Archives available at journals.mriindia.com

Open Access International Journal of Science and Engineering

ISSN: 2456-3293

Volume 9 Issue 05, 2026

Privacy-Preserving Federated Learning Framework for Intelligent Cyber Threat Detection in Distributed Networks

[1]Rekha Bajirao Bankar, [2]Anupama Shankarrao Budhewar

[1,2]Department of Computer Science Engineering, JSPM University Pune, School of Computational Science, Wagholi, Pune, Maharashtra, India – 412207

[1]rekhasuresh29112@gmail.com, [2]budhewar.anupama@gmail.com

Peer Review Information

Submission: 30 April 2026
Revision: 09 May 2026
Acceptance: 26 May 2026

Keywords

Federated Learning, Intrusion Detection System, Cyber Threat Detection, Privacy Preservation.

Abstract

Cyber threats keep going up in modern distributed network environments, so yeah, classic intrusion detection systems are, kind of, less effective when attacks keep changing. A lot of machine learning security methods depend on centralized data gathering, and that ends up raising real worries about privacy, confidentiality, and data security, especially when there is sensitive traffic involved. So in order to deal with these issues, this paper puts forward a privacy-preserving federated learning setup for intelligent cyber threat detection across distributed networks. With the proposed idea, several client nodes can work together to train one global intrusion detection model, but they don't need to share raw network data between them, which is the main point.

Machine learning techniques are then used with benchmark intrusion detection datasets, for efficiently separating normal from malicious network traffic. On top of that, feature preprocessing along with model optimization are included, to boost detection performance and also trim down computational complexity. Overall, the federated learning architecture improves data privacy, scalability, and collective intelligence, while still keeping a high detection accuracy. And finally, the framework offers a secure and practical path for real world cybersecurity work, like IoT networks, enterprise infrastructure, and cloud based distributed systems.

Introduction

The fast advancement of digital technologies cloud computing, Internet of Things (IoT) and distributed communication systems has really reshaped modern network infrastructures. In healthcare, banking, education, smart industries, and government, organizations more and more rely on linked services for data exchange and everyday delivery. But then again this increasing connectedness also boosts the frequency and the overall complexity of cyber threats like malware attacks, phishing , ransomware, denial-of-service attacks, and even unauthorized network intrusions [1]. And since cyberattacks keep evolving, guarding network infrastructures and protecting sensitive organizational data is now a major hurdle, for both cybersecurity researchers and the people practicing it.

Intrusion Detection Systems are widely used to keep an eye on network traffic and to spot suspicious actions inside communication environments. Traditional IDS approaches mostly lean on signature based and rule driven detection methods, where network activity is checked against already defined attack patterns [2]. Even if these systems work pretty well for threats that are already known, they can fall short when it comes to zero-day attacks and

those more elaborate, continuously evolving cyber threats. Because of that, machine learning and deep learning have been slipped into intrusion detection frameworks, since they can automatically pick up traffic patterns and sort out malicious behavior with stronger accuracy [3].

Machine learning based intrusion detection systems usually lean on centralized learning setups, where huge amounts of network traffic are gathered from many devices or organizations, and later stored on one server for training. Even if these approaches still reach strong detection results, they tend to bring in a bunch of privacy and security worries. For example, sensitive traffic logs can carry private organizational traces, user behavior fingerprints, and even communication records , which basically cannot be shared freely because of privacy rules and security policies [4]. On top of that centralized storage behaves like a single chokepoint, so when something goes wrong, the whole thing suffers, and the chance of data breaches, insider misuse, or unauthorized access goes up. These issues get even more intense in distributed settings like IoT networks, cloud infrastructures, smart cities, and industrial cyber physical systems, where massive volumes of assorted data are continuously produced [5].

To overcome the drawbacks of the centralized learning system, Federated Learning (FL) has been proposed as another distributed machine learning system. As opposed to the conventional machine learning methods, in Federated Learning, many clients or devices train a global model together without data being moved to a single central point [6]. In this method, each client will conduct local training with its own dataset and only send the parameters of models or weight updates to the aggregation server in the central system. The server uses aggregation algorithms like Federated Averaging (FedAvg) to create a global model from these updates. This distributed learning technique preserves data personal privacy and allows for collaborative intelligence of many distributed participants [7].

In recent years, there has been a surge of research on using Federated Learning in cybersecurity and intrusion detection. It has been proved that the IDS framework based on FL can be used to achieve the accuracy of IDSs of centralized systems, without compromising data confidentiality [8]. Furthermore, Federated Learning enhances scalability and enables distributed learning across geographically different organizations, making it well suited for real-world cybersecurity applications. Some works have combined machine learning algorithms like Random Forest, Decision Tree, Support Vector Machine, and Multi-Layer Perceptron (MLP) in federated system to classify network traffic successfully [9].

Even though Federated Learning-based IDSs have their benefits, there are still some practical problems. The most important problem is that the data at various client nodes is non-identically distributed (Non-IID). In real-world settings, each organization has its own unique traffic pattern and attack distribution, which may affect the performance of global model convergence and detection. [10] Another key issue is the communication overhead when the model parameters are transmitted repeatedly from the central server to clients. The problem is important in situations where there are lots of devices with poor network bandwidth and computational capabilities in large IoT systems [11].

Security concerns also are of huge importance in federated systems. It is possible for malicious clients to purposefully push malformed or corrupted models to affect the performance of the global model and thus affect the accuracy of threat detection [12]. Moreover, privacy leakage attacks like model inversion and gradient leakage may be able to reveal sensitive information in shared model updates. Hence, the incorporation of strong aggregation methods, secure communication channels, and privacy protection methods is crucial for enhancing the reliability of FL-enabled cybersecurity systems [13].

A Federated Learning framework for intelligent cyber threat detection in distributed networks, while preserving privacy is proposed. The framework provides a way to share information about the intrusions while preserving the confidentiality of any sensitive network traffic data. The federated architecture incorporates machine learning techniques in order to efficiently classify the network traffic into normal and malicious traffic. The planned system is based on the distributed local training and aggregation of models at the central level, which helps to enhance cybersecurity intelligence without compromising data confidentiality and also reduces the risks associated with centralized storage.

The proposed framework can be applied in real-world distributed systems including enterprise systems, IoT network, smart infrastructure, and platforms. Benchmark intrusion detection datasets are used to compare the performance of the proposed model based on its accuracy, precision, recall, F1 score and detection capability. A combination of Federated Learning and intelligent IDS mechanisms is a scalable,

secure and privacy-preserving approach for current cyber security applications [15].

Literature Review

In recent years, innovative developments in cybersecurity have prompted the exploration of combining Federated Learning (FL) with Intrusion Detection Systems (IDS) for enhanced privacy and distributed threat detection.In recent years, innovative developments in cybersecurity have encouraged researchers to look into combining Federated Learning (FL) with Intrusion Detection Systems (IDS) for better privacy and distributed threat detection. Traditional machine learning-based IDS approaches are based on centralized datasets, thereby causing privacy and security issues. To address these disadvantages, a few studies suggested FL frameworks that simultaneously train models without sharing raw network traffic information among multiple clients. Mahmud et al. [4] proposed a privacy-preserving FL-based IDS for IoT networks and Wang et al. [6] used homomorphic encryption and gradient similarity to protect model updates. These approaches have shown to achieve better intrusion detection ability, with the simultaneous preservation of the user data privacy in distributed environments.

Some studies on lightweight and scalable FL architectures for IoT and industrial cyber threats detection have also been conducted. Lazzarini et al. [7] analyzed the practical implementation of FL for IoT IDS and pointed out that FL is effective in a collaborative environment. Likewise, Sharma et al. [14] suggested an IDS model for IoT networks based on FL which demonstrated the high detection accuracy through the distributed learning approach. But current works still have some problems like communication overhead and non-IID data distribution and are still vulnerable to poisoning attacks. Recent studies highlight the importance of developing effective aggregation techniques, privacy-preserving approaches, and model optimization algorithms that enhance the reliability and scalability of FL-based IDS systems in practical cybersecurity deployments.In practical cybersecurity deployments, it is crucial to design effective aggregation techniques, privacy-preserving mechanisms, and efficient model optimization strategies to ensure the reliability and scalability of FL-based IDS systems in real-world settings.

Table 1: Comparative Analysis of Existing Federated Learning-Based Intrusion Detection Systems

Sr. No.	Author(s) & Year	Title	Technique Used	Key Contribution	Limitation
1	I. B. Ababio *et al.* (2025)	Blockchain-Assisted Federated Learning Framework for Trustworthy Malware Analysis in IIoT	FL + Blockchain + FedAvg	Improved trust and secure model sharing in IIoT environments	Increased computation and synchronization overhead
2	M. Rahmati (2025)	Federated Learning-Driven Cybersecurity Framework for IoT Networks	FL + Homomorphic Encryption + RNN	Achieved secure and privacy-preserving intrusion detection	Scalability in real-time deployment not fully tested
3	B. Buyuktanir *et al.* (2025)	Federated Learning in Intrusion Detection: Advancements and Challenges	Federated IDS Survey	Identified major challenges and future research directions	Lacks implementation-based evaluation
4	S. A. Mahmud *et al.* (2024)	Privacy-Preserving Federated Learning-Based Intrusion Detection in IoT	FL + Differential Privacy	Enhanced privacy protection in distributed IDS systems	Differential privacy reduced accuracy slightly

5	N. Sun *et al.* (2024)	Blockchain-Based Federated Learning for Intrusion Detection Systems	FL + Blockchain Consensus	Prevented insider attacks and improved decentralized security	High communication and storage overhead
6	J. M. Wang *et al.* (2024)	NIDS-FGPA: Federated Intrusion Detection Using HE	FL + Homomorphic Encryption	Reduced malicious update influence during aggregation	Heavy computational requirements
7	R. Lazzarini *et al.* (2023)	Federated Learning for IoT Intrusion Detection	Comparative FL Models	Demonstrated feasibility of FL in IoT IDS environments	Limited evaluation in real-world scenarios
8	S. K. Das *et al.* (2023)	Federated Learning Assisted IoT Malware Detection Using Random Forest	FL + Random Forest	Achieved effective malware detection using lightweight models	Did not address poisoning attacks or privacy leakage

Research Methodology

The proposed research methodology is kind of centered on building a privacy-preserving Federated Learning based Intrusion Detection System IDS for those distributed network settings. It mixes machine learning tricks with a more decentralized federated setup, so cyber threats can be spotted without actually sharing sensitive network traffic data. At first, the chosen intrusion detection dataset gets preprocessed through a mix of data cleaning, normalization, and encoding in order to make the data quality better and also help model efficiency. After that, the prepared dataset is split up and sent across several client nodes, so it can feel like a true federated environment[16].

Then, each client node trains a local model on its own private data, independently, and they don't send raw records back. Instead, only the model parameters are shared with a central aggregation server which just collects updates and moves things along. The overall, global model is produced using the Federated Averaging (FedAvg) aggregation algorithm, where local updates from different clients get combined together. To make detection practical, machine learning classifiers are used to recognize malicious and benign traffic patterns quite efficiently[17].

Finally, performance evaluation is done using metrics like accuracy precision recall, F1-score, and confusion matrix analysis, maybe to check how well normal versus attack types are handled. In short, this methodology supports data privacy, lowers the dangers of centralized storage, and it enhances cooperation for cyber threat detection across distributed environments such as IoT networks, cloud infrastructures, and enterprise systems[18].

1. Federated Learning (FL)

Federated Learning (FL) is a decentralized form of machine learning where multiple client devices or organizations can work together to train a unified global model without sharing raw data. In the conventional centralized ML setups, the data from all network traffic is centralized and stored on a central server for training models. But it opens up significant privacy, security and data confidentiality issues. To address this, Federated Learning enables local training on each client node while only transmitting model parameters, or weight updates, to a central aggregation server.

The figure exemplifies the structure of the Federated Learning based Intrusion Detection System (IDS) for privacy-preserving cyber threat detection. In this case, several client nodes train local machine learning models with their private network traffic. Only the updates of the models are sent to the central aggregation server, not the raw data. These updates are then merged by the server to create an enhanced global model and re-sent to all clients. In this way, data privacy, scalability, collaborative learning, and intrusion detection in distributed network environments are improved.

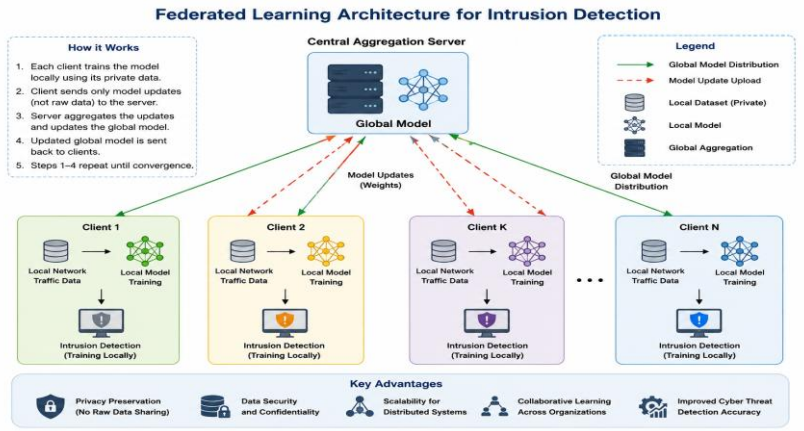

Figure 1: Federated Learning Architecture for Intrusion Detection

The training of the model is done independently by each client node and with its private network traffic. Once the local training is complete, the learning model parameters are sent to the global server rather than the sensitive raw data. The server then merges all the updates from clients to create an enhanced global model. This process is repeated an infinite number of times for a number of rounds of communication until the model attains a stable performance[19].

Some of the benefits of Federated Learning include better privacy protection, fewer risks of centralized storage, better scalability, and collaborative cyber threat detection in distributed environments. It is suitable for applications such as enterprise cybersystems, healthcare systems, cloud systems, and IoT networks where sensitive data sharing is constrained. Also, FL enhances security intelligence by learning attack patterns from various distributed sources while protecting security data confidentiality[20].

2. Federated Averaging (FedAvg)

Federated Averaging, or FedAvg, is a commonly used aggregation approach in Federated Learning which mixes the model updates coming from several client devices to make one global model. In the system we propose, each client trains the intrusion detection model on its own, using a local dataset, and it sends only the learned weight values to the central server. Then the server takes those received parameters, computes a weighted average over them and uses that result to refresh the global model. After that the newly updated model gets sent back out to every participating client, so they can start the next training round. Overall FedAvg helps the whole collaborative learning process run more efficiently, cuts down communication cost, and keeps data privacy safer since raw data is never directly shared between the distributed network nodes.

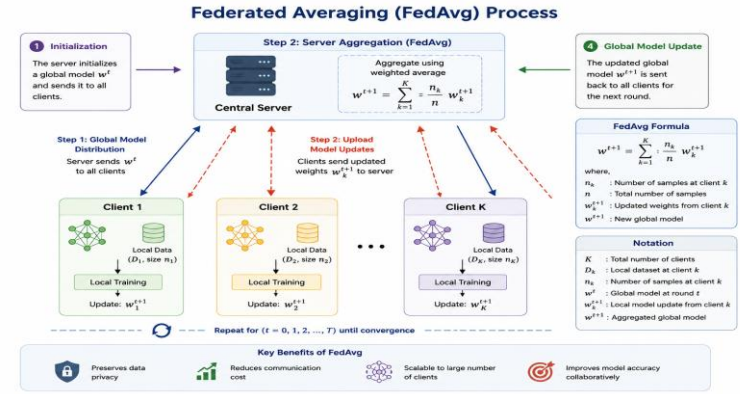

Figure 2: Federated Averaging (FedAvg) Process

This figure shows how the Federated Averaging, or FedAvg, method works inside a Federated Learning setting. At first, the central server hands out a global model to a bunch of client nodes. Then each client trains that model on its own private dataset and later it returns updated model weights back to the server. After that, the server combines those received updates with a weighted averaging rule, so it can craft a better global model overall. This loop just keeps going

until the process settles, meaning convergence is reached. In practice, FedAvg supports cooperative learning, keeps data confidential, lowers transmission expenses, and can even strengthen intrusion detection in distributed network systems.

3. Data Preprocessing

However, due to the presence of missing values, redundant information, categorical attributes, and inconsistent distribution of features among the raw network traffic data sets, data preprocessing is an important step in the proposed intrusion detection framework. Effective preprocessing ensures that data is clean and well-structured for model training, which can lead to better results.The preprocessing can help to ensure that the data is clean and well-structured, which can improve the efficiency of the model training process. Before the Federated Learning and machine learning models are applied to the selected intrusion detection dataset, it is subjected to multiple pre-processing steps in this research[21].

First, duplicated records and unnecessary attributes are eliminated, refining the data. Label encoding is used to map categorical features like protocol type, service, and attack category into numerical values to enable machine learning algorithms to work efficiently with them. These methods, known as feature scaling and normalization, ensure that all the numerical features have a similar range, so that a model does not favor a particular feature. Moreover, the dataset is also divided into seven subsets at the local level to emulate the distributed setup of the clients during Federated Learning.

In addition, feature selection techniques are used in the preprocessing stage to select the most relevant features in the network traffic for intrusion detection. This is useful for reducing training time and for increasing the accuracy of classification. The data preprocessing substantially increases the reliability, efficiency and effectiveness of the proposed private intrusion detection system overall[22].

4. Feature Selection

Feature selection is, honestly, a crucial step in intrusion detection systems since network traffic datasets have a lot of attributes, and many are kinda irrelevant or redundant. If you just use every single feature, the computational burden grows, and model efficiency can drop. In the proposed system, we apply feature selection methods to single out the most meaningful network traffic attributes for cyber threat detection. Recursive Feature Elimination (RFE) ranks features according to their importance and then trims away the less relevant ones, kind of iteratively though not exactly in that wording. Because of this, training usually becomes faster, overall model accuracy can improve, and detection performance gets stronger. On top of that, feature selection helps curb overfitting and also makes Federated Learning more practical in distributed network environments, where bandwidth and coordination matter more than people expect[23].

5. Random Forest Classifier

Random Forest is basically a ensemble machine learning method, pretty popular for intrusion detection, due to its strong accuracy , and robustness and also because it can deal with large datasets quite efficiently. Under the hood, it builds many decision trees that sort network traffic into something normal, or something malicious, at least in practice. Each tree makes its own guess, and then the system decides the overall outcome by majority voting which is kinda straightforward. In the suggested setup, Random Forest gets used to spot cyber threats, based on chosen network traffic features. The approach tends to manage tangled attack patterns well, it helps lower overfitting, and it can boost the classification results. Also, Random Forest can train relatively faster, and it gives dependable intrusion detection when you run it in Federated Learning based distributed environments.

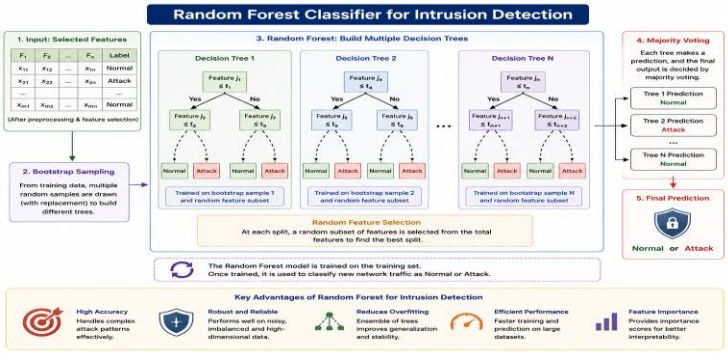

Figure 3: Random Forest Classifier for Intrusion Detection

The figure show the working architecture of the Random Forest Classifier, that is used for intrusion detection in the proposed system. At first, the chosen network traffic features are pushed through bootstrap sampling, so several training subsets get formed. After that, many decision trees get built on their own using random selection of features, sort of like each tree gets a slightly different view. Then each individual tree outputs a verdict, whether the network traffic looks normal or malicious, and in the end the system decides via majority voting, depending on what most trees agree on[24]. The diagram also points out some key benefits of Random Forest, like high accuracy, strong robustness, less overfitting, efficient performance, and more informative feature importance analysis, which is helpful for cyber threat detection.

6. Multi-Layer Perceptron (MLP)

Multi-Layer Perceptron (MLP) is, basically a deep learning artificial neural network that's used for intelligent intrusion detection across distributed network environments. The MLP setup has an input layer, one or more hidden layers and an output layer, and together they chew through network traffic features so it can tell malicious from normal behavior. In the proposed approach, the MLP classifier ends up learning rather intricate traffic patterns, through repeated training rounds, and this improves the classification accuracy for attacks. During training, they rely on activation functions , and also the backpropagation algorithm for tuning the whole model so it performs better. Overall the MLP can deal with nonlinear relationships in the traffic data and it tends to boost the ability to catch cyber threats. And yes it can also fit Federated Learning too, because it allows efficient local training on distributed client nodes without all data needing to move around.

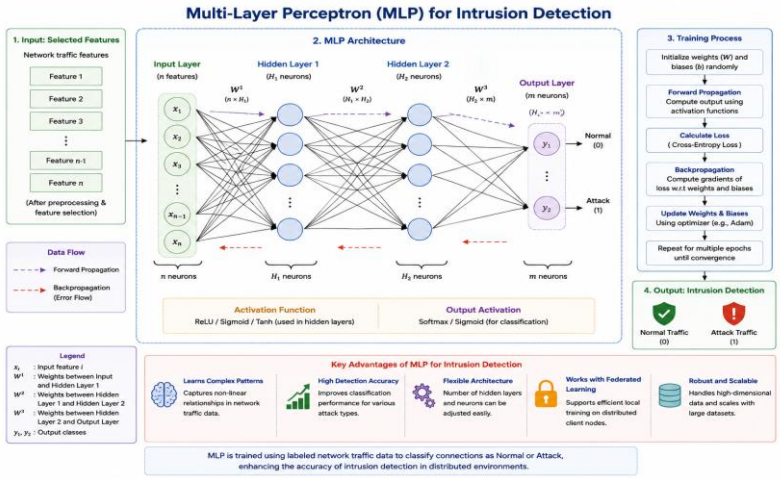

Figure 4: Multi-Layer Perceptron (MLP) for Intrusion Detection

The figure shows the architecture and how the Multi-Layer Perceptron (MLP) model, kinda used for intrusion detection, really works in practice. First, some chosen network traffic features are fed into the neural network as the input, then inside it there are a few hidden layers and finally an output layer that does traffic classification. During training, the system runs forward propagation then backpropagation, in order to tune the weights and, yes, boost prediction accuracy. After that, the output layer decides whether the traffic is normal or it is attack traffic, kinda like a gate keeper[25]. Also, the diagram points out the big benefits of MLP, for example strong detection accuracy , scalability, ability to learn nonlinear patterns, and it fits well with Federated Learning settings , so it can cooperate in distributed environments without too much friction.

7. Secure Model Aggregation

Secure Model Aggregation is kind of an essential privacy-preserving mechanism in Federated Learning settings, where the main idea is to protect local client updates while they are being sent to the central server, or well near it. In the intrusion detection framework we are discussing, every client node trains its model on site and then shares only encrypted model parameters, rather than sending raw network traffic data directly. The aggregation server then combines these updates in a secure way, so a global model can be produced, yet it does not actually peek at private client information. Because of that the overall risk drops, like data leakage, unauthorized access, and even model inversion attacks become less likely. In practice, Secure aggregation can make the participating organizations more confident, and it strengthens the confidentiality for distributed learning

systems overall. It also supports stronger cybersecurity collaboration while still keeping privacy, plus secure communication, in those distributed network environments.

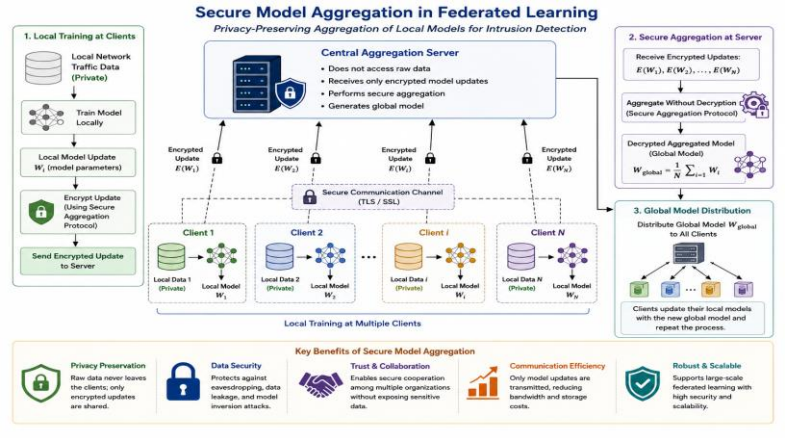

Figure 5: Secure Model Aggregation in Federated Learning

The figure shows the sort of secure model aggregation process that the proposed Federated Learning framework uses, specifically for intrusion detection . In a nutshell, multiple client nodes will locally train machine learning models using their own private network traffic datasets, and afterward they produce encrypted model updates. Those encrypted parameters are then sent to the central aggregation server via protected communication pathways , so nothing sensitive leaks in transit.

On the server side, aggregation happens while it does not directly touch any raw client data, and it outputs an updated global model. After that, the global model gets redistributed back out to all clients again. Overall, the figure is meant to emphasize several key benefits, privacy preservation included , secure communication, scalability , lower chances of data leakage, and the idea of shared cybersecurity intelligence, kind of in a coordinated way.

Result and Discussion

The proposed Federated Learning-based Intrusion Detection System was successfully implemented using a Streamlit-based interactive dashboard environment. The developed system consists of multiple modules including dataset upload, federated model training, intrusion prediction, and performance monitoring. The implementation demonstrates the practical application of privacy-preserving machine learning for intelligent cyber threat detection in distributed network environments.

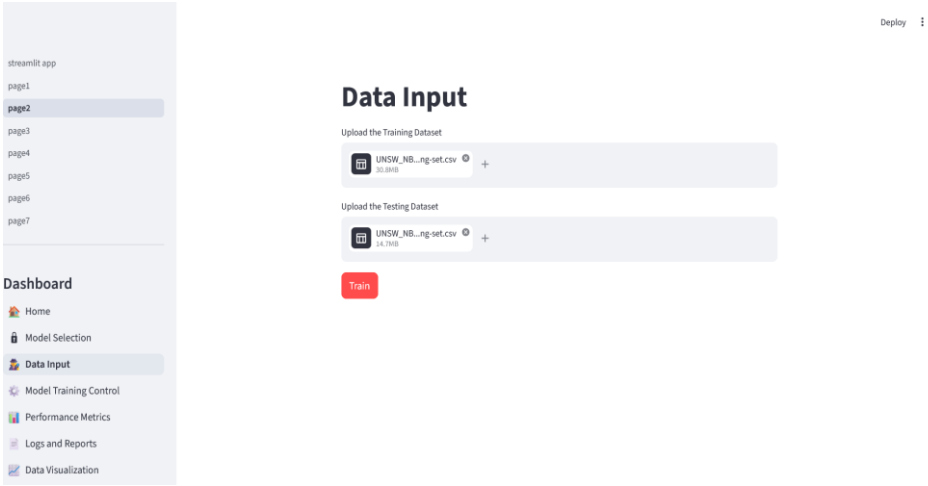

Figure 6: Dataset Upload Interface

The dataset upload interface, shown in the implementation dashboard, lets users drop in training and testing files, separately for both model training and evaluation. For the UN SW-NB15 dataset, it was actually loaded successfully into the system, by using the “Data Input” module. There is this kind of organized place there, for dataset management before kicking off

the federated learning process. Overall, the fact that the datasets were loaded works as a confirmation that the system can manage large scale intrusion detection datasets, which are needed for distributed model training.

Figure 7: Federated Learning Training Control

The “Model Training Control” interface sort of shows how the Federated Learning environment gets kicked off and initialized. In the system view, it looks like the federated learning server is configured properly, and several client nodes get brought online for distributed learning too. Each client in turn runs its own local model training, using private network traffic data, without dumping anything raw. The dashboard also offers a way to start the federated learning process , so the parties can work together in a protected manner for collaborative training while still keeping sensitive raw data out of reach. Overall, this setup confirms the decentralized training architecture, is integrated into the intrusion detection framework in a working way.

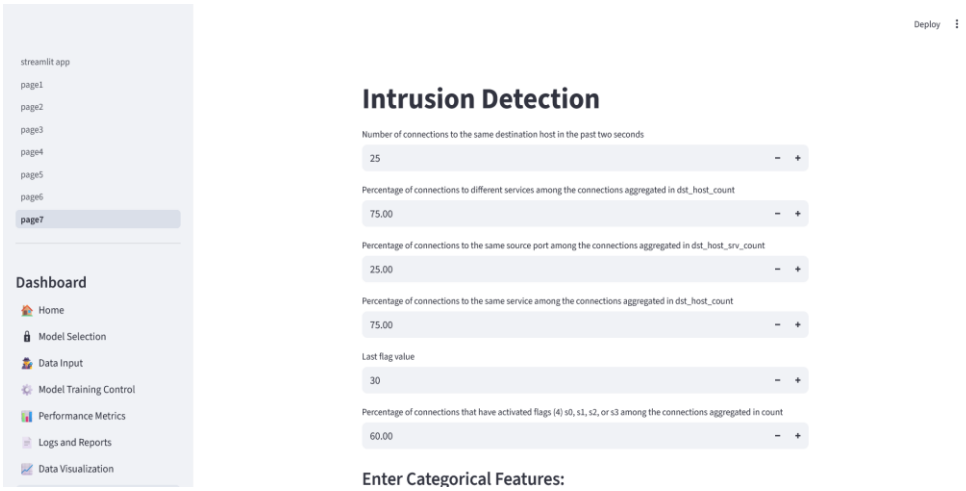

Figure 8: Intrusion Detection Input Interface

The intrusion detection interface lets users type in numerical network traffic features like destination host count, service percentage, source port percentage, and flag values. Those numbers are, well pretty useful for peeking at how the network behaves, and catching malicious traffic patterns. Beyond that the system supports multiple configurable input parameters, so the whole thing feels more flexible, and honestly more effective for cyber threat detection. Overall this interface shows a real world way of doing intelligent attack prediction, by combining machine learning with Federated Learning methods.

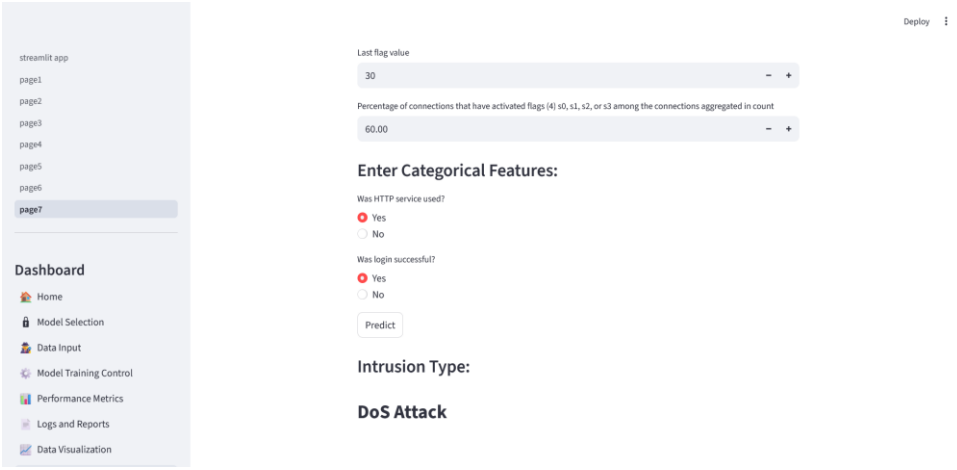

Figure 9: Intrusion Prediction Output

The final output interface shows the predicted intrusion category from the trained Federated Learning model. After the system processed the given network traffic parameters along with the categorical features, it managed to classify the attack type as a "DoS Attack" so, in the end, the output is pretty clear. This result also confirms that the proposed framework is able to analyze network traffic and spot malicious actions across distributed settings. The successful prediction points out that the combined machine learning model can detect cyber threats reliably, while still keeping data privacy handled, and it also supports secure distributed intrusion detection , overall.

In general, the implementation outcomes indicate that the proposed Federated Learning-based Intrusion Detection System delivers solid cyber threat detection. It also provides secure collaborative learning, better privacy preservation, and scalable distributed model training which is useful for modern cybersecurity environments.

Conclusion

The paper in this regard proposes an Intrusion Detection System (IDS) based on Federated Learning to achieve privacy protection for intelligent intrusion detection in a distributed network environment. This study aimed to address shortcomings of existing centralized intrusion detection models by identifying a way to collaboratively build a model while avoiding disclosure of sensitive network traffic information. With the primary goal of preserving data privacy and security, Federated Learning was combined with machine learning algorithms like the Random Forest and Multi-Layer Perceptron to enhance the accuracy of detecting attacks. Distributed learning architecture was applied to classify the normal and malicious network traffic with the proposed framework, which showed good performance. Additionally, secure model aggregation and preprocessing techniques helped enhance system efficiency and reliability. The results indicate that Federated Learning can provide scalable, secure, and efficient cybersecurity solutions for modern IoT, cloud, and enterprise environments. In general, proposed system is a practical solution for collaborative cyber threat detection while maintaining privacy preservation, decreasing the threats of centralization, and augmenting the capability of intrusion detection in distributed networks.

References

I. B. Ababio, K. Agyekum, and M. Frimpong, "Blockchain-Assisted Federated Learning Framework for Trustworthy Malware Analysis in IIoT," *Future Internet*, vol. 17, no. 2, pp. 1–18, 2025.

M. Rahmati, "Federated Learning-Driven Cybersecurity Framework for IoT Networks," *arXiv preprint arXiv:2502.04125*, pp. 1–12, 2025.

B. Buyuktanir, A. Keskin, and H. Polat, "Federated Learning in Intrusion Detection: Advancements, Challenges, and Future Directions," *Journal of Network and Systems Management*, vol. 33, no. 1, pp. 1–29, 2025.

S. A. Mahmud, M. Hasan, and T. Rahman, "Privacy-Preserving Federated Learning-Based Intrusion Detection in IoT," *Mathematics*, vol. 12, no. 4, pp. 1–20, 2024.

N. Sun, J. Li, and Y. Wang, "Blockchain-Based Federated Learning for Intrusion Detection Systems," in *Proceedings of the ACM International Conference on Cybersecurity*, 2024, pp. 125–132.

J. M. Wang, H. Zhao, and K. Liu, "NIDS-FGPA: Federated Learning Intrusion Detection Based on

Gradient Similarity and Homomorphic Encryption," *PLOS ONE*, vol. 19, no. 3, pp. 1–22, 2024.

R. Lazzarini, F. Khan, and A. Rehman, "Federated Learning for IoT Intrusion Detection: A Practical Implementation Survey and Evaluation," *AI*, vol. 4, no. 3, pp. 455–478, 2023.

S. K. Das, P. Sharma, and R. Patel, "Federated Learning Assisted IoT Malware Detection Using Random Forest," in *Proceedings of the ACM International Conference on Internet of Things Security*, 2023, pp. 210–217.

J. L. Hernandez-Ramos, P. Moreno, and A. J. Jara, "Intrusion Detection Based on Federated Learning: A Survey and Taxonomy," *arXiv preprint arXiv:2305.10214*, pp. 1–25, 2023.

A. Agrawal, R. Gupta, and V. Sharma, "Federated Learning for Intrusion Detection Systems: Concepts, Challenges, and Future Directions," *Computer Communications*, vol. 188, pp. 16–35, 2022.

H. Xu, A. Glick, and K. Greene, "Edge-Enabled Federated Learning for Cybersecurity Applications," *IEEE Internet of Things Journal*, vol. 10, no. 5, pp. 4210–4221, 2023.

Z. Yang, J. Liu, and F. Chen, "A Survey on Federated Machine Learning: Challenges and Opportunities," *ACM Computing Surveys*, vol. 54, no. 4, pp. 1–37, 2023.

T. Li, A. Smith, and S. Wang, "Federated Optimization for Heterogeneous Distributed Systems," *Journal of Machine Learning Research*, vol. 24, no. 86, pp. 1–50, 2023.

R. Sharma, P. Singh, and A. Jain, "Federated Learning-Based Intrusion Detection for IoT Networks," *IEEE Access*, vol. 11, pp. 45212–45225, 2023.

M. Chen, U. Challita, W. Saad, and C. Yin, "Machine Learning for Large-Scale Wireless Networks," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 6–35, 2022.

Y. Zhao, J. Li, and H. Wang, "Privacy-Aware Federated Deep Learning Framework for Cybersecurity Applications," *IEEE Access*, vol. 12, pp. 11245–11259, 2024.

A. Verma, S. Gupta, and R. Mehta, "Secure Federated Intrusion Detection System for Distributed IoT Networks," *Journal of Information Security and Applications*, vol. 78, pp. 1–14, 2024.

K. Patel and M. Shah, "Hybrid Federated Learning Approach for Intelligent Network Intrusion Detection," *Computer Networks*, vol. 245, pp. 1–13, 2025.

L. Chen, Y. Zhou, and P. Kumar, "Lightweight Federated Learning Model for Real-Time Cyber Threat Detection," *IEEE Internet of Things Journal*, vol. 11, no. 7, pp. 6521–6534, 2024.

S. Roy and D. Banerjee, "Privacy-Preserving Deep Federated Learning for Smart Network Security," *Future Generation Computer Systems*, vol. 154, pp. 220–233, 2025.

M. Ali, T. Ahmed, and R. Hassan, "Federated Learning-Based Secure Communication Framework for IoT Intrusion Detection," *Sensors*, vol. 24, no. 3, pp. 1–19, 2024.

P. Singh and V. Kumar, "Efficient Federated Cyberattack Detection Using Ensemble Learning Techniques," *Expert Systems with Applications*, vol. 254, pp. 1–15, 2025.

J. Brown, E. Wilson, and A. Clark, "Distributed Federated Deep Learning for Privacy-Preserving Malware Detection," *IEEE Transactions on Network and Service Management*, vol. 21, no. 2, pp. 1450–1463, 2024.

H. Kim and S. Lee, "Robust Federated Learning Architecture Against Poisoning Attacks in Intrusion Detection Systems," *Computers & Security*, vol. 138, pp. 1–17, 2025.

F. Ahmad, M. Usman, and N. Rehman, "Scalable Federated Learning Framework for Intelligent Threat Detection in Cloud Environments," *Journal of Cloud Computing*, vol. 13, no. 1, pp. 1–16, 2024.