Detecting Cyberattacks to Federated Learning on Software-Defined Networks

Himanshi Babbar[1], Shalli Rani[1], Aman Singh[2,3,4], and Gabriele Gianini[5]

[1] Chitkara University Institute of Engineering and Technology, Chitkara University,
Rajpura,140401 Punjab, India
{himanshi.babbar,shalli.rani}@chitkara.edu.in
[2] Universidad Europea del Atlantico, 39011 Santander, Spain
[3] Universidad Internacional Iberoamericana, Arecibo, PR 00613 - USA
[4] Universidad Internacional Iberoamericana, Campeche 24560 - Mexico
amansingh.x@gmail.com
[5] Universita degli Studi di Milano, via Celoria 18, 20133, Milano - Italy
gabriele.gianini@unimi.it

Abstract. Federated learning is a distributed machine-learning technique that enables multiple devices to learn a shared model while keeping their local data private. The approach poses security challenges, such as model integrity, that must be addressed to ensure the reliability of the learned models. In this context, software-defined networking (SDN) can play a crucial role in improving the security of federated learning systems; indeed, it can provide centralized control and management of network resources, enforcement of security policies, and detection and mitigation of network-level threats. The integration of SDN with federated learning can help achieve a secure and efficient distributed learning environment. In this paper, an architecture is proposed to detect attacks on Federated Learning using SDN; furthermore, the machine learning model is deployed on a number of devices for training. The simulation results are carried out using the N-BaIoT dataset and training models such as Random Forest achieves 99.6%, Decision Tree achieves 99.8%, and K-Nearest Neighbor achieves 99.3% with 20 features.

Keywords: Software Defined Networks · Federated Learning · Cyber Security

1 Introduction

Cybersecurity in federated learning and SDN refers to the measures taken to protect sensitive or confidential data and prevent unauthorized access, theft, or tampering. Federated learning involves sharing model updates between decentralized devices and a central controller, which creates security concerns as sensitive data may be transmitted over the network [14].

Some of those concerns can be addressed through software-defined networks (SDNs). SDN is a type of networking architecture that separates the control plane (which manages network traffic flow) from the data plane (which forwards traffic to its destination). Whereas in traditional networking, network devices, such as switches and routers, perform both control and data plane functions, in SDNs, the control plane is moved to a centralized controller, which can be programmed using software to manage the network traffic flow, allowing for more efficient and flexible network management [19].

SDN can help address some of the cybersecurity challenges in federated learning by providing tools and techniques for securing the network and the data transmitted over it [4]. Some notable resources made available by SDN are the following:

1. **Network Segmentation:** SDN can be used to segment the network and restrict access to sensitive data, helping to prevent unauthorized access or data breaches.
2. **Traffic Monitoring:** The central controller in an SDN-based federated learning architecture can monitor network traffic and detect any malicious activity, such as malware or network attacks [9].
3. **Secure Data Transport:** The central controller can also dynamically configure the network to ensure secure data transport between federated devices and the central controller [12]. This can include using encryption, secure protocols, and firewalls to protect against data tampering or theft.
4. **Access Control:** The central controller can implement access control to ensure that only authorized devices are allowed to participate in the federated learning process. This can include authentication and authorization mechanisms, such as digital certificates or biometric authentication.
5. **Model Integrity:** The central controller can monitor the model updates generated by federated devices and detect any malicious modifications. This can help to ensure the integrity and trustworthiness of the federated learning model.

It is important to note that while SDN can enhance the security of federated learning, it is not a silver bullet [10]. Adequate security measures, including proper network configuration, strong access controls, and secure data storage, must also be in place to ensure the security and privacy of sensitive data in a federated learning system. Rather, it is important to prioritize cybersecurity in federated learning to ensure the integrity and trustworthiness of the models generated and to prevent unauthorized access to or misuse of sensitive data [6].

1.1 Integration of cybersecurity with federated learning in SDN

Federated learning and SDN have the potential to revolutionize the way data is processed and managed. Integrating cybersecurity into these technologies can help secure networks and prevent sensitive data from being compromised. In federated learning [8], multiple decentralized devices collectively contribute to building a shared machine-learning model without sharing the raw data with a central server. Integrating cybersecurity measures into this process can ensure that the data is protected while in transit and at rest. This can be achieved through encryption, secure communication protocols, and access control mechanisms. There are various contributions to integration:

1. **Decentralized Model Training:** Federated learning enables decentralized training of machine learning models, reducing the risk of a single point of failure and increasing the overall robustness of the system.
2. **Enhanced Privacy:** By allowing models to be trained locally on edge devices, federated learning enhances the privacy of the data and reduces the risk of sensitive information being intercepted or leaked during the model training process.
3. **Dynamic Resource Allocation:** By leveraging SDN, federated learning can dynamically allocate network resources based on the needs of the system, improving network efficiency and reducing the risk of network congestion.
4. **Improved Security:** SDN enables the centralized management and control of network security policies, improving the overall security of the system.
5. **Efficient Data Collection:** By using federated learning, data collection can be optimized to reduce data transmission overhead and improve the accuracy of the models.
6. **Improved Performance:** The integration of federated learning and SDN can lead to improved system performance through the efficient use of network resources and the efficient training of models.

7. **Resilience:** Federated learning and SDN can make cybersecurity systems more resilient by enabling the efficient distribution of tasks and resources across the system, reducing the risk of single-point failures.

The main contributions of the paper are:

1. To enhance the security, efficiency, and overall performance of these systems, the potential of combining federated learning and SDN in cybersecurity is discussed.
2. Integrating cybersecurity into Federated Learning and SDN can help secure the network infrastructure, prevent sensitive data from being compromised, and ensure the reliability and availability of the network.
3. The architecture is proposed to detect attacks with federated learning in SDN to build a secure cybersecurity architecture for networks.
4. The N-BaIoT dataset is deployed for IDS in IoT networks and contains benign and malicious traffic generated from real IoT devices.
5. The performance is evaluated on different models (Decision tree, Random forest, and K-nearest neighbor) based on the dataset using metrics such as accuracy, precision, recall, and F1-score.

The rest of the paper is organized as: Section 2 describes the related work to secure federated learning from any kind of attack; Section 3 describes the methodology of encrypted algorithms and proposed architecture for cybersecurity; Section 4 represents the performance analysis and evaluation of the proposed work; Section 5 showcase the results; and Section 6 concludes the paper.

2 Related Work

The various researchers are focusing on the work to secure federated learning from any kind of attack. Secure federated learning [20] Researchers have proposed various methods for securing the federated learning process, such as homomorphic encryption, secure multiparty computation, and secure aggregation. These methods aim to protect the privacy of sensitive data and ensure the integrity of the federated learning model updates. In Privacy-preserving federated learning [1], there has been a growing interest in developing privacy-preserving methods for federated learning, such as differential privacy and secure aggregation. These methods aim to protect the privacy of sensitive data while still allowing for the training of accurate machine-learning models. In this, the centralized control plane, i.e., SDN, provides a single point of management for the network, which can be leveraged to implement privacy-preserving techniques in Federated Learning. This can include the use of encryption and secure communication protocols to protect data in transit, as well as access control mechanisms to ensure that only authorized devices can access the data. In Federated Learning, privacy preservation can be achieved through the use of differential privacy techniques, such as adding random noise to the data before it is shared, to protect the privacy of individual users. Additionally, privacy-preserving aggregation methods can be used to combine the model updates from multiple devices in a way that protects the privacy of each individual device.

In [7], the authors have used machine learning for IDS to detect malicious activity on networks. However, this requires collecting large amounts of network traffic data, which can be a privacy risk. Federated learning is a new approach that allows IDS to train models without sharing data. This can help protect privacy while still providing effective security. The authors of [13] proposed a new federated deep learning scheme called DeepFed to detect cyber threats against industrial CPSs. First, we design a new deep learning-based intrusion detection model for industrial CPSs using a convolutional neural network and a gated recurrent unit. Second, we develop a federated learning framework that allows multiple industrial CPSs to collectively build a comprehensive intrusion detection model in a privacy-preserving way. Finally, we create a secure communication protocol based on the Paillier cryptosystem to protect the security and privacy of model parameters during the training process.

There have been efforts to develop privacy-preserving methods for software-defined networking, such as secure multiparty computation, homomorphic encryption, and differential privacy. These methods aim to protect the privacy of sensitive data while still allowing for the efficient and secure management of network resources. According to the authors of [5], researchers have proposed various methods for securing software-defined networking, such as using encryption and authentication techniques, network segmentation, and access control. These methods aim to prevent unauthorized access to sensitive data and network attacks, such as malware or hacking.

This is a rapidly growing field, and there is a lot of ongoing research into improving the security and privacy of federated learning and software-defined networking. The goal of this research is to create secure and scalable systems for machine learning that can effectively protect the privacy and security of sensitive data.

3 Methodology

Encryption algorithms are mathematical algorithms that are used to convert plain text into an encrypted, or ciphertext, form that can only be deciphered by someone who has the appropriate decryption key. The purpose of encryption algorithms is to ensure that sensitive data is kept confidential and secure, even if it is intercepted by an unauthorized third party.

There are many different encryption algorithms available, including symmetric and asymmetric algorithms. Very few of the most commonly used algorithms in encryption include **Advanced Encryption Standard (AES):** is defined as the symmetric-based encryption algorithm that is widely used for the encryption of data. It uses 122 bits for fixed block size and assists key sizes of 128, 192, and 256 bits; **RSA** is an asymmetric encryption algorithm that is widely used for transmitting data securely. It uses a public key for the encryption and a private key for decryption, respectively [16]; **Blowfish** is a symmetric encryption algorithm that uses variable-length key sizes and is known for its speed and efficiency; **Twofish** is a symmetric encryption algorithm that is similar to Blowfish but uses a 128-bit block size and supports key sizes up to 256 bits; **Triple DES** is a symmetric encryption algorithm that utilizes three rounds of the DES encryption algorithm to provide increased security.

We chose RSA as the most effective in countering the attacks with federated learning in SDN.

3.1 Federated Learning with RSA

RSA, a public-key cryptography algorithm that is based on the mathematical properties of prime numbers was first described in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman, whose first letters of their last names form the acronym RSA.

The RSA algorithm utilizes two keys, a public and a private key, for encryption and decryption purposes [17]. The public key is used for encrypting the messages, and the private key is used for decrypting the messages. This means that anyone can encrypt a message employing the recipient's public key, but the message can be decrypted by only the recipient, who holds the private key.

The security of public-key cryptography algorithms, such as RSA, is based on the computational infeasibility of certain mathematical problems, such as factoring large numbers or computing the discrete logarithm. This makes it difficult for an attacker to obtain the private key, even if they have access to the public key.

3.2 Proposed Architecture for cybersecurity with federated learning in SDN

The integration of Federated Learning and SDN can be used to build a secure cybersecurity architecture for networks as shown in Fig. 1. The following steps outline a general architecture for this integration [18]:

1. **Secure Data Collection:** Sensitive data from various sources is collected and securely stored in a centralized data repository. This data can then be used for training machine learning models.
2. **Federated Learning:** The decentralized devices in the network contribute to building a shared machine learning model without sharing the raw data with a central server. The devices only share the model updates with each other, which helps to protect the privacy of the data.
3. **Secure Communication:** Encryption and secure communication protocols, such as SSL/TLS, are used to protect the data in transit between the devices and the central server. This helps to prevent eavesdropping and tampering with the data.
4. **Access Control:** Access control mechanisms are implemented to ensure that only authorized devices have access to the sensitive data. This can be achieved through the use of authentication and authorization protocols.
5. **Intrusion Detection and Prevention:** Intrusion detection and prevention systems are implemented in the network to detect and prevent malicious traffic from spreading through the network. This can be achieved through the use of firewalls, intrusion detection systems, and other security measures.
6. **Network Management:** The centralized control plane in SDN is used to manage the network, including the implementation of security policies and configuration of security-related devices.

In the architecture, there are three IoT clusters which consist of various switches, routers and various IoT devices [11]. The attacks that are included in federated learning can be evaluated by deploying local data of IoT devices or the

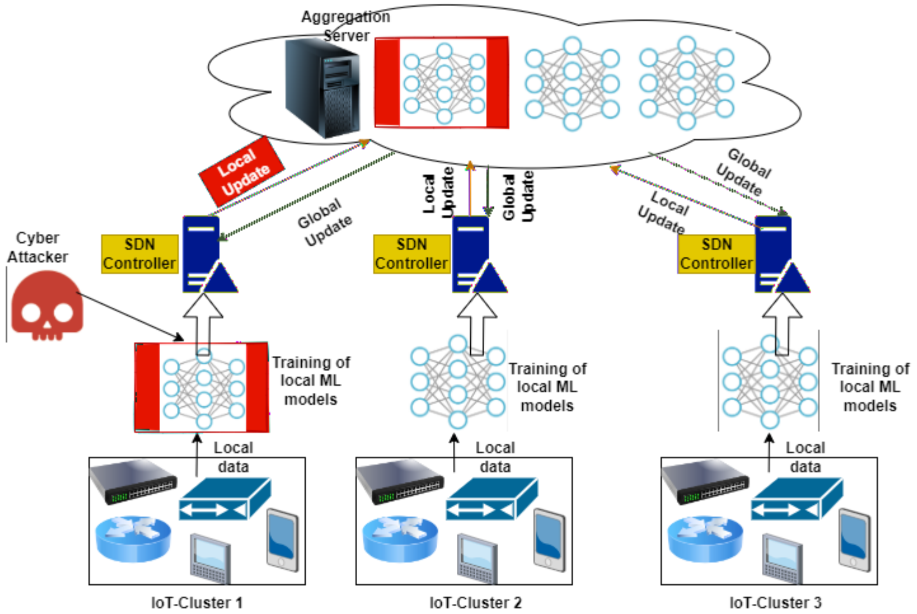

Fig. 1. Proposed architecture for detection of attacks with federated learning in SDN

model parameters that are there on the client or at the side of the server. The question arises ”how an attacker may execute the different attacks in federated learning?” By observation, it is said that the cyber attacker may control the IoT devices to compromise the local machine learning model to produce the model. The attacks did not degrade the performance of the machine learning model, but also exposed the user’s private information. The machine learning model is applied to the local data of IoT clusters for training, then the data is sent to the SDN controller and local updates are transferred to the aggregation server. The cyber attack happened on the model in IoT cluster 1, if the adversary gains control of the aggregation server, the attacker can get complete knowledge of the history of updated devices’ parameters and the structure of the global model during the training process done in a local model with the intention of generating a biased model. With this information, attackers can get the device’s privacy through the use of reverse engineering attacks. Federated learning is vulnerable to attacks that are carried out through devices in IoT clusters.

The proposed architecture offers a number of advantages, such as:

1. Since the data is encrypted before being delivered to the federated learning server, it safeguards the privacy of the data sources.
2. It enables the training of a federated learning model using data from different sources without requiring data sharing.
3. Due to the fact that the attack detection model is trained using data from many sources, it may increase accuracy.

There are other difficulties with the proposed architecture, such as:
1. One must have trust that the federated learning server to properly gather the data and train the model.
2. The output of the federated learning model must be accurately interpreted by the attack detection application.

Therefore, the architecture is a potential strategy for SDN network attack detection. In addition to allowing many data sources to train a federated learning model and preserving the privacy of the data sources, it can increase the precision of the attack detection model. Before the suggested architecture can be widely used, there are a few issues that must be resolved. Overall, the integration of Federated Learning and SDN in a cybersecurity architecture helps to ensure that sensitive data is protected and that the network is secure and reliable. This can help to promote trust in the network and ensure that it is used in a responsible and secure manner.

4 Performance analysis and Evaluation

4.1 Dataset

The N-BaIoT dataset [15] is a widely used dataset for intrusion detection in IoT networks. It contains benign and malicious traffic generated from real IoT devices such as smart homes, IP cameras, and smart plugs. The dataset was created by the National Institute of Standards and Technology (NIST) and can be used to train machine learning models for intrusion detection in IoT networks. The dataset contains over 140 million network flows and covers a wide range of attack types, including Distributed Denial of Service (DDoS), unauthorized access, and malware infections.

The N-BaIoT dataset is commonly used for detecting attacks in IoT domains for federated learning in SDN. It contains a large number of network-based attacks on IoT devices, including DDoS attacks, unauthorized access, and data manipulation. The dataset is designed to be used in a federated learning scenario, where multiple IoT devices can collaborate to train a shared machine learning model for attack detection.

4.2 Data Preprocessing

Data preprocessing is an important step in using the N-BaIoT dataset for intrusion detection. The following are some common preprocessing steps that can be applied to the N-BaIoT dataset:

Data cleaning: This involves removing any irrelevant, missing, or inconsistent data from the dataset. **Data normalization:** This involves scaling the data to a common range, such as [0, 1], to ensure that the features have equal importance in the analysis. **Data balancing:** This involves ensuring that the dataset has a balanced distribution of benign and malicious samples. An imbalanced dataset can lead to biased results and poor performance of machine learning models. **Feature selection:** This involves choosing a subset of relevant features from the dataset to improve the performance of the machine learning models and reduce the dimensionality of the data [2]. **Data splitting:** This involves splitting the dataset into training and testing sets. The training set is employed for training the machine learning models, the validation set is employed for tuning the model hyperparameters, and the testing set is employed for evaluating the performance of the models.

4.3 Feature Extraction

Extraction of features is defined as the process of transforming raw data into a set of relevant and informative features that can be used to train machine learning models. In the case of the N-BaIoT dataset, the following features can be extracted for intrusion detection:

Flow-based features: These features capture information about the flow of network traffic, including the number of packets, the size of the payload, and the duration of the flow.

Protocol-based features: These features capture information about the protocols used in the network, including the type of the protocol, the source and destination ports, and the flags set in the protocol header.

Content-based features: These features capture information about the con-

tent of the network traffic, including the frequency of specific keywords or patterns.
Statistical features: These features capture information about the distribution of network traffic, including the mean and standard deviation of the flow size and duration.

4.4 Models used for evaluation

The following are some common models used for intrusion detection on the N-BaIoT dataset:

1. **Decision Trees:** This is a tree-based model that can be used to model the relationships between the features and the target variable. Decision Trees divide the feature space into a series of regions, with each region corresponding to a different class [3]. They're simple to interpret and can be used for both binary and multiclass classification problems.
2. **Random Forest:** This is an ensemble of decision trees that can be used to improve the accuracy of the predictions by combining the outputs of multiple trees. Random Forest creates multiple decision trees on randomly sampled subsets of the training data, and the final prediction is made by combining the predictions of all the trees.
3. **K-Nearest Neighbor (KNN):** The choice of the value of K is an important decision in KNN. A larger value of K means that the prediction is based on a larger number of neighbors, which may improve accuracy but may also introduce more noise. A smaller value of K means that the prediction is based on a smaller number of neighbors, which may be more sensitive to outliers but may also be more accurate in some cases.

5 Results and Discussions

The performance of different models on the N-BaIoT dataset can be compared using various performance metrics, including accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve (AUC-ROC).

There are two types of attacks: Mirai and Bashlite. The main aim of binary classification is to categorize malicious and normal traffic. The dataset is trained with three classifiers: Random Forest, K-Nearest Neighbor and Decision tree. The number of features changes as the accuracy improves. Random Forest is considered as the most effective because of its performance in comparison with K-Nearest Neighbor and Decision Tree. The selected number of features are 2, 3, 5, 10 and 20, demonstrating unique results.

The table 1 focus on the binary classification of Mirai attacks, in which there are 20 features with regard to the various models. The minimum number of attacks in a decision tree with three features achieves 98.8% as compared to the other models. The figure 2 shows the comparative analysis in graphical form to highlight the accuracy amongst different models. As depicted, decision tree with green line showing the minimum number of attacks.

Table 1. Binary Classification with Mirai Attack

Number of features	**Random Forest**	**K-Nearest Neighbor**	**Decision Tree**
2	99.1	99.3	98.8
3	99.1	99.3	98.8
5	99.3	99.5	99.0
10	99.5	99.6	99.0
20	99.6	99.8	99.3

Table 2 focuses on the binary classification of Bashlite attacks, in which there are 10 features with regard to the various models. The minimum number of attacks in a decision tree with two features achieves 94.6% as compared to the other models. The figure 3 shows the comparative analysis in graphical form to highlight the accuracy amongst different models. As depicted, decision tree with green line showing the minimum number of attacks.

Table 2. Binary Classification with Bashlite Attack

Number of features	**Random Forest**	**K-Nearest Neighbor**	**Decision Tree**
2	96.3	96.5	94.6
3	96.5	95.5	95.5
5	96.6	95.5	94.8
10	97.8	97.3	96.3
20	97.6	96.6	96.5

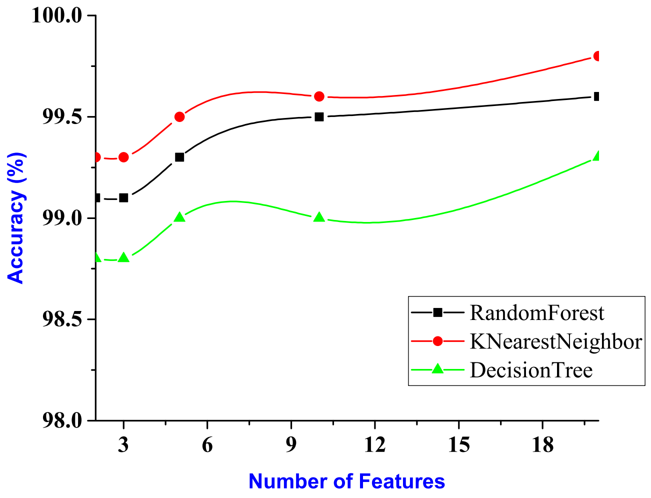

Fig. 2. Binary Classification with Mirai Attack

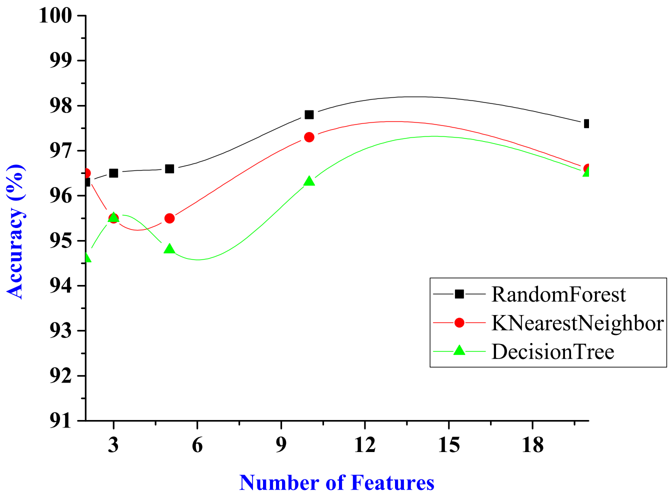

Fig. 3. Binary Classification with Bashlite Attack

6 Conclusions

In conclusion, the use of federated learning in SDN can provide an effective solution for detecting attacks in cyber security. FL allows for the aggregation of data from multiple sources without requiring data to be centralized, thereby maintaining the privacy and security of individual data sources. In the context of SDN, this approach can enable more accurate and efficient detection of attacks, while minimizing the risk of false positives and false negatives. The architecture is proposed for the detection of attacks with FL in SDN for secure data transfer. The performance analysis has been done using the NBaIoT dataset with baseline models of decision tree, random forest, and K-nearest neighbor, in which the proposed model achieved maximum accuracy as compared to the state-of-the-art models. However, further research is essential to evaluate the performance of FL-based solutions in real-world scenarios and address potential challenges such as communication overhead and privacy concerns.

Acknowledgements

The work was partially supported by the project MUSA ”Multilayered Urban Sustainability Action” project, funded by the European Union ”NextGenerationEU”, (CUP G43C22001370007, Code ECS00000037). The work was also partially supported by the project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NextGenerationEU.

References

1. Abou El Houda, Z., Hafid, A.S., Khoukhi, L.: Mitfed: A privacy preserving collaborative network attack mitigation framework based on federated learning using sdn and blockchain. IEEE Transactions on Network Science and Engineering (2023)
2. Alazab, M., RM, S.P., Parimala, M., Maddikunta, P.K.R., Gadekallu, T.R., Pham, Q.V.: Federated learning for cybersecurity: concepts, challenges, and future directions. IEEE Transactions on Industrial Informatics **18**(5), 3501–3509 (2021)
3. Ali, M.N., Imran, M., din, M.S.u., Kim, B.S.: Low rate ddos detection using weighted federated learning in sdn control plane in iot network. Applied Sciences **13**(3), 1431 (2023)
4. Anand, A., Rani, S., Anand, D., Aljahdali, H.M., Kerr, D.: An efficient cnn-based deep learning model to detect malware attacks (cnn-dma) in 5g-iot healthcare applications. Sensors **21**(19), 6346 (2021)

5. Balasubramanian, V., Aloqaily, M., Reisslein, M., Scaglione, A.: Intelligent resource management at the edge for ubiquitous iot: An sdn-based federated learning approach. IEEE network **35**(5), 114–121 (2021)
6. Balyan, A.K., Ahuja, S., Lilhore, U.K., Sharma, S.K., Manoharan, P., Algarni, A.D., Elmannai, H., Raahemifar, K.: A hybrid intrusion detection model using ega-pso and improved random forest method. Sensors **22**(16), 5986 (2022)
7. Duy, P.T., Hung, T.V., Ha, N.H., Hoang, H.D., Pham, V.H.: Federated learning-based intrusion detection in sdn-enabled iiot networks. In: 2021 8th NAFOSTED Conference on Information and Computer Science (NICS). pp. 424–429 (2021). https://doi.org/10.1109/NICS54270.2021.9701525
8. Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H.: Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning. IEEE Access **10**, 40281–40306 (2022)
9. Gebremariam, G.G., Panda, J., Indu, S., et al.: Blockchain-based secure localization against malicious nodes in iot-based wireless sensor networks using federated learning. Wireless Communications and Mobile Computing **2023** (2023)
10. Hbaieb, A., Ayed, S., Chaari, L.: Federated learning-based ids approach for the iov. In: Proceedings of the 17th International Conference on Availability, Reliability and Security. pp. 1–6 (2022)
11. Huong, T.T., Bac, T.P., Long, D.M., Luong, T.D., Dan, N.M., Thang, B.D., Tran, K.P., et al.: Detecting cyberattacks using anomaly detection in industrial control systems: A federated learning approach. Computers in Industry **132**, 103509 (2021)
12. Kapoor, K., Rani, S., Kumar, M., Chopra, V., Brar, G.S.: Hybrid local phase quantization and grey wolf optimization based svm for finger vein recognition. Multimedia Tools and Applications **80**, 15233–15271 (2021)
13. Li, B., Wu, Y., Song, J., Lu, R., Li, T., Zhao, L.: Deepfed: Federated deep learning for intrusion detection in industrial cyberphysical systems. IEEE Transactions on Industrial Informatics **17**(8), 5615–5624 (2021). https://doi.org/10.1109/TII.2020.3023430
14. Ma, X., Liao, L., Li, Z., Lai, R.X., Zhang, M.: Applying federated learning in software-defined networks: A survey. Symmetry **14**(2), 195 (2022)
15. Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., Elovici, Y.: N-baiotnetwork-based detection of iot botnet attacks using deep autoencoders. IEEE Pervasive Computing **17**(3), 12–22 (2018)
16. Mothukuri, V., Khare, P., Parizi, R.M., Pouriyeh, S., Dehghantanha, A., Srivastava, G.: Federated-learning-based anomaly detection for iot security attacks. IEEE Internet of Things Journal **9**(4), 2545–2554 (2021)
17. Preuveneers, D., Rimmer, V., Tsingenopoulos, I., Spooren, J., Joosen, W., Ilie-Zudor, E.: Chained anomaly detection models for federated learning: An intrusion detection case study. Applied Sciences **8**(12), 2663 (2018)

18. Rahman, S.A., Tout, H., Talhi, C., Mourad, A.: Internet of things intrusion detection: Centralized, on-device, or federated learning? IEEE Network **34**(6), 310–317 (2020)
19. Ramesh, T., Lilhore, U.K., Poongodi, M., Simaiya, S., Kaur, A., Hamdi, M.: Predictive analysis of heart diseases with machine learning approaches. Malaysian Journal of Computer Science pp. 132–148 (2022)
20. Zhang, L., Xu, J., Vijayakumar, P., Sharma, P.K., Ghosh, U.: Homomorphic encryption-based privacy-preserving federated learning in iot-enabled healthcare system. IEEE Transactions on Network Science and Engineering (2022)