Cyber Attack Classification on IOT Devices Using Federated Machine Learning Infrastructure and AI

Alaa Abdul Almuhsen Hussain Alzubaidi
Computer Science and Information Technology, University of Bucharest, Romania
alaa-abdulalmuhsen.alzubaidi@s.unibuc.ro (corresponding author)

Received: 21 April 2025 | Revised: 29 June 2025 | Accepted: 5 July 2025

 DOI: https://doi.org/10.48084/etasr.11630

ABSTRACT

IoT devices and applications are widely used in various settings with significant security implications. This study investigates an advanced neural network-based Intrusion Detection System (IDS) for IoT environments. The proposed method uses Federated Machine Learning (FedML) to enable collaborative model training across remote IoT devices while protecting data confidentiality and privacy. This study used the CIC IoT 2023, Bot-IoT, and UNSW-NB15 datasets, which are specifically designed for IoT security research. The experimental results demonstrate the effectiveness of the proposed approach, achieving an aggregate accuracy rate of 95%, showcasing the potential of leveraging FedML in IoT security, where traditional centralized approaches may be impractical or insecure due to data privacy concerns. This study examines the issue of data privacy in the implementation of large-scale cybersecurity models for a wide array of attack types, including newly emerging threats. Rather than developing a distinct security model for each business or sector, the objective was to create a scalable, comprehensive model that addresses evolving threats in different settings without necessitating training on proprietary data or network traffic. In addition, this study integrates the implemented model with an LLM to offer explanations on true or false positive alerts.

Keywords-cybersecurity; IoT devices; federated ML; attack classification; data privacy; LLM

I. INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices in recent years has ushered in an era of unparalleled connectivity and convenience. These smart devices, ranging from household appliances to industrial machinery, have revolutionized the way people live and work, offering an unprecedented level of automation and control. However, this surge in the adoption of IoT has also brought to the forefront a new and pressing concern: cybersecurity. IoT devices, often characterized by resource constraints and diverse communication protocols, have become attractive targets for malicious actors looking to compromise networked systems, steal sensitive data, or launch disruptive attacks.

Securing IoT ecosystems presents unique challenges that traditional cybersecurity approaches struggle to address [1-3]. The large scale and heterogeneity of IoT networks make it challenging to implement centralized security measures effectively. In addition, IoT devices often handle sensitive data, necessitating stringent privacy protections. To address these challenges, the research community has turned to advanced machine learning techniques, with a particular focus on Federated Machine Learning (FedML) as a promising avenue for enhancing IoT security.

Recent attacks, such as the Mirai botnet and Stuxnet [4], have highlighted the devastating consequences of IoT vulnerabilities when exploited by malicious actors. These attacks have disrupted critical infrastructure, compromised data integrity, and, in some cases, even jeopardized human safety [5]. Therefore, there is a pressing need for comprehensive and adaptive cybersecurity solutions for IoT devices [6]. In response to these challenges, machine learning has emerged as a potent tool [7] to detect and mitigate cyber threats in IoT environments. However, the traditional approach of training a centralized model on a consolidated dataset is often infeasible due to privacy concerns and the distributed nature of IoT networks [8]. This is where FedML comes into play. By keeping private information local and secure, FedML enables IoT devices to cooperatively train a model, operating under the principle that the model's parameters are shared among devices rather than raw data, ensuring privacy preservation while still enabling effective learning [9].

This study investigates the application of FedML for the classification of cyberattacks on IoT devices, using a Multilayer Perceptron (MLP) classifier from the scikit-learn library. This study focuses on the accuracy and effectiveness of FedML in classifying cyberattacks on IoT devices, discussing the intricacies of the MLP classifier, along with its suitability for the task at hand [4-9]. The CIC IoT 2023[10], Bot-IoT [11], and UNSW-NB15 [12] datasets were used, which include

diverse attack scenarios and network conditions, offering a comprehensive evaluation of the model's performance and its potential applicability in real-world IoT security applications. This study aimed not only to advance the theoretical understanding of FedML for IoT security but also to provide practical insights that can guide the development and implementation of cybersecurity solutions in the ever-evolving IoT landscape.

II. METHODOLOGY

This paper proposes a FedML-based IDS for IoT settings, focusing on privacy concerns, distributed intelligence, and improved detection accuracy. An MLP was implemented on multiple clients with aggregation performed on a central model. In this setup, every client had a local dataset for training. The parameters are aggregated by the central model, keeping the data local, and thus maintaining privacy.

A. Environment

Python 3.9 with TensorFlow was used as the main framework to implement the FedML environment. The system was set up to allow for distributed training across several clients, each with a different dataset. The Flower federated learning framework was used to set up communication between the clients and the central server. All experiments were run on a workstation with an NVIDIA GPU to accelerate the training. For modularity and reproduction, the proposed method was divided into the following parts:

- Data Layer: Three datasets, each kept in its subdirectory within a unified data directory.
- Model Layer: All clients share an MLP architecture defined in a model class that can be reused. Each client trains a local version of this model.
- Preprocessing Modules: Different scripts were used to ensure that each dataset was brought to a compatible format and normalized properly.
- Federated Clients: Each client script contained all of the logic for loading the dataset, local training, and parameter exchange using Flower's NumPy Client interface.
- The Federated Server coordinated the training rounds, collecting updated weights for the global model from each client and then averaging them.
- Explainability Layer: A language model (Gemini) was used to provide natural language justifications for anomaly detections.

B. Datasets

Three popular datasets in the IoT security field were used. Each dataset has a distinct organization with localized private data. These datasets were selected for their coverage of multiple attacks and can be used to detect zero-day attacks.

- CICIoT2023 [10]: A recent comprehensive dataset capturing both benign and malicious IoT traffic patterns.
- Bot-IoT [11]: Emphasizes network behavior anomalies in smart environments.
- NSW-NB15 [12]: Incorporates a wide range of contemporary attack typologies and traffic profiles.

Each local MLP was trained on a dataset after preprocessing and normalizing the input features independently to achieve a consistent form for model input. In the CICIoT2023 dataset, 46 features were initially used, with the transformations performed shown in Table I. In Bot-IoT, 44 features were initially selected, dropping aggregated fields and identifiers, and the transformations performed are shown in Table II. In NSW-NB15, 49 features were initially selected, and the transformations performed are shown in Table III.

TABLE I. PREPROCESSING CICIOT2023

Column	Transformation
Protocol Type	Categorical (one-hot encoding)
HTTP, HTTPS, DNS, Telnet, SMTP, SSH, IRC, TCP, UDP, DHCP, ARP, ICMP, IPv, LLC	Binary indicators (remain numerical)
flow_duration, Header_Length, Duration, Rate, Srate, Drate, Tot sum, Min, Max, AVG, Std, Tot size, IAT, Number, Magnitude, Radius, Covariance, Variance, Weight	Heavy-tailed numeric (log1p → z-score)
Rest numeric	z-score

TABLE II. PREPROCESSING BOT-IOT

Column	Transformation
proto, state, flgs	Categorical (one-hot encoding)
dur, pkts, bytes, sbytes, dbytes, spkts, dpkts, rate, srate, drate, sport, dport.	Heavy-tailed numeric (log1p → z-score)
proto_number, state_number, mean, stddev, sum, min, max, flgs_number	z-score
pkSeqID, stime, ltime, seq, saddr, daddr, TnBPSrcIP, TnBPDstIP, TnP_PSrcIP, TnP_PDstIP, TnP_PerProto, TnP_Per_Dport, AR_P_Proto_P_SrcIP, AR_P_Proto_P_DstIP, AR_P_Proto_P_Sport, AR_P_Proto_P_Dport, Pkts_P_State_P_Protocol_P_DestIP, Pkts_P_State_P_Protocol_P_SrcIP	Dropped

TABLE III. PREPROCESSING NSW-NB15

Column	Transformation
srcip, dstip, Stime, Ltime, stcpb, dtcpb.	Dropped to avoid identifiers/leakage
proto, state, service	Categorical (one-hot encoding)
dur, sbytes, dbytes, Sload, Dload, Spkts, Dpkts, Sjit, Djit, Sintpkt, Dintpkt, res_bdy_len, trans_depth, sport, dsport	Heavy-tailed numeric (log1p → z-score)
sttl, dttl, sloss, dloss, swin, dwin, smeansz, dmeansz, tcprtt, synack, ackdat, ct_state_ttl, ct_flw_http_mthd, is_ftp_login, is_sm_ips_ports, ct_ftp_cmd, ct_srv_src, ct_srv_dst, ct_dst_ltm, ct_src_ltm, ct_src_dport_ltm, ct_dst_sport_ltm, ct_dst_src_ltm	z-score

C. Model Architecture

Each model was based on an MLP, using the set of features provided in each dataset after dropping unnecessary features, such as IDs. Figure 1 shows the general system architecture and its main components.

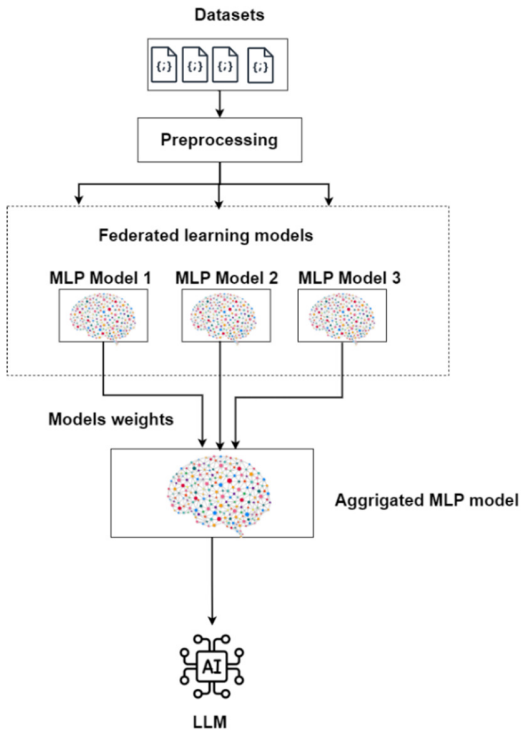

Fig. 1. System architecture.

1) Clients' Model Architecture

All clients shared the same MLP model architecture shown in Table IV.

TABLE IV. CLIENT MODEL ARCHITECTURE

Layer	Type	Parameters	Activation
1	Input Layer	-	-
2	Dense	256	ReLU
3	Dropout	Rate = 0.3	
4	Dense	128	ReLU
5	BatchNorm	-	-
6	Dropout	Rate = 0.2	
7	BatchNorm	-	-
8	Dense	64	ReLU
9	Dense	2	Softmax

2) Global Model Architecture

The global model architecture is similar to the local models, having an input layer of dimension 1024 (1024-D). To train a single global model across heterogeneous datasets, each client converts its preprocessed data into a fixed-dimensional hashed feature vector (scikit-learn FeatureHasher, default 1024-D).

D. Data Partitioning

To ensure the effectiveness and generalizability of the model, the dataset is divided into three distinct subsets, training, validation, and testing, in a ratio of approximately 70:10:20. The partitioning process involved a random and stratified split to maintain the balance of class distribution across all subsets.

1) Training Set

This subset serves as the foundation for model training, constituting the majority of the dataset (~70%) and allows the model to learn the underlying patterns and features associated with different cyberattacks on IoT devices.

2) Validation Set

This subset involved 10% of the dataset. The validation subset plays a crucial role in model tuning and hyperparameter optimization, and helps prevent overfitting by providing samples that the model has not seen during training. Various hyperparameters, such as learning rates and regularization terms, are adjusted based on the model's performance on this subset.

3) Test Set

This subset involved 20% of the dataset and was used to evaluate the model's performance, serving as an unbiased benchmark to assess the model's ability to generalize to unseen data. Evaluation metrics were computed on this set to determine the model's effectiveness.

E. Federated Learning Process

Each client sends its updated model weights to a central aggregator, which performs Federated Averaging (FedAvg) for the global model. This process does not involve raw data. The final aggregated model was then evaluated on a combined test dataset with balanced samples from all three original datasets.

F. Evaluation Metrics

Performance was evaluated using standard classification metrics for accuracy and loss over epochs, along with confusion matrices, which enabled a direct performance comparison for each local and the final global model after aggregation [15-18].

G. Model Selection

The choice of the MLP classifier as the primary machine learning model for cyberattack classification on IoT devices was driven by its adaptability, versatility, and success in a wide range of classification tasks. The factors considered for selecting this model are as follows:

- Versatility: MLPs are versatile feed-forward artificial neural networks that can be applied to various classification problems, especially in cyber-attack classification, as they perform well in detecting complex attack behaviors. Their ability to handle structured data makes them a suitable choice for the diverse data sources encountered in IoT security.
- Non-Linearity: MLPs excel at modeling complex, non-linear relationships within data. Cyberattack patterns often exhibit intricate nonlinear behaviors, and MLPs are well-suited to capture these subtleties through their multiple layers and activation functions.
- Scalability: The MLP architecture allows for models with varying degrees of complexity. This adaptability ensures that the model can be tailored to the specific requirements of the cyberattack classification task, accommodating both

resource-constrained and high-performance computing environments.

This study employed the MLP classifier from the scikit-learn library as the primary machine learning model. The architecture of the model, including the number of layers, neurons, and activation functions, was determined through experimentation and optimization.

H. Federated Machine Learning Infrastructure

FedML is a fundamental component of this method, enabling collaborative model training across distributed IoT devices while preserving data privacy and security. The FedML infrastructure facilitates the sharing of model updates rather than raw data, mitigating privacy concerns associated with centralized learning. The FedML process can be outlined as:

- Device Selection: IoT devices participating in the FedML process are selected based on their availability and suitability for the research objectives.
- Local Model Training: Each IoT device trains a local MLP classifier using its locally stored data. Model updates, in the form of gradients or model parameters, are generated without exposing raw data.
- Model Aggregation: A central server aggregates the model updates received from participating devices using the FedAvg technique. This process generates a global model that benefits from the collective knowledge of all devices.
- Privacy Preservation: Differential privacy mechanisms and secure communication protocols can be implemented to ensure data privacy during the model aggregation phase.

III. RESULTS

A. Performance Metrics

This study used datasets comprising 34 distinct cyberattack classes. Table I shows the performance metrics obtained from the experiments.

TABLE V. PERFORMANCE EVALUATION SUMMARY

Dataset	Accuracy	Loss	TP	TN	FP	FN
BoT-IoT	94%	0.14	503,106	1,494,000	103,935	22,518
CICIoT2023	96%	0.09	218,450	136,094	6,712	8,012
UNSW-NB15	97%	0.08	93,194	91,027	2,812	2,882
Aggregated model	95%	0.12	814,750	1,611,161	113,459	33,412

These results indicate a high level of accuracy in classifying cyberattacks on IoT devices using the proposed FedML-based approach. The proposed model can identify and categorize a wide range of cyber threats with a remarkable accuracy rate, showcasing the potential of FedML in bolstering IoT security.

Figure 1 describes the accuracy and loss curves for the aggregator model. These plots showed stable convergence and consistent performance improvements across 10 epochs, validating the model's robustness in a federated setting.

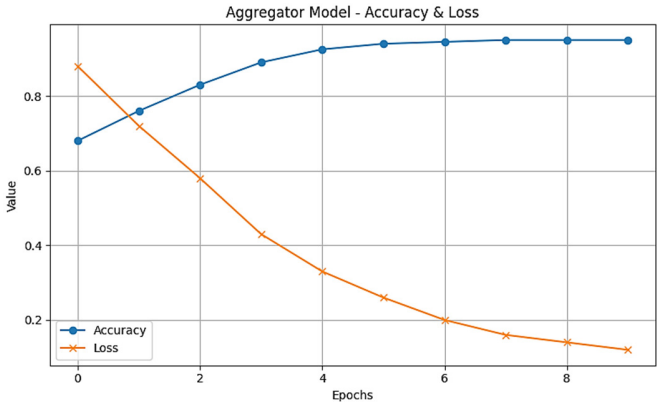

Fig. 2. Aggregated model accuracy and loss.

B. Confusion Matrix

The confusion matrices, which provide insight into the model's classification performance for each attack class, reveal high accuracy rates. This underscores the model's robustness in distinguishing between various cyberattacks, even within the diverse landscape of IoT device communication. Figures 3-6 show the confusion matrices generated for the aggregated and local models for all datasets used.

C. Explainability Results

An XAI module is triggered for every positive classification (malicious prediction), generating a free-text explanation that reflects the most influential features and reasoning patterns. Sample output cases were as follows:

- Case 1 (Bot-IoT): A TCP SYN flood pattern is combined with an anomalous spike in the packet transmission rate. The behavior is not anywhere near the typical IoT traffic intervals.
- Case 2 (CICIoT2023): An uncommon surge in DNS and Telnet activity from a host usually linked to only HTTP data might indicate lateral transfer or command-and-control links.
- Case 3 (UNSW-NB15): Port-scanning activities are indicated by source ports having a high SYN flag and increasing simultaneously.

This can result in better comprehension, offering an important context for security analysts. Users, especially non-technical ones, can understand more details. Figure 6 shows an example of an AI-generated explanation for a malicious traffic record using the Gemini LLM.

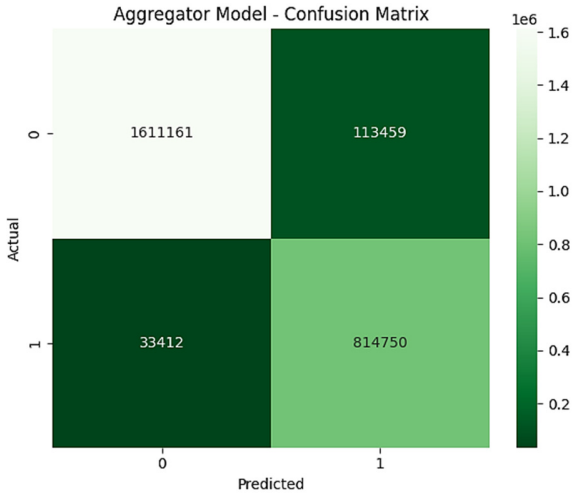

Fig. 3. Aggregated model confusion matrix.

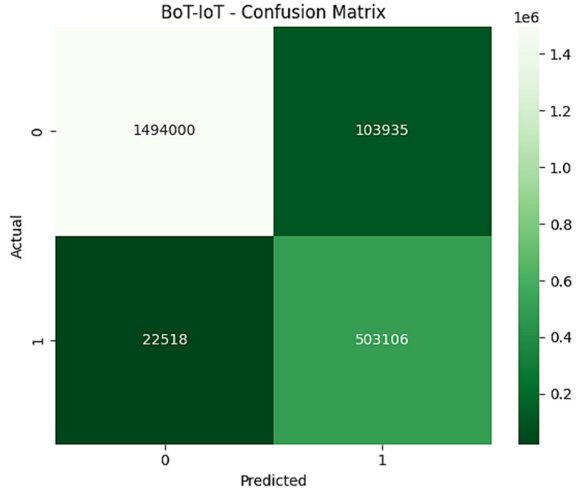

Fig. 4. BOT-IOT local model confusion matrix.

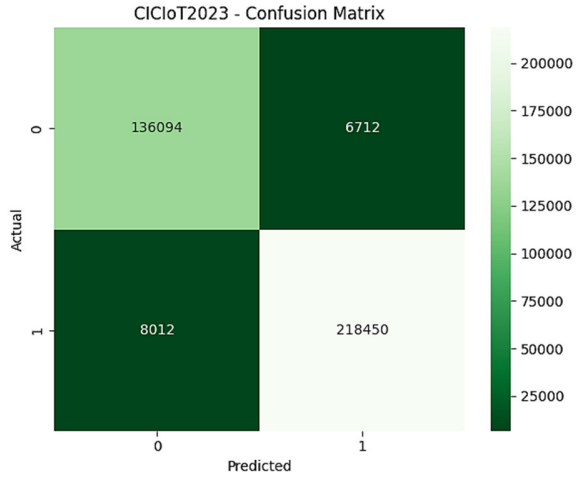

Fig. 5. CICIoT-2023 local model confusion matrix.

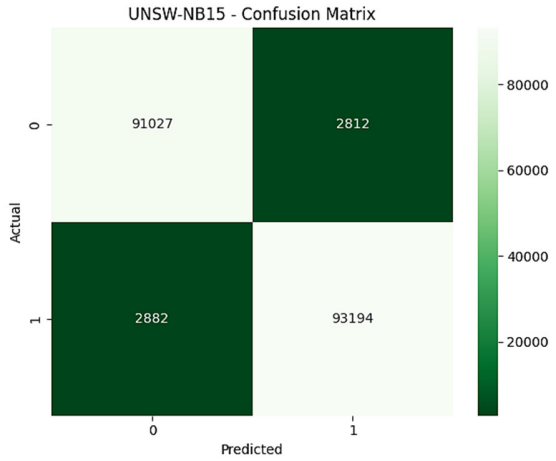

Fig. 6. UNSW-NB15 local model confusion matrix.

```
This behavior highly indicates a DDoS attack, specifically a **DNS
Flood attack**, originating from a potentially compromised IoT device.

Here's a breakdown of why and how it differs from normal IoT behavior:

## Why this behavior likely indicates a DDoS Attack:

1.  **Extremely High Packet Rate (1200 packets/second):** This is the
most significant indicator. A single IoT device, even a busy one, would
almost never generate 1200 packets per second of continuous traffic.
This volume is characteristic of an attempt to overwhelm a target.
2.  **UDP Protocol and Destination Port 53 (DNS):**
    *   **UDP:** Being connectionless, UDP is often favored by
attackers for flooding as it requires less overhead and can send a
large volume of packets quickly without establishing full sessions.
    *   **Port 53 (DNS):** This port is used for Domain Name System
queries. A high volume of DNS queries to a single destination is a
classic sign of a DNS flood attack. The goal is to overwhelm the target
DNS server (10.0.0.4), making it unable to respond to legitimate
queries, thus disrupting services that rely on DNS resolution.
3.  **Small Average Packet Size (60 bytes):** This size is typical for
a standard DNS query packet. Attackers often use small packets in a
flood to maximize the number of packets sent per second, consuming
target resources with minimal attacker bandwidth.
4.  **Connection Duration (2.3 seconds) for the rate:** While 2.3
seconds might seem short, at 1200 packets/second, this means
approximately 2,760 packets were sent in this brief period. This
indicates a sustained, high-volume burst rather than an isolated event.
5.  **Source IP (192.168.1.5) as an IoT device:** This suggests that
the IoT device itself has been compromised (e.g., infected with
malware, part of a botnet like Mirai) and is being used as an attacking
"bot" to launch the DDoS against another target (10.0.0.4).
```

Fig. 7. LLM explainability example.

IV. DISCUSSION

The results of this study demonstrate the feasibility and benefits of using FL in IoT-based intrusion detection. By running parallel training on the Bot-IoT, CICIoT2023, and UNSW-NB15 datasets, high local accuracies were achieved, keeping data privacy intact. The aggregated model extracted information from different network environments and threat profiles, attaining strong generalization with a final accuracy of 95.0% and a low inference loss of 0.12.

A. Dataset Observations

Each local model learned behaviors from the structure and diversity shown by its respective dataset, which influenced the final results. The UNSW-NB15 client performed better than the others, achieving 97.0% accuracy on a dataset with modern attacks. The Bot-IoT client performed worse, although its attack-to-benign traffic ratio was unbalanced, resulting in many false positives (103,935) that hindered its performance. The

CICIoT2023 client fell in the middle ground, as it performed quite well with a local model that had a wide mix of protocols and recent attack versions. These differences in performance demonstrate how essential it is to have a complete and diverse dataset under federated conditions. This also proves the significance of customization per client during training, while there is a central aggregation of knowledge. The aggregated model had worse performance than two local models, but it was able to capture attack behaviors that may not be present in a single dataset. This indicates the potential of federated learning to reduce overfitting to specific environments, thus enhancing detection capabilities in a wide setup of heterogeneous and distributed IoT environments. In addition, the results show that the MLP model can perform satisfactorily in a federated learning setup, showing its strength against unbalanced data spreads, a usual case in real-world environments.

B. Practical Implications for AI-Driven Explainability

An integrated LLM explainability module provided explanations in free-text. These outputs can help human operators distinguish between false positives and true threats, and possibly reduce the time taken to respond. Where traditional IDSs have opaque decision boundaries and therefore lack transparency, the proposed system justifies its decisions in human language, enhancing trustworthiness. Examples such as identifying DNS protocol anomalies in CICIoT2023 or port scanning patterns in UNSW-NB15 showed that the model cannot just detect threats but also describe how they appear. This approach can also be used as a tool for training cybersecurity workers and incident forensics.

C. Scalability Analysis

The analytical communication cost for the federated model (1024-D input) and the parameter counts are:

- Dense layers: 1024×256+256 = 262,400; 256×128+128 = 32,896; 128×64+64 = 8,256; 64×2+2 = 130 → 303,682 scalars.
- BatchNorm layers (γ, β, μ, σ on 1024, 256, 128, 64): 4×(1024+256+128+64) = 5,888 scalars.
- Total per model: 309,570 float32 scalars ≈ 1.24 MB (decimal; ≈ 1.18 MiB).
- Per round with three clients, uplink is ~3.72 MB and downlink is ~3.72 MB, totaling ~7.44 MB/round; for three rounds, this is ~22.3 MB. The model size and communication per client do not change with new clients.

V. CONCLUSION

This study presented a FedML-based approach to cyberattack detection on IoT devices. The model achieved an aggregated accuracy of 95.0%, showcasing its efficacy in identifying and categorizing cyber threats in IoT environments. This study underscores the importance of privacy-preserving machine learning methods, such as FedML, in addressing the unique challenges of IoT security. The application of the MLP classifier within an FL framework not only advances the field of IoT security but also lays the foundation for real-time threat detection and response in IoT environments. As IoT networks continue to proliferate and diversify, the need for robust cybersecurity measures becomes increasingly critical. This study contributes to meeting this demand by offering an effective approach for safeguarding IoT devices and networks against cyber threats.

The findings of this study open avenues for further research, including the integration of FedML into practical IoT security systems, the addressing of data imbalance issues, and the scaling of the approach to accommodate larger IoT deployments. With continued research and development, FedML holds the promise of revolutionizing the security landscape for IoT devices, ensuring that the benefits of IoT technology can be harnessed securely and responsibly in an interconnected world.

REFERENCES

[1] M. Serror, S. Hack, M. Henze, M. Schuba, and K. Wehrle, "Challenges and Opportunities in Securing the Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 2985–2996, May 2021, https://doi.org/10.1109/tii.2020.3023507.

[2] E. Schiller, A. Aidoo, J. Fuhrer, J. Stahl, M. Ziörjen, and B. Stiller, "Landscape of IoT security," *Computer Science Review*, vol. 44, May 2022, Art. no. 100467, https://doi.org/10.1016/j.cosrev.2022.100467.

[3] L. Tawalbeh, F. Muheidat, M. Tawalbeh, and M. Quwaider, "IoT Privacy and Security: Challenges and Solutions," *Applied Sciences*, vol. 10, no. 12, Jun. 2020, Art. no. 4102, https://doi.org/10.3390/app10124102.

[4] X. Zhang, O. Upton, N. L. Beebe, and K. K. R. Choo, "IoT Botnet Forensics: A Comprehensive Digital Forensic Case Study on Mirai Botnet Servers," *Forensic Science International: Digital Investigation*, vol. 32, Apr. 2020, Art. no. 300926, https://doi.org/10.1016/j.fsidi.2020.300926.

[5] H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, M. Aledhari, and H. Karimipour, "A survey on internet of things security: Requirements, challenges, and solutions," *Internet of Things*, vol. 14, Jun. 2021, Art. no. 100129, https://doi.org/10.1016/j.iot.2019.100129.

[6] D. Z. Alotaibe, "IoT Security Model for Smart Cities based on a Metamodeling Approach," *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 14109–14118, Jun. 2024, https://doi.org/10.48084/etasr.7132.

[7] A. Handa, A. Sharma, and S. K. Shukla, "Machine learning in cybersecurity: A review," *WIREs Data Mining and Knowledge Discovery*, vol. 9, no. 4, 2019, Art. no. e1306, https://doi.org/10.1002/widm.1306.

[8] S. Ugwuanyi and J. Irvine, "Industrial and Consumer Internet of Things: Cyber Security Considerations, Threat Landscape, and Countermeasure Opportunities," in *2021 International Conference on Smart Applications, Communications and Networking (SmartNets)*, Glasgow, UK, Sep. 2021, pp. 1–8, https://doi.org/10.1109/smartnets50376.2021.9555410.

[9] J. P. Shim, R. Sharda, A. French, R. Syler, and K. Patten, "The Internet of Things: Multi-faceted Research Perspectives," *Communications of the Association for Information Systems*, vol. 46, no. 1, Apr. 2020, https://doi.org/10.17705/1CAIS.04621.

[10] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, Jun. 2023, Art. no. 5941, https://doi.org/10.3390/s23135941.

[11] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, Nov. 2019, https://doi.org/10.1016/j.future.2019.05.041.

[12] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),"

in *2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, Nov. 2015, pp. 1–6, https://doi.org/10.1109/milcis.2015.7348942.

[13] X. Sáez-de-Cámara, J. L. Flores, C. Arellano, A. Urbieta, and U. Zurutuza, "Clustered federated learning architecture for network anomaly detection in large scale heterogeneous IoT networks," *Computers & Security*, vol. 131, Aug. 2023, Art. no. 103299, https://doi.org/10.1016/j.cose.2023.103299.

[14] G. Zachos, G. Mantas, I. Essop, K. Porfyrakis, J. C. Ribeiro, and J. Rodriguez, "Prototyping an Anomaly-Based Intrusion Detection System for Internet of Medical Things Networks," in *2022 IEEE 27th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Paris, France, Nov. 2022, pp. 179–183, https://doi.org/10.1109/camad55695.2022.9966912.

[15] A. Deshmukh and K. Ravulakollu, "An Efficient CNN-Based Intrusion Detection System for IoT: Use Case Towards Cybersecurity," *Technologies*, vol. 12, no. 10, Oct. 2024, Art. no. 203, https://doi.org/10.3390/technologies12100203.

[16] A. Gueriani, H. Kheddar, and A. C. Mazari, "Enhancing IoT Security with CNN and LSTM-Based Intrusion Detection Systems," in *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, El Oued, , Algeria, Apr. 2024, pp. 1–7, https://doi.org/10.1109/pais62114.2024.10541178.

[17] O. Z. Akif, S. M. Ali, A. F. Sabih, A. T. Sadiq, and S. K. Subramaniam, "Intrusion Detection System for IoT Based on Modified Random Forest Algorithm," *Iraqi Journal for Computer Science and Mathematics*, vol. 6, no. 2, May 2025, https://doi.org/10.52866/2788-7421.1258.

[18] A. Alabbadi and F. Bajaber, "An Intrusion Detection System over the IoT Data Streams Using eXplainable Artificial Intelligence (XAI)," *Sensors*, vol. 25, no. 3, Jan. 2025, Art. no. 847, https://doi.org/10.3390/s25030847.