33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

ICAS 2022 SWEDEN

A FEDERATED LEARNING BASED SECURITY FOR CONTROLLER-PILOT DATA LINK COMMUNICATION

Suleman Khan, Gurjot Singh Gaba & Andrei Gurtov

Department of Computer and Information Science (IDA), Linköping University, Sweden

Abstract

The safety of the passengers and goods in airplanes depends upon a number of combined factors. An airplane's condition and the pilot's experience are pivotal but another very crucial element is the synchronization among the pilots and the air traffic controller (ATC). The communication link between the two carries many uncertain aspects. The aviation sector often tends to give more priority to safety rather than cybersecurity. Although the controller-pilot data communication link (CPDLC) system has been proposed for consistent and reliable communication recently, it has some serious drawbacks. In this paper, we highlight the shortcomings of the CPDLC system from a cyber security perspective. We propose a federated learning-based privacy-preserving intrusion detection system (IDS) to protect the CPDLC from uplink and downlink cyber attacks. To ensure a realistic and viable solution, we created our own training dataset by eavesdropping on the air-ground communication at a site near Arlanda airport, Sweden. The anomaly detection model constructed through federated learning has achieved higher accuracy, precision, recall and F1 score as compared to the centrally and locally trained models, enabling higher security. Due to the lower training loss and time, the proposed approach is highly suitable for the sensitive aviation communications.

Keywords: Aviation, CPDLC, Cyber-Attacks, Federated Learning, Intrusion Detection System.

1. Introduction

The aviation industry was severely hit by the coronavirus pandemic. As per the EUROCONTROL, the aviation sector is regaining momentum, and it is predicted that the air traffic will rise from 74 % to 105% by 2024 [1]. As the traffic density has increased gradually, the aeronautical engineers have been analysing the shortcomings of the current Air Traffic Management (ATM) system and protocols in mild and intense traffic scenarios. The rising saturation of the Very High Frequency (VHF) band in some parts of the world, especially Europe, and the lack of digitization, bandwidth, and cyber security therein, is creating problems in the growth of the civil aviation industry [2].

One of the significant challenges is reliable communication between various entities in civil aviation. Increased aviation traffic at one airport poses a threat to reliable communication, as the air traffic controller (ATC) has to communicate with more pilots on the same frequency channel. To avoid chaos, the aviation experts recommend the use of data link communication over the legacy analog voice communication [3]. Controller pilot data link (CPDLC) complements VHF radio voice communication by handling non-critical communication and has reduced miscommunications and increased communication effectiveness since its introduction. CPDLC enables the ATC and the pilots to exchange level assignments, crossing constraints, lateral deviations, route adjustments, route clearances, speed assignments, radio frequency assignments and other requests [4].

Air traffic control requires a high level of situational awareness, and the safety of the flight depends upon the communication between the pilot and the ATC. CPDLC is adopted at various levels across the world, and it is presently in its last stages of implementation [4]; however, its reliability in compromised conditions and robustness against attacks is still a big concern. CPDLC provides a

sensitive interface, and any exploitation of it could result in disastrous consequences, including premature landing, landing on the wrong runway, crashes, etc. Low-cost hardware like software-defined radios are easily available in the market, allowing ordinary people to access the sophisticated radio manipulation tools, thus bypassing the technical complexity that previously protected aircraft communication.

In the CPDLC's early-stage development, easy availability of software-defined networking equipment and non-consideration of security aspects has created doubts ifor aviation experts regarding the strength of CPDLC against modern-day cyber threats. The authors' investigation reports [5] that declare the CPDLC conceptually insecure have shaken the aviation industry and experts. Likewise, authors in [6], [7], and [8] show that the CPDLC system is vulnerable to attacks such as eavesdropping, injection, replay, man-in-the-middle (MITM) and impersonation.The consequences of these attacks could be mild to severe, like minor delays in flights schedule and crashes. Consequently, the aeronautical telecommunications network requires a high level of security for protecting the air to ground communication and vice-versa.

1.1 Problem Statement and Motivation

Aviation has empowered countries in many aspects, including the exchange of skilled labour and trade. However, the associated security threats of using airplanes for transportation and commutation always make the stakeholders anxious. Many crash incidents have occurred in the past, whose investigations revealed that they occurred due to miscommunication rather than technical or mechanical system failure. Aviation experts proposed CPDLC for reliable communication to reduce crash incidents. To some extent, CPDLC has been a blessing to the ATC and the cockpit crew. However, One of shortcomings of the CPDLC protocol is that it uses plaintext communications between air-ground terminals without any built-in security features, exposing the controller and pilot to an enormous attack vector. For instance, an adversary can manipulate CPDLC's communications, such as route adjustments, speed allocations and radio-frequency assignments. Such falsified communications can result in devastating outcomes for passengers, cockpit crew, aircraft and the aviation industry as a whole. As per an investigation by Eurocontrol, the number of cyber-attacks has risen to 530% since the deployment of CPDLC in 2017. In 2020, there were 775 cyber-abuses recorded against airlines, and 150 were reported against airports. Around 95% of these attacks were carried out for financial gain, with 55% resulting in a loss of monetary value, whereas 35% were related to data breach [9]. An airplane transports several valuable entities, including humans. Hijacks are relatively common. Therefore, the security issues of CPDLC need to be addressed to ensure a high degree of security and safety.

1.2 Our Contributions

- We propose a federated learning-based privacy-preserving intrusion detection system (IDS) to prevent the CPDLC from uplink and downlink attacks.
- To develop a practical realizable anomaly detection model, we created our own training dataset by eavesdropping on the air-ground communication at a site near Arlanda airport, Sweden.
- We have deployed a deep neural network architecture that trains the model with realistic and malicious data in minimal time and loss.
- We investigated the performance of centralized, localized, and federated learning-based anomaly detection models in terms of detection accuracy, precision, recall, F1-score, and training/testing time.

1.3 Paper Organization

The rest of the paper is organized as follows: *Section 2* discusses the related work. *Section 3* contains the proposed solution, whereas *Section 4* elaborates on the results and discussion. Finally, *Section 5* summarizes the paper and sheds some light on the future directions.

2. Related Work

Several researchers have doubted the cyber-security of the CPDLC system, and many also agree that it is insufficient. Stromheimer et al. [8] discussed the security vulnerabilities in the most aircraft communication systems, including CPDLC. Di Marco et al. complemented Stromheimer et al. [8] observations by performing injection and manipulation attacks in CPDLC [5]. Likewise, Wernberg discovered the cyber-threats on CPDLC and suggested preventive measures [10].

Smailes et al. [7] discussed the impact of the MITM attack on the handover phase in the CPDLC drove ground-air communication. The successful MITM can allow the attacker to hijack the communication link between the pilot and the ATC, thereby enabling the attacker to send arbitrary malicious CPDLC commands towards the target without alerting the legitimate controller. On the contrary, the authors in [7] also provided countermeasures to protect the air-ground interface from MITM attacks.

Strohmeier in [11] investigated the aviation ecosystem and identified various possible threats (e.g., jamming, eavesdropping, message injection, message deletion) to ground-air communications and vice versa. Eskilsson et al. [12] demonstrated several cyber-attacks using software-defined radio (SDR) that can compromise the confidentiality and integrity of communications besides giving unauthorized access to ADS-B and CPDLC systems.

Gurtov et al. [6] emphasized the technical aspects and properties of the CPDLC system from a cyber-security perspective. The authors developed a threat model to highlight the security insufficiency in the CPDLC system. An attempt has been made by [13] and [14] to protect the illegitimate penetrations in the CPDLC system using asymmetric and elliptic-curve cryptography. In another research [15] the investigators designed and implemented a security model that offers several important security features like mutual authentication, confidentiality, and secure handover in the ground-air aviation network.

The cyber and aviation experts have put a lot of effort into determining and fixing the vulnerabilities; however, a sustainable solution that can address the futuristic mutations of cyber abuses has not been developed yet. Most existing research articles provide cryptography-based solutions that can withstand sophisticated modern-day attacks, but they may not be competent enough to prevent zero-day and mutated cyber attacks. The state-of-the-art methods lacks a sustainable and resource-efficient approach that can prevent attacker intrusions in the CPDLC communication.

3. Proposed Model

The behavior of the airplane changes as it moves from one terrain to another. Therefore, it is essential to examine the overall conduct of the airplane before considering any random and unseen event as an anomaly. Global modeling can enable the IDS to efficiently analyze the behavior of an aircraft with every ATSC/ATC and vice versa. We propose a privacy-preserving federated learning model for the CPDLC system to protect against cyber abuses. Figure 1 illustrates the framework of the proposed model. As a use case, we have considered Sweden, but this framework applies to all instances where CPDLC is used for communication.

Sweden has 11 ATSCs, out of which, one can be designated as the central node for constructing the global model. The ATSCs prepare a featured dataset for themselves and all the airplanes entering their designated area. The ATSC also puts together a local model based on the local dataset after data pre-processing, and then shares the local gradients for itself and each aircraft with the global ATSC. The global ATSC, Arlanda Stockholm in this case, receives the local gradients from every ATSC. The federated cloud server at Global ATSC uses a four-layer deep neural network architecture to prepare a global model. These global gradients are next shared with every single one of the ATSC and then with the airplanes. This global model has the tendency to protect the CPDLC communications from cyber intrusions. The supervisory bodies, including the Swedish Civil Aviation Administration, can also access the global model, dataset and other relevant details from the federated cloud server.

3.1 Dataset and Data Pre-processing

In this sub-section, our data collection and generation methods will be discussed.

Figure 1 – Federated learning based IDS for the CPDLC system

3.1.1 CPDLC Data Generation

In order to collect the CPDLC real messages, we established an experimental setup which includes an RTL-SDR dongle (R820T2), an antenna, and a Chromebook running Crouton. RTL-SDR is a hardware-based software-defined radio that has been used to capture real-time communications between the airplanes and ATC. We used Dumpvdl2 to decode the captured messages. We collected the data at a site near Stockholm's Arlanda Airport and used dumpvdl2 for the interpretation of messages. We collected 4040 messages in 3 days, which were later categorized into ATC and airplane messages. Finally, the decoded data was stored as plain text for further processing and analysis. CPDLC messages are unencrypted, and anybody with a radio receiver can intercept, read, and alter legitimate messages. Our investigation on CPDLC messages revealed that CPDLC messages adhere to the generic guidelines of the civil aviation; however, it does not comply to the security standards. Table 1 provides the information on features (e.g., address, status, header) and their characteristics (e.g., numeric and non-numeric).

3.1.2 Dataset Description

As seen in the Table 1, the first feature is *src_addr*, which represents the unique address of the source. The next feature in the table *src_type* defines the source type, i.e., *ground station* or *aircraft*. The features *dst_addr*, *dst_type* and *dst_status* provides information about the CPDLC message destination. Similarly, the features *header_msg_id*, *header_msg_ref*, *header_datetime*, and *header_logical_ack* represent the identity of the message, the identity of the message being answered, timestamp and response expectancy. The feature *data_choice* specifies the direction of the message, i.e., uplink or downlink, whereas *data_choice_label* provides a meta-data. The feature *data_choice* explains the message type, whereas *data* contains the message content. These three features — *data_choice*, *data_choice_label* and *data* appear five times in the dataset, with indexes ranging from 0 to 4.

Table 1 – Features extracted from CPDLC messages

Feature	Type	Feature	Type
s.no	Numeric	data_0	Non-Numeric
src_addr	Non-Numeric	data_choice_1	Non-Numeric
src_type	Non-Numeric	data_choice_label_1	Non-Numeric
src_status	Non-Numeric	data_1	Non-Numeric
dst_addr	Non-Numeric	data_choice_2	Non-Numeric
dst_type	Non-Numeric	data_choice_label_2	Non-Numeric
dst_status	Non-Numeric	data_2	Non-Numeric
cr	Non-Numeric	data_choice_3	Non-Numeric
header_msg_id	Numeric	data_choice_label_3	Non-Numeric
header_msg_ref	Numeric	data_3	Non-Numeric
header_datetime	Non-Numeric	data_choice_4	Non-Numeric
header_logical_ack	Non-Numeric	data_choice_label_4	Non-Numeric
data_choice_0	Non-Numeric	data_4	Non-Numeric
data_choice_label_0	Non-Numeric	genuine	Non-Numeric

3.1.3 Attack Data Generation Using GAN

To mimic the distribution of authentic CPDLC messages, we have used Generative Adversarial Network (GAN) [16] to learn the normal data distribution and generate attack data. The attack data is inspired from the original dataset for every aircraft and ground stations with a unique International Civil Aviation Organization (ICAO) or source address. GANs consists of two neural networks, that generates attack CPDLC messages from input noise and the discriminator.

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{1}$$

Equation (1) shows the objective function of a GAN, where D and G stand for the discriminator and generator respectively. The probability distribution of generated data is $p_z(z)$, whereas that for the real data it is $p_{data}(x)$. The generator aims to decrease the objective function to a minimum, whereas the discriminator aims to increase it to its maximum. The dataset comprises of 4040 legitimate CPDLC messages that were exchanged between the aircraft and ATSC. Another set of 4040 messages is generated using GAN. These 4040 attack messages are combined with the 4040 legitimate messages to form the dataset required for training the DNN model. Figure 2 depicts the absolute mean and standard deviation of all the features extracted from the actual CPDLC data. The line represents an approximation developed by our GAN model to match the CPDLC message features closely.

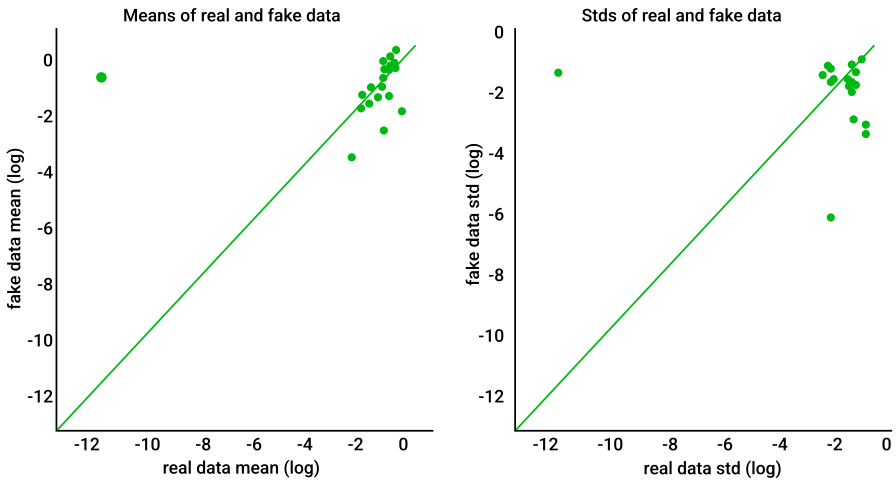


Figure 2 – Absolute log mean and standard deviations of real and simulated CPDLC data

3.2 Data Preprocessing

Before sending the data to the federated learning-based model, it is necessary to clean the data. Some of the captured CPDLC messages had missing feature values. We replaced the missing values with the mean value of that feature. Post insertion, we performed label encoding. Our dataset has both numeric and categorical values. Machine learning algorithms produce improved results on the numeric data. Therefore, we converted categorical values to an integer value. We used LabelEncoder() function from sklean package in python [17]. The final step of our data preprocessing is data normalization. We used the min-max normalization method to scale the data between 0 and 1.

$$\overline{F} = \frac{f - \min(F)}{\max(F) - \min(F)} (new\_\min(F) - new\_\max(F)) + new\_\min(F) \tag{2}$$

F in equation (2) represents a feature in the given dataset, whereas $min(F)$ and $max(F)$ represent the minimum and maximum values of the features respectively. $\overline{F}$ is the updated value of every entry in a dataset, f is the previous value in the dataset, $new\_max(F)$ and $new\_min(F)$ are the upper and lower boundaries of the given range. In this paper, we used 0 as the lower and 1 as the upper boundary. After cleaning the dataset, we used deep neural network (DNN) to train the local and global models.

3.3 Deep Neural Network (DNN)

DNN is a popular supervised learning method that needs a significant amount of labeled training data. It consists of three layers: input, hidden and output layers. Each layer has a certain number of nodes alias neurons. The number of input nodes is equal to the number of sample features, while the number of output nodes is typically equal to the number of labels or classes. Tunable hyper-parameters are the number of hidden layers and nodes. The proposed framework employs the DNN in a federated learning environment to detect anomalies in an air-ground communication.

Figure 3 depicts the implemented DNN structure, which includes input layers, hidden layers and an output layer. The proposed framework has an input vector F with 26 features. To classify anomalous and real messages, we used the output vector $Y = [y]$, which contains the probability values $[0, 1]$. Our model consists of 4 hidden layers followed by the ReLu activation function.

For each hidden layer (h), the output is mathematically expressed in equation 3.

$$h_i(f) = A\left(w_i^T f + b_i\right) \tag{3}$$

$A(.)$ is the nonlinear activation function, and w_i and b_i are the hidden layer weight and bias, respectively. The activation functions used in this work are *ReLU* for hidden layers and *sigmoid* for the output layer, which is computed using equations 4 and 5 respectively.

$$\mathrm{ReLU}(x) = \max(0, x) \tag{4}$$

$$\mathrm{sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

3.4 Federated Learning Based IDS

Federated Learning recently gained a lot of attention due to its inherent advantages including resource efficiency and data privacy. It is a new kind of learning that avoids centralized data collection and model training. In Federated Learning (FL), instead of exchanging heavy datasets with the centralized server for model training, the edge devices prepare a local model and only share the gradients with the centralized server. The centralized server, in turn, aggregates the various models and disseminates a generalized model to the clients. On the contrary, all the edge nodes propagate the entire data toward a centralized server in centralized learning. The centralized server then executes a machine learning model on the large dataset and produces a global model.

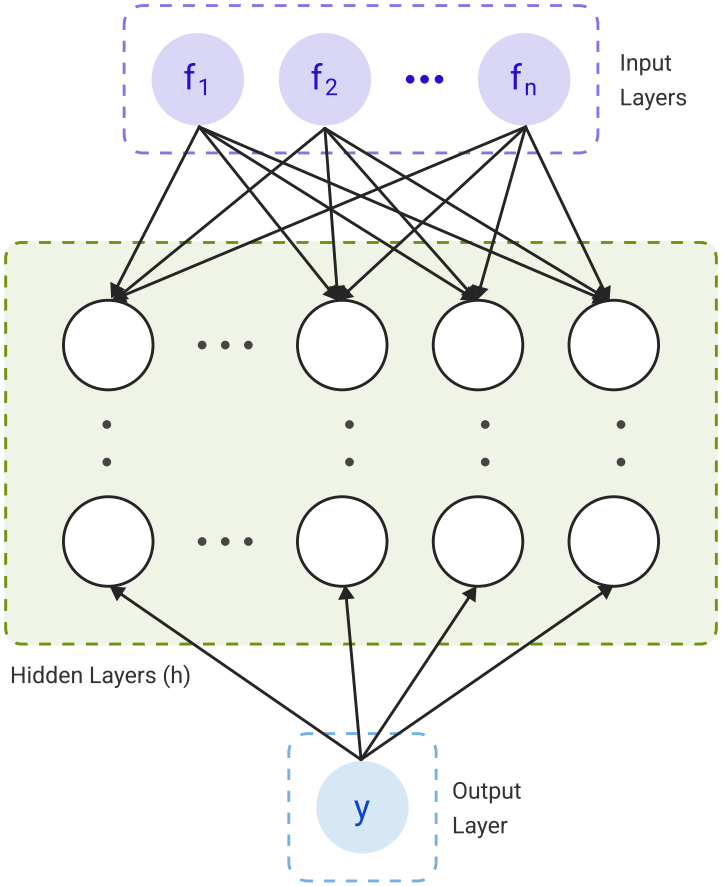


Figure 3 – Deep Neural Network Architecture

Centralized Learning (CL) is resource-expensive as the edge nodes need to send the entire dataset to the server; moreover, the server needs to perform extensive computations due to the size of the data. Besides, CL also mandates sharing sensitive private information with the centralized server. Gathering and sending large amounts of data in CL incurs more bandwidth expenses and is susceptible to adversarial data attacks. Also, keeping the data private to the operational ground station and transmitting the model parameters is a far more efficient and secure approach. Due to the aforementioned problems in centralized learning, we propose federated learning based IDS for the CPDLC system. Figure 4 illustrates the procedural flow of our proposed FL based IDS framework.

3.5 Model Training

An FL-specific framework called PySyft has been proposed in [18] to implement PyTorch. PySyft is a Pythonic version of FL for secure and private Deep Learning. Algorithm 1 discusses the operational workflow of our proposed IDS. Initially, the CPDLC messages are fed as an input to the DNN model. Afterward, encoding is performed to convert non-numerical features to numeric ones and a standard scalar to scale the data. After converting the data into tensors, the entities train their model using the processed data. The ATSCs exchange the local gradients with the global ATSC, which is aggregated by the global ATSC. The global ATSC disseminates the global gradients with the other ATSCs and the airplanes. Finally, the constructed local and global model is tested for reliability and robustness.

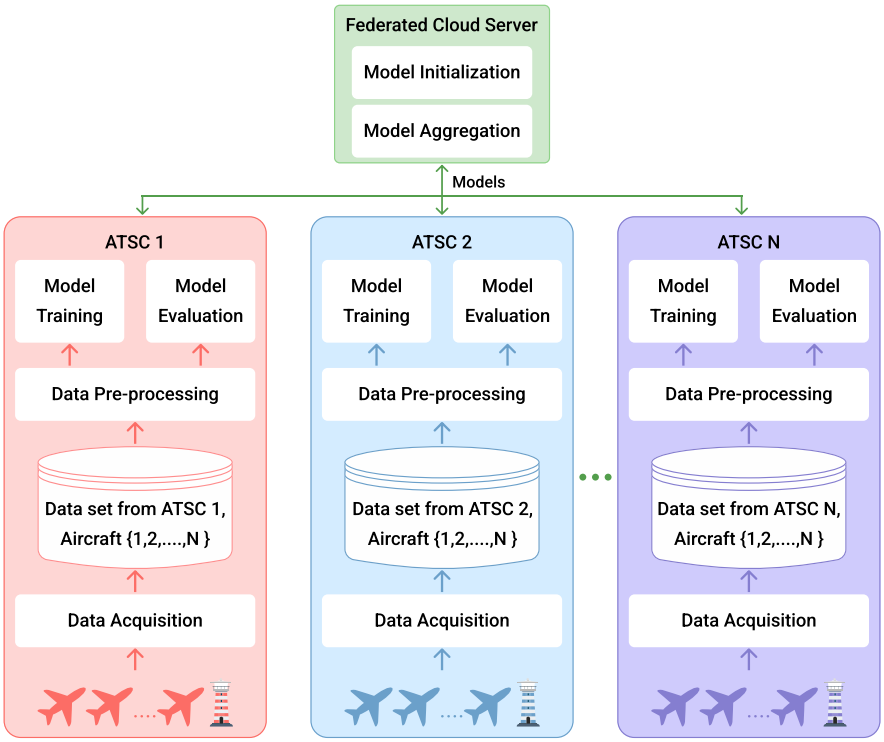


Figure 4 – Proposed IDS framework

Algorithm 1: FL based IDS for CPDLC

1. **Input**: I $\leftarrow f1, ...f26$
2. **Output**: Y $\leftarrow y_1$
3. Data Pre-processing
   3.0 Replace Missing Values $A = \frac{1}{n}\sum_{i=1}^{n} f_i = \frac{f_1 + f_2 + \cdots + f_n}{n}$
   3.1 Label Encode: $I[C] \leftarrow L(C)$
   3.2 Apply Standard Scalar: $\frac{f - \min(F)}{\max(F) - \min(F)})$
4. Convert train data to PyTorch tensor
   4.1 I $\leftarrow$ Tensor(I')
5. Split data among workers
   5.1 A1 $\leftarrow$ I[a1]
   5.2 A2 $\leftarrow$ I[a2]
   5.3 ATSC1 $\leftarrow$ I[atsc1],..,ATSC3 $\leftarrow$ I[atsc3]
6. Initialize w_0
7. Model, $M = DNN()$
8. *For* epoch in E
   8.1. $S_t \leftarrow$ (select aircraft or ATSC from C = 1,2..)
9. *For* each aircraft or ATSC in S_t
   9.1 $c.to(M)$
   9.2 send model to aircraft or ATSC
10. $w_{t+1}^k \leftarrow w - \eta * \nabla l(w; b)$
    10.1 update aircraft and ATSC weights
11. Aggregate
    11.1 $w_{t+1} \leftarrow \sum_{c=1}^{c} \frac{n_c}{n} * w_{t+1}^c$
    11.2 $\sum_{c=1}^{C} w_{t+1}^k \leftarrow w_{t+1}$
12. Update local aircraft and ATSC with the global parameters
13. Test the model accuracy $(\%)$ on each aircraft and ATSC

4. Results and Discussion

We executed and investigated the proposed approach on the $8th$ generation Intel core $i7$ processor GPU. Python 3.7 has been used to develop DNN with four hidden layers, and these layers have 26, 100, 250 and 100 nodes respectively. As a use case, we have considered two airplanes and three ATSCs, out of which one ATSC also performs the aggregation required to prepare the global model.

4.1 Experimental setup

We have considered the DNN model with input layer having 26 features, four hidden layers, with ReLU activation at each layer, and an output layer with sigmoid activation as shown in table 2. We have trained our federated-learning model with ten local epochs and the secure FedAvg method. An SGD optimizer with a 0.01% learning rate is used to train the models. We calculated training and validation loss using the cross-entropy function (CEF) using which we can determine the difference between the actual value of the final dependent and our predicated variable.

Table 2 – Model Parameters

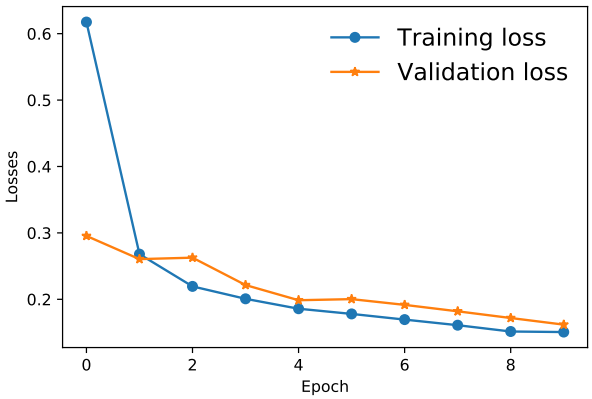
Parameters	**Description**
FL Model	FedAvg
FL library	PySyft
Input size	26
Output size	2
Number of hidden layers	4
Neurons per layer	26/100/250/100
Batch size	32
Activation function	ReLU and Sigmoid
Loss function	Cross-entropy function
Optimization	SGD
Learning rate	0.01
Number of epochs	10

4.2 Centralized vs. Federated Learning

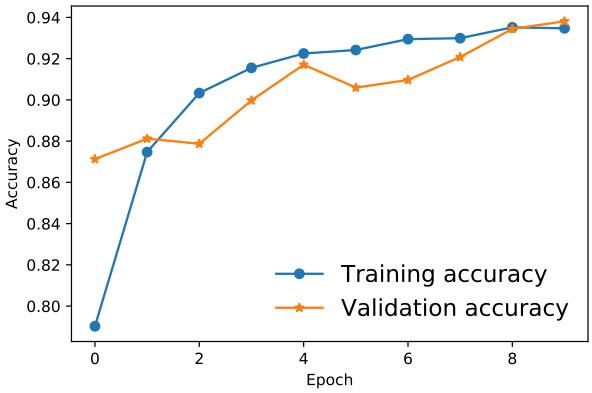
In centralized learning, we have used 90% of data for training and the remaining 10% for the testing purposes. For training purposes, we have used 7272 real and malicious messages, whereas 808 messages were used for the testing. Likewise, to train and test our global model in the federated learning, we have used 6543 messages for training and 900 messages for testing the performance. Table 3 demonstrates that federated learning-based IDS (FIDS) is more robust and efficient in contrast to the centralized learning-based IDS (CIDS). The training and testing accuracy of FIDS is approximately 5% higher than CIDS, which makes the FIDS approach more reliable than CIDS. Interestingly, as presented in figure 5 and figure 6, training and validation loss for FIDS is 0.01, whereas it is 0.11 and 0.12 for CIDS; hereby proving the efficiency of FIDS over CIDS. Similarly, precision, recall, and f1-score for FIDS are much better than CIDS, thus ascertaining the robustness of FIDS.

Table 3 – Classification summary for both Centralized and Federated Learning

Method	**Training Accuracy**	**Testing Accuracy**	**Training Loss**	**Testing Loss**	**Precision**	**Recall**	**F1-Score**
Centralized Learning	95.00	94.18	0.11	0.12	92.51	95.61	94.04
Federated Learning	99.64	99.77	0.01	0.01	99.63	100	99.81

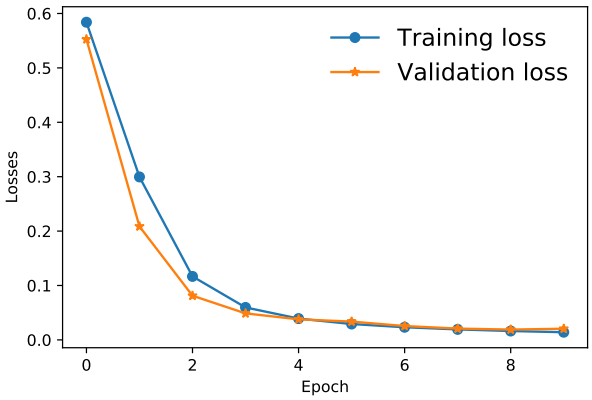


(a) Training and validation loss

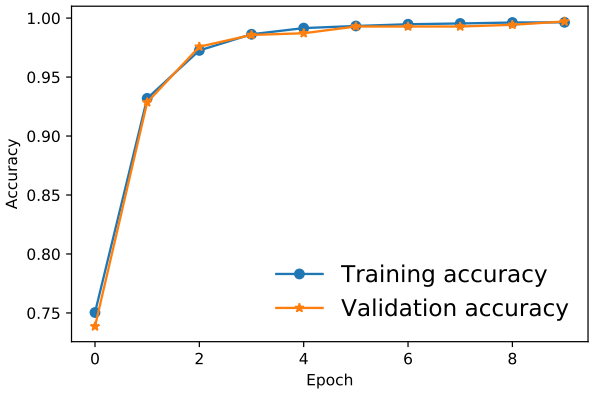


(b) Training and validation accuracy

Figure 5 – Centralized learning loss and accuracy



(a) Training and validation loss



(b) Training and validation accuracy

Figure 6 – Federated learning loss and accuracy

From table 4, we can conclude that CIDS is less trustworthy than FIDS because CIDS has a high misclassification rate of 5.81% in contrast to 0.25% in FIDS. The key reason behind the better performance of FIDS over CIDS is due to the fact that in CIDS, all the clients send their data to the centralized server for computations making it time and compute expensive. Whereas in FIDS, clients prepare their local models and send the parameters to the centralized server; the centralized server averages the parameters and disseminates the global model to all the clients. Averaging and limited data sharing makes the federated learning approach reliable and resource inexpensive.

Table 4 – Confusion matrix for both Centralized and Federated Learning

Method	Label	TP	FP	FN	TN
Centralized Learning	Real\Attack	390	17	30	371
Federated Learning	Real\Attack	358	2	0	540

4.3 Prior and Post-Federated Learning

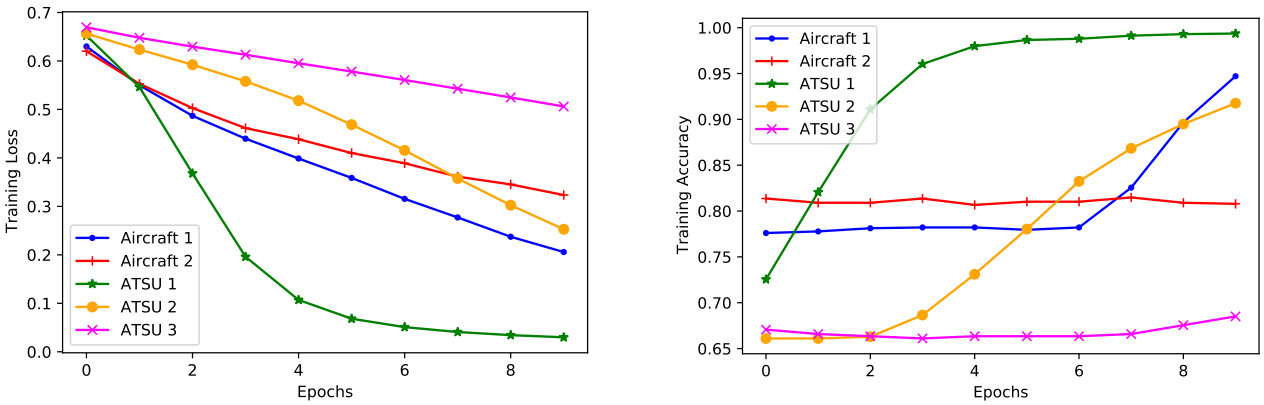
In this sub-section, we discuss the local client IDS performance before and after the federated learning. For aircraft 1, we have used 1177 and 100 CPDLC messages for training and testing, respectively. Similarly, for aircraft 2, we have used 894 and 100 CPDLC messages for training and testing respectively. Figure 7a and figure 7b illustrate the training loss and accuracy of 2 aircraft's and 3 ground stations prior to federated learning. It is evident from the comparison of figure 6 and figure 7 that FIDS have low loss and high accuracy as compared to locally trained IDS (LIDS). The local training accuracy and loss at ATSC 3 was even worst compared to others and FIDS, probably due to the

fewer training data. On the contrary, FIDS performs better even if the training data is less at one client because the global model is prepared after averaging the parameters received from all clients.

Table 5 – Confusion matrix prior and post Federated Learning

Entities	**Prior to federated learning**					**Post federated learning**			
	Label	TP	FP	FN	TN	TP	FP	FN	TN
Aircraft 1	Real\Attack	8	0	2	90	9	0	1	90
Aircraft 2	Real\Attack	0	0	22	90	21	0	1	78
ATSC 1	Real\Attack	65	0	0	35	63	0	2	35
ATSC 2	Real\Attack	54	9	0	34	56	0	1	43
ATSC 3	Real\Attack	73	24	0	3	72	0	1	27

It is apparent from table 5 that the performance of clients (ATSC 1, etc.) has significantly improved after training the local models with the global parameters. As a piece of evidence, the misclassification rate was 4.8% before federated learning, which got reduced to 1.2% post federated learning. From table 6, we can see a substantial increase in the detection accuracy, especially for aircraft 2, ATSC 2, and ATSC 3. Figure 8 depicts that federated learning-based IDS are highly responsive. For instance, ATSC 3 and aircraft 1 consumed 0.1354 and 0.0959 seconds to detect unseen normal and anomaly messages. But after training with the global parameters, the detection time consumed by ATSC 3 and aircraft 1 was reduced to 0.0434 and 0.0489 seconds respectively. All these advantages make FIDS a better alternative to CIDS and LIDS.



(a) Training loss

(b) Training accuracy

Figure 7 – Local models training loss and accuracy

Table 6 – Classification summary prior and post Federated Learning

Entities	**Prior to federated learning (%)**				**Post federated learning (%)**			
	Testing Accuracy	Precision	Recall	F1-Score	Testing Accuracy	Precision	Recall	F1-Score
Aircraft 1	98.00	100	97.82	98.90	99.00	100	98.90	99.44
Aircraft 2	78.00	100	78	87.64	99	100	98.73	99.36
ATSC 1	100.00	100	100	100	98	100	94.59	97.22
ATSC 2	91.00	79	100	88.31	99	100	97.72	98.85
ATSC 3	76	11.11	100	19.99	99	100	96.42	98.18

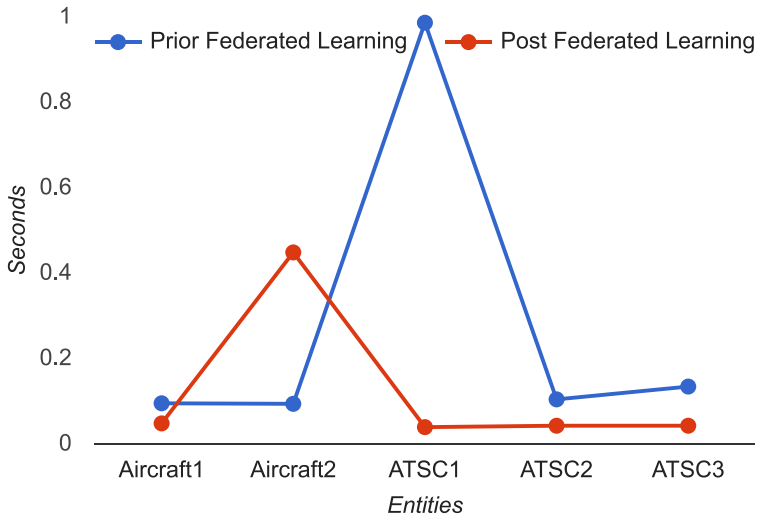


Figure 8 – Detection time prior and post federated learning

5. Conclusion and Future Work

Channel congestion in large airports was usually caused by the excessive usage of analog voice communications. CPDLC has significantly reduced the communication traffic on the VHF band by transforming the analog voice model into a digital text method. However, some serious security concerns were identified in the working of the CPDLC that an attacker could exploit to perform uplink and downlink attacks. We proposed an FL-based IDS to detect the anomalies in the messages received by the airplanes and ATCs. The presented model can be trained with minimal training and loss. The performance investigation revealed that the devised approach achieved better accuracy, precision, recall and F1-score than centralized and local model-driven IDSs. In the future, we plan to enhance the dataset for more practical modeling. We will extend this work using auto-encoders to further assess the reliability of the approach in hostile environments.

References

[1] International Civil Aviation Organization. Global operational data link document (gold). 2013. Available at `https://icao.int/APAC/Documents/edocs/GOLD_2Edition.pdf`. (17.12.2020).

[2] Nils Mäurer, Thomas Gräupl, Christoph Gentsch, Tobias Guggemos, Marcel Tiepelt, Corinna Schmitt, and Gabi Dreo Rodosek. A secure cell-attachment procedure of ldacs. In *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 113–122. IEEE, 2021.

[3] André Lehto, Isak Sestorp, Suleman Khan, and Andrei Gurtov. Controller pilot data link communication security: A practical study. In *2021 Integrated Communications Navigation and Surveillance Conference (ICNS)*, pages 1–11. IEEE, 2021.

[4] Gaurav Dave, Gaurav Choudhary, Vikas Sihag, Ilsun You, and Kim-Kwang Raymond Choo. Cyber security challenges in aviation communication, navigation, and surveillance. *Computers & Security*, 112:102516, 2022.

[5] Doris Di Marco, Alessandro Manzo, Marco Ivaldi, and John Hird. Security testing with controller-pilot data link communications. In *2016 11th International Conference on Availability, Reliability and Security (ARES)*, pages 526–531. IEEE, 2016.

[6] Andrei Gurtov, Tatiana Polishchuk, and Max Wernberg. Controller–pilot data link communication security. *Sensors*, 18(5):1636, 2018.

[7] Joshua Smailes, Daniel Moser, Matthew Smith, Martin Strohmeier, Vincent Lenders, and Ivan Martinovic. You talkin'to me? exploring practical attacks on controller pilot data link communications. In *Proceedings of the 7th ACM on Cyber-Physical System Security Workshop*, pages 53–64, 2021.

[8] Martin Strohmeier, Matthias Schäfer, Rui Pinheiro, Vincent Lenders, and Ivan Martinovic. On perception and reality in wireless air traffic communication security. *IEEE transactions on intelligent transportation systems*, 18(6):1338–1357, 2016.

[9] Ten major cyberattacks against the airport industry. `https://www.stormshield.com/news/ten-major-cyberattacks-against-the-airport-industry/`. Accessed: 2022-05-22.

[10] Max Wernberg. Security and privacy of controller pilot data link communication, 2018.

[11] Martin Strohmeier. *Security in next generation air traffic communication networks*. PhD thesis, University of Oxford, 2016.

[12] Sofie Eskilsson, Hanna Gustafsson, Suleman Khan, and Andrei Gurtov. Demonstrating ADS-B and CPDLC attacks with software-defined radio. In *2020 Integrated Communications Navigation and Surveillance Conference (ICNS)*, pages 1B2–1. IEEE, 2020.

[13] Tom McParland, Vic Patel, and WJ Hughes. Securing air-ground communications. In *20th DASC. 20th Digital Avionics Systems Conference (Cat. No. 01CH37219)*, volume 2, pages 7A7–1. IEEE, 2001.

[14] James H Griner. An elliptic curve based authentication protocol for controller-pilot data link communications. 2005.

[15] Suleman Khan, Andrei Gurtov, An Breaken, and Pardeep Kumar. A security model for controller-pilot data communication link. In *2021 Integrated Communications Navigation and Surveillance Conference (ICNS)*, pages 1–10. IEEE, 2021.

[16] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[17] sklearn.preprocessing.labelencoder. `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder`. Accessed: 2022-05-22.

[18] Theo Ryffel, Andrew Trask, Morten Dahl, Bobby Wagner, Jason Mancuso, Daniel Rueckert, and Jonathan Passerat-Palmbach. A generic framework for privacy preserving deep learning. *arXiv preprint arXiv:1811.04017*, 2018.

6. Acknowledgement

This work was supported by Trafikverket and Luftfartsverket under Automation Program II. This work was also partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP).

7. Contact Author Email Address

Corresponding Author: Prof. Andrei Gurtov (andrei.gurtov@liu.se)

8. Copyright Statement