.2

2) ROC Curve

The AUC value of the ROC curve is a method to measure how well the classifier performs. AUC is expressed as the area underneath the ROC curve, representing the area calculated by plotting TPR on the y axis against FPR on the x axis. The AUC is computed as the integral of the TPR with respect to the FPR (3), thus producing a single measure summarizing how well the model could classify classes.

$$AUC = \sum_{i=1}^{n-1} \frac{(FPRi+1-FPRi) \times (TPRi+1+TPRi)}{2} \quad (3)$$

The ROC curve is shown in Fig. 2, illustrating the model's performance in terms of TPR and FPR.

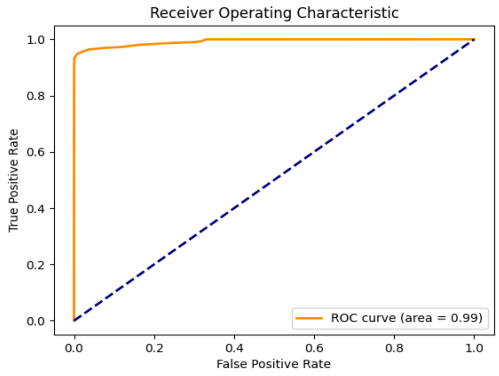

Fig.2.ROC Curve

The ROC curve in Fig. 2 demonstrates the model's strong performance, with an AUC of 0.99 indicating its ability to effectively distinguish between attack and normal traffic. This near-perfect AUC suggests the model minimizes false negatives while maintaining a low false positive rate, crucial for an effective IDS.

3) Confusion Matrix

The confusion matrix for the Attention-Augmented CNN model in our Fl system is displayed in Fig. 3

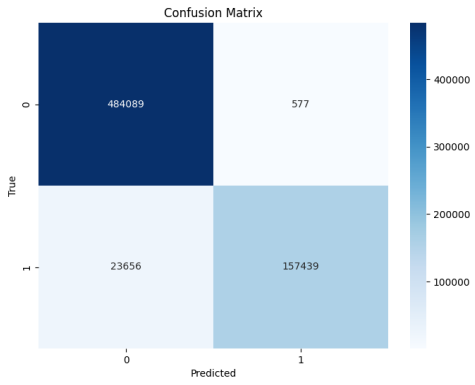

Fig.3. Confusion Matrix

Fig. 3 presents the confusion matrix for the Attention-Augmented CNN model. Low counts of false positives (577) and false negatives (23656) demonstrate the model's strong performance, with 157439 true positives and 484089 true negatives indicating effective classification across both categories.

4) Precision-Recall Curve

The Precision-Recall curve of Fig. 4 plots the trade-off amid precision and recall of the model. A high area underneath the curve shows good balance in maximizing both precision and recall and thereby the ability of the model to detect positive instances with fewer false positives.

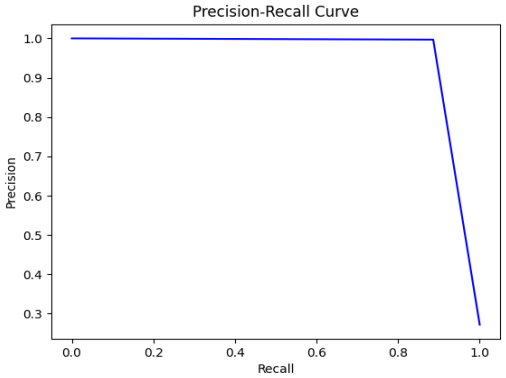

Fig .4. Precision-Recall Curve

Fig .4 shows the curve's shape, with its extended high-precision region, indicates a well-balanced model capable of effectively managing this trade-off. This suggests the model is adept at correctly identifying intrusions while minimizing false positives, crucial for practical deployment.

B. Novel Contributions

The novelty of this project is the attention-augmented CNN model that is designed to enhance intrusion detection in edge environments. This model integrates a dynamic attention mechanism to prioritize subtle and complex attack patterns in network traffic data, distinguishing it from conventional models that may overlook such anomalies. Another is through the application of Federated Learning (FL) to ensure privacy in data through collaborative training at the edge, without sharing raw data. It forms a scalable, precise intrusion detection system personalized for the obstacles presented by IoT and edge computing environments, combining the robustness of an attention-augmented CNN with the power of privacy-preserving FL.

V. CONCLUSION

As proved this project addresses some of the pressing issues concerning the challenges associated with intrusion detection within IoT and edge environments. An attention-augmented CNN will allow a model to place more attention to significant patterns or anomalies and avoid overlooking minute or hard-to-discern patterns and anomalies while considering data privacy through the federated learning framework, promoting scalability. By utilizing methods, such as FedAvg for aggregation, it

improves accuracy in performance and robustness of the system and satisfies the specific requirements of edge computing environments. It has a possibility of extending federated learning to improve security cyber systems, which is confirmed by some studies "Federated Learning for Distributed IIoT Intrusion Detection Using Transfer Approaches" [6].

VI. FUTURE WORK

The future work will be in the evaluation of the framework on diverse datasets to increase its applicability to a wide range of cybersecurity domains. The system can be further robust and adaptable by extending the range of datasets. Further, more privacy-preserving methods such as homomorphic encryption and secure multi-party computation can be incorporated into the federated learning environment to strengthen data protection. Lowering the computational overhead of the attention-augmented CNN and using hybrid models could also help to improve efficiency as well as detection accuracy on resource-constrained edge devices.

REFERENCES

[1] Jun-Min Shao, Guo-Qiang Zeng, Kang-Di Lu, Guang-Gang Geng, Jian Weng. (2024). Automated federated learning for intrusion detection of industrial control systems based on evolutionary neural architecture search. ScienceDirect Computers & Security, ISSN 0167-4048.

[2] Danish Javeed, Muhammad Shahid Saeed, Muhammad Adil, Prabhat Kumar, Alireza Jolfaei. (2024). A federated learning-based zero trust intrusion detection system for Internet of Things. ScienceDirect Computers & Security, ISSN 1570-8705.

[3] Syed Muhammad Salman Bukhari, Muhammad Hamza Zafar, Mohamad Abou Houran, Zakria Qadir, Syed Kumayl Raza Moosavi, Filippo Sanfilippo. (2024). Enhancing cybersecurity in Edge IIoT networks: An asynchronous federated learning approach with a deep hybrid detection model. ScienceDirect Internet of Things.

[4] Thin Tharaphe Thein, Yoshiaki Shiraishi, Masakatu Morii. Personalized federated learning-based intrusion detection system: Poisoning attack and defense. *Future Generation Computer Systems,* ISSN 0167-739X.

[5] M. H. Bhavsar, Y. B. Bekele, K. Roy, J. C. Kelly, and D. Limbrick. "FL-IDS: Federated Learning-Based Intrusion Detection System Using Edge Devices for Transportation IoT," in *IEEE Access,* vol. 12.

[6] J. Zhang, C. Luo, M. Carpenter, and G. Min. "Federated Learning for Distributed IIoT Intrusion Detection Using Transfer Approaches," in *IEEE Transactions on Industrial Informatics,* vol. 19, no. 7, pp. 8159-8169, July 2023.

[7] L. Cui et al. "Security and Privacy-Enhanced Federated Learning for Anomaly Detection in IoT Infrastructures," in *IEEE Transactions on Industrial Informatics,* vol. 18, no. 5, May 2022.

[8] O. Aouedi, K. Piamrat, G. Muller, and K. Singh. "Federated Semisupervised Learning for Attack Detection in Industrial Internet of Things," in *IEEE Transactions on Industrial Informatics,* vol. 19, no. 1, pp. 286-295, Jan. 2023.

[9] Y. Cheng, J. Lu, D. Niyato, B. Lyu, J. Kang, and S. Zhu. "Federated Transfer Learning With Client Selection for Intrusion Detection in Mobile Edge Computing," in *IEEE Communications Letters,* vol. 26, no. 3, pp. 552-556, March 2022.

[10] Man, D., Zeng, F., Yang, W., Yu, M., Lv, J., & Wang, Y. (2024). Intelligent Intrusion Detection Based on Federated Learning for Edge-Assisted Internet of Things. *IEEE Access*, 12, 12345-12356.

[11] I. Bello, B. Zoph, Q. Le, A. Vaswani and J. Shlens, "Attention Augmented Convolutional Networks," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 3285-3294, doi: 10.1109/ICCV.2019.00338.

[12] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras and H. Janicke, "Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning," in *IEEE Access*, vol. 10, pp.

[13] K. Srinivas, G. Sundaramoorthy, and S. S. Ramakrishnan. "Federated learning-based approach for efficient intrusion detection system in IoT networks," in *Future Generation Computer Systems*, vol. 105, pp. 253-265, 2020.

[14] X. Zhang, Y. Wang, Z. Wei, and W. Li. "A lightweight federated learning model for

intrusion detection in industrial IoT networks," in *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 3982-3991, 2022.

[15] B. Mohammadi, M. A. Ganaie, M. H. Anisi, and M. S. Hossain. "Federated learning for intrusion detection in IoT networks: Challenges and opportunities," in *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3321-3330, April 2020.

[16] J. Wang, Y. Wang, Z. Wei, and J. Zhang. "Federated learning for intrusion detection in industrial IoT networks using lightweight models," in *International Journal of Computer Science and Network Security*, vol. 20, no. 9, pp. 34-42, 2020.

[17] McMahan, B. et al. "Communication-Efficient Learning of Deep Networks from Decentralized Data." Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), 2017.