"DFSat: Deep Federated Learning for Identifying Cyber Threats in IoT-based Satellite Networks"

by Nour Moustafa, Izhar Ahmed Khan, Mohammed Hassanin, David Ormrod, Dechang Pi, Imran Razzak, and Jill Slay

published in the *IEEE Transactions on Industrial Informatics* Early Access

Digital Object Identifier: 10.1109/TII.2022.3214652

It has been recommended by the authors and Editor-in-Chief of the *IEEE Transactions on Industrial Informatics* that this article will not be published in its final form and should not be considered for citation purposes. Two of the authors, Jill Slay and David Ormrod, were not informed of all included coauthors of the article, and therefore do not agree to the final publication of this article.

We regret any inconvenience this may have caused.

Ren Luo
Editor-in-Chief
IEEE Transactions on Industrial Informatics

DFSat: Deep Federated Learning for Identifying Cyber Threats in IoT-based Satellite Networks

Nour Moustafa,Izhar Ahmed Khan, Mohammed Hassanin, David Ormrod, Dechang Pi, Imran Razzak, Jill Slay

***Abstract*—The integration of satellite systems with smart computing and networking technologies, such as the Internet of Things (IoT), has intensely augmented sophisticated cyberattacks against satellite environments. Resisting cyber threats to complex and large-scale satellite configurations has been enormously challenging, owing to the deficiency of high-quality samples of attack data collected from distributed satellite networks. This study proposes a novel federated learning-based deep learning framework for intrusion detection, named DFSat, to identify cyberattacks from IoT-integrated satellite networks. We develop a distributed deep learning-enabled attack detection method using a recurrent neural network. We then build a federated learning architecture which, utilizes several IoT-integrated satellite networks to preserve the privacy and security of DFSat's parameters throughout the learning process. Extensive experiments have been conducted using communication rounds on an IoT-based network dataset to validate the efficiency of DFSat. The results revealed that the proposed framework significantly distinguishes complex cyberattacks, outperforming recent state-of-the-art intrusion detection techniques, validating its usefulness as a viable deployment framework in IoT-integrated satellite networks.**

***Index Terms*—Cyber security, Smart Enterprise Systems, Federated learning, Intrusion Detection, Internet of Things (IoT), Satellite Systems.**

I. Introduction

A. IoT Integrated Satellite Networks

SATELLITES are devices that are sophisticated and often perform multiple tasks. They are launched into Earth's orbit to extend the wireless communications medium to areas and locations that are unreachable using the terrestrial infrastructure network [1]. Satellites are categorised based on the trajectory they assume, for example, Low Earth Orbit (LEO), Medium Earth Orbit (MEO), and Geostationary Earth Orbit (GEO). The rotations of LEO and MEO satellites are faster than the GEO satellites since they can rotate around the Earth and complete an Earth orbital circle in about $1.5h$ and $5h$, respectively. While the rotation of GEO satellites is in sync with Earth's axis and rotation, it seems to be motionless in the sky.

Satellites differ in their functionality and purpose; nevertheless, some shared architectural constituents are common between diverse applications [1], [2]. For example, the computational constituent handles the onboard computations, such as data preprocessing of the collected sensory data and the algorithm execution. Satellites are equipped with different sensors to gather data regarding the position and orientations, run diagnostics, and keep track of the satellite's state to prevent it from malfunctioning. Similarly, satellites are equipped with actuators that help regulate their orientation and trajectory. Also, these actuators empower satellites to adjust their solar panels by interacting and assessing their surroundings.

The development of satellite systems needs effective cyber security schemes. Recent and ongoing advancement in communication technologies allows the hardware resources to be integrated with Internet of Things (IoT) ecosystems and thus these ecosystems have become larger and more complex. This has resulted in the development of smart systems capable of making intelligent decisions in smart enterprise management systems. Parallel developments have occured with networks of satellites [1]. Thus Smart Satellite Networks (SSN) are satellite system with IoT tools and devices, such as sensors and actuators, connected over network communications. The data collection from such SSN and IoT integrated satellite networks can thus be analysed using Machine Learning (ML) algorithms in the same way that this can be achieved with terrestrial enterprise networks. This leads to automation, improvements to satellite positioning techniques, managing resources, troubleshooting, communications and measurements, and the identification of security events [2].

IoT-integrated satellite network security events can include breach by cyber-attacks such as botnet and ransomware. Because of low power and compute capacity such satellite networks needs a lightweight intrusion detection system (IDS) [3]. In [4], the authors examined over 1350 businesses and revealed that 57% of the IoT tools and devices were exposed to cyber-space attacks, eliciting a sizable risk to the integrity of these businesses. Such vulnerabilities are easily exploitable by sophisticated attack techniques to covertly amend for example the firmware of a device, resulting in modified behaviour in the device. Attackers can also completely deactivate IoT devices using Distributed Denial of Service (DDoS) attacks, with a botnet attack specific to IoT devices such as Mirai. These types of cyberattacks are very challenging for SSNs.

B. Cyber Attacks in Satellite Networks

The function of most traditional satellites is in transmission or as a communication link, supporting or offering connectivity to out-of-reach or distant locations. SSNs are equipped

Nour Moustafa is with School of Engineering and Information Technology, University of New South Wales at ADFA, Australia. E-mail: nour.moustafa@unsw.edu.au

I.A. Khan, D. Pi are with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, People's Republic of China. E-mail: izhar@nuaa.edu.cn, dc.pi@nuaa.edu.cn

Imran Razzak is with the the University of New South Wales, Sydeny Australia. E-mail: z5131416@unsw.edu.au

Mohammed Hassanin, David Ormrod and Jill Slay are with the University of South Australia, Australia. E-mail: Mohammed.Hassanin@unisa.edu.au, Dave.Ormrod@unisa.edu.au, jill.slay@unisa.edu.au

with the supplementary operations of IoT-based sensory nodes fused into their design. Using the Internet in this design, the interconnected nature of these networks exposes their vulnerabilities to cyber threats with ramifications fluctuating from minor to grave risk. SSNs can be influenced by both conventional (pre-established) cyberspace threat vectors and the weaknesses of IoT tools and devices (i.e., the actuators and sensors) [1]. Furthermore, vulnerable satellites can become targets for malicious satellites because of poor network security in inter-satellite networks.

DDoS attack can be initiated through network communications, manipulation with the weaknesses of specific protocols used for satellite communication, or interfering with the firmware and operating system, allowing the satellite to be "bricked". Another major attack in IoT-driven networks, known as a power depletion attack, could affect the functioning of smart satellites. Its restricted nature via needlessly growing the load capacity of the sensors utilized by the satellite can be accomplished by dedicated cyberspace attacks. Moreover, various eavesdropping attacks, such as data manipulation and Man-In-The-Middle (MITM) attacks, could affect the integrity and confidentiality of SSNs data sent to ground stations.

If an attack is able to modify the sensing data gathered via satellite sensors this is likely to impact other networks or systems that depend on such data, such as weather measurement systems, Global Positioning Systems (GPS), and satellite imagery systems. In worst cases, a satellite could be crash-landed or thrown off its flight route if its gyroscope or height sensors are compromised. Therefore, developing an effective security framework that can be deployed onboard satellites is vital to ensure resilience and stability.

C. Research Contribution

This study aims to develop a federated learning model to detect cyber-attacks from satellite networks while preserving data privacy. This research presents a federated learning (FL)-based IDS framework to discover advanced attacks from Smart Satellite Networks (SSN). The proposed framework utilises federated learning-assisted recurrent networks to implement the IDS approach for SSNs augmented with IoT devices. Specifically, this study offers an intelligent FL-assisted recurrent network IDS model (DFSat) to detect intrusions from SSNs in real time. The proposed DFSat enables multiple SSN owners to cooperatively form a broad model of IDS in a privacy-preserving manner. We utilise the LSTM architecture because of its effectiveness in processing satellite data, as evident from the studies of other scholars published in top venues such as [5], [6], [7]. The ***key contributions*** of this study are summarised below.

- We develop a distributed deep learning architecture to detect intrusions from SSN. This method efficiently identifies numerous cyber-attacks, such as reconnaissance, fuzzers, and denial of service (DoS) attacks in smart satellite systems.
- Our proposed federated architecture framework empowers multiple SSNs to build a comprehensive IDS method utilizing data sources of several satellite networks. Moreover, it supports processing data sources at each network's own premise, preserving data privacy.
- We evaluate an extensive network data collection to inspect attack patterns and their underlying traces to build a reliable security framework in IoT-integrated satellite networks.
- We conduct an extensive evaluation of our proposed framework, which validates its superiority compared with compelling benchmark methods.

The rest of this study is structured as follows: the literature review is presented in Section II, and Section III presents the architecture and methodology of the proposed framework. Section IV and V presents the results and conclusion, respectively.

II. Literature Review

A. Intrusion Detection for Satellite Networks

Satellites are complicated and versatile devices orbited around the earth to cover wireless communications to distant or unreachable areas. Security and privacy are vital functions for SSNs because of the confidential nature of the data involved in these systems. Various studies have been conducted to achieve the privacy and security of IoT-integrated satellite networks [8], [9]. N. Zhenyu *et al*. [10] proposed distributed routing design using an ML technique for Low Earth Orbit (LEO) satellite networks. They employed extreme learning machines to predict network traffic and undertake intelligent routing decisions. To detect anomalies and reduce the rate of false alarms in satellite systems, G. Lachlan *et al*. [11] proposed an DL-based attach detection model. They used the LSTM network to train their model and claimed that their model could detect unseen anomalies.

In [12], the authors proposed an anomaly detection model using an ML technique to monitor the spacecraft's health. They used dictionary learning and sparse representation to detect anomalies in satellite telemetry data. To detect abnormalities in power system parameters of satellites, the authors of [13] used a Long Short Term Memory (LSTM) model. Their model can predict the satellite power system parameters using the error between the actual and the expected parameters. Likewise, in [14], the authors proposed a method based on a data-driven technique to detect point anomalies from in-orbit satellite telemetry data. They developed their model using the DDMN technique to identify the problem of fake anomalies due to errors in satellite data. Following this, a combination of the LSTM method and Gaussian model was utilized for training from multivariate time-series data and detecting anomalies. To solve the problems of poor interpretability and high rates of false positives, the authors of [5] proposed a data-driven method named CN-FA-LSTM for anomaly detection using satellite telemetry data. Similarly, the authors of [6] also developed a data-driven method for the prediction of Voltage and Current in LEO satellites. In [7], the authors developed a model using the LSTM model for boundary localization problem.

B. Federated Learning-based Intrusion Detection

The concept of distributed learning or FL has been widely implemented in numerous domains to offer privacy preserva-

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2022.3214652

tion and train models from data of various clients. Several research studies have been conducted to develop IDS methods using FL. For example, in [15], the authors developed an FL-driven blockchain approach to classify cyber-attacks. The FL-based contributing parties are responsible for audit model updates and can be held accountable. To identify compromised IoT devices, the authors of [16] proposed a distributed FL-driven autonomous self-learning method to discover intrusions.

Yang *et al*. [17] proposed a method named FDAGMM to detect anomalies from large-scale data sources. Recently, Sawsan *et al*. [18] proposed FL-based for detecting intrusions in IoT networks utilizing NSL-KDD data source to evaluate the effectiveness of their proposed model. Xinhong *et al*. [19] also suggested an FL-based method to detect malicious attacks from cloud servers. They employed blockchain technology to store the trained parameters and applied a filter identification module-based alert method to reduce false alarm rates. In [20], the authors developed a security method using FL architecture named DFF-SC4N for protecting supply chain 4.0 networks.

To the best of our knowledge, the literature does not offer sufficient security solutions for protecting IoT-integrated satellite networks that use FL and allow for the cooperative building of security models without losing privacy. Hence, we propose an FL-based intrusion detection model and a viable deployment framework for attack identification in SSNs in this work. Although privacy is not the key focus, due to the integration of a federated learning architecture, the model data is not shared throughout the communication networks, making the proposed model privacy preserved compared to the traditional centralized learning approach.

III. The Proposed DFSat Framework

This section describes the proposed DFSat framework, including the core mechanism of security-by-design for attack detection in smart satellite networks.The proposed DFSat framework is presented in Fig. 1, which illustrates its key components. In this case, an anomalous party communicates with a compromised satellite, which could be used to target other satellites by launching inter-satellite cyber-attacks and could harm any network device that is approachable to it.

The proposed framework can evaluate the satellite-enabled systems and discover the abnormal data patterns in the network traffic. It depends on FL and recurrent network architecture for detecting cyber-attacks from smart satellite networks. An overview of the proposed model is depicted in Fig. 2, which is flexible to accommodate new devices and applications. New applications and devices could be added to the proposed model at any time by updating the model parameters in real-time.

A. Data Processing

It is essential to filter and clean feature values before the detection process. The steps involved in data preparation are dimensionality reduction and feature normalization. The former is implemented to decrease the feature dimension by removing unwanted features to enhance model efficiency and save computational time. The latter normalizes the data into a specific scale, i.e., between 0 and 1. This step is vital as it helps the classifier clear any bias from source data.

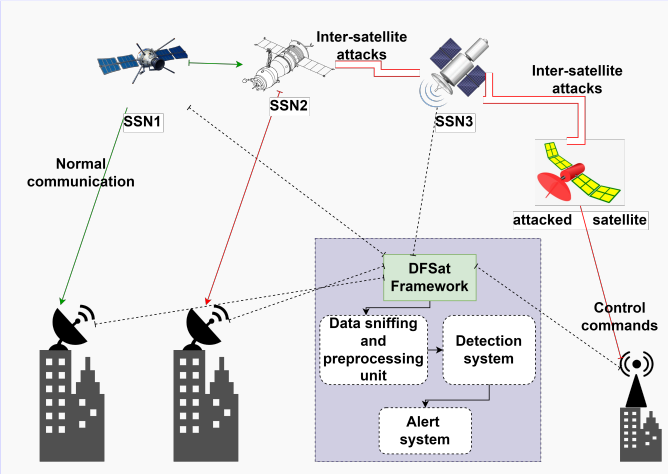

Fig. 1. Proposed intrusion detection framework for satellite networks

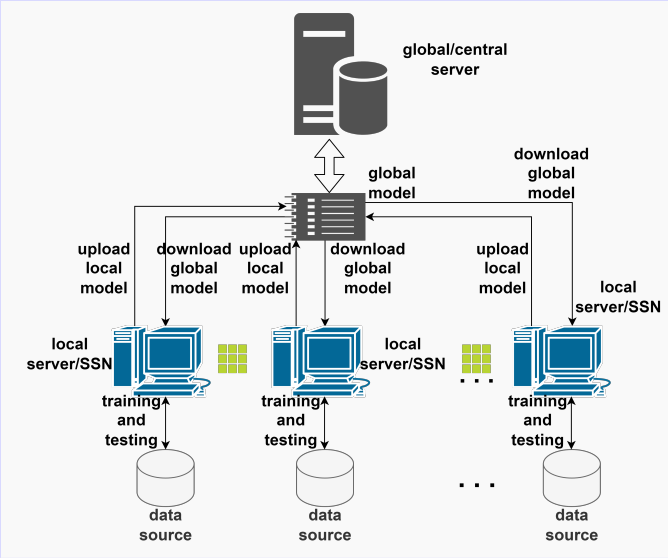

Fig. 2. Proposed architecture of federated learning for intrusion detection

B. Deep Learning-based Intrusion Detection

In this work, we employ deep learning-based intrusion detection using Bi-directional Long Short-Term Memory (BidLSTM) method. BidLSTM is a Recurrent Neural Network (RNN) type, and because of its distinct characteristics, it is considered suitable for handling time series data. The network comprises forward and backward propagation, which is commonly applied to model tasks related to natural language processing. This study employs the bidirectional relation between the SSNs data before and after processing. The LSTM units retain the basic continuous RNNs' structure. But, the cell state (cs) is additional in the LSTM network, which is transferred to the subsequent cell with t time. This cs is considered key to the LSTM architecture because it represents long-term memory. Throughout the computation of the cell series, the cell autonomously chooses to keep the previous information or not use the inputs on the current state and the estimations of the prior state.

To control the contents of cs, the input gate (ig) and forget gate (fg) are used, where ig determines the amount of input to be stored at t in cs and fg. They are responsible of determining how much of cs (cs_{t-1}) at $t-1$ time that can be retained at current time cs_t. In addition, to control how much cs of the unit is forwarded to the current output value hds_t, LSTM uses an output gate (og). During the update process in every iteration of LSTM, the cell receives the input x_t of the preceding cell and the hidden state (hds) hds_{t-1} in the cell series. After that, the cell decides whether to keep the information for output calculation or not, as formulated by:

$$fg_t = \sigma(wm_{fg} \cdot [hds_{t-1}, x_t] + bv_{fg}) \tag{1}$$

The activation function is represented by σ, wm defines the weight matrix, and the bias vector is represented by bv and $[hds_{t-1}, x_t]$, where the vectors are associated with a longer vector (i.e., the input and previous states). The current cs_t is determined by the ig, where the basic idea decides the amount of information to be retained over an activation function layer and then produces the cs_t at the current time t. The computational steps can be reflected as an integration of two parts, where one part is related to computing what input x_t information in the current cell to be stored in cs_t of the LSTM unit. The activation function layer decides which information needs to be upgraded, as formulated in Eq. 2. Eq. 3 formulates the second part in which the *tanh* layer produces a vector which is the substitute updated content and is given as:

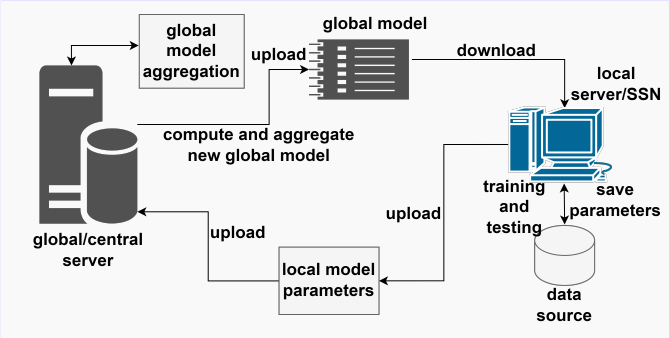

Fig. 3. Overview of FL-based intrusion detection architecture

$$ig_t = \sigma(wm_{ig} \cdot [hds_{t-1}, x_t] + bv_{ig}) \tag{2}$$

$$\tilde{cs}_t = tanh(wm_{cs} \cdot [hds_{t-1}, x_t] + bv_{cs}) \tag{3}$$

The cs_t of the new LSTM unit can be measured by joining the outputs of ig and fg, is given by:

$$cs_t = fg_t * cs_{t-1} + ig_t * \tilde{cs}_t \tag{4}$$

Lastly, the og of the LSTM unit identifies the current hds_t of the current unit cell at t time as:

$$hds_t = tanh(cs_t) * \ldots (wm_{og} \cdot [hds_{t-1}, x_t]) + bv_{og} \tag{5}$$

where the weight matrix of og is denoted by WM_{og} and the bias vector of og is denoted by bv_{og}.

BidLSTM processes the data in both directions by combining forward and backward propagation. This bi-directional architecture can improve the detection rate by capturing the semantic dependencies in both ways. The purpose of adding a dropout layer is to randomly neglect some neurons throughout the learning process during each iteration so that the weight of the corresponding neurons stays the same as in the preceding stage while updating the weights of other neurons. This phenomenon helps avoid the overfitting problem by reducing the number of neurons in the hidden layer and keeping the proposed method from going into a local optimum solution.

C. Federated Learning-based Intrusion Detection

FL is founded on edge computing and distributed ML concepts, whose weights are the same as distributed ML algorithms. But, unlike distributed ML, every client in FL has comprehensive discretion upon the data on local servers and can independently pick to adopt FL for modelling. In addition, FL highlights the process of model training in a privacy-preserving fashion, as data privacy protection for SSN owners is significantly essential. Conventional privacy-preserving methods, e.g., l-diversification, k-anonymity and differential privacy, use generalized methods or apply noisy data to transform sensitive data features until a third party or an attacker cannot differentiate the original from noisy data.

These methods transmit original data, and a potential breach or outbreak is probable. Since no data transmission occurs in the FL environment there is no probability of sensitive data leakage at the data level, nor does it disrupt rigorous data protection regulations. The straightforward architecture of the FL-enabled IDS method is presented in Fig. 2. There are several IoT-based SSN owners/firms (M), and each one has a local server to build a local model using its data (DS_j). Initially, the global or central server sends the global model (gm_t) to every local server at every round of communication (cr).

The local server then builds the local model using their respective training data if the global model gm_t is not found; otherwise, the local server loads the global model gm_t first and then starts the training process. After the model training is executed at the local server, the weight parameters we_j^t of the trained model by each local machine are transferred to the central server by the local servers. Upon receiving all the trained models from local servers through communication round at t, the weights of the model are updated as:

$$we_{t+1} = \sum_{j=1}^{M} \frac{m_j}{m} we_{t+1}^m \tag{6}$$

The updated weights of the parameters are averaged and uploaded to the global server, where mj represents every local server data, m represents the sum of all the local servers' datasets dsm_j. After the computation of average weights, the new weights we_{t+1} are loaded by the global server, the compilation is executed, and a global model gm_t is formed at the central server. Lastly, the updated global model is instantly redeployed to every local server, and this process is done iteratively.

To better understand the communication process, the schematic diagram of the proposed method is shown in Fig. 3. Throughout every communication round, four steps are performed: i) transmitting a global model to local servers; ii) training at the local server; iii) uploading updated model weights to the central server; and iv) generating a new global model by aggregating local trained models. To prevent the method from going into a local optimum solution, the epoch is set to 1 to enhance the overall performance of the proposed FL-assisted IDS.

IV. Evaluation Results and Analysis

This section discusses the experiments conducted to evaluate the efficiency of proposed DFSat using ToN_IoT and UNSW-NB 15 datasets [21]. Firstly, the experimental environment, dataset and evaluation metrics are outlined. Then, the performance of the proposed method is explained using different experiments under the FL environment. A comparison with benchmark methods is constructed to evaluate the results.

A. Experimental Environment

Experimental evaluation of the proposed DFSat framework is executed using the assessment criteria mentioned in Section IV-C. The details of the practical environment are outlined in Table I. The proposed intrusion detection model was trained using an Adam optimizer with $lr = 0.001$ as a learning rate, with the sigmoid activation and cross-entropy loss functions. Throughout the training process, the hyper-parameters of the learning model were appropriately tuned to accomplish improved results.

TABLE I
Experimental setup of the development environment

System configuration	Model
OS	Ubuntu 18.04.2 LTS
Working environment	Python
CPU	Silver 4110 CPU 2.10 GHz (Intel (R) Xeon (R))
Ram	128Gb
Working Libraries	Latest versions of Numpy Keras TensorFlow Scikit-learn
GPU	NVIDIA RTX2080 GPU

B. Dataset description

Several experiments were conducted on a data source [22] created from real-world IoT networks designed at the IoT Security Lab, UNSW Canberra, to evaluate the framework's performance. This data source has different attack events, such as DDoS, XSS, Injection, Backdoor attacks, etc. The abnormal data was collected during the attack to acquire various normal and cyber-attack events from IoT environments. Table II lists the feature description of this dataset.

To further validate the performance of the proposed model, we also assess the efficiency of the DFSat framework on the UNSW-NB 15 dataset [23], which includes various security events collected from network traffic. The description of some of the features of this dataset is outlined in Table III.

TABLE II
Description of some features of ToN_IoT data

Feature	Type	Description
time	time	logging time of data
date	date	logging date of data
motion_status	number	status of a motion sensor where 1 indicates on and 0 indicates off
light_status	boolean	indicates the staus of ligh as on or off
temperature	number	measurement of temperature sensor attached to network
pressure	number	reading of pressure sensor attacked to network
humidity	number	reading of humidity sensor attacked to network
sphone_signal	boolean	receiving door signal status on phone
latitude	number	GPS tracker latitude value
longitude	number	GPS tracker longitude value
FC1_Read_Input_Register	number	code of Modbus function for handling input register readings
label	number	indicates the class of record as attacked data or normal data

TABLE III
Description of some features of UNSW-NB 15 data

Feature	Type	Description
srcip	nominal	IP address of the source
dstip	nominal	IP address of the destination
proto	nominal	Protocol type used in Transaction
Sload	Float	BPS from the source
Dload	Float	BPS from the destination
Stime	Timestamp	Starting time of the record
Ltime	Timestamp	Ending time of the record
Spkts	Integer	Number of packets from source to destination
label	Binary	indicates the class of record as attacked data or normal data

C. Evaluation Metrics

Four evaluation metrics, including Recall (Reca), Precision (Prec), F-measure (f-mea), and Accuracy (Accy), are utilized for evaluation of the proposed approach because of their wide usage in assessing ML/DL models [16], [24]. They are briefly defined below:

Accy is the most intuitive efficiency factor and universally employed performance evaluation metric. It specifies the ratio of the correct data $T_p + T_n$ to the total data, which shows the Accy of the overall outcome.

$$Accy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{7}$$

The Prec metric is used to calculate the score of actual samples in entire positive cases of recognition.

$$Prec = \frac{T_p}{T_p + F_p} \tag{8}$$

The recall metric defines the percentage of positive samples refereed as positive for all positive occurrences in the data source. The rate of recall reveals the detection rate of security events.

$$Reca = \frac{T_p}{T_p + F_n} \tag{9}$$

F-mea is one of the precise and concise indicators for performance evaluation. The micro averaged values are exploited for comprehensive performance evaluation of all deliberated IDS models.

TABLE IV
PERFORMANCE COMPARISON OF ALL METHODS

Model	Accy	Prec	Reca	F-mea
Local server 1/(SSN 1)	93.90	94.35	88.95	93.49
Local server 2/(SSN 2)	94.43	94.68	90.89	94.15
Local server 3/(SSN 3)	94.08	94.58	88.89	93.66
CNN (centralized)	92.64	95.67	66.82	80.11
LSTM (centralized)	99.87	99.87	99.74	99.87
DFSat (federated)	**99.66**	**99.66**	**99.58**	**99.60**

$$F - mea = 2 \times \frac{Prec \times Reca}{Prec + Reca} \tag{10}$$

where the true positives are denoted by T_p, true negatives are represented by T_n, F_p represents false positives, and false negatives are denoted by F_n.

D. Comparative Results and Discussion

1) Federated Learning (F) vs. Centralized Learning (C): The comparative performance of the proposed framework is discussed, where the comparison is comprised of two cases. The first case uses all the local serves datasets for performance evaluation in the training process. The second case of the comparison model is for performance evaluation using the local dataset in the training process. The proposed federated learning model contains four steps in each communication round, i.e., the transmission of global mode, training at the local server, uploading local-trained parameters, and generating a new global model by aggregating local model parameters.

The total time consumed by local servers in the training process is approximately $17s$ in every round of communication. It is similar to the training time of the centralized method. The trajectory of accuracy curves and model losses via FL-based LSTM and centralized LSTM are shown in Fig. 4 and Fig. 5. From Table IV and Fig. 6, it is realized that even after the aggregation and transmission of the model by the global/central server throughout the training process in every communication round. It can still preserve the results of learned features in the local servers' dataset. Thus, the loss value gradually decreases, and the overall accuracy rate inclines.

The comparison of performance efficiency between the F-LSTM model and the C-LSTM model is demonstrated in Table IV. The accuracy rate for the F-LSTM method reached as high as 99.66%, with 99.66% for precision, 99.58% for recall, and 99.60% for f-measure, respectively. The C-LSTM model's accuracy rate reaches 99.87%, about 0.2% enhancement over the F-LSTM method. Since the idea is to learn from the full features of the dataset in the training process without any communication round, it is adequate to accomplish this outcome. Also, the performance of the F-CNN model is not desirable to be compared, as the experimental results display that the F-LSTMs' performed better than the C-CNN method. It can also be observed from Table IV. The proposed F-LSTM method did not considerably lessen the LSTM model performance and precisely identified real-time intrusions from IoT-augmented satellite networks.

The second case contains the training process of local serves using a local dataset. Table IV represents the results of the conducted tests, and the results of the F-LSTM are displayed in Fig. 4. In this figure, the blue line shows the training accuracy, and the red indicates the validation accuracy using LSTM as a training method. It can be observed from Table IV that the average accuracy rate for detecting intrusions and malicious activities by local servers using the local dataset is about 93%. Hence, the detection performance of F-LSTM compared to the local methods and other models have apparent advantages by achieving a higher accuracy rate and a lesser and more enduring value of the loss.

Lastly, the results of experiments conducted to evaluate the performance of the proposed model are summarized in Table IV. The best values achieved by each model during the testing process are shown in terms of accuracy, precision, recall (can also be termed as the rate of detecting intrusions), and F-measure. From the results of Table IV, we can observe that the F-LSTM model performed better in detecting intrusions than the other models. The possible reason for the reduction of recall score could be the prediction process of the proposed framework, where the proposed framework predicted more false negatives than false positives. However, the precision score is more vital than the recall score since we want a lower false-positive rate than false negatives. Thus, the recall score of

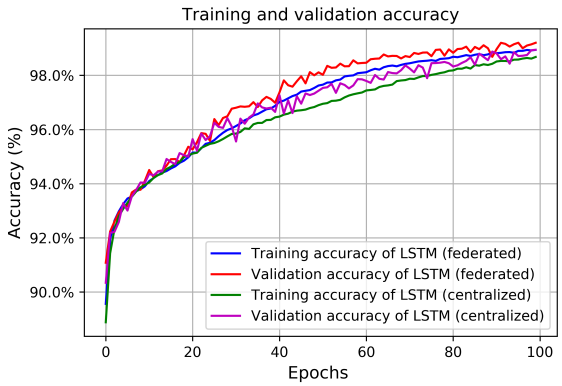

Fig. 4. Comparison of training and validation accuracy for LSTM architecture in centralized and federated mode

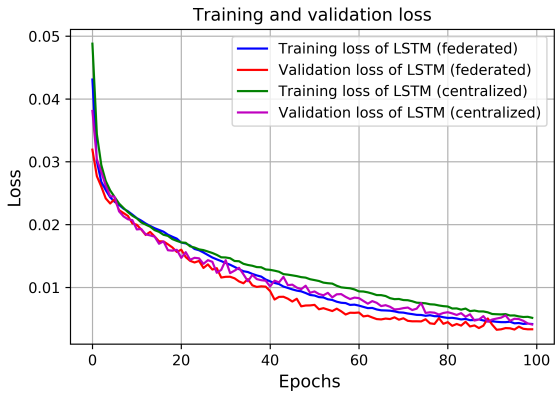

Fig. 5. Comparison of training and validation loss for LSTM architecture in centralized and federated mode

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2022.3214652

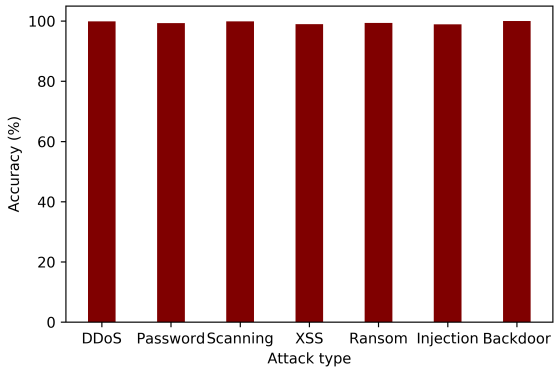

Fig. 6. Comparison of accuracy score for different categories of attacks

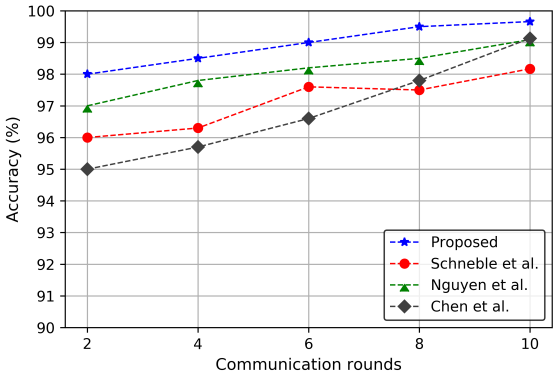

Fig. 7. Comparison of accuracy with existing researches using different value of communication rounds

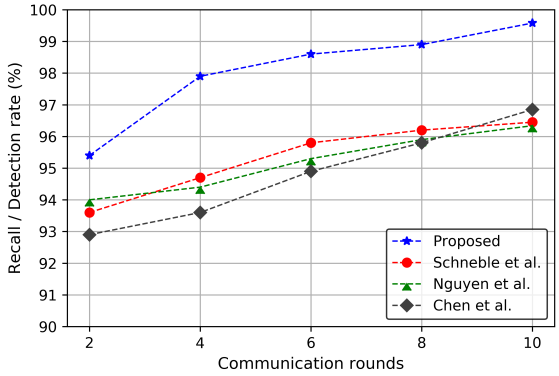

Fig. 8. Comparison of detection rate with existing researches using different value of communication rounds

the proposed framework is adequate since false positives are more costly than false negatives. Nonetheless, the proposed model accomplished a high precision score of 99.60%.

2) CNN VS LSTM: After the pre-processing process, different algorithms, such as CNN and LSTM, were trained for testing and comparison. The binary cross-entropy function was used, utilizing the Adam optimizer by 0.001 as initial learning rates. The hyperparameters of the testing algorithms were adjusted during this study to attain enhanced performance. The performance testing for these experiments is done over the entire data source set in the training process. The experimental results reveal the prediction ability of the proposed FL-based method using the complete dataset, which is the target of this study.

The outcomes of this comparison are outlined in Table IV, concerning the accuracy, precision, recall, and f-measure. From Table IV, we can observe that the LSTM model performed better than the CNN model by achieving a 99.66% rate of accuracy, 99.58% detection rate, and a lower loss value. The overall influence of identifying intrusions using the CNN model achieved lower results, achieving a 92.64% rate of accuracy, 66.82% detection rate, and higher loss value. The LSTM model delivers better outcomes through its adaptability to time series detection.

3) Performance Comparison: We also compare the performance of the proposed framework with the existing state-of-the-art methods [16], [24], [25]. This comparison shows the advantages of the DFSat framework against contemporary studies and quantifies the extent to which the proposed framework beats the compelling ones. These approaches have been chosen for comparison because of their appropriateness for research.

The experimental results of the evaluation are displayed in Table V in terms of rates for accuracy, precision, recall, and f-measure while considering diverse amounts of local serves/industrial firms under different rounds of communication (such as 2, 4, 6, 8, and 10). From Table V, we can observe that the developed FL-based IDS method outperforms existing effective benchmark methods using all the performance metrics.

TABLE V
COMPARISON OF PERFORMANCE WITH BENCHMARK MODELS

Model	Accy	Reca	Prec	F-mea
C. Yiqiang *et al.* [25]	99.13	96.85	99.03	97.90
S. William *et al.* [24]	98.17	96.45	98.79	97.57
N. Thien *et al.* [16]	99.09	96.34	99.91	97.77
DFSat (ToN_IoT data)	**99.66**	**99.58**	**99.66**	**99.60**
DFSat (UNSW-NB 15 data)	**99.87**	**99.74**	**99.87**	**99.84**

The performance of each detection method generally improves with the increasing number of communication rounds. It can also be noticed that the performance stabilizes gradually with a sufficiently large number of rounds. It is worth mentioning that the proposed model obtained 99.66%, 99.66%, 99.58%, 99.60% of accuracy, precision, recall, and F-measure, respectively, with three local servers/industrial firms. The results are also visually presented in Figs. 7 and 8 for all compared detection methods with varying rounds of communication. All the detection methods converge once adequate communication rounds are performed with the central/global server. The proposed FL-based IDS method achieved better performance than the benchmark methods.

Conclusively, the DFSat framework's advantage is developing a robust security solution for SSNs. This study delivers valuable, applicable implications for the acclimation of a distributed training-driven security framework for the safety of communication networks of IoT-driven satellite systems. However, the proposed DFSat framework must be evaluated on satellite data from diverse domains and IoT environments

to validate its generalization capability. A technique is required to be formulated for automatically tuning the hyper-parameters to upsurge the recall and f score.

V. Conclusion and Future Work

This study has proposed an effective attack detection model using a deep federated learning framework, DFSat, to mitigate and identify cyber-attacks in IoT-driven satellite networks. The proposed framework achieves a better detection rate as compared to state-of-the-art models while protecting the privacy of diverse satellite networks. The developed federated learning architecture enables multiple SSNs to collectively build a comprehensive intrusion detection system against contemporary cyber threats in a privacy-preserving manner. In addition, we designed a novel intrusion detection method named DFSat, which allows the efficient discovery of numerous types of cyber-attacks. The results on real datasets indicated that the proposed DFSat technique works well over state-of-the-art methods. This is because the LSTM architecture can offer enhanced semantic information in feature vectors combined with the learning characteristics through the weighted average parameters of each dataset in the training process. In the future, we plan to address security concerns in the satellite industry by federating different data resources from SSNs in diverse-domain by combining other feature vectors to improve the detection process and accomplish a more operational and wide-ranging system for intrusion detection.

References

[1] G. Falco, "When satellites attack: Satellite-to-satellite cyber attack, defense and resilience," in *ASCEND 2020*, 2020, p. 4014.

[2] M. Usman, M. Qaraqe, M. R. Asghar, and I. Shafique Ansari, "Mitigating distributed denial of service attacks in satellite networks," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 6, p. e3936, 2020.

[3] A. Gharanjik, M. B. Shankar, F. Zimmer, and B. Ottersten, "Centralized rainfall estimation using carrier to noise of satellite communication links," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 5, pp. 1065–1073, 2018.

[4] P. A. Networks, "The connected enterprise: Iot security report 2020," [Online] Available: hhttps://Www.Paloaltonetworks.Com/Resources/Research/Connected-Enterprise-Iot-Security-Report-2020.

[5] Z. Zeng, G. Jin, C. Xu, S. Chen, Z. Zeng, and L. Zhang, "Satellite telemetry data anomaly detection using causal network and feature-attention-based lstm," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–21, 2022.

[6] S.-T. Yun and S.-H. Kong, "Data-driven in-orbit current and voltage prediction using bi-lstm for leo satellite lithium-ion battery soc estimation," *IEEE Transactions on Aerospace and Electronic Systems*, 2022.

[7] Y.-T. Liu, J.-J. Chen, Y.-C. Tseng, and F. Y. Li, "An auto-encoder multi-task lstm model for boundary localization," *IEEE Sensors Journal*, 2022.

[8] M. Woźniak, J. Siłka, M. Wieczorek, and M. Alrashoud, "Recurrent neural network model for iot and networking malware threat detection," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5583–5594, 2020.

[9] M. Woźniak, A. Zielonka, A. Sikora, M. J. Piran, and A. Alamri, "6g-enabled iot home environment control using fuzzy rules," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5442–5452, 2020.

[10] Z. Na, Z. Pan, X. Liu, Z. Deng, Z. Gao, and Q. Guo, "Distributed routing strategy based on machine learning for leo satellite network," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[11] L. Gunn, P. Smet, E. Arbon, and M. D. McDonnell, "Anomaly detection in satellite communications systems using lstm networks," in *2018 Military Communications and Information Systems Conference (MilCIS)*. IEEE, 2018, pp. 1–6.

[12] B. Pilastre, L. Boussouf, S. d'Escrivan, and J.-Y. Tourneret, "Anomaly detection in mixed telemetry data using a sparse representation and dictionary learning," *Signal Processing*, vol. 168, p. 107320, 2020.

[13] F. Cheng, X. Guo, Y. Qi, J. Xu, W. Qiu, Z. Zhang, W. Zhang, and N. Qi, "Research on satellite power anomaly detection method based on lstm," in *2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*. IEEE, 2021, pp. 706–710.

[14] Y. Wang, J. Gong, J. Zhang, and X. Han, "A deep learning anomaly detection framework for satellite telemetry with fake anomalies," *International Journal of Aerospace Engineering*, vol. 2022, 2022.

[15] D. Preuveneers, V. Rimmer, I. Tsingenopoulos, J. Spooren, W. Joosen, and E. Ilie-Zudor, "Chained anomaly detection models for federated learning: An intrusion detection case study," *Applied Sciences*, vol. 8, no. 12, p. 2663, 2018.

[16] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and 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.

[17] Y. Chen, J. Zhang, and C. K. Yeo, "Network anomaly detection using federated deep autoencoding gaussian mixture model," in *International Conference on Machine Learning for Networking*. Springer, 2019, pp. 1–14.

[18] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: Centralized, on-device, or federated learning?" *IEEE Network*, vol. 34, no. 6, pp. 310–317, 2020.

[19] X. Hei, X. Yin, Y. Wang, J. Ren, and L. Zhu, "A trusted feature aggregator federated learning for distributed malicious attack detection," *Computers & Security*, vol. 99, p. 102033, 2020.

[20] I. A. Khan, N. Moustafa, D. Pi, Y. Hussain, and N. A. Khan, "Dff-sc4n: A deep federated defence framework for protecting supply chain 4.0 networks," *IEEE Transactions on Industrial Informatics*, 2021.

[21] N. Koroniotis, N. Moustafa, and J. Slay, "A new intelligent satellite deep learning network forensic framework for smart satellite networks," *Computers & Electrical Engineering*, vol. 99, p. 107745, 2022.

[22] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "Ton_iot telemetry dataset: a new generation dataset of iot and iiot for data-driven intrusion detection systems," *IEEE Access*, vol. 8, pp. 165 130–165 150, 2020.

[23] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 military communications and information systems conference (MilCIS)*. IEEE, 2015, pp. 1–6.

[24] W. Schneble and G. Thamilarasu, "Attack detection using federated learning in medical cyber-physical systems," in *2019 28th International Conference on Computer Communication and Networks, ICCCN*, 2019, pp. 1–8.

[25] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 83–93, 2020.