Design of Privacy-Preserving AI-Driven SDN Controller Using Federated Learning for Intrusion Detection

Dr. Zainab Ali Abbood

zainab.a.abbood@muc.edu.iq

Abstract: The rapid development of network technologies and the growing sophistication of cyber threats have emphasized the necessity of smart, scalable and privacy-sensitive security solutions. Software-Defined Networking (SDN) is a system that offers centralized control and visibility of the network globally and as such is a fitting platform on which to build complex security mechanisms. Nevertheless, SDN-based traditional intrusion detection methods usually involve a centralized amount of data collection, posing serious questions about the privacy of data and scaling. This paper suggests designing a privacy-sensitive Artificial Intelligence (AI)-controlled SDN controller with Federated Learning (FL) to detect intrusions in a distributed manner. The SDN controller in the proposed architecture is complemented with smart modules, comprising of intrusion detection system (IDS), federated learning aggregation unit, and a decision engine. The distributed controllers are trained on their own data to obtain a local model (Deep Neural Network) and only the model parameters are communicated to a central controller to be aggregated. The model is tested with the help of the NSL-KDD dataset, in which the network traffic is categorized as normal and malicious. Experimental outcomes prove that the suggested FL-based model has high detection capabilities, with the accuracy of 96% and precision of over 90, which is better than the performance of the traditional non-federated models. The suggested system is successful in terms of data privacy, lowering the communication overhead, and increasing the scalability with a high detection accuracy. These findings suggest the implementation of federated learning in an AI-based SDN controller offers a powerful and effective framework to current network security.

Keywords: Software-Defined Networking (SDN), Federated Learning (FL), Intrusion Detection System (IDS), Deep Neural Networks (DNN), Privacy-Preserving Security.

Zianb Ali ABBOOD, Computer Technology Engineering Department, Al-Mansour University College, Baghdad, Iraq

1. Introduction

The rapid evolution of network technologies and the proliferation of Internet-based services have significantly the high rate of development of network technologies and the spread of Internet-based services has made contemporary communication infrastructures much more complicated and susceptible. Software-Defined Networking (SDN) has become a promising paradigm, which makes the network control plane independent of the data plane, which can be centrally managed, programmed, and dynamically configured [1-5]. This architectural malleability renders SDN especially appropriate to incorporate complex security controls and smart decision-making frameworks. Thus far, centralization of SDN controllers also presents significant security threats, since they become the target of cyberattacks like Distributed Denial-of-Service (DDoS), intrusion attempts, and data breaches [6]. Therefore, the provision of strong and smart security systems in SDN networks has become a key necessity of new-generation networks.

Simultaneously, cybersecurity threats have become increasingly common and sophisticated and are threatening the network reliability, data integrity, and user privacy seriously. The more conventional intrusion detection systems (IDS) are based on centralized data collection and analysis which in most cases results in scalability constraints, excessive communication overhead and exposes sensitive data [7]. The use of machine learning (ML) and deep learning (DL) algorithms has become popular to improve intrusion detection abilities as it can identify intricate patterns on network traffic data [8]. Nevertheless, the vast majority of traditional ML-based solutions demand centralization of large amounts of data in a central server, which poses a major privacy risk and regulatory issues, particularly in distributed and heterogeneous networks [9-11].

To overcome them, Federated Learning (FL) has been proposed as a decentralized machine learning paradigm that allows collaborative training of a model without any participants exchanging raw data [12]. Local models in FL are trained using distributed data, and model updates are only shared and combined to create a global model, thus ensuring data privacy and minimizing the risk of communication. This is an especially appropriate method in SDN environments, in which various controllers and network domains can engage in distributed learning with locality of data. Although it has its benefits, the concept of FL being integrated with SDN-based security systems is a relatively new research topic, and scanty studies have been done on the implementation of intelligence as a part of the SDN controller to make real-time decisions and mitigate the threat.

Current literature has examined the use of machine learning and federated learning to SDN-based intrusion detection systems, but there are still a number of research gaps. The majority of existing solutions consider intrusion detection to be an independent application-layer functionality, instead of being integrated into the SDN controller architecture [13,14]. Also, there are numerous studies that do not have a common framework that integrates privacy protection, distributed intelligence, and adaptive control into one system. Moreover, minimal focus has been directed towards developing AI-based SDN controllers capable of conducting traffic control, threat detection, and collaborative learning in a privacy-compliant manner. The drawbacks indicate the necessity of an innovative model that will combine FL and SDN control systems effectively to increase the level of security and efficiency [15,16].

To address these issues, the paper will suggest how to design a privacy-preserving AI-based SDN controller based on the use of federated learning to detect intrusions efficiently. The suggested system augments the SDN controller by incorporating a distributed learning system that facilitates collaborative learning in multiple network domains without revealing sensitive data. The fundamental detection model is a deep neural network (DNN), which is designed to be used to classify network traffic based on whether it is normal or malicious.

2. Proposed System Architecture

2.1 Overall Architecture

The suggested system presents a privacy-aware AI-based Software-Defined Networking (SDN) controller that incorporates the Federated Learning (FL) to provide efficient and distributed intrusion detection. Compared to the conventional SDN-based security solutions, where the intrusion detection is introduced as an independent application, the proposed architecture integrates intelligence into the SDN controller and allows making real-time decisions and take an adaptive response to mitigate threats.

Overall architecture is modeled as a hierarchical and distributed structure whereby there are various SDN controllers that learn by means of a federated learning. With this architecture, the main SDN controller in this architecture serves as global aggregation server and secondary or domain level controllers in this architecture are distributed clients. All client controllers locally observe traffic on the networks in their domains and train a local deep learning model to detect intrusion using their own data. Raw data are not sent but only model parameters and updates are sent to the central controller, hence maintaining data privacy, and minimizing communication risks.

The SDN controller at the center collects the local model updates received and combines them with a federated averaging approach to create a global detecting intrusion model. This international model is then shared to all the controllers involved and they are then able to increase their detection levels through shared information without jeopardizing sensitive information. The process is repeated through numerous rounds of communication, which will guarantee the further improvement of the model and its response to the emerging cyber threats.
Moreover, the architecture proposed takes advantage of the programmability of SDN to intimately couple the intrusion detection mechanism with network control activities. The controller is powered by AI and therefore, in addition to identifying malicious traffic, it is capable of dynamically implementing security policies, including blocking traffic or rerouting or limiting traffic rate, depending on the identified threat. Such close-knit between detection and response plays a key role in enhancing the responsiveness and effectiveness of the entire security framework. Figure 1 shows the general setup of the AI-based federated SDN controller architecture to demonstrate the organization of the proposed system and how the SDN controller interacts with distributed learning entities.

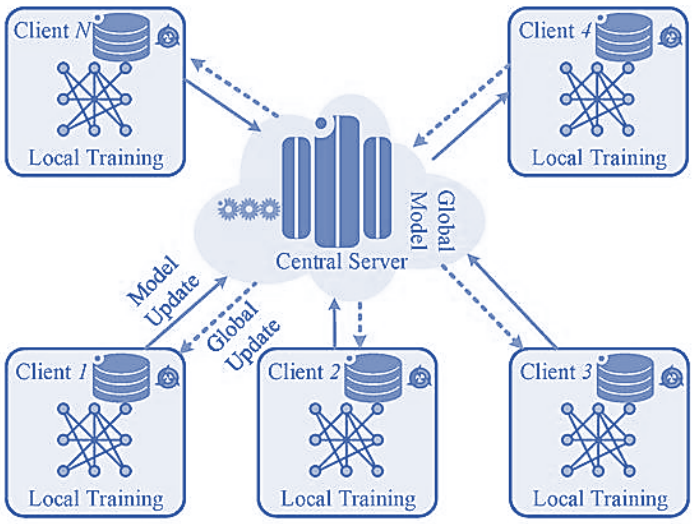

Figure 1: Proposed AI-driven SDN controller architecture integrating federated learning for privacy-preserving intrusion detection.

2.2 AI-Driven SDN Controller Design

The suggested AI-enhanced SDN controller builds upon the original SDN functionality by introducing smart security measures, allowing real-time intrusion

detection and dynamic response. It combines three main modules: an IDS module, which uses a deep neural network (DNN) to process network traffic and label it as normal or malicious behavior; a Federated Learning (FL) aggregation module, which enables the use of multiple controllers to jointly train their IDS models and does not require sharing raw data, thus preserving privacy and enhancing overall model performance through federated averaging; and a decision engine, which translates the This close collaboration of detection, learning and control makes the SDN controller an entirely autonomous and intelligent security agent, greatly increasing network resilience, shortening response time, and proactively defending against cyber threats. Figure 2 shows the architecture of the controller with the IDS module, FL aggregation module, and decision engine as its functional modules.

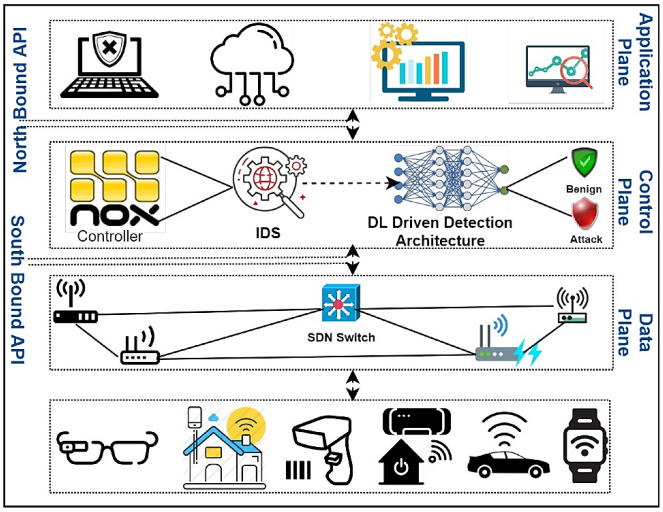

Figure 2: Internal architecture of the proposed AI-driven SDN controller.

2.3 Federated Learning Integration

The suggested system incorporates federated learning (FL) into the SDN controller to support distributed and private model training to train intrusion detection. The SDN controller in this architecture is localized and is a model that is independently trained on locally monitored network traffic data to a deep neural network (DNN) model. The FL structure makes sure that only sensitive data are within their respective domains, and only model parameters are collaboratively learned.

2.3.1 Local Training

In the local training phase, every SDN controller gathers traffic flow statistics on its known data plane devices and uses the statistics to train a local intrusion detection model. The model is trained through the supervised learning methods with traffic instances being classified as either normal or malicious. Stochastic gradient descent

(SGD) is used to perform the training process, which enables the model to update the weights of the model repetitively using local data samples. At each controller, the update of the local model can be written as:

$$w_i^{t+1} = w_i^t - \eta \nabla Fi\left(w_i^t\right) \quad (1)$$

where (w_i^t) represents the local model parameters at iteration (t), η is the learning rate, and $\nabla Fi\left(w_i^t t\right)$ is the gradient of the local loss function. This process enables each controller to learn domain-specific traffic patterns while maintaining data locality.

2.3.2 Global Aggregation

Once the local training is done, a controller sends its new model parameters to the central SDN controller, which serves as the federated aggregation server. The global model is built through local model merging by weighted averaging method, or more often termed Federated Averaging (FedAvg). The global model update is given by:

$$w^{t+1} = \sum_{i=1}^{N} pi\, w_i^{t+1} \quad (2)$$

where $w^{t+1}$is the global model, pi represents the contribution weight of each client, and (N) is the total number of participating controllers. The aggregated model is then redistributed to all controllers, allowing them to update their local models with improved global knowledge.

2.3.3 Iterative Learning Process

The federated learning process is carried out in several communication rounds. Local training and global aggregation is repeated in every round till the model converges or attains the performance desired. This feedback process allows the system to keep changing with the changing network traffic patterns and the new cyber threats. Figure 3 shows the local training and global aggregation process as an iterative process to better demonstrate the federated learning workflow in the proposed SDN controller architecture.

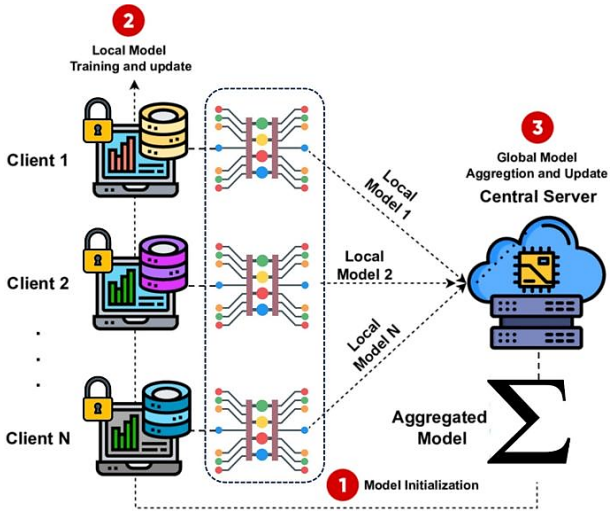

Figure 3: Federated learning workflow.

2.4 Workflow of the Proposed System

The proposed AI-driven SDN controller has an operational workflow based on a federated learning paradigm that allows distributed intrusion detection without compromising data privacy. The system works in a series of synchronized operations that include data gathering, local model training, parameter sharing, global aggregation and model redistribution. This workflow sustains a lifelong learning and adaptation to changing cyber threats in distributed network spheres.

The data on network traffic is gathered at the data plane in the first stage and monitored using SDN-enabled forwarding devices and switches. The SDN controller extracts flow-level features, such as packet statistics and connection attributes, which are used as input for the intrusion detection model. This is a locally performed data collection process and thus the sensitive information is limited to its own environment. The second stage involves local training in which the SDN controllers use their own data. A deep neural network (DNN) model is trained to either classify network traffic as normal or malicious. The training mechanism uses local data to learn domain-specific traffic and attack patterns and enhances detection accuracy without violating the privacy of data.

Each controller sends the modified model parameters (weights) only to the central SDN controller, the federated aggregation server, after local training. The step will remove the necessity of moving raw data, therefore, minimizing the overheads of communication and the privacy threats of centralized data sharing.

During the aggregation step, the central controller takes the local model updates received and uses a federated averaging algorithm to create a global intrusion

detection model. The aggregated model synthesizes the information of various domains of the networks, making it more general and resistant to various types of cyber threats.

Lastly, the new global model is redispatches to the controllers involved and they can revise their local models. This is an iterative process, which repeats between several communication rounds, which allows one to learn continuously, to enhance performance in detecting, and to respond adaptively to new network threats. Figure 4 shows the workflow of the AI-based SDN controller that incorporates federated learning to detect intrusions to demonstrate how the proposed system works sequentially.

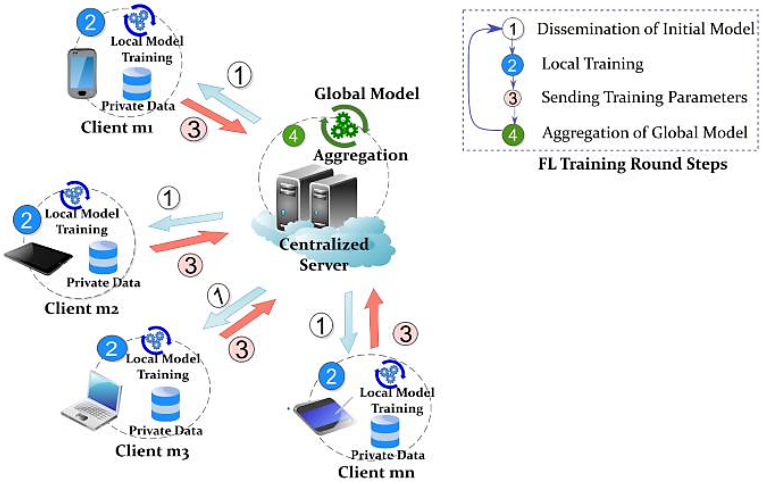

Figure 4: Workflow of the proposed AI-driven SDN.

3. Mathematical Model

This part will provide the mathematical model of the federated learning (FL) model that is built into the suggested AI-based SDN controller. The global model aggregation and the loss function, as well as the local training process, are formulated. All these elements determine the distributed learning mechanism employed to detect intrusion and maintain privacy of data to several network domains.

3.1 Global Model Update

In the proposed federated learning, the global model is aggregated by summing locally trained models of various SDN controllers. Every controller is a client and adds its updated model parameters on a local scale to the central aggregation process. The Federated Averaging (FedAvg) algorithm is used to update the global model with an average of local models weighted.

$$w^{t+1} = \sum_{i=1}^{N} pi \, w_i^t \quad (3)$$

where w^{t+1} represents the updated global model at iteration t+1, w_i^t denotes the local model parameters of client i, pi is the weight associated with each client (typically proportional to the dataset size), and N is the total number of participating controllers. This aggregation mechanism enables the system to combine knowledge from distributed environments without sharing raw data.

3.2 Loss Function

The federated learning model aims to reduce the global loss objective of the participating clients. The global loss is the mean of all data sample losses to the network entities.

$$F(W) = \frac{1}{D} \sum_{i=1}^{N} \sum_{j=1}^{NDi} L(W, x_{ij}) \quad (4)$$

where $F(W)$ denotes the global loss function, D is the total number of samples across all clients, Di is the number of samples at client $\boldsymbol{i}$, and $L(W, x_{ij})$ represents the loss for sample x_{ij}. This formulation ensures that the global model is optimized based on the collective contribution of all distributed datasets while maintaining data privacy.

3.3 Local Training

Local training is completed in isolation on each SDN controller on its own data. Stochastic Gradient Descent (SGD) is used to update the model parameters by minimizing the local loss function through the use of gradient updates.

$$w_i^{t+1} = w_i^t - \eta \nabla Fi\left(w_i^t\right) \quad (5)$$

where w_i^{t+1} represents the local model parameters at iteration t, η is the learning rate, and $\nabla Fi\left(w_i^t\right)$ is the gradient of the local loss function at client i. After completing local updates, the model parameters are transmitted to the central controller for aggregation.

4. Machine Learning Model

4.1 DNN Architecture

The intrusion detection model proposed relies on the Deep Neural Network (DNN) architecture that is expected to be effective in separating normal and malicious network traffic. The DNN model is chosen because it can be used to identify the complex and nonlinear relationship between high-dimensional network traffic data, which is applicable in cybersecurity use in SDN environments. The features in the input layer of the model are based on the NSL-KDD dataset and have 41 features consisting of basic, content-based, and traffic-based features. They are extensive features of network behavior, and are commonly employed in intrusion detection. These inputs are then processed by the model using several fully connected hidden layers to get meaningful patterns and representations.

The architecture is made up of five hidden layers of a decreasing number of neurons namely 30, 20, 10, 5 and 2 neurons respectively. This hierarchical architecture allows gradual features abstraction, with more abstract representations being acquired at lower levels. The decrease of the number of neurons in the layers also contributes to reducing the overfitting and enhancing the performance of generalization. Every hidden layer is followed by a sigmoid activation function that adds nonlinearity to the model enabling it to learn more complex decision boundaries between normal and malicious traffic. The output layer is a single neuron, which is used to classify binary, where the output is whether the input traffic is benign or an attack.

Training is performed on the DNN model under supervised learning where the data is labeled and used to train the model. Backpropagation and gradient-based optimization techniques are applied to optimize the model parameters, making them efficient to converge. It is built into the federated learning structure, where every SDN controller will be trained on its individual model, yet it will contribute to the global model. As shown Figure 5.

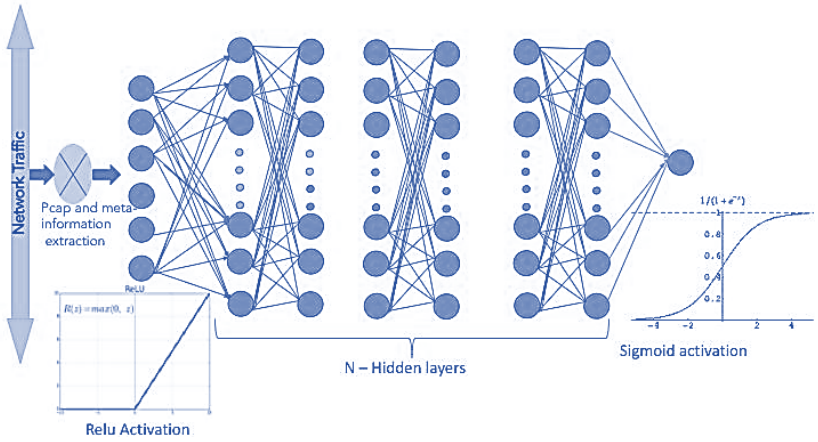

Figure 5: Deep neural network (DNN) architecture used for intrusion detection.

5. Dataset and Preprocessing

5.1 Dataset Description

The suggested model is tested on the NSL-KDD data, a more mature incarnation of the original KDD Cup 1999 data, which is commonly used to test intrusion detection systems. Compared with its predecessor, the NSL-KDD dataset mitigates a number of limitations such as redundant records, imbalanced data distribution,

and is better applicable in the assessment of machine learning models when used in cybersecurity applications. The dataset includes 41 features that describe various aspects of network traffic, including basic features (e.g., protocol type and connection duration), content features (e.g., number of failed login attempts), and traffic ones (e.g., connection statistics). These characteristics offer a holistic description of network behaviour and are commonly used in research on intrusion detection.

Moreover, the dataset contains various types of cyberattacks, including Denial-of-Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R) attacks. The intrusion detection task in this work is defined as a binary classification problem, where all types of attacks are classified as one malicious category, the normal traffic is the benign category. Although the NSL-KDD dataset is relatively old, it is still widely used as a benchmark dataset for intrusion detection due to its balanced structure and reduced redundancy compared to KDD'99. It allows fair comparison with existing studies.

5.2 Data Preprocessing

Prior to the model training, the dataset is preprocessed in a number of steps to guarantee the quality of data and enhance the performance of the models. To prevent bias on training, first, redundant and duplicate records are eliminated. Next, the irrelevant or low-variance features are eliminated to minimize the complexity of the computations and improve the process of learning.

Then, categorical characteristics like protocol type and service are encoded in numbers by using relevant encoding methods. This conversion is needed to allow the neural network to work with non-numeric data. Moreover, the feature scaling is used to normalize the input values, so that all features are equally contributing to the training process and no features have higher numbers that dominate the training process.

5.3 Data Partitioning for Federated Learning

To simulate a federated learning environment, the dataset is partitioned into multiple subsets corresponding to different SDN controllers. The data are divided as follows:

a) 40% of the dataset is allocated to Client 1
b) 40% of the dataset is allocated to Client 2
c) 20% is reserved for testing and validation

This allocation will make sure that every client can get enough data to train locally and that a separate collection of data is used to assess performance. Clients separately train their local model on its own data, and model parameters are only exchanged with the central controller to aggregate them. This design is representative of the real-world distributed network settings in which data are decentralized and sensitive to privacy. The main features of the NSL-KDD dataset used in this paper are summarized in Table 1.

Table 1: Summary of the NSL-KDD dataset features and characteristics.

Category	Description	NO. of Features
Basic Features	General information about connections (e.g., protocol, duration)	9
Content Features	Information from packet payload (e.g., login attempts)	13
Traffic Features	Statistical traffic properties (e.g., connection count)	19
Total Features	—	**41**

6. Experimental Setup

6.1 Environment

The proposed AI-based SDN controller is experimentally tested in MATLAB R2023b as a main simulation platform. MATLAB is chosen because it has a strong numerical computing platform, machine learning platform, and is highly efficient at manipulating matrices, which is crucial to training deep neural networks and simulating federated learning processes.

The experiments are run to a high-performance computing platform with an Intel Core i7 processor with a speed of 3.2GHz, 32GB of RAM, and an NVIDIA GeForce graphic card, having 6GB of dedicated memory. This structure allows the training of models efficiently and facilitates parallel processing when training models through federated learning. GPU acceleration drastically shortens the training time, especially when it comes to computing deep neural networks and iterative aggregation algorithms.

6.2 FL Configuration

The federated learning structure is implemented using a client–server architecture within the SDN environment, where two distributed SDN controllers act as clients and a central controller serves as the aggregation server. Each client trains a local model using its own dataset and periodically shares only the model parameters with the central server, ensuring data privacy. The training process is conducted over multiple communication rounds, with local updates performed at the client side and global aggregation at the server side in each round. A total of 15 rounds are used to achieve model convergence and improve detection accuracy.

Additionally, the dataset is introduced during training, starting with smaller subsets and expanding to the full dataset, allowing the evaluation of model performance under different data availability scenarios and simulating realistic distributed environments. This design balances computational efficiency, communication overhead, and model performance, as summarized in Table 2.

Table 2: Experimental setup and FL configuration parameters.

Parameter	Value
Simulation Tool	MATLAB R2023b
CPU	Intel Core i7 (3.2 GHz)
RAM	32 GB
GPU	NVIDIA GeForce (6 GB)
Number of Clients	2
Aggregation Server	SDN Controller
Number of FL Rounds	15
Dataset	NSL-KDD

.7 Results and Discussion

In this section, the performance analysis of the proposed AI-based SDN controller combined with federated learning (FL) to detect intrusions will be provided. It is assessed based on the key performance measures, such as accuracy, precision and true positive rate (TPR). The aim is to determine how effectively the proposed system is in identifying cyber threat and maintaining data privacy in distributed network settings.

7.1 Accuracy

Accuracy is a performance measure that is used to assess the general accuracy of the intrusion detection model. The experimental findings indicate that the suggested FL-based DNN model has a high-detection accuracy of about 96, which indicates that it is effective in classifying network traffic into normal and malicious.

Moreover, the precision is enhanced gradually with several rounds of federated learning as the accumulation of knowledge is done by distributed SDN controllers. This feedback learning approach improves the generalizability of the model in a variety of network settings, and results in more consistent and robust performance. The accuracy grows gradually in the course of training as demonstrated in Figure 6, which shows that the federated learning mechanism is able to play a substantial role in increasing performance. The fact that the values of accuracy converge in the subsequent rounds also indicates that the model is in a stable state, which proves the strength of the suggested approach in distributed intrusion detection situations.

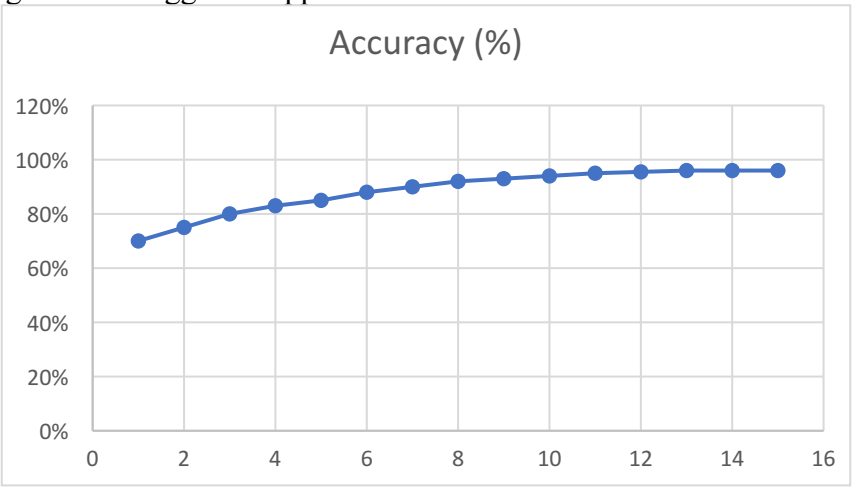

Figure 6: Accuracy progression of the proposed FL-based DNN model across multiple federated learning rounds.

In addition to accuracy and precision, further evaluation metrics are used to provide a more comprehensive assessment of the model performance. These include Recall, F1-score, Confusion Matrix, and Receiver Operating Characteristic (ROC) analysis.

7.2 Precision

Precision is a measure of the percentage of attack instances that were correctly identified out of the total number of attacks that are detected. A precision value of more than 90 per cent is obtained by the proposed model, which is low false positive rate. This finding validates that the model can be trusted in differentiating normal traffic and malicious traffic. The precision is especially crucial in a real-world network setup, where the false alarms may cause the waste of resources and system inefficiency. To further assess the applicability of the proposed model in reducing

false positives, Figure 7 shows the precision performance with repeated rounds of federated learning.

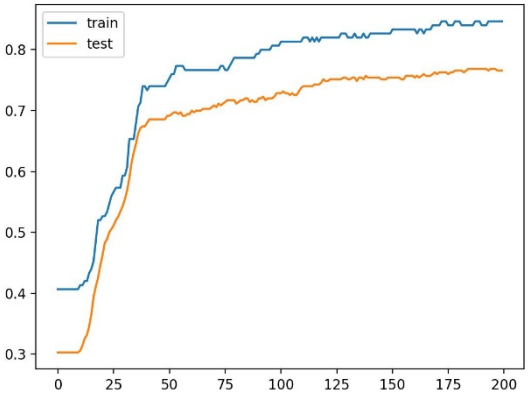

Figure 7: Precision performance of the proposed FL-based intrusion detection model across federated learning rounds.

7.3 True Positive Rate (TPR)

One of the key performance measures, which determines the performance of the model in those cases when it correctly recognizes real attack cases, is the true positive rate (TPR) or recall. As the experimental data shows, the proposed FL-based DNN model has a high TPR at all times, which is indicative of the high ability of the model to identify malicious actions in network traffic.

The findings also indicate that the TPR increases progressively with the rounds of federated learning, which indicates the effectiveness of the collaborative learning process in increasing the detection sensitivity. This enhancement underscores the fact that the model can learn through distributed sources of data, and can also adjust to varying and changing patterns of attacks. Figure 8 shows that the TPR has a gradual increase with the training process, which ultimately levels to higher values. This convergence pattern shows that the model has a high detection rate and does not deteriorate in terms of performance, which is essential to guarantee reliable and robust cybersecurity within a dynamic and distributed network setting.

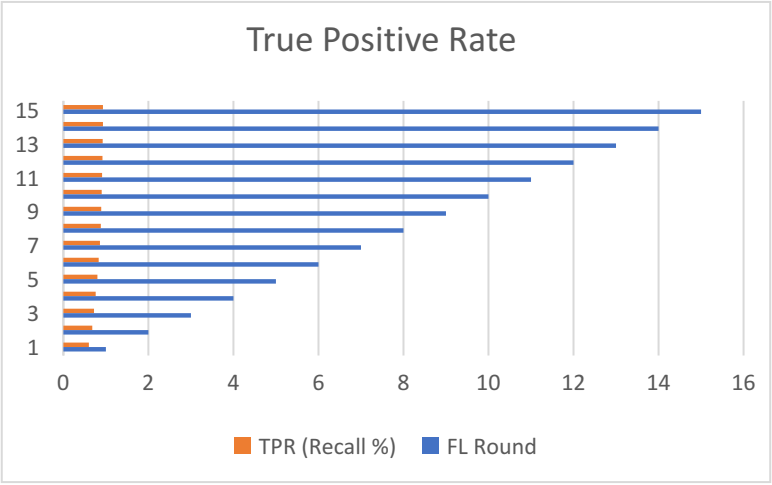

Figure 8: True Positive Rate (Recall) performance of the proposed FL-based intrusion detection model across federated learning rounds.

7.4 Performance Comparison

In order to further confirm the effectiveness of the proposed approach, the federated learning-based DNN model is compared with the traditional non-federated learning models. The purpose of this comparison is to emphasize how federated learning can be used to enhance the performance of models in a distributed network setting.

As shown in Table 5, the proposed FL-DNN model has a much better detection accuracy of about 96 per cent, as compared to 76 per cent and 83 per cent of the non-federated models. This gain is an indication of a significant performance improvement of up to 20% which shows that federated learning is effective in improving model generalization and detection ability. The high-performance of the proposed model is explained by the possibility to utilize distributed data sources with the help of collaborative training which allows the model to acquire more diverse and representative patterns of traffic. Conversely, conventional centralized models are based on limited datasets, which can limit their extrapolative capabilities to other network conditions. The suggested FL-based model, Table 3. Performance comparison between federated and non-federated intrusion detection models.

Table 3: Performance comparison

Model	Accuracy
FL-DNN (Proposed)	**96%**
Non-FL Model 1	76%

Non-FL Model 2	83%

The results demonstrate that the proposed model achieves a balanced performance across all evaluation metrics. The high F1-score indicates an effective trade-off between precision and recall, while the AUC value confirms strong classification capability. As shown Table 4

Table 4: Performance Evaluation Metrics of the Proposed Model

Metric	Value
Accuracy	96%
Precision	91%
Recall	93%
F1-score	92%
AUC	0.95

7.5 Analysis and Discussion

The excellent results of the suggested FL-based model could be explained by a number of factors associated with the paradigm of learning and the architecture. To begin with, federated learning allows sharing knowledge among several SDN controllers without the necessity of a central collection of the data. This distributed learning model enables the model to learn various traffic patterns across different domains of the network, leading to a better generalization ability and a much greater detection accuracy than traditional centralized methods.

Second, federated learning is privacy-preserving, which means that sensitive information will be stored at each controller, limiting the likelihood of data leakage. This feature is especially important in contemporary network settings where stringent data protection policies have to be implemented. The proposed system does not require to transfer raw data, which results in a tradeoff between privacy and model performance, hence it can be applied to real-world, privacy-sensitive deployments.

Moreover, the intrusion detection mechanism directly integrated into the SDN controller will improve responsiveness and efficiency of the system. In contrast to the traditional architectures where the external detection modules are used, the proposed design allows detecting the threat in real-time and automatically implementing mitigation policies. This close connection between detection and control will greatly decrease the response time and the overall resiliency of the network against cyber threats.

Moreover, federated learning is iterative, which adds to the constant improvement of the model. The more the global model is updated in various communication rounds, the more robust it will be to the changes of attack patterns and dynamic network conditions. This flexibility is vital in ensuring a steady performance within settings with swiftly evolving cybersecurity threats. Nevertheless, even with these

benefits, there are still some challenges. The overhead inherent to the federated learning process is the communication overhead caused by regular parameters exchanges between distributed controllers that can be a limitation in large-scale deployments. In addition, federated systems may be vulnerable to adversarial attacks like model poisoning and inference attacks, which may adversely affect model integrity. To overcome these shortcomings, it is necessary to incorporate superior security measures, such as differential privacy, secure aggregation, and effective model validation strategies.

8. Conclusion

In this paper, a privacy-conscious AI-based SDN controller is suggested, which incorporates federated learning to facilitate effective and distributed intrusion detection. This framework mitigates the shortcomings of the classic centralized security systems by enabling multiple SDN controllers to jointly train models without transferring sensitive information to keep privacy intact. With the ability to detect and react in real-time by incorporating intelligence into the SDN controller, the system enhances the security and resilience of the network. The federated learning framework includes a deep neural network (DNN) to learn the traffic behavior and effectively differentiate between normal and malicious behavior. Experimental tests demonstrate good performance with up to 96% detection accuracy and more than 90 percent precision, which is better than the classic non-federated models. These results prove that federated learning improves the generalization of models through the use of distributed data and ensures confidentiality. Scalability is also enhanced by the system as the training process is distributed among multiple controllers, thus the individual computational load is also decreased. Its design that focuses on privacy is consistent with current data protection needs, such that it can be appropriate in the field of the real-world applications in sensitive settings. Nevertheless, there are certain issues such as communication overhead on model updates and possible adversarial attacks within federated environments. Future research can be aimed at the combination of more sophisticated privacy methods like differential privacy and secure multi-party computation, and the use of explainable AI in order to enhance transparency. On the whole, the suggested framework can provide an effective approach to the creation of intelligent, scalable, and secure SDN-based network architectures.

References

[1]. Raza, M., Saeed, M. J., Riaz, M. B., & Sattar, M. A. (2024). Federated learning for privacy-preserving intrusion detection in software-defined networks. *IEEE Access*, *12*, 69551-69567.

[2]. Alazab, A., Khraisat, A., Singh, S., & Jan, T. (2023). Enhancing privacy-preserving intrusion detection through federated learning. *Electronics*, *12*(16), 3382.

[3]. AlQaruty, S., Al Qaruty, R., Al-Tkhayneh, K. M., & Hadi, S. A. (2025, December). Enhancing Security and Performance in Software-Defined Networks Using AI-Driven Automation. In *2025 12th International Conference on Software Defined Systems (SDS)* (pp. 67-73). IEEE.

[4]. Chaganti, R., Suliman, W., Ravi, V., & Dua, A. (2023). Deep learning approach for SDN-enabled intrusion detection system in IoT networks. *Information*, *14*(1), 41.

[5]. Krishnan, P., Jain, K., Aldweesh, A., Prabu, P., & Buyya, R. (2023). OpenStackDP: a scalable network security framework for SDN-based OpenStack cloud infrastructure. *Journal of Cloud Computing*, *12*(1), 26.

[6]. Halman, L. M., & Alenazi, M. J. (2023). MCAD: A Machine learning based cyberattacks detector in Software-Defined Networking (SDN) for healthcare systems. *IEEE Access*, *11*, 37052-37067.

[7]. Wang, X., Wang, Y., Javaheri, Z., Almutairi, L., Moghadamnejad, N., & Younes, O. S. (2023). Federated deep learning for anomaly detection in the internet of things. *Computers and Electrical Engineering*, *108*, 108651.

[8]. Haqmal, R., Safi, M. W., & Mohammad, F. (2026). Enhancing Security in Software-Defined Networks Using Artificial Intelligence Techniques. *Journal of Advanced Computer Knowledge and Algorithms*, *3*(1), 37-54.

[9]. Nasir, Z. U. I., Iqbal, A., & Qureshi, H. K. (2024). Securing cyber-physical systems: A decentralized framework for collaborative intrusion detection with privacy preservation. *IEEE Transactions on Industrial Cyber-Physical Systems*, *2*, 303-311.

[10]. Soy, A. (2025). Secure and Intelligent Collaboration Frameworks for Online Learning Platforms. *Transactions on Internet Security, Cloud Services, and Distributed Applications*, 56-65.

[11]. Vibhute, A. D., Patil, C. H., Mane, A. V., & Kale, K. V. (2024). Towards detection of network anomalies using machine learning algorithms on the NSL-KDD benchmark datasets. *Procedia Computer Science*, *233*, 960-969.

[12]. Yu, D., Xie, Z., Yuan, Y., Chen, S., Qiao, J., Wang, Y., ... & Zhang, X. (2023). Trustworthy decentralized collaborative learning for edge intelligence: A survey. *High-Confidence Computing*, *3*(3), 100150.

[13]. Hamad, N. A., Bakar, K. A., Qamar, F., Jubair, A. M., Mohamed, R. R., & Mohamed, M. A. (2025). Systematic analysis of federated learning approaches for intrusion detection in the Internet of Things environment. *IEEE Access*.

[14]. Li, P., & Wang, J. (2025, June). Rethinking Networks for the Digital Age: The Rise and Impact of Software-Defined Networking. In *2025 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)* (pp. 1-6). IEEE.

[15]. Wen, J., Zhang, Z., Lan, Y., Cui, Z., Cai, J., & Zhang, W. (2023). A survey on federated learning: challenges and applications. *International journal of machine learning and cybernetics*, *14*(2), 513-535.

.[16] Alsamhi, S. H., Myrzashova, R., Hawbani, A., Kumar, S., Srivastava, S., Zhao, L., ... & Curry, E. (2024). Federated learning meets blockchain in decentralized data sharing: Healthcare use case. *IEEE Internet of Things Journal*, *11*(11), 19602-19615.

تصميم متحكم شبكات معرّفة برمجيًا (SDN) قائم على الذكاء الاصطناعي مع الحفاظ على الخصوصية باستخدام التعلم الاتحادي لاكتشاف التسلل

د. زينب علي عبود

zainab.a.abbood@muc.edu.iq

المستخلص: إن التطور السريع في تقنيات الشبكات والتزايد المستمر في تعقيد التهديدات السيبرانية قد أكدا الحاجة إلى حلول أمنية ذكية، قابلة للتوسع، ومراعية للخصوصية. تُعد الشبكات المعرفة برمجيًا (SDN) نظامًا يوفر تحكمًا مركزيًا ورؤية شاملة للشبكة، مما يجعلها منصة مناسبة لبناء آليات أمنية متقدمة. ومع ذلك، فإن طرق كشف التسلل التقليدية المعتمدة على SDN غالبًا ما تعتمد على جمع مركزي للبيانات، مما يثير تحديات كبيرة تتعلق بخصوصية البيانات وقابلية التوسع. تقترح هذه الدراسة تصميم متحكم SDN مدعوم بالذكاء الاصطناعي (AI) ومراعٍ للخصوصية باستخدام التعلم الاتحادي (FL) بهدف كشف التسللات بشكل موزع. يتضمن الهيكل المقترح دمج وحدات ذكية داخل متحكمSDN ، تشمل نظام كشف التسلل(IDS) ، ووحدة تجميع التعلم الاتحادي، ومحرك اتخاذ القرار. يتم تدريب المتحكمات الموزعة على بياناتها المحلية لتوليد نموذج محلي (شبكة عصبية عميقة)، ويتم فقط مشاركة معاملات النموذج مع المتحكم المركزي لدمجها. تم تقييم النموذج باستخدام مجموعة بياناتNSL-KDD ، حيث تم تصنيف حركة الشبكة إلى طبيعية وخبيثة. أظهرت النتائج التجريبية أن النموذج المقترح المعتمد على التعلم الاتحادي يمتلك قدرة عالية على الكشف، حيث بلغت الدقة 96%، وتجاوزت قيمة الدقة النوعية (Precision) نسبة 90%، متفوقًا بذلك على النماذج التقليدية غير المعتمدة على التعلم الاتحادي. كما أثبت النظام المقترح كفاءته في الحفاظ على خصوصية البيانات، وتقليل الحمل على قنوات الاتصال، وزيادة قابلية التوسع مع الحفاظ على دقة كشف مرتفعة. وتشير هذه النتائج إلى أن دمج التعلم الاتحادي ضمن متحكم SDN المعتمد على الذكاء الاصطناعي يوفر إطارًا قويًا وفعالًا لتعزيز أمن الشبكات الحديثة.

الكلمات المفتاحية: الشبكات المعرفة برمجيًا (SDN)، التعلم الاتحادي (FL)، نظام كشف التسلل (IDS)، الشبكات العصبية العميقة (DNN)، الأمن المحافظ على الخصوصية.

د. زينب علي عبود-قسم تقنيات هندسة الحاسوب -كلية المنصور الجامعة - بغداد، العراق 1