This article has been accepted for publication in GIIS2022. Content may change prior to final publication. Citation information: DOI 10.1109/GIIS56506.2022.9936912, 2022 Global Information Infrastructure and Networking Symposium (GIIS)

Enhancing Cyber Security in IoT Systems using FL-based IDS with Differential Privacy

Zacharias Anastasakis
Synelixis Solutions SA
Athens, Greece
anastasakis@synelixis.com

Konstantinos Psychogyios
Synelixis Solutions SA
Athens, Greece
psychogios@synelixis.com

Terpsi Velivassaki
Synelixis Solutions SA
Athens, Greece
terpsi@synelixis.com

Stavroula Bourou
Synelixis Solutions SA
Athens, Greece
bourou@synelixis.com

Artemis Voulkidis
Synelixis Solutions SA
Athens, Greece
voulkidis@synelixis.com

Dimitrios Skias
Netcompany-Intrasoft S.A.
Athens, Greece
Dimitrios.Skias@netcompany-intrasoft.com

Antonis Gonos
Entersoft S.A.
Athens, Greece
agn@entersoft.gr

Theodore Zahariadis
National & Kapodistrian University of Athens
Chalkida, Greece
zahariad@uoa.gr

***Abstract*—Nowadays, IoT networks and devices exist in our everyday life, capturing and carrying unlimited data. However, increasing penetration of connected systems and devices implies rising threats for cybersecurity with IoT systems suffering from network attacks. Artificial Intelligence (AI) and Machine Learning take advantage of huge volumes of IoT network logs to enhance their cybersecurity in IoT. However, these data are often desired to remain private. Federated Learning (FL) provides a potential solution which enables collaborative training of attack detection model among a set of federated nodes, while preserving privacy as data remain local and are never disclosed or processed on central servers. While FL is resilient and resolves, up to a point, data governance and ownership issues, it does not guarantee security and privacy by design. Adversaries could interfere with the communication process, expose network vulnerabilities, and manipulate the training process, thus affecting the performance of the trained model. In this paper, we present a federated learning model which can successfully detect network attacks in IoT systems. Moreover, we evaluate its performance under various settings of differential privacy as a privacy preserving technique and configurations of the participating nodes. We prove that the proposed model protects the privacy without actually compromising performance. Our model realizes a limited performance impact of only ~ 7% less testing accuracy compared to the baseline while simultaneously guaranteeing security and applicability.**

Keywords—Federated Learning, Internet of Things, Differential Privacy, Privacy Preservation, Cyber Security

I. INTRODUCTION

Artificial Intelligence (AI) plays a significant role in Internet of Things (IoT) applications. The main added value of AI lies in its ability to provide insights, by automatically identifying patterns and detecting anomalies on data collected from IoT sensors and other devices. Machine Learning (ML), as a specific area of AI that trains machines on how to learn from data, provides significant benefits across application domains, arising from proactive intervention, tailored experiences and intelligent automation. ML is almost everywhere, from small wearable devices and smartphones to powerful super-computers ensuring fast and accurate data analysis. IoT devices generate large amounts of data, which are a real wealth for ML applications.

On the other hand, IoT systems and devices are vulnerable to a wide range of attacks, including those at the network level. Modern connected systems may significantly increase their cybersecurity levels via ML-based intrusion detection systems. Specifically, ML applied on the network logs within a corporate network is able to learn data patterns and thus identify potential attacks as anomalies in the network traffic. The performance of such models could be significantly enhanced by combining the insights from different administrative domains. However, traditional cloud computing applications would need the data to be uploaded and processed on a central server giving data access to third parties, which would raise significant concerns about privacy and ownership of those data.

To address these concerns Google introduced Federated Learning [1] which is a distributed machine learning approach. FL aims to build and train global models based on training datasets that are distributed across different remote devices while avoiding data leakage. The data is never processed on central servers, decoupling the machine learning process from the data sources. In practice, FL solutions train an initial, generic machine learning model in a central server, which is a baseline to start with. Afterwards, the server sends this model to the user's devices, where the local copy of the model is trained using its own data. Then, the updated model parameters are sent back to the central server and the global model is updated. Therefore, FL approaches are capable of learning robust models from a huge amount of distributed data across IoT devices without transferring and/or processing it on a central server. However, FL systems may suffer, as well, from malicious activity, which may affect the training process. Including privacy preservation techniques within the FL system introduces a trade-off between the privacy preservation level and the model performance.

This article has been accepted for publication in GIIS2022. Content may change prior to final publication. Citation information: DOI 10.1109/GIIS56506.2022.9936912, 2022 Global Information Infrastructure and Networking Symposium (GIIS)

That said, we introduce an anomaly attack detection model trained on the widely used NSL-KDD dataset [2], which combines the federated learning architecture and privacy preserving techniques in order to enhance the cyber security in IoT systems by classifying network attacks to "DoS" or "Not DoS". Training our model in a federated manner, knowledge is transferred from one IoT device to another resulting in better understanding of possible attack types in the future. We perform an extensive analysis of how differential privacy [3], [4] is affecting the attack detector's accuracy under several, real-world scenarios, resulting in creating a robust model for attack detection. Our proposed model can successfully be adopted to [5], where the authors propose a novel and intelligent Mobile Ad-Hoc Network for the detection, identification and recording events on a given traffic network.

The rest of the paper is organized as follows. In section II, we provide a literature review of related works. In section III, we describe our experimental components and we present our results acquired from training procedures. Finally in section IV we conclude the paper.

II. Related Work and Backround

A. Differential Privacy

While FL is resilient and resolves, up to a point, data governance and ownership issues, it does not guarantee security and privacy by design. A lack of encryption can allow adversaries to abduct personally identifiable data directly from the processing nodes or interfere with the communication process, expose network vulnerabilities, and perform attacks. In addition, the decentralized nature of the data complicates data handling and curation. Federated Learning can be vulnerable to various backdoor threats (bug injection, inference & model attacks) on different processing steps. Therefore, additional measures are essential to protect data from adversarial attack strategies such as data poisoning and model poisoning attacks. The major approaches that can be employed in FL for data protection are differential privacy, homomorphic encryption [6], and secure multiparty computation [7], [8]. In this paper, we will examine the differential privacy approach. Differential Privacy (DP) is a method that randomizes part of the mechanism's behavior to provide privacy [3], [4]. The motivation behind adding randomness (Gaussian noise) into a learning algorithm is to make it impossible to reveal data patterns or insights that correspond either to the model and the learned parameters or to the training data. Therefore, the DP provides privacy against a wide range of attacks (e.g., differencing attacks, linkage attacks) [9]. The method of introducing noise to the data can result in great privacy but may compromise accuracy. In differential privacy techniques, a central aggregator exists which has access to the raw data. Generally, differential privacy may be divided into Local Differential Privacy (LDP) [10], and global differential privacy (GDP) [11]. LDP is a state-of-the-art approach which allows statistical computations while simultaneously protecting each individual user's privacy, as shown in Fig. 1. No trust limitations to a central authority or a third party are necessary since noise is added to the individual inputs locally.

In global differential privacy techniques, a central aggregator exists (i.e., a trusted curator) which has access to the raw data, as depicted in Fig. 2. In particular, each user sends their data to the aggregator node without adding noise. The aggregator then considers the input data and transforms it with a differentially private mechanism, by adding Gaussian noise. When an untrusted querier makes a specific query on the trusted aggregator node, an answer shall be provided, however, this answer is mathematically impossible to be reverse-engineered, and consequently it is impossible to know the precise answer about the private raw data.

Generally, global private systems are more accurate, since all the analysis is implemented on "clean" (i.e. noise-free) data, and only a small amount of noise is added at the end of the process. However, the efficiency of global privacy models lies in the users' amount of trust in the trusted curator.

Abadi [12] introduced a Differentially Private Stochastic Gradient Descent (DP-SGD) algorithm which aims to control the effect of the training data during the optimization operation (GD). For each step, the DP-SGD algorithm computes the gradient for a random set of data, calculates the clipped l_2 norm of each gradient computes the average, adds noise to preserve privacy and takes a step in the opposite direction of this SGD.

B. Network Intrusion Detection Systems

In the past few years, the variety and complexity of cyber-attacks and malicious events has grown tremendously. For that reason, the design of a robust Intrusion Detection System becomes a high priority need. Intrusion detection systems monitor networks for potentially malicious activity and policy violations. Several approaches have been proposed in the related literature [13], [14]. The detection method of IDS can be divided into two categories, the signature-based method and the anomaly-based method. The signature-based method detects on the basis of the already known malicious instruction sequence that is used by the malware. The detected patterns in the IDS are known as signatures. The anomaly-based IDS was introduced to detect unknown malware attacks as new malware are developed rapidly. In anomaly-based IDS there is use of machine learning to create a trustful activity model and anything coming is compared with that model and it is declared suspicious if it is not found in the model.

In [15] the authors investigate the possibilities enabled by federated learning concerning IoT malware detection, study security issues and propose a framework that uses federated learning to detect malware affecting IoT devices.

Rahman [16] proposed a Federated Learning based intrusion detection scheme for IoT systems that maintains data privacy and detects network threads could occur over an FL setting. The authors evaluate their method across multiple use-cases in order to simulate real-world scenarios and conclude that the federated learning setting can achieve results in accuracy as high as a centralized approach.

Authors in [17], introduced an FL approach for malicious activity detection in IoT devices. They created and trained an intrusion detection model for the security of IoT devices and

This article has been accepted for publication in GIIS2022. Content may change prior to final publication. Citation information: DOI 10.1109/GIIS56506.2022.9936912, 2022 Global Information Infrastructure and Networking Symposium (GIIS)

manage to keep private sensitive data. Their method has been evaluated in several use-cases similar to [16].

The contribution of the present paper is to introduce new attack detection model architectures that would achieve increased accuracy, while preserving privacy of the local data.

III. Proposed Model Architecture

This paper proposes an FL-based, privacy preserving intrusion detection system (IDS) for the detection of DoS attacks in IoT networks. Our model consists of 2 fully connected Dense layers, an input layer and an output layer. The output layer is responsible for the binary classification task, and is selected to be activated with the SoftMax activation function. In addition, we use the Stochastic Gradient Descent (SGD) as our optimizer with 0.001 learning rate and the loss function is the Binary Cross-Entropy (BCE) since our goal is binary classification ("DoS" or "Not DoS" attacks). Our proposed model architecture follows the standard Federated Learning training process. A central Server loads the initial, generic global model. For each federated round, the Server sends the model to its clients and each individual client trains the model in their corresponding local, sensitive dataset. After the training procedure of all clients is finished, each client transforms its model's weights with the Local Differential Privacy mechanism by adding Gaussian noise. Finally, the clients send back to the central Server the noisy-aggregated model updates, where the Server updates the global model by averaging the updated weights using the FedAvg [1] method. After model's training process is finished, the model can successfully classify if there is a "DoS" attack or not and can be applied to IoT systems to detect anomaly attacks in networks as shown in Fig. 3.

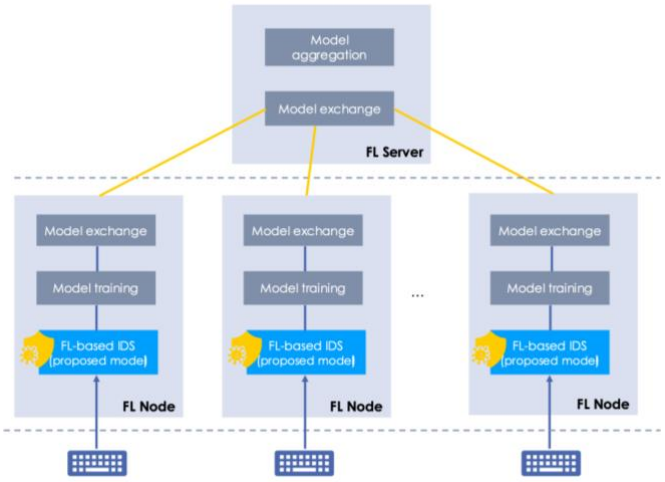

Figure 3: IoT System with our proposed model applied.

The following paragraphs are dedicated to the binary classification problem of network records under the following scenarios: the number of clients in the Federated System, the quantity of noise addition in Differential Privacy and the scenario of imbalance client's dataset.

A. Number of clients in the FL System

In this scenario the effect of the number of clients on model's performance is examined. As the total number of clients in our IoT system is increased, the security of each individual IoT device (client) is enhanced because each client transfers her knowledge to the others while they are training in the form of federated learning.

B. Noise addition in Differential Privacy

The main component of our proposed architecture is the Differential Privacy. Adding noise to our model can lead to higher data privacy resulting in keeping the data of each IoT device secure. Therefore, there is a trade-off between applying differential privacy and achieving a high level of model accuracy. In this scenario, the effect of the noise addition on model's accuracy is investigated.

C. Imbalance Client's dataset

In real-world scenarios the local dataset size of each IoT device may differ from each other affecting the entire IoT systems' performance. To define the imbalance level of each client's dataset we introduce a new component *a*, a factor that defines how imbalanced each client's dataset will be. A value of 1 means no imbalance, whereas higher values than 1 assign datasets to clients with higher imbalance.

IV. Experimental Evaluation

A. Experimental Setup

Dataset**.** We use the NSL-KDD [2] dataset for our experiments. For its features, it is widely used for the testing and evaluation of intrusion detection systems. The dataset is composed of 257673 cyber-attacks labeled in 5 different classes as *Dos*, *Probe*, *U2R*, *R2L* (attack types) and *Normal* (no attack). We modify the dataset in a way that we have 2 classes, converting our problem to binary classification problem, where the labels are "No DoS" and "DoS". That said, "No DoS" encapsulates normal and other kind of attack samples. Consequently, our goal is only to detect Denial of Service attacks.

B. Environment Setup

Google proposed TensorFlow Federated [18] an open-source framework for machine learning and other computations on decentralized data. TensorFlow Federated includes TensorFlow Privacy, a python library for applying privacy techniques and provides two different APIs, one for model training and one for creating custom federated algorithms. That said, we choose Tensorflow Federated framework for our experiments.

C. Metrics

The overall performance of our model is based on the accuracy metric. The accuracy measures the impact of the differential privacy on the efficiency of the model. The accuracy is defined as:

$$accuracy = \frac{\sum_{i \in Dt}(pred(x_i) = j | y_i = j)}{n_t}$$

This article has been accepted for publication in GIIS2022. Content may change prior to final publication. Citation information: DOI 10.1109/GIIS56506.2022.9936912, 2022 Global Information Infrastructure and Networking Symposium (GIIS)

where D_t is the test set, n_t is the size of the D_t, x_i is the input and y_i is the true label.

D. Experimental Results

As described in section II, our model consists of two fully connected layers, the input and the output layer (i.e., classification layer). Specifically, the output layer is activated with the SoftMax function and has 1 unit (neural). It assigns the 0 or 1 depending on the class it classifies, 0 for "Not DoS" class and 1 for "DoS" class. The number of hyper-parameters were evaluated via an extended validation approach. The number of batch size was set to 512, the number of epochs was set to 2 and the number of federated rounds was set to 25. We observe that within the interval of 2 epochs and 25 rounds we achieve highly robust results in terms of accuracy. Our baseline model was trained in a federated learning scenario with one client (centralized case), whereas the others are all trained in a federated learning manner.

Regarding the first scenario, where we investigate the number of clients in the FL system, we experiment with the values 1 (baseline model), 10, 25 and 50. Fig. 4 illustrates the impact that the number of clients has on the model's accuracy. The total accuracy achieved for the testing phase is 79.06%, 78.95%, 77.88% and 75.54% respectively. We observe that a higher number of clients results in accuracy degradation. This is to be expected since splitting the dataset in smaller sets eliminates certain relationships that define the data. The difference can be as high as 3.52%.

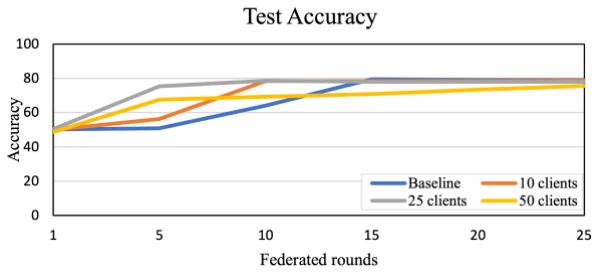

Figure 4: Impact of clients' number.

In the second scenario, we experiment with the differential privacy that our model preserves. The parameter which controls the noise addition quantity is called noise multiplier [18]. The parameter noise multiplier is the ratio of the noise standard deviation to the clipping norm and we experiment with the values 0, 0.5, 1, 1.5 and 2 where the value 0 refers to a model trained with no DP, a classical federated learning model as introduced to its original paper [1]. As the value of noise multiplier is increased, the more Gaussian noise is added to the model and so more privacy is preserved. In Fig. 5, we observe that adding more noise has a negative impact on the model's accuracy and adding too much may cause the model to collapse. In particular, our model achieves 79.06%, 75.85%, 73.49%, 63.33% and 50.58% test accuracy with respect to the values of the *noise multiplier* mentioned above. There is a trade-off between privacy preservation and accuracy and this is something that has to be fine-tuned for each specific situation. For small portions of noise, the difference in accuracy is between 3.21-15.73 % bounds, whereas for bigger ones can be as high as 28.48%.

For the last scenario, we experiment with the values of 1 (no imbalance), 1.2, 1.5 and 2 for the component a. The quantity and quality of samples that each client has for these values of the component a during their training procedure, are shown in Fig. 6, while Fig. 7 reflects the impact of the component a (imbalanced level) on model's accuracy. In particular, our model achieves 79.06%, 77.42%, 76.98% and 75.11% with respect to the values for the component a mentioned above. As expected, while a is being increased, the model's accuracy becomes worst. It can be seen that higher imbalance (higher a) leads to worse results. Specifically, the range of accuracy degradation is 1.42-3.95%.

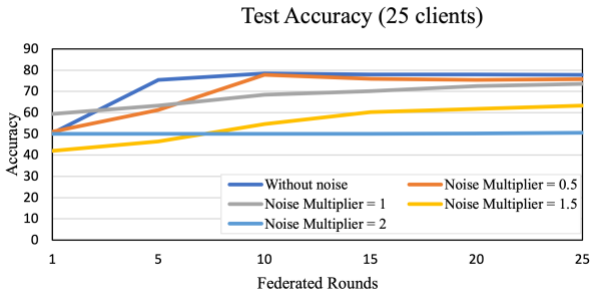

Figure 5: Impact of noise on model's accuracy.

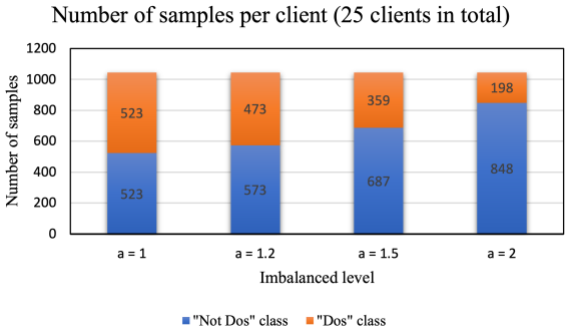

Figure 6: Number of samples per client for different values of a.

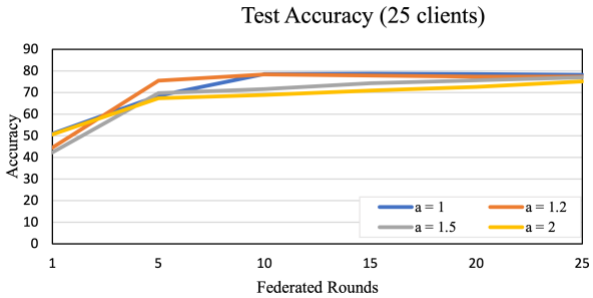

Figure 7: Impact of the component a on model's accuracy.

Considering all the scenarios mentioned above, we propose an FL-based IDS which performs accurately under real world settings. We assume that a scenario of a = 2 (imbalance ratio ~

This article has been accepted for publication in GIIS2022. Content may change prior to final publication. Citation information: DOI 10.1109/GIIS56506.2022.9936912, 2022 Global Information Infrastructure and Networking Symposium (GIIS)

2.5:1) represents the real-world circumstances since malicious packets appear less frequently in network traffic logs. Furthermore, a value of *noise multiplier* = 1.5 is sufficient to provide both security guaranties and viable performance since experiments show that higher values cause model failure. This combination achieves high privacy concerning data as well as minimizes the risk of sacrificing model's performance. Thus, the experiment described below reflects realistic applications of IDS in IoT Systems. Fig. 8 plots the proposed model and the baseline model. Our proposed model consists of 25 clients. In such a case, we can see that the model performs decently compared to the baseline and is a viable solution for real world applications. The resulting model after 25 rounds achieves accuracy that is only 7% lower compared to the baseline.

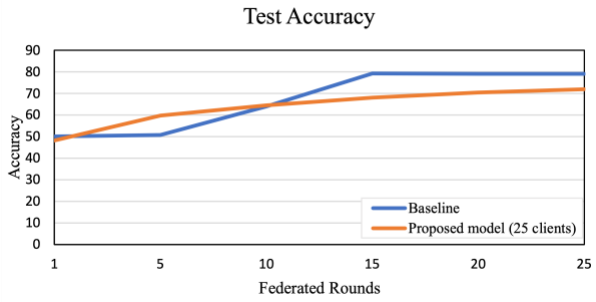

Figure 8: Proposed model vs baseline model.

V. Conclusion

Within Federated Learning, intrusion detection systems can be trained and deployed to enhance IoT cybersecurity. However, such an application faces many challenges, including imbalanced classes on clients' data, privacy concerns regarding sensitive data and number of clients. In this paper, we examine those challenges and provide a privacy preserving IDS which can successfully detect attacks on IoT devices, while it can be applied in real world problems and scenarios. Additionally, we evaluate the model performance under different settings and configurations. More specifically, we show how the model performance is affected when (i) client data are imbalanced regarding the labels, (ii) number of clients increases, (iii) additive noise of differential privacy is changed. Through experiments, we demonstrate that the proposed privacy preserving IDS achieves an acceptable performance compared to the baseline, namely 7% accuracy drop. In the future, we aim to expand the ideas of this paper to different datasets that contain additional types of cyberattacks and further test whether and how different types of additive noise (e.g., Laplacian) of differential privacy or greater number of clients, affect the model's performance.

Acknowledgment

The work presented in this document was funded through H2020 IoT-NGIN project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 957246.

References

[1] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson and Blaise Aguera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data".

[2] UNB, "Nsl-kdd dataset," :https://www.unb.ca/cic/datasets/nsl.html///, Accessed April 20, 2021.

[3] M. Abadi et al., "Deep learning with differential privacy," in Proceedings of the ACM Conference on Computer and Communications Security, 2016, vol. 24-28-Octo, no. Ccs, pp. 308–318. doi: 10.1145/2976749.2978318.

[4] F. McSherry and K. Talwar, "Mechanism Design via Differential Privacy," pp. 94–103, 2008, doi: 10.1109/focs.2007.66.

[5] Kalapodi, Angeliki & Sklavos, Nicolas & Zaharakis, I. & Kameas, Achilles. (2018). A Safe Traffic Network Design and Architecture, in the Context of IoT.

[6] C. Lefebvre, "On data Banks and Privacy Homomorphisms," Journal of Pidgin and Creole Languages, vol. 15, no. 2, pp. 313–337, Dec. 2000, doi: 10.1075/jpcl.15.2.04lef.

[7] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," ACM Transactions on Intelligent Systems and Technology, vol. 10, no. 2, pp. 1–19, 2019, doi: 10.1145/3298981.

[8] C. Zhao et al., "Secure Multi-Party Computation: Theory, practice and applications," Information Sciences, vol. 476, pp. 357–372, 2019, doi: 10.1016/j.ins.2018.10.024.

[9] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," Foundations and Trends in Theoretical Computer Science, vol. 9, no. 3–4, pp. 211–487, 2013, doi: 10.1561/0400000042.

[10] P. C. Mahawaga Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman, "Local Differential Privacy for Deep Learning," IEEE Internet of Things Journal, vol. 7, no. 7, pp. 5827–5842, 2020, doi: 10.1109/JIOT.2019.2952146.

[11] J. Lee and C. Clifton, "How much is enough? Choosing ε for differential privacy," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7001 LNCS, pp. 325–340, 2011, doi: 10.1007/978-3-642-24861-0_22.

[12] M. Abadi et al., "Deep learning with differential privacy," in Proceedings of the ACM Conference on Computer and Communications Security, 2016, vol. 24-28-Octo, no. Ccs, pp. 308–318. doi: 10.1145/2976749.2978318.

[13] Samrin, R.; Vasumathi, D. Review on anomaly based network intrusion detection system. In Proceedings of the 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, India, 15–16 December 2017.

[14] Sultana, N.; Chilamkurti, N.; Peng,W.; Alhadad, R. Survey on SDN based network intrusion detection system using machine learning approaches. Peer-Peer Netw. Appl. 2019, 12, 493–501.

[15] Rey, Valerian, Pedro Miguel S'anchez S'anchez, Alberto Huertas Celdr'an, Gérôme Bovet and Martin Jaggi. "Federated Learning for Malware Detection in IoT Devices." Comput. Networks 204 (2022): 108693.

[16] S. A. Rahman, H. Tout, C. Talhi and A. Mourad, "Internet of Things Intrusion Detection: Centralized, On-Device, or Federated Learning?," in IEEE Network, vol. 34, no. 6, pp. 310-317, November/December 2020, doi: 10.1109/MNET.011.2000286.

[17] O. Shahid, V. Mothukuri, S. Pouriyeh, R. M. Parizi and H. Shahriar, "Detecting Network Attacks using Federated Learning for IoT Devices," 2021 IEEE 29th International Conference on Network Protocols (ICNP), 2021, pp. 1-6, doi: 10.1109/ICNP52444.2021.9651915.

[18] K. Bonawitz et al., "TensorFlow Federated: Machine Learning on Decentralized Data." 2020. [Online]. Available: https://www.tensorflow.org/federated