F-BIDS: Federated-Blending based Intrusion Detection System

Ons Aouedi, Kandaraj Piamrat

F-BIDS: Federated-Blending based Intrusion Detection System

Ons Aouedi[a,*], Kandaraj Piamrat[a]

[a]Nantes Université, École Centrale Nantes, CNRS, INRIA, LS2N, UMR 6004, F-44000, Nantes, France

ARTICLE INFO

Keywords:
Data privacy
Federated Learning
ensemble learning
Deep Learning
Intrusion Detection System

ABSTRACT

The rapid development of network communication along with the drastic increase in the number of smart devices has triggered a surge in network traffic, which can contain private data and in turn affect user privacy. Recently, Federated Learning (FL) has been proposed in Intrusion Detection Systems (IDS) to ensure attack detection, privacy preservation, and cost reduction, which are crucial issues in traditional centralized machine-learning-based IDS. However, FL-based approaches still exhibit vulnerabilities that can be exploited by adversaries to compromise user data. At the same time, meta-models (including the blending models) have been recognized as one of the solutions to improve generalization for attack detection and classification since they enhance generalization and predictive performances by combining multiple base models. Therefore, in this paper, we propose a Federated Blending model-driven IDS framework for the Internet of Things (IoT) and Industrial IoT (IIoT), called `F-BIDS`, in order to further protect the privacy of existing ML-based IDS. The proposition consists of a Decision Tree (DT) and Random Forest (RF) as base classifiers to first produce the meta-data. Then, the meta-classifier, which is a Neural Networks (NN) model, uses the meta-data during the federated training step, and finally, it makes the final classification on the test set. Specifically, in contrast to the classical FL approaches, the federated meta-classifier is trained on the meta-data (composite data) instead of user-sensitive data to further enhance privacy. To evaluate the performance of `F-BIDS`, we used the most recent and open cyber-security datasets, called *Edge-IIoTset* (published in 2022) and *InSDN* (in 2020). We chose these datasets because they are recent datasets and contain a large amount of network traffic including both malicious and benign traffic.

1. Introduction

Recently, strict laws such as the General Data Protection Regulation (GDPR) in European Union[1] have completely redefined the data management policy. As a result, Federated Learning (FL) has appeared as a valuable approach in enabling collaborative training of ML-based models among several clients/devices under privacy restrictions. More specifically, it has been introduced by Google as a decentralized approach [1] in order to decouple the model training from the need for direct access to the end-user data. Since FL was introduced in order to keep the data where they are generated and to preserve data privacy, several solutions have attempted to integrate FL into real-life applications [2] including Intrusion Detection Systems (IDS) [3] [4].

In fact, IDS is considered an important tool for network security as it is used to detect attacks from incoming network traffic. However, despite that FL-based approaches exchange the model parameters instead of the raw data, the recent attacks (e.g., membership inference attack) demonstrate that such an approach does not provide sufficient privacy guarantee [5] [6]. This is because the FL training process is based on the communication between the clients and the FL server that can expose the model and in turn be a target for several security threats. For example, Shokri *et al.* [7]

*Corresponding author
ons.aouedi@ls2n.fr (O. Aouedi)
ORCID(s): 0000-0002-2343-0850 (O. Aouedi)

[1]https://gdpr-info.eu/issues/data-protection-officer/

demonstrated inference attacks and in particular membership inference attacks can leak private information about their data owners. The membership inference attacks aim to determine whether or not target data was used to train the target model (victim model) [8]. To solve these issues, FL's serious privacy concerns motivated several privacy-preservation approaches against membership inference attacks such as Differential Privacy (DP) and encryption. Unfortunately, these approaches produce federated models with unacceptable computation overhead and classification performance. As a consequence, the FL-based IDS requires a new solution with a better guarantee of security and privacy while keeping good performance.

Contribution

Based on the above motivations, in this paper, we design a Federated Blending model for IDS, called `F-BIDS`, in order to reduce the reverse engineering attack on the FL training process, while keeping costs low. The main purpose behind our model is to incorporate only the meta-classifier into the FL framework in order to further enhance data privacy. In particular, the base classifiers, which are Random Forest (RF) and Decision Tree (DT), will be trained in the first communication round only on the client's side using their private data. After that, the meta-classifier, which is Neural Network (NN) model, uses only the meta-data generated by the base classifiers for the training process. Then, the NN models of each device are transferred to the FL server for global aggregation and the creation of the new global model. Finally, the global model is returned to the clients for further updates. To the best of our knowledge, `F-BIDS` model is the first model that combines the FL with the meta-classifier and blending ensemble for attack classification. The proposed `F-BIDS` supports heterogeneous architectures across the local and the global model.

In brief, the key contributions of this paper can be summarized as follows:

- An FL-based blending scheme using deep learning (DL) that can analyze and generalize a huge amount of sensitive data rapidly.
- A reduction in privacy risk was brought by blending ensemble learning and the classical FL process.
- An evaluation of the proposed scheme using recent and realistic datasets (*Edge-IIoTset* and *InSDN*).
- A comparative analysis of the proposed scheme, the centralized machine learning (ML) model, and the state-of-the-art.

2. Background

This section details the theoretical background of the basic concepts, which are useful to understand our proposition.

2.1. Federated Learning

Given the privacy concerns and data governance challenges, FL has been proposed as an alternative solution for centralized learning [9]. It attempts to answer the main question: *Can we train the model without needing to transfer data over a central location?* [10]. The learning process of FL is divided into three phases: initialization, local training on end-user devices, and central model aggregation. In the beginning, the FL server sends the initial global model to the selected clients/devices. Then, each client calculates a local update of the global model using its own private data. Once all the updates are received, the FL server aggregates the model's parameters. This process is repeated several rounds until the desired performance is achieved. Within the FL, the learning phase occurs locally at each client or device, and only model parameters are transferred. Also, it comes with another benefit of having a model trained on larger landscape data. It is an iterative process, wherein each communication round the model performance can be improved. The Federated Average (FedAvg) algorithm is the most popular algorithm for model aggregation, which is the core of FL [9]. At each communication round, the FL server uses the FedAvg algorithm in order to aggregate the local model after the local update by the clients. The FedAvg can be calculated as follows.

Definition: Global Model Aggregation

$$\theta_{t+1} = \sum_{k=1}^{K} \frac{D_k}{D} \theta_{t+1}^{k} \tag{1}$$

Given a client's model update, the equation (1) performs the global aggregation at each communication round. θ corresponds to the model parameters at iteration $t+1$, D is the amount of data from all the K clients, D_k is the amount of the data of the client k.

2.2. Blending Ensemble

Ensemble learning is one of the promising directions where it has established its superiority in recent years. In practice, it combines the prediction of several heterogeneous or homogeneous models. In other words, it tries to imitate nature by seeking different opinions before making any decision where we combine weighted and various individual opinions to find a final decision. It can be generally divided into two groups: (i) *homogeneous* ensemble (such as bagging or boosting) and (ii) *heterogeneous* ensembles such as blending. The blending ensemble tries to take advantage of several heterogeneous models, which is not the case with bagging and boosting models [11]. As a result, it can achieve a remarkable generalization performance [12]. Moreover, individual models in blending can be trained in parallel and hence requires less training time than boosting models. Therefore, this can be used to provide an efficient mechanism for the Internet of Things (IoT) environment. In particular, the blending ensemble consists of two levels known as *base-classifiers* and *meta-classifier*. The objective of the base classifiers is to train ML models using the raw data and produce the meta-data through the validation set. It is important to note here that data need to be split into

training, validation, and test sets. Then, the meta-classifier training process uses the meta-data generated by the base classifiers to make the final classification. Therefore, the meta-classifier uses the intermediate data (i.e., meta-data) whereas the original data is used only by the base classifiers in the first level.

As mentioned above, blending is a type of ensemble learning model, which has been originally introduced in the Netflix competition [13] in 2009. The main objective of blending is to combine and improve the prediction of several models (base-classifiers), through the meta-classifier. In other words, the base classifiers are used to provide base predictions as new features, which is the meta-data. Then the meta-classifier is trained on these new features to give the final decision. A general overview of the training and testing process is shown in Figure 1. In practice, the blending ensemble is based on several steps as described in the following.

1. Apply a hold-out method to divide the training set into a new training set and validation set.
2. Train the base classifiers through the new training dataset, and the prediction of the base classifiers on the validation set forms the meta-training dataset.
3. Join the prediction on the validation set of the base classifiers to form the meta-training dataset.
4. Train the meta-classifier using the meta-training dataset.
5. Use the meta-classifier to make the final prediction through the intermediate test produced by the base classifiers.

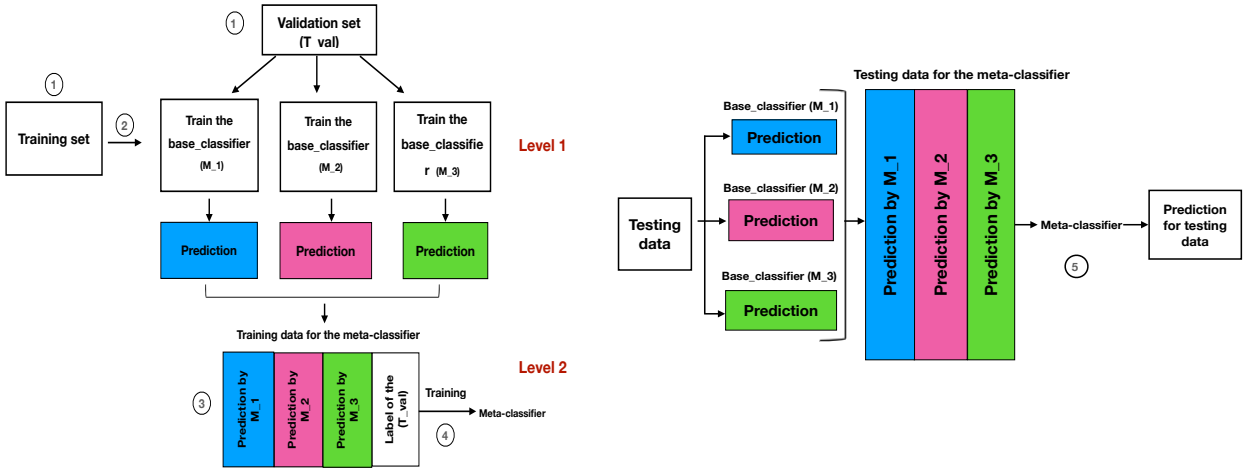

Figure 1: Training process (*left*) and Test process (*right*)

3. Related Work

Due to the continuous changes in the behaviors of cyber-attacks within the IoT environment, several ML/DL-based approaches have been proposed. In this section, we first provide a brief overview of some IDS based on conventional ensemble learning (Section 3.1). Then, we present the recent achievements of state-of-the-art approaches that proposed

IDS using FL-based architecture (Section 3.2). Finally, we present recent works on improving the privacy of FL-based solutions in (Section 3.3).

3.1. Conventional ensemble models

Kumar *et al.* [14] proposed an ensemble learning for intrusion detection using fog nodes. It is composed of two levels. In the first level, three supervised models are trained on the initial data (XGBoost, K-Nearest Neighbor, Naive Bayes). Then, in the second level, Random Forests (RF) have been used to combine the output of the first level and provide the final classification ("attack" or "benign"). Kumar *et al.* [15] also proposed another ensemble model by combining several conventional ML models on the Internet of Medical Things (IoMT) environment. Specifically, the Decision Tree, Naive Bayes, and RF are used in the first level, and in the second level, the classification results are obtained by XGBoost for identifying normal and attack instances. To evaluate the performance of these ensemble models the authors used ToN_IoT, DS2OS, and UNSW-NB15 datasets which are obsolete and can lack modern IoT-based attacks.

Also, Folino *et al.* [16] combined four base DNN classifiers, trained on disjoint chunks of the data instances' stream, and the meta classifier uses both the base classifiers predictions and original instance features for training and the final prediction tasks. Using two datasets the experimental results showed that the proposed ensemble model can act as a methodological basis robust and scalable enough for intelligent systems for the analysis of streaming IDS. Furthermore, in our previous work [11], a blending ensemble has been proposed for network traffic classification within centralized learning using a decision tree-based blending method and DL as a meta-classifier. The evaluation results demonstrate that the proposed ensemble can prevent overfitting and reduces bias simultaneously to some extent. Specifically, the use of a meta-classifier corrects the errors that occur during the learning process of the base classifiers. The presented results also confirm that using neural networks as a meta-classifier enables to discover the nonlinear relationships with very little handcrafted engineering and increases the learning ability of the whole model. However, this ensemble is limited to a centralized version and is not dedicated to intrusion/attack detection.

The aforementioned approaches are based on a centralized model where the data are collected on a central entity. Despite the effectiveness of these approaches, they can cause problems with information security and leakage of confidential data. In contrast, our `F-BIDS` approach takes advantage of both ensemble learning and FL to detect the attack without data leakage and reduce communication overhead as much as possible by sending only the meta-classifier model parameters to the central entity instead of the end-user data.

3.2. Federated Learning-based models

As a solution to privacy problems, FL has been proposed to ensure data privacy during ML/DL model training. In this context, Nguyen *et al.* [17] introduced an FL system for detecting compromised IoT devices, called `DÏoT`. This

was the first system that deployed FL for IDS. It consists of two components, which are security gateways and IoT security services. Security Gateways use the local data to train the local models with Gated Recurrent Unit (GRU) and IoT security service aggregates the local models into a global model. Another solution proposed by Friha *et al.* [18] uses federated DL-based IDS for the cybersecurity of agricultural-IoT networks, called `FLIDS`. It uses three DL-based models, namely, Deep Neural Network, Convolutional Neural Network, and Recurrent Neural Network; and the experimental results show that it achieved a competitive performance compared to the centralized model.

As each DL model has its own strengths and weaknesses, the researchers started to use the combination of several models in order to overcome their individual shortcomings and in turn improve the classification performance. In such context, Aouedi *et al.* [19] proposed a federated Semi-Supervised Learning for Attack Detection in Industrial IoT, called `FLUIDS`. An AutoEncoder (AE) model has been used for feature representation learning and dimensionality reduction. After the AE global aggregation, the global encoder is directly linked to a NN layer for fine-tuning and supervised learning. The experiments conducted on a real industrial dataset demonstrate the performance of FL against centralized versions of the DL model (non-federated learning). Moreover, Mothukuri *et al.* [20] proposed an ensemble FL model for IDS in an IoT environment. More specifically, after the GRU model has been trained locally on the client for different window sizes, the FL server calculates the global model at a central server. Then, in the FL server, it applies an ensemble model (i.e., RF) to improve the classification performance. Similarly, Attota *et al.* [21] proposed an ensemble FL-based intrusion detection, called `MV-FLID`. Specifically, the authors have developed three Feedforward Neural Network (FNN) models for three views (i.e., Biflow View, Packet View, and Uniflow View), and the outcomes of these models are sent to an ensemble model (RF), which combines the predictions of these models and classifies the instances. The results show that the FL approach can outperform the non-FL one. Although the proposed models have improved the IDS through the use of an FL model and the combination of different models, they can leak sensitive data through the communication of model parameters directly. Therefore, the FL-based model still needs further protection of parameters as well as investigations on the trade-offs between data privacy and model system performance for attack detection. In this direction, Al-Marri *et al.* [22] proposed a federated mimic learning by combining FL and mimic learning in order to better preserve the privacy of end-users. Using the NSL-KDD dataset, the results show that the federated mimic learning-based method can achieve 98% detection accuracy, which is close performance to the centralized DL while improving the privacy preservation of the user data significantly. However, the NSL-KDD dataset is obsolete and can lack modern IoT-based attacks.

3.3. Privacy enhancement

To enhance FL security and better preserve the privacy of the end-user data, some techniques have been used such as differential privacy (DP) and data encryption. For example, Li *et al.* [23] proposed an FL-based IDS for Industrial

Cyber-Physical Systems, called DeepFed. Then, to preserve the security and privacy of model parameters during the training process, the encryption-based algorithm has been used. Although the good performance of such algorithms, they have some drawbacks when applying directly to IDS based on the FL model. First, encryption-based techniques are computationally expensive because the users have to store the set of encryption keys and run the encryption process, which requires more time and CPU usage. Second, using DP with the FL model may add more noise to the model parameters and in turn decrease the performance of the whole system. Third, the clients can have different amounts of data, and using the same noise with all clients can be unfair and inefficient. In contrast to these techniques, as the meta-classifier used with F-BIDS consists of only a few layers, it requires less computation. Moreover, the use of a meta-classifier corrects the errors that occur during the learning process of the base classifiers and in turn improves the classification performance.

Table 1
Comparison of related works and our proposition.

Ref.	Ensemble leaning	Federated approach	Privacy enhancing
[14]	✓	×	×
[15]	✓	×	×
[11]	✓	×	×
[16]	✓	×	×
DÏoT[17]	×	✓	×
FLIDS[18]	×	✓	×
FLUIDS[19]	×	✓	×
[20]	×	✓	×
MV-FLID[21]	×	✓	×
[22]	×	✓	✓
F-BIDS(our proposition)	✓	✓	✓

4. F-BIDS methodology

This section presents our federated blending model for IDS. We will describe the data pre-processing step and learning process of the proposed model.

4.1. Data pre-processing

In this subsection, we detail a discussion of feature pre-processing steps used with our model.

- **Feature mapping and normalization**

As the network traffic often contains categorical features, feature mapping needs to be used to transform these features. To do so, we have used one-hot encoding to encode the categorical features as a one-hot numeric array. Also, as the dataset consists of different features with values on different scales, it needs to be normalized. This can be done by *Min-Max* normalization to perform feature scaling. It is a technique that scales every feature of our dataset between

0 and 1 as expressed by equation (2), the maximum value of that feature gets transformed to 1 and the minimum value gets transformed to 0.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2}$$

where X is the feature to be scaled down, X_{max} is the maximum value, and X_{min} is the minimum value for this feature.

- **Feature selection**

To improve the performance of the proposed model as well as reduce its training and classification time, we have used the feature selection method. The feature selection method tries to identify the most relevant features and discard the irrelevant ones. In the proposed model, Recursive Feature Elimination (RFE) has been used. It is a wrapper method that recursively evaluates the performance of a specific model with a different set of features. In other words, starting from all the feature sets, RFE recursively removes features in order to maximize accuracy. Then it ranks the features based on the order of their elimination.

4.2. Learning Process

In the proposed detection system, the Federated blending model is used for designing security mechanisms. The proposed model is mainly composed of two levels, which are base classifiers (level-1) and meta-models (level-2). As

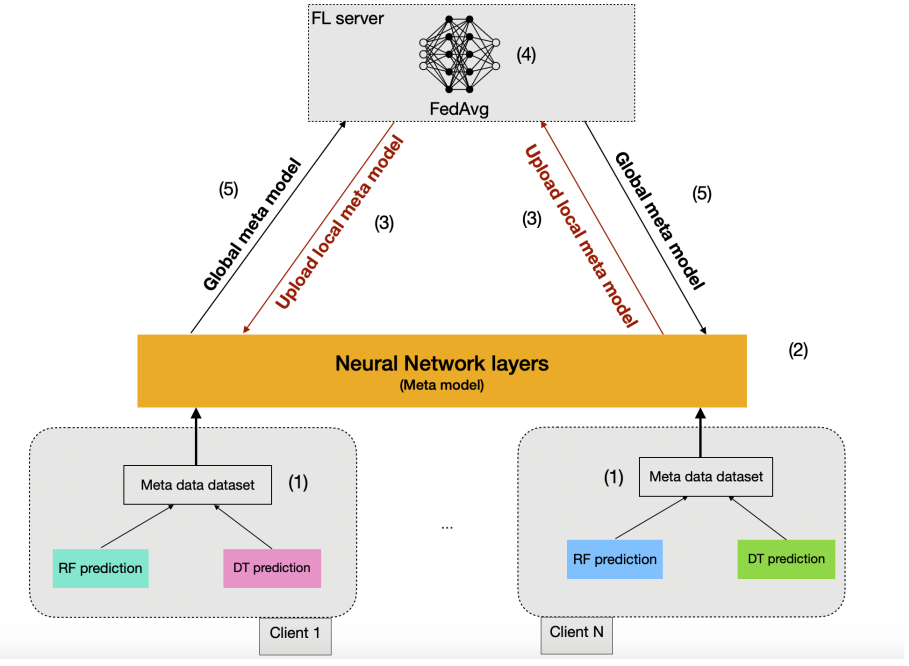

Figure 2: F-BIDS learning process

shown in Figure 2, two simple and well know ML-based models (DT and RF) have been used as base classifiers and local models that generate the meta-data **(step 1)**. We select DT-based models because they are considered one of the most suitable learning algorithms for network traffic classification [24]. Also, the main advantage of DT and RF compared to other models (e.g., XGBoost) is that they are easy to train and require fewer hyper-parameters tuning [25]. Then, NN layers (meta-model) are used as meta-classifier and a global model in order to learn the non-linear relationship among the base-classifiers **(step 2)**. Next, the client sends the meta-model to the FL sever for the global aggregation **(step 3)**.

The FL server aggregates the weights of the meta-model from the different client **(step 4)**. Finally, the FL server sends back the aggregated meta-model to the clients **(step 5)**. Consequently, `F-BIDS` takes advantage of both ensemble learning and FL, and their relationship is win-win. In particular, `F-BIDS` uses FL to enhance the privacy concerns for the blending model by sending the model parameters instead of row data and the blending model, whereas the blending model can reduce the membership inference attacks performance on the FL model because, during the FL communication round, the clients and the FL server exchange the parameters of the meta-classifier instead of the base classifiers. In other words, this is due to the fact, that the FL meta-classifiers are trained on the meta-data that describe prior learning tasks and previously learned DT and RF models. Algorithm 1 describes the steps involved in designing a blending-based ensemble design, and Table 2 describes the notations used in Algorithm 1.

Algorithm 1: Learning step of the proposed F-BIDS model

Input: Training Dataset $DT = \{x_i, y_i\}$ $(x_i \in X, y_i \in Y)$;
Output: Prediction of Ensemble Algorithm $H(X)$;
/* -- Train the base classifiers used in the level 1 -- */
for j=1 to 2 **do in parallel**
    Train M_j from D_train;
end for
/* -- Construct meta-data -- */
D'=$\{P, Y\}$, where P= `Concatenate` $M_1\_prediction$, $M_2\_prediction$;
/* -- Train the meta-classifier MC in federated way -- */
for i=1 to R **do**
    n= r_k * K
    for c=1 to n do
        for e=1 to E do
            train MC parameters θ_c using D_c
        end for
        Send θ_c to the FL server
    end for
    Aggregate $\{\theta\}_{c=1\ldots n}$ with FedAvg into θ
    Send meta global model θ to the clients
end for
return $H(X) = MC(M_1(X), M_2(X))$

Table 2
List of notations used in F-BIDS model.

List of notations	Meaning
X	Set of features
Y	Class label set
M	Base classifier
H	Ensemble classifier
MC	Meta classifier
D_train	training set of the base classifier (level-1)
P	Base classifier prediction
D'	meta-data used to train the meta-classifier (level-2)
θ	meta classifier parameters
R	Total number of rounds
E	meta classifier training epochs
K	Total number of clients
n	The number of clients selected at each round

5. Experimental study and results analysis

In this section, we first describe the datasets, and the experimental parameters, and then evaluate our proposed model.

5.1. Dataset description

To evaluate the performance of the proposed model, we have chosen two of the most recent datasets. Specifically, the first dataset is *Edge-IIoTset* published in 2022 [26]. This dataset consists of 157,800 observations, 61 features, and 1 label, the label contains 15 possible values, benign and seven different types of attacks [2]. The statistics of the selected dataset are summarized in Table 3. The second dataset is *InSDN* published in 2020 [27]. It consists of 343890 observations and 84 features. The label contains 8 possible values, benign and seven different types of attacks. We also chose this dataset because it contains a large amount of network traffic including both malicious and benign traffic.

Summarized descriptions of both datasets are presented in Table 3 and Table 4 respectively. In this experiment, we drop unnecessary flow features as done with its original paper [26]. Then, we separated the dataset into 80% for training, 10% for validation, and 10% for testing.

5.2. Experimental setup

We conducted our experiments with `Python 3` as a programming language, `Scikit-learn` for the conventional models, `PyTorch` for the federated blending model. Also, all experiments were run using four core Intel Core i7-6700 CPU@3.40GHz processor, and 32.00 GB of RAM.

5.3. Classification performance

Our experiments objectives are the following: 1) Evaluate the performance of the proposed ensemble blending model trained in a centralized way, where the training data are located in a central entity; 2) Evaluate the performance

[2]https://www.kaggle.com/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot

Table 3
Edge-IIoTset description.

Label	Total
Normal	24,301
DDoS_UDP attack	14,498
DDoS_ICMP attack	14,090
Ransomware attack	10,925
DDoS_HTTP attack	10,561
SQL_injection attack	10,311
Uploading attack	10,269
DDoS_TCP attack	10,247
Backdoor attack	10,195
Vulnerability_scanner attack	10,076
Port_Scanning attack	10,071
XSS attack	10,052
Password attack	9,989
MITM attack	1,214
Fingerprinting attack	1,001

Table 4
InSDN description.

Label	Total
Normal	68,424
DDoS	121,942
DoS	53,616
Probe	98,129
BFA	1,405
Web-attack	192
Botnet	164
U2R	17

of F-BIDS model, trained in a federated way; and 3) Comparing F-BIDS against the centralized model, all these in terms of several evaluation metrics.

5.3.1. Centralized Blending performance evaluation

In this subsection, we compare the performance of different centralized models. Figure 3 and Figure 4 present the centralized blending model performance against support vector machines (SVM), Multi-layer Perceptron (MLP), the proposed model by Ferrag *et al.* [26] (the original paper of Edge-IIoT dataset), as well as the model proposed by Friha *et al.* [18] in terms Precision, Recall, and F1-score under multi-class classification (6 and 15 class). To note here, Ferrag *et al.* [26] and Friha *et al.* [18] used fully connected layers for IDS using the Edge-IIoT dataset. They evaluated the performance of the fully connected layers in both centralized and federated learning modes.

It is worth noting that the centralized blending outperforms all the other models for 6-class and 15-Class in terms of all the evaluation metrics. For example, using Edge_IIoT dataset the F1-score is increased by 8.17 (3.23%), 12.15% (7.58%), and 15.19% (5.36%) for MLP, Ferrag *et al.* [26] model, and SVM, respectively. These results illustrate the high performance of our model, especially the 15-class classification scenario. Moreover, as shown in Figure 4 using

InSDN dataset, the F1-score of our model is 9.71%, 14.29%, and 51.15% better than MLP, Friha *et al.* [18], and SVM, respectively. This may be attributed to the fact that the combination of DT, RF, and NN-based models helps to yield far superior results compared to other models.

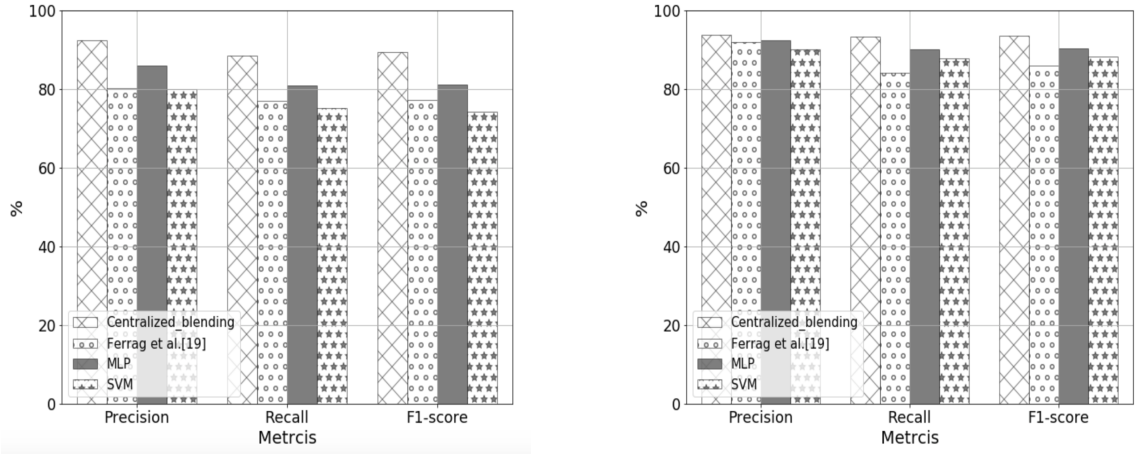

Figure 3: Centralized models performance for 15-class (*left*) and for 6-class (*right*) with Edge-IIoTset dataset

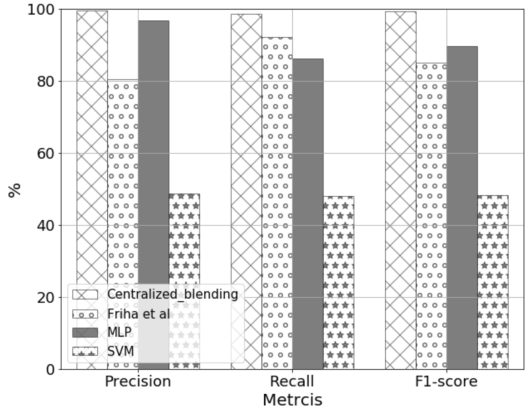

Figure 4: Centralized model performance with InSDN dataset

In order to further validate the effectiveness of the centralized blending model, we evaluate its performance by attack classification. Table 5 and Table 6 present the obtained results of the centralized models including SVM, MLP, the model proposed by Ferrag *et al.* [26], and our blending ensemble model, in terms of Precision (Pr), Recall (Rc), F1-score (F1), under multi-class classification (6 and 15 class). It can be seen from these Tables that our model has a good performance in detecting different attacks as well as in detecting benign network traffic. More specifically, as shown in Table 5, our centralized blending model has better or equal results to other models but is never less effective. For example, our model gives the highest F1-score for Normal traffic 100%, Injection attacks 84%, Scanning attacks

Table 5
Classification performance for 6 classes of simple models centralized blending vs Ferrag *et al.* [26] with Edge-IIoTset dataset (Centralized model performance)

Model	Metrics	Normal	DDoS attacks	Injection attacks	MITM attacks	Malware attacks	Scanning attacks
SVM	Pr	1.0	0.86	0.64	1.0	0.96	0.93
	Rc	1.0	0.89	0.84	1.0	0.65	0.85
	F1	1.0	0.88	0.73	1.0	0.78	0.89
Neural Networks	Pr	1.0	0.94	0.65	1.0	0.96	**0.96**
	Rc	1.0	0.87	**0.94**	1.0	0.69	0.88
	F1	1.0	0.90	0.77	1.0	0.81	0.92
Ferrag *et al.* [26]	Pr	1.0	0.92	0.66	1.00	**0.97**	**0.96**
	Rc	1.0	**0.99**	0.90	0.94	0.48	0.74
	F1	1.0	**0.95**	0.76	0.97	0.64	0.84
Centralized Blending	Pr	1.0	**0.94**	**0.86**	1.0	0.84	0.93
	Rc	1.0	0.89	0.83	1.0	**0.95**	**0.96**
	F1	1.0	0.91	**0.84**	1.0	**0.88**	**0.95**

Table 6
Classification performance for 15 classes of centralized blending vs simple models and Ferrag *et al.* [26] with Edge-IIoTset dataset (Centralized model performance)

Model	Metrics	Normal	Back	HTTP	ICMP	TCP	UDP	Fing	MITM	Pwd	Port	Rans	SQL	Upload	Scan	XSS
SVM	Pr	1.00	**0.99**	0.41	1.00	0.88	0.96	**1.00**	**1.00**	0.05	0.92	0.92	0.29	0.80	0.76	0.98
	Rc	0.99	0.92	0.26	0.96	0.97	1.00	0.52	1.00	0.00	0.90	0.96	**0.88**	0.17	0.98	0.75
	F1	0.99	0.93	0.32	0.98	0.92	0.97	0.68	1.00	0.01	0.91	0.94	0.43	0.28	0.86	0.81
MLP	Pr	1.00	**0.99**	0.57	1.00	0.92	0.99	**1.00**	**1.00**	0.43	0.95	0.96	0.41	0.76	**0.98**	0.87
	Rc	**1.00**	0.95	0.59	0.99	0.99	1.00	0.59	**1.00**	0.36	0.94	0.97	0.57	0.29	0.85	**0.99**
	F1	1.00	0.97	0.58	0.99	0.96	0.99	0.74	1.00	0.30	0.94	0.96	0.43	0.42	0.91	0.92
Ferrag *et al.* [26]	Pr	1.00	0.95	0.76	1.00	0.82	1.00	0.59	1.00	0.55	**1.00**	0.73	0.47	0.67	0.96	0.53
	Rc	**1.00**	0.86	**0.92**	0.99	**1.00**	1.00	0.64	**1.00**	0.38	0.50	0.85	0.71	0.48	0.85	0.37
	F1	1.00	0.90	**0.83**	0.99	0.90	1.00	0.61	1.00	0.45	0.66	0.79	0.57	0.56	0.90	0.43
Centralized Blending	Pr	1.00	0.95	**0.82**	1.00	**1.00**	1.00	0.64	1.00	**0.63**	0.95	**0.97**	**0.67**	**0.63**	0.96	**0.99**
	Rc	**1.00**	**0.99**	0.56	**1.00**	0.93	1.00	**1.00**	1.00	**0.76**	**0.95**	**0.97**	0.81	**0.91**	**0.99**	0.96
	F1	1.00	0.97	0.65	**1.00**	**0.96**	1.00	**0.78**	1.00	**0.66**	**0.95**	**0.97**	**0.72**	**0.74**	**0.97**	**0.98**

95%, MITM attacks 100%, and Malware attacks 88%, while for DDoS attacks the highest F1-score is given by Ferrag *et al.* [26] with 95%. On the other hand, it can be seen from Table 6 that our model obtained the highest F1-score in 14 out of 15 classes. In particular, our centralized blending model gives the highest F1-score for Normal, Backdoor attacks, ICMP DDoS attacks, TCP DDoS attacks, UDP DDoS attacks, OS Fingerprinting, MITM attacks, Password attacks, Port Scanning attacks, Ransomware attacks, SQL Injection, Upload attacks, Vulnerability scanning attack, and XSS attack are 100%, 97%, 100%, 96%, 100%, 78%, 100%, 66%, 95%, 97%, 92%, 74%, 97%, 98%, respectively.

In addition, it can be seen from Table 7 that our model obtained the highest F1-score for the Normal traffic 100%, DDoS attack 100%, DoS 100%, Probe attack 99%, Brute Force Attack 94%, Web Attack 100%, Botnet attack 100%, and U2R attack 100%. Consequently, these results can validate that our blending model is a promising model and can better differentiate the applications. Although the competitive performance of the proposed blending model raises privacy issues as confidential data might need to be shared in the process. As mentioned in the introduction, we proposed `F-BIDS` in order to ensure the privacy of the data as maximum as possible.

Table 7
Centralized model evaluation with InSDN dataset

Model	Metrics	Normal	DDoS	DoS	Probe	BFA	Web-Attack	Botnet	U2R
SVM	Pr	0.99	0.98	0.95	0.90	0.05	0.00	0.00	0.00
	Rc	0.97	0.96	0.93	0.96	0.00	0.00	0.00	0.00
	F1	0.98	0.97	0.94	0.93	0.00	0.00	0.00	0.00
MLP	Pr	0.99	0.99	0.99	0.99	**0.90**	0.84	**1.00**	**1.00**
	Rc	0.99	0.99	0.99	0.99	0.85	0.49	0.80	0.75
	F1	0.99	0.99	0.99	**0.99**	0.87	0.58	0.87	0.83
Friha *et al.* [18]	Pr	**1.00**	**1.00**	0.98	**1.00**	0.45	0.71	0.97	0.33
	Rc	**1.00**	**1.00**	0.99	0.97	0.91	**1.00**	**1.00**	0.50
	F1	**1.00**	**1.00**	0.99	**0.99**	0.60	0.83	0.99	0.40
Centralized Blending	Pr	**1.00**	**1.00**	**1.00**	**1.00**	0.89	**1.00**	**1.00**	**1.00**
	Rc	**1.00**	**1.00**	**1.00**	**0.99**	**1.00**	**1.00**	**1.00**	**1.00**
	F1	**1.00**	**1.00**	**1.00**	**0.99**	**0.94**	**1.00**	**1.00**	**1.00**

5.3.2. F-BIDS performance evaluation

By taking advantage of the blending model and the FL, a federated blending model-driven IDS framework (`F-BIDS`) has been proposed. As a continuation from the previous subsection, using the same two different datasets (Edge_IIoT and InSDN) we also evaluated the performance of `F-BIDS`. The evaluation results are presented in Table 8 and Table 9, which present the performance of the global model, the worst client, and the best client. Also, in this experiment, we train `F-BIDS` under different numbers of clients K, where $K = 5$, $K = 10$, and $K = 15$, and the different numbers of rounds R. Using Edge_IIoT dataset and for 15-class, the best global model results in the first and 10^{th} round of `F-BIDS` achieved when the number of clients $K = 5$, where the best client accuracy achieves 90.18% and 90.91% in the first and 10^{th} rounds, respectively, whereas the worst client accuracy is more than 82% and 85% in the first and 10^{th} rounds. For the 6-class, the best global model results in the first and 10^{th} round of `F-BIDS` also achieved when the number of clients $K = 5$, where the best client accuracy achieves 90.91% and 90.93% in the first and 10^{th} rounds, respectively, whereas the worst client accuracy is more than 86% and 88% in the first and 10^{th} rounds. Furthermore, using InSDN dataset, the best global model results in the first and 50^{th} round of `F-BIDS` achieved when the number of clients $K = 5$, where the best client accuracy achieves 99.89% and 90.92% in the first and 50^{th} rounds, respectively, whereas the worst client accuracy is more than 99% and 88% in the first and 50^{th} rounds.

From these results, we can notice that accuracy decreases as the number of clients increases. This is because, with a larger number of clients, the diversity of the models across different users increases significantly [28]. Also, we can notice that `F-BIDS` converges quickly since the first round has relatively good accuracy. This is due to the federated meta-classifier (NN layers) that uses the new features generated by the base classifiers. Consequently, the features deployed by the meta-classifier are relevant features and can help to converge fast and better than the original features. Moreover, the experimental results show that the global performance of `F-BIDS` is competitive with the centralized blending model.

Table 8
F-BIDS performance evaluation with Edge-IIoTset

	client	1^{st} round			10^{th} round		
		B	W	G	B	W	G
15-class	K=5	**90.18**	86.25	**89.56**	**90.91**	**88.66**	**89.91**
	K=10	89.03	**87.01**	88.15	89.69	85.87	88.45
	K=15	88.69	82.18	85.18	89.05	85.24	87.82
6-class	K=5	**90.91**	**90.27**	**90.86**	**90.93**	**90.67**	**90.91**
	K=10	90.15	86.71	89.07	89.93	88.53	89.87
	K=15	90.06	87.83	89.57	90.07	88.72	89.94

(B): Best client accuracy; (W): Worst client accuracy; (G): Global model accuracy

Table 9
F-BIDS performance evaluation with InSDN dataset

client	1^{st} round			50^{th} round		
	B	W	G	B	W	G
K=5	**99.89**	**99.54**	**99.86**	**99.93**	99.66	99.90
K=10	99.83	99.48	99.74	99.92	**99.70**	**99.91**
K=15	99.65	99.48	99.50	99.52	99.43	99.49

(B): Best client accuracy; (W): Worst client accuracy; (G): Global model accuracy

5.4. Communication overhead

Further, in this subsection, we compare the communication overhead of `F-BIDS` against different schemes, which are a centralized learning process, Ferrag *et al.* [26] and Friha *et al.* [18]. In particular, for the centralized model, the raw data need to be communicated to a central entity for the training process. On the other hand, we simulated the Ferrag *et al.* [26] and Friha *et al.* [18] in a federated way. As shown in Figure 5, the FL-based models (F-BIDS, FL (Ferrag *et al.* [26] and Friha *et al.* [18])), reduce the communication overhead in comparison to the centralized model. This advantage becomes even more significant in the case of larger training data like InSDN dataset. This is mainly due to the fact that FL avoids transferring raw data samples to the central entity and sends only model parameters. In addition, we can observe that `F-BIDS` performs better than the Federated version of Ferrag *et al.* [26] and Friha *et al.* [18]. This is attributed to the fact that with `F-BIDS`, the federated meta-model (i.e. NN layers) trained on the meta-data generated by the base classifiers whereas the other models used the Federated version of Ferrag *et al.* [26] and Friha *et al.* [18] raw data, which are larger than the meta-data as well as more complicated to train.

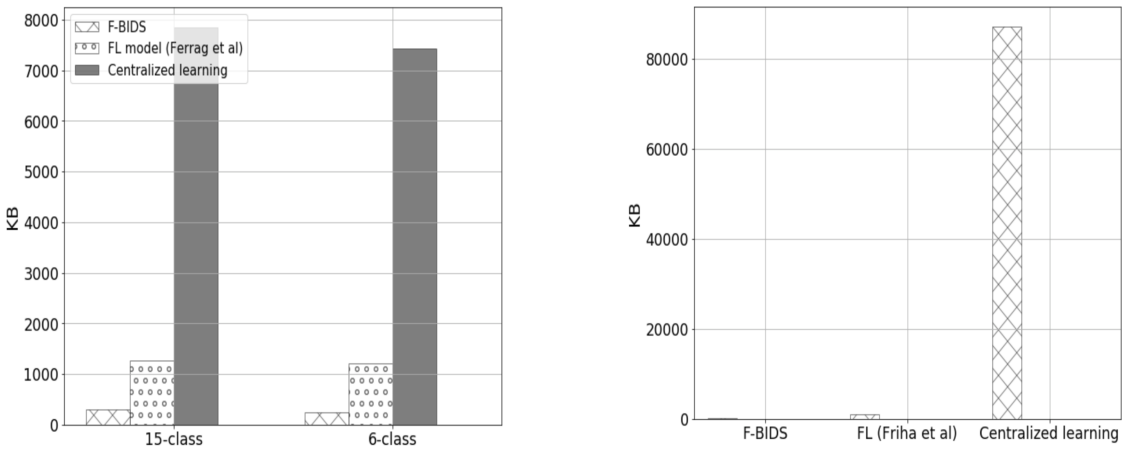

Figure 5: Communication overhead with Edge_IIoT dataset (*left*) and for InSDN dataset (*right*)

6. Conclusion

In this paper, we propose a federated blending model-driven IDS framework to improve the classification performance as well as reduce privacy risk against some attacks, e.g., membership inference attacks, as maximum as possible. Using two different and recent datasets, the results demonstrate that by taking advantage of different models, centralized blending and `F-BIDS` are shown to be accurate and efficient for traffic classification tasks. Also, the presented results also confirm that `F-BIDS` achieves competitive results, compared to the state-of-the-art model. In addition, the FL server benefits from the blending learning process in order to reduce privacy risks brought by centralized learning and the classical FL process. Finally, `F-BIDS` helps to reduce the communication overhead and hence reduce the network congestion if all traffic has to be sent to the FL server.

For future works, we plan to implement the model, study its network performance, and quantify the privacy risk by using some recent software like *ML-Doctor*. Also, the black-box nature of `F-BIDS` may raise an issue of trust; hence, an explainable solution is required for our proposed IDS system. In this context, we will extend `F-BIDS` models by adding an explainability module such as SHapley Additive exPlanations (SHAP) to further explain the model.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[2] T. R. Gadekallu, Q.-V. Pham, T. Huynh-The, S. Bhattacharya, P. K. R. Maddikunta, M. Liyanage, Federated learning for big data: A survey on opportunities, applications, and future directions, arXiv preprint arXiv:2110.04160 (2021).

[3] S. Agrawal, S. Sarkar, O. Aouedi, G. Yenduri, K. Piamrat, S. Bhattacharya, P. K. R. Maddikunta, T. R. Gadekallu, Federated learning for intrusion detection system: Concepts, challenges and future directions, arXiv preprint arXiv:2106.09527 (2021).

[4] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, L. Shu, Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis, IEEE Access 9 (2021) 138509–138542.

[5] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, G. Srivastava, A survey on security and privacy of federated learning, Future Generation Computer Systems 115 (2021) 619–640.

[6] M. Nasr, R. Shokri, A. Houmansadr, Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning, in: 2019 IEEE symposium on security and privacy (SP), IEEE, 2019, pp. 739–753.

[7] R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership inference attacks against machine learning models, in: 2017 IEEE symposium on security and privacy (SP), IEEE, 2017, pp. 3–18.

[8] H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, X. Zhang, Membership inference attacks on machine learning: A survey, ACM Computing Surveys (CSUR) (2021).

[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial intelligence and statistics, PMLR, 2017, pp. 1273–1282.

[10] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, Y. Zhou, A hybrid approach to privacy-preserving federated learning, in: Proceedings of the 12th ACM workshop on artificial intelligence and security, 2019, pp. 1–11.

[11] O. Aouedi, K. Piamrat, B. Parrein, Decision tree-based blending method using deep-learning for network management, in: IEEE/IFIP Network Operations and Management Symposium, 2021.

[12] Y. Xiao, J. Wu, Z. Lin, X. Zhao, A deep learning-based multi-model ensemble method for cancer prediction, Computer methods and programs in biomedicine 153 (2018) 1–9.

[13] A. Töscher, M. Jahrer, R. M. Bell, The bigchaos solution to the netflix grand prize, Netflix prize documentation (2009) 1–52.

[14] P. Kumar, G. P. Gupta, R. Tripathi, A distributed ensemble design based Intrusion Detection System using fog computing to protect the Internet of Things networks, Journal of Ambient Intelligence and Humanized Computing (2020) 1–18.

[15] P. Kumar, G. P. Gupta, R. Tripathi, An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for iomt networks, Computer Communications 166 (2021) 110–124.

[16] F. Folino, G. Folino, M. Guarascio, F. S. Pisani, L. Pontieri, On learning effective ensembles of deep neural networks for intrusion detection, Information Fusion 72 (2021) 48–69.

[17] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, A.-R. Sadeghi, Dïot: A federated self-learning anomaly detection system for iot, in: 2019 IEEE 39th International conference on distributed computing systems (ICDCS), IEEE, 2019, pp. 756–767.

[18] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, K.-K. R. Choo, M. Nafaa, Felids: Federated learning-based intrusion detection system for agricultural internet of things, Journal of Parallel and Distributed Computing 165 (2022) 17–31.

[19] O. Aouedi, K. Piamrat, G. Muller, K. Singh, Federated semi-supervised learning for attack detection in industrial internet of things, IEEE Transactions on Industrial Informatics (2022).

[20] V. Mothukuri, P. Khare, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, G. Srivastava, Federated learning-based anomaly detection for iot security attacks, IEEE Internet of Things Journal (2021).

[21] D. C. Attota, V. Mothukuri, R. M. Parizi, S. Pouriyeh, An ensemble multi-view Federated Learning Intrusion Detection for IoT, IEEE Access (2021).

[22] N. Al-Marri, B. Ciftler, M. Abdallah, Federated mimic learning for privacy preserving intrusion detection, 2020, pp. 1–6. doi:10.1109/BlackSeaCom48709.2020.9234959.

[23] B. Li, Y. Wu, J. Song, R. Lu, T. Li, L. Zhao, Deepfed: Federated deep learning for intrusion detection in industrial cyber–physical systems, IEEE Transactions on Industrial Informatics 17 (2020) 5615–5624.

[24] T. T. Nguyen, G. Armitage, A survey of techniques for internet traffic classification using machine learning, IEEE communications surveys & tutorials 10 (2008) 56–76.

[25] O. Aouedi, K. Piamrat, B. Parrein, Performance evaluation of feature selection and tree-based algorithms for traffic classification, in: 2021 IEEE International Conference on Communications Workshops (ICC Workshops), IEEE, 2021, pp. 1–6.

[26] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, H. Janicke, Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning (2022).

[27] M. S. Elsayed, N.-A. Le-Khac, A. D. Jurcut, Insdn: A novel sdn intrusion dataset, IEEE Access 8 (2020) 165263–165284.

[28] Z. Zhang, Z. Yao, Y. Yang, Y. Yan, J. E. Gonzalez, M. W. Mahoney, Benchmarking semi-supervised federated learning, arXiv preprint arXiv:2008.11364 17 (2020) 3.