A federated learning framework for enhanced data security and cyber intrusion detection in distributed network of underwater drones

Singh Popli, Mansahaj; Singh, Rudra Pratap; Kaur Popli, Navneet; Mamun, Mohammad

Publisher's version / Version de l'éditeur:
https://doi.org/10.1109/ACCESS.2025.3530499
IEEE Access, 13, pp. 12634-12646, 2025-01-16

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 7 January 2025, accepted 14 January 2025, date of publication 16 January 2025, date of current version 23 January 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3530499

RESEARCH ARTICLE

A Federated Learning Framework for Enhanced Data Security and Cyber Intrusion Detection in Distributed Network of Underwater Drones

MANSAHAJ SINGH POPLI[1], (Student Member, IEEE), RUDRA PRATAP SINGH[1], (Student Member, IEEE), NAVNEET KAUR POPLI[1], (Senior Member, IEEE), AND MOHAMMAD MAMUN[2], (Senior Member, IEEE)

[1]Faculty of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8P 5C2, Canada
[2]National Research Council Canada, Fredericton, NB E3B 9W4, Canada

Corresponding author: Rudra Pratap Singh (rudrapratap s0110@gmail.com)

This work was supported by the National Research Council Canada (NRC).

ABSTRACT Underwater drones are vital for scientific research, environmental monitoring, and maritime operations, allowing data collection in challenging environments. However, their deployment faces issues such as low bandwidth, high latency, signal attenuation, and intermittent connectivity due to mobility and water currents. Traditional centralized data processing approaches are inefficient under these conditions as they require transmitting large volumes of raw data to a central location. To address these challenges, this study proposes a Federated Learning (FL) framework specifically tailored for underwater networks. Unlike centralized approaches, FL enables underwater drones to collaboratively train a global intrusion detection model by processing data locally and sharing only model updates with the central server. This approach significantly improves data security by ensuring that sensitive information never leaves the local devices, reducing the risk of interception or compromise during transmission. Furthermore, FL's decentralized architectures inherently aligns with the dynamic and distributed nature of underwater drone networks. The proposed framework improves cyber intrusion detection by leveraging localized insights from individual drones to detect threats, including zero-day attacks, without directly exposing sensitive data. By preserving privacy and enabling collaborative anomaly detection, FL addresses key cybersecurity challenges in the Internet of Underwater Things (IoUT).

INDEX TERMS Internet of Underwater Things (IoUT), Underwater Drones, Federated Learning Model, Collaborative Intrusion Detection, DDoS attack, Cyber-physical System (CPS).

I. INTRODUCTION

Underwater drones are increasingly being deployed in underwater wireless sensor networks (UWSN) for various applications, including environmental monitoring, scientific exploration, and maritime operations [1]. These networks rely on efficient decision-making and tracking capabilities, facilitated by the extensive communication infrastructure of UWSNs. However, underwater data transmission poses significant challenges, such as transmission loss, high latency, limited bandwidth, and short network lifetime [2], [3].

Coupled with these inherent communication challenges, UWSNs are highly susceptible to various cybersecurity threats, including man-in-the-middle (MITM) and Distributed denial of service (DDoS) attacks [4]. Further, the data held or sensed on these devices is private in nature. Sharing it over the networks also makes it vulnerable to a variety of attacks causing critical consequences. Eventually, attackers may take control of connected devices connected to the Internet of Things.

Recent advancements in cyber-physical systems (CPS) have introduced control strategies aimed at addressing

The associate editor coordinating the review of this manuscript and approving it for publication was Mouquan Shen.

systemic challenges in distributed and resource-constrained environments. For instance, mismatched quantized output-feedback control addresses quantization-induced errors and mismatched uncertainties, which are critical in scenarios where precise data measurements are challenging, such as underwater networks. This control strategy enhances robustness and ensures stability under quantization constraints [5]. Similarly, guaranteed cost event-triggered control provides an optimal mechanism to reduce communication overhead while maintaining performance within a cost boundary. This approach has been effectively applied to dynamic systems like wind turbines and demonstrates potential for resource optimization in decentralized and distributed networks [6]. These strategies underscore the importance of precision and efficiency in CPS and highlight the growing need for decentralized frameworks that operate reliably under challenging conditions, such as those encountered in UWSNs

In centralized intrusion detection systems (IDS), data from all devices in the network are transmitted to a main server, where ML algorithms are employed for intrusion detection. However, even after the successful transfer of data to the central server, performing ML algorithms on such a massive and centralized dataset can be computationally expensive and resource-intensive, as traditional centralized ML approaches often require significant computational power, memory, and storage resources to process and analyze the aggregated data effectively [7].

In underwater drone networks, drones can be widely spread across vast water bodies, resulting in significant distances between individual drones and the central server. Transmitting the raw data generated by these drones to a centralized location for intrusion detection can be highly inefficient and resource-intensive, especially considering the limited bandwidth and high latency of underwater communication channels. Moreover, even after successful intrusion detection using centralized ML techniques, identifying the specific drone or location affected by an attack or anomaly becomes a daunting task due to the centralized nature of the data processing. This lack of granularity and context awareness can hinder timely and targeted response efforts, potentially compromising the security and operational effectiveness of the underwater drone network.

To address these challenges, we will explore FL in underwater drone networks for cyber intrusion detection. FL was first proposed in 2017 by Google researchers to solve the problem of updating models locally by distributed Android devices, with the design goal of carrying out an efficient ML process among multiple clients while safeguarding information security and privacy [8].

FL involves training ML models on local data across multiple devices or nodes. Each node trains the model on its local dataset and shares only the model updates (e.g., gradients or parameters) with a central server. The central server then aggregates these updates to form a global model. This process ensures that the raw data never leaves the local devices, enhancing privacy and security approach eliminates the need for data sharing, thus preserving data privacy while enhancing the detection capabilities against cybersecurity threats.

While solving these problems, FL also improves on the existing intrusion detection techniques:

- **Enhanced Data Security:** Underwater drones generate a substantial amount of data, including navigation and operational information. Using FL, each drone can locally train an intrusion detection model on its data, thereby reducing the need to transmit sensitive data over potentially insecure networks. This localized data processing enhances the overall security of the system [9].
- **Distributed Intrusion Detection:** Implementing FL for intrusion detection across a network of underwater drones allow each drone to identify anomalies and potential security breaches based on its localized data. The aggregated global model, which incorporates insights from all drones, can detect a wider range of threats and improve the overall accuracy of the intrusion detection system, [10] and give results in a shorter time period.
- **Efficiency and Adaptability:** FL is particularly suitable for applications requiring real-time adaptability and learning from diverse, distributed datasets [11] making it suitable for detecting different types of attacks.
- **Real-Time Anomaly Detection:** FL enables continuous learning and updating of models, which is crucial for maintaining effective security measures against evolving threats. This capability is particularly important for underwater drones, which operate in dynamic and often hostile environments where timely detection and response to security threats are essential [12].

II. LITERATURE REVIEW

The security of CPS has been extensively studied, with recent advancements emphasizing AI-driven and decentralized approaches to enhance resilience against cyber threats. For instance, Fatorachian and Kazemi proposed an AI-enhanced fault-tolerant control framework for transportation and logistics systems, addressing both physical malfunctions and cyber threats through advanced ML algorithms and real-time data analytics [13]. This approach is particularly relevant to underwater drone networks, where resource constraints and dynamic environments necessitate robust fault detection and recovery mechanisms.

Similarly, Yan et al. developed a fusion-based event-triggered H infinity state estimation method for networked autonomous surface vehicles, effectively mitigating the impact of measurement outliers and cyber-attacks [14]. Their methodology leverages sensor data fusion and event-triggered communication to maintain accurate state estimation, even under adversarial conditions. Such

approaches align with the challenges in the Internet of Underwater Things (IoUT), where intermittent connectivity and limited bandwidth increases the risks of undetected anomalies.

While these studies provide valuable insights into CPS security, their centralized data processing often raises privacy concerns and inefficiencies in underwater networks. In contrast, FL offers a decentralized approach, enabling devices to collaboratively train models without sharing raw data.

Researchers have explored various applications of FL in developing IDS for IoT networks. One notable study by Sharma et al. proposed an FL-based IDS specifically designed for smart home environments. Their approach leveraged the decentralized nature of FL to allow smart devices within a home, such as security cameras, smart thermostats, and lights, to collaboratively train an IDS model. By keeping the data local to each device, the system enhanced privacy while still benefiting from the collective intelligence of the entire network. This approach demonstrated the potential of FL to detect and mitigate various types of cyber threats in smart homes without compromising user privacy [15].

In another study, Nguyen et al. applied FL to an industrial IoT setting, where the system needed to detect anomalies in the operational data collected from industrial sensors. The researchers found that FL not only preserved the privacy of sensitive industrial data but also improved the system's ability to detect a wide range of attacks, including those that were previously unknown. By training on locally generated data, the IDS could adapt to the specific operational characteristics of different industrial environments, leading to a more robust and accurate detection system [16].

A different approach was presented by Bagdasaryan et al., who focused on using FL for IDS in vehicular ad hoc networks (VANETs). In their research, they utilized FL to train IDS models across multiple vehicles, each acting as a client in the federated system. This allowed the system to detect and respond to cyber threats in real-time while vehicles were on the move. The study highlighted the importance of timely and accurate detection in VANETs, where delayed responses could lead to severe consequences. FL was shown to be effective in maintaining high detection accuracy while minimizing the communication overhead typically associated with centralized IDS [17].

While FL provides significant benefits for IoT-based IDS, it also introduces new challenges. One of the primary issues is the heterogeneity of data across different IoT devices. IoT devices often generate data that is highly diverse and specific to their particular use cases, leading to non-IID (non-independent and identically distributed) data distributions. This can affect the performance of the global IDS model, as traditional aggregation techniques like FedAvg may not be well-suited for handling such diversity [18].

To address these challenges, several advanced techniques have been proposed. For example, Yang et al. introduced a personalized FL approach, where each IoT device trains a model that is partially personalized to its own data while still contributing to the global model. This technique improves the ability of the IDS to generalize across different devices while maintaining high accuracy for specific types of data [18].

Another significant advancement is the integration of FL learning, as explored by Liu et al. This approach allows models trained in one domain or on one set of devices to be adapted to another domain or set of devices, thus enhancing the flexibility and applicability of FL-based IDS. FL learning is particularly useful in scenarios where new devices are frequently added to the network, or where the nature of the data changes over time [19].

The aggregation of model updates in FL is a pivotal process that determines the overall performance of the global model. The FedAvg algorithm, which computes a weighted average of the updates from participating devices, is the most widely used aggregation technique. However, in environments where the data is non-IID or devices have varying amounts of data, FedAvg may not always be the optimal choice.

To address the shortcomings of FedAvg, several alternative aggregation methods have been developed:

- **FedProx:** This technique modifies the local objective function by adding a proximal term, which helps stabilize the training process in non-IID settings. By doing so, FedProx ensures that the local models do not diverge too far from the global model, improving the robustness of the training process [20].
- **FedNova:** FedNova addresses the problem of uneven contributions by normalizing the local updates before they are aggregated. This normalization helps balance the influence of devices with different amounts of data, leading to a more equitable and effective aggregation process [21].
- **Federated Matched Averaging (FedMA):** FedMA takes a unique approach by aligning and averaging the layers of local models rather than directly averaging the model parameters. This method is particularly effective in handling non-IID data, as it allows the global model to better reflect the diverse data distributions present across the devices [22].
- **q-FedAvg:** This algorithm enhances FedAvg by weighting the contributions of each device based on the quality of their updates, as determined by the loss function. Devices that achieve lower losses are given more influence in the aggregation process, which helps improve the convergence speed and accuracy of the global model [23].

These advanced aggregation methods enhance the robustness and flexibility of FL, making it a viable approach for a wide range of applications, including Intrusion Detection Systems in IoT networks. By selecting the appropriate aggregation method, FL can be tailored to address the specific challenges

of different environments, such as the diversity of data and the computational constraints of IoT devices [24].

III. METHODOLOGIES

A. FEDERATED LEARNING PROCESS

The process can be generalized by three major steps as follows.

- **Selection of model:** A central ML model, called global model, is pretrained with initial parameters, and then, is shared with all the clients in the entire FL environment.
- **Training locally:** The global model, which is shared along with all parameters to the clients, is trained locally at the client side with their individual data.
- **Aggregating the local model:** After training locally in the client environment, the updated parameters are forwarded to the central server. Using the updated parameters, the global model is updated. This updated global model is then shared with the clients to start a new iteration.

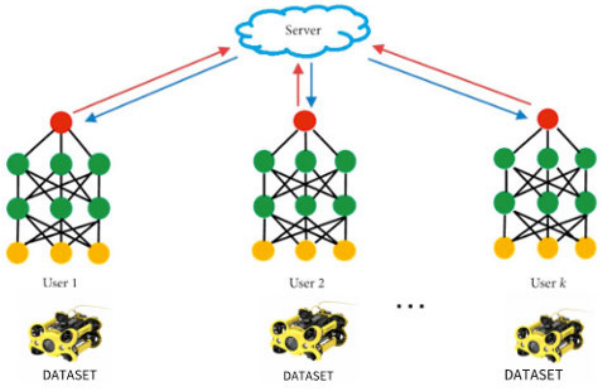

FIGURE 1. Workflow in FL.

Although a lot of research has been done where fl is used for IDS in IOT systems and Aerial drones, there has not been any research for creating a FL framework for a network of underwater sensor suites. In this paper, we will first use the CIC IDS 2017 dataset to perform fl and get results to gain a better understanding of the fl process. Then try to replicate those results on the SOLIDS lab underwater dataset. We will explore different scenarios for the IDS and try to prove that fl works just as good as centralized ml if not better.

B. DATASETS USED

1) CIC IDS 2017

In this research, we utilize the CIC IDS 2017 dataset, a widely recognized benchmark dataset for intrusion detection systems (IDS) that was developed by the Canadian Institute for Cybersecurity (CIC) at the University of New Brunswick. This dataset is designed to replicate real-world network traffic, encompassing both benign and malicious activities. The dataset includes a wide variety of modern cyberattacks, making it a valuable resource for training ML models to detect intrusions.

The CIC IDS 2017 dataset comprises seven days of network traffic data, during which various attack scenarios were executed on a simulated network environment. The attack types include Brute Force Attacks, Denial of Service (DoS) and Distributed Denial of Service (DDoS) Attacks, Heartbleed, Botnet Activity, Web Attacks such as SQL injection and cross-site scripting, Port Scanning, and Infiltration of the Network from an Inside Attacker. This wide range of attack types allow for comprehensive training and testing of intrusion detection systems, ensuring that models can generalize across different types of cyber threats [25].

Additionally, the dataset contains more than 80 flow-based network traffic features, extracted using CICFlowMeter, covering various attributes such as packet counts, byte counts, TCP flags, and timestamps. These features allow the dataset to represent real-world network behavior, both in terms of benign traffic and malicious intrusions. The CIC IDS 2017 dataset is particularly useful for evaluating ML models in intrusion detection due to its comprehensiveness and diversity of attack scenarios [26].

2) SOLIDS DATASET

The SOLIDS Lab Underwater Dataset was developed in our lab to address the specific challenges of detecting cyber threats in underwater networks, particularly those involving underwater drones and IoUT devices. This dataset serves as a crucial asset in studying and enhancing cybersecurity measures for underwater communication networks, which are inherently different from terrestrial or traditional wireless networks [27].

Key Properties of the SOLIDS Lab Underwater Dataset:

- **Realistic Underwater Network Conditions:** The SOLIDS dataset captures network traffic characteristics under realistic underwater conditions, simulating challenges such as limited bandwidth, higher latency, and increased signal attenuation. This focus on underwater-specific scenarios makes it uniquely positioned for developing intrusion detection systems (IDS) that target underwater sensor suites and networks.
- **Diverse Attack Scenarios:** The dataset includes comprehensive records of both benign network traffic and simulated Distributed Denial of Service (DDoS) attacks, which are among the most common threats to underwater networks. It comprises multiple variations of DDoS attacks, capturing different behaviors and patterns in network traffic during these malicious activities. This diversity is crucial for training ML models to recognize both known and previously unseen attacks, increasing the robustness of the IDS.
- **Rich Set of Features:** The dataset consists of various network traffic attributes, including packet-level and flow-level features such as Total Forward Packets, Flow Bytes/s, Flow IAT Mean, and Active Mean.

These features are particularly relevant for detecting anomalies in underwater network communication, as sudden changes in these attributes often correlate with cyberattacks like DDoS. The carefully selected features provide high-quality input data for training machine learning models in detecting anomalies and intrusions.

Since the SOLIDS Lab Underwater Dataset was developed in-house, it is tailored to match the specific requirements and objectives of this research. The dataset reflects real-world underwater network conditions and includes comprehensive attack scenarios relevant to intrusion detection in underwater sensor networks. It is specifically designed to be used with ML techniques that target underwater networks' unique challenges, making it a perfect fit for the FL framework implemented in this research.

C. DATA PREPROCESSING

1) PREPROCESSING OF THE SOLIDS LAB UNDERWATER DATASET

The SOLIDS Lab Underwater Dataset, which simulates network behavior in underwater environments, contains extensive network traffic data that were captured using Packet Capture (PCAP) files. The preprocessing steps for this dataset included:

- **Data Extraction and Conversion:** Raw PCAP files were converted into CSV format using the CICFlowMeter tool. This conversion extracted over 80 attributes related to network flow, including features such as Total Forward Packets, Flow Bytes per second (Flow Bytes/s), and Flow Inter-Arrival Time Mean (Flow IAT Mean). These features were chosen based on their relevance in identifying Distributed Denial of Service (DDoS) attacks within an underwater network context.
- **Feature Selection:** The dataset was manually curated to focus on specific features that indicated attack patterns, such as sudden changes in packet rates or inter-arrival times. This careful selection of features aimed to reduce computational overhead while maximizing the relevance of the selected attributes for intrusion detection.
- **Labeling and Cleaning:** Attack scenarios were clearly labeled as either benign or malicious (DDoS) based on the experimental setup. Instances with missing or inconsistent values were removed, ensuring a clean and standardized dataset for training ML models.

2) PREPROCESSING OF THE CIC-IDS-2017 DATASET

The CIC-IDS-2017 dataset, published by the University of New Brunswick, is widely used in intrusion detection research due to its realistic representation of network traffic. This dataset includes several days of network data with various attack scenarios, including DDoS, Brute Force, Botnet, and SQL Injection attacks. The preprocessing of this dataset involved:

- **Cleaning and Removing Duplicates:** Instances with missing or duplicated data entries were removed to ensure data quality. Non-relevant attributes such as source and destination IP addresses were discarded, focusing instead on traffic flow features and statistics.
- **Feature Engineering and Transformation:** Similar to the SOLIDS Lab dataset, CIC-IDS-2017 was converted into a CSV format with over 80 network traffic attributes using CICFlowMeter. Key features such as Total Fwd Packets, Flow Bytes/s, and Flow IAT Mean were retained, given their importance in detecting DDoS patterns.
- **Balancing and Normalization:** The dataset underwent normalization to scale feature values, which was crucial for training an artificial neural network (ANN) model. The normalization process ensured that no single attribute disproportionately influenced the model training.

D. ML STRATEGY USED

We have utilized an ANN as the model that runs on individual drones for the task of intrusion detection within a FL framework. The ANN model was constructed using TensorFlow's Keras library, structured as a Sequential model. This architecture allows for a simple and effective stacking of layers that transform the input data into a binary classification output. The specific details of the model are as follows:

- **Input Layer:** The first layer in the ANN model is a dense (fully connected) layer consisting of 128 neurons. This layer employs the ReLU (Rectified Linear Unit) activation function, which introduces non-linearity to the model and allows it to capture complex relationships in the input data. The input shape is defined by the number of features in the dataset, which in the case of intrusion detection consists of numerous flow-based network traffic features such as packet counts, flags, and byte counts.
- **Hidden Layer:** The second layer contains 64 neurons and also uses the ReLU activation function. This hidden layer serves as a transformational layer that reduces the dimensionality of the feature space while maintaining critical patterns that distinguish benign network traffic from malicious activities.
- **Output Layer:** The final layer is a single neuron with a Sigmoid activation function. The Sigmoid function is appropriate for binary classification tasks as it outputs a probability score between 0 and 1. This output can be interpreted as the likelihood of a network flow being malicious or benign.

The model is compiled using the Adam optimizer, which is widely recognized for its efficiency in adjusting learning rates during training. The loss function used is binary cross-entropy, ideal for binary classification problems where

the goal is to minimize the divergence between predicted probabilities and true binary labels.

The selection of an ANN for this FL framework is driven by several key factors, including its ability to process high-dimensional data, model non-linear relationships, and its compatibility with the decentralized nature of FL. Intrusion detection systems (IDS) typically analyze large volumes of network traffic data, which consist of numerous features, such as packet counts, flags, and byte counts. ANN models are well-suited to this task because of their ability to learn from high-dimensional feature spaces. Buczak and Guven note that ANNs have been widely used in cybersecurity applications, particularly for identifying complex patterns in network traffic [28].

E. FL FRAMEWORK USED

For our experimentation, we have used the Flower Federated Learning Python library [29] One of the primary reasons for choosing Flower is its ease of use. Flower provides a clear and intuitive API for defining the behavior of both clients and the server in a FL setup, making it accessible for researchers and practitioners alike.

While other frameworks such as TensorFlow Federated and PySyft are powerful, they are often more suited for environments where the specific library (TensorFlow or PyTorch) is the core of the solution. These frameworks can be more complex to set up and have a steeper learning curve, making them less ideal for scenarios where ease of experimentation and rapid prototyping are needed [30]. Flower, on the other hand, is designed to work across multiple libraries and simplifies many aspects of FL, such as client-server communication, strategy customization, and experiment management. For research settings where quick iterations and scalability are critical—such as developing intrusion detection systems across distributed environments—Flower provides a balance of simplicity and flexibility.

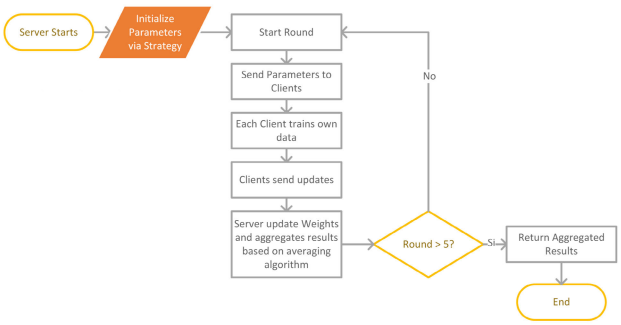

FIGURE 2. Flower framework.

IV. RESULT

A. BINARY CLASSIFICATION

In this experiment, we utilize the CIC IDS 2017 dataset to conduct a comparative analysis between traditional centralized ML algorithms and federated learning. We approach this as a binary classification task, labeling benign rows as 0 and attack rows as 1. The chosen model is the previously explained ANN model, trained with 3 epochs and a batch size of 32. To evaluate the model's behavior, we examine both loss and accuracy metrics. In machine learning, loss measures the error between the predicted and actual values, while accuracy indicates the percentage of correct predictions made by the model.

As shown in the first graph (Fig. 3), the training loss decreases sharply to nearly zero after just one epoch, indicating that the model quickly learns from the training data. Meanwhile, the validation loss remains low and stable, which suggests that the model consistently performs well on unseen validation data. In the second graph (Fig. 4), the model's accuracy improves rapidly, surpassing 97% for both training and validation. This stable, high validation accuracy indicates robust model performance on unseen data. The final accuracy for the centralized ML approach is 98.09%.

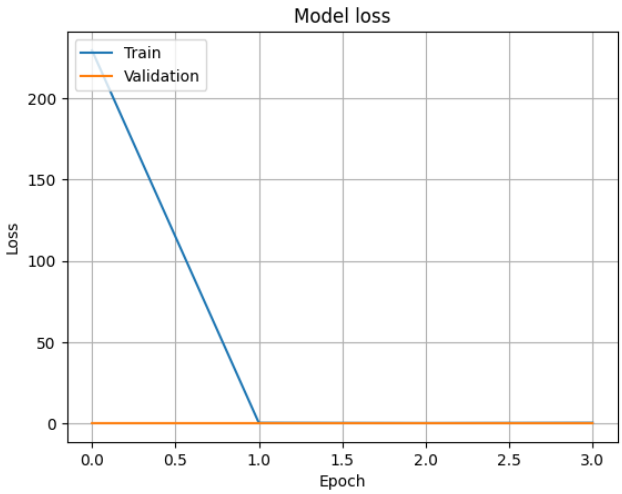

FIGURE 3. Loss for ML.

Next, we conduct the same experiment using a federated learning approach. In this setup, the dataset is separated into three clients based on attack type, creating a non-IID data distribution. Each client runs the FL algorithm using the same ANN model, with 1 epoch and a batch size of 32. The FL workflow is illustrated in Figure 5. The server-side strategy employed is FedProx due to its effectiveness in handling non-IID datasets and managing data heterogeneity [20].

In Fig. 6, the accuracy trends for the three clients and the global model (server) are depicted over five iterations. The client accuracies vary, reflecting the non-IID nature of their local datasets. For example, Client 1 achieves a peak accuracy of approximately 99.4% by the second iteration but then experiences a slight decline. In contrast, Client 3 consistently shows lower accuracy relative to the other clients, likely

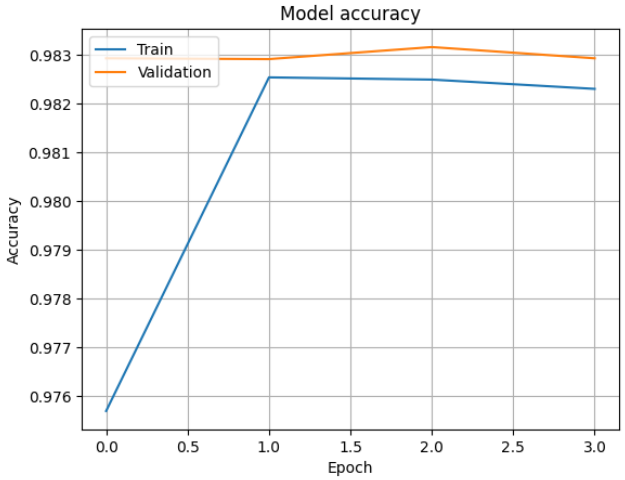

FIGURE 4. Accuracy for ML.

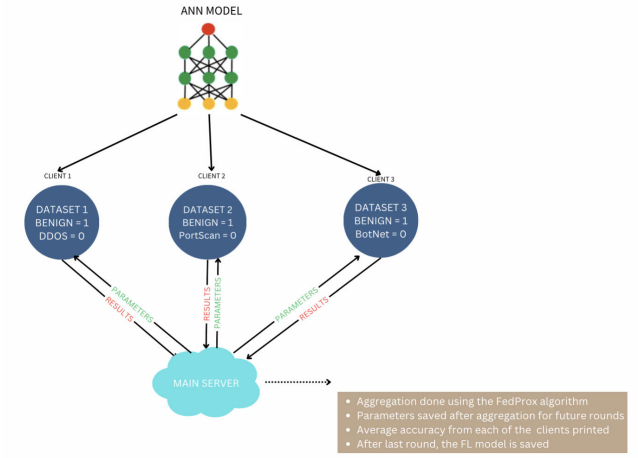

FIGURE 5. FL in binary classification.

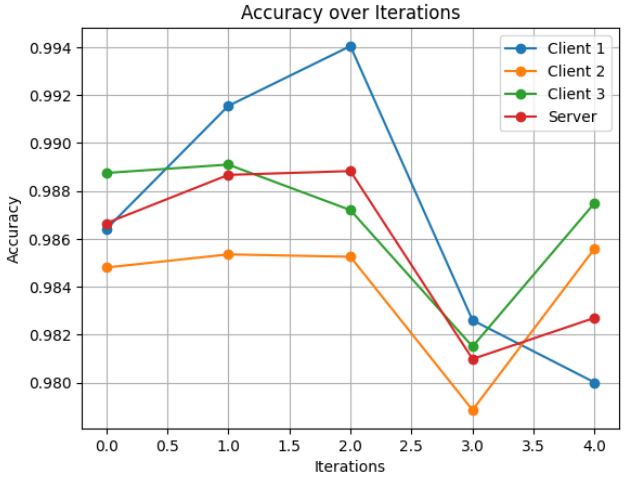

FIGURE 6. FL binary classification accuracy.

due to differences in local data characteristics or sample sizes. Despite these variations, the global model's accuracy stabilizes between 98.6% and 98.8% by the fourth iteration,

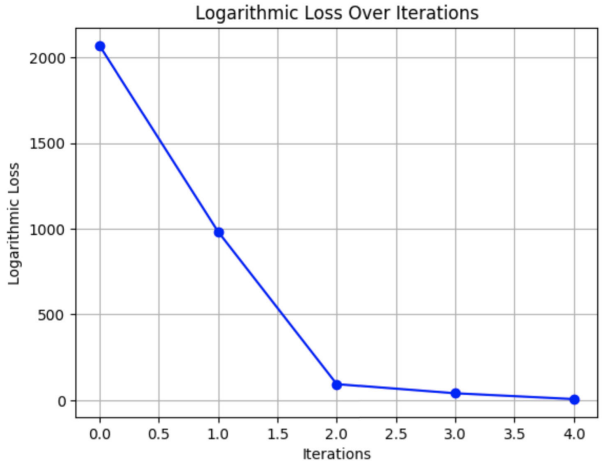

FIGURE 7. FL binary classification loss.

demonstrating the robustness of the federated aggregation mechanism in synthesizing diverse client updates into a reliable global model.

Fig. 7 presents the logarithmic loss of the global model across these iterations. The initial loss value, around 2000, decreases sharply to below 500 within the first two iterations, indicating that the global model quickly adapts based on the client contributions. Following this sharp decline, the loss stabilizes at approximately 200, suggesting that the model has reached a state of convergence, where additional iterations result in diminishing improvements in performance.

After training the model, it is evaluated using the centralized dataset to provide a direct comparison between centralized ML and FL. The centralized ML approach achieves an accuracy of 98.01%, while the FL approach achieves a slightly higher accuracy of 98.26%. This comparison illustrates that, under the same conditions, model architecture, and hyperparameters, federated learning performs comparably or even slightly better than traditional centralized approaches. This demonstrates the feasibility and effectiveness of FL in scenarios where data privacy and distribution are critical considerations.

To demonstrate the effectiveness of our proposed framework, we performed a comparative analysis against several existing methodologies commonly employed for intrusion detection systems (IDS) in distributed networks. These include FedAdam, FedAvg, FedAvg with Autoencoder, and centralized ML.

FedAdam, an optimization strategy for FL, builds upon the Adam optimizer by integrating adaptive learning rates for server-side updates. Unlike FedAvg, which uses a simple weighted average of client updates, FedAdam takes into account the first- and second-order moments of the gradients, enabling more precise model updates, particularly in non-IID data scenarios. While this approach often leads to improved accuracy, its computational cost is significantly higher, making it less practical in

resource-constrained environments. FedAdam has been extensively explored in the work of Reddi et al., where its advantages in heterogeneous federated settings are highlighted [31].

FedAvg, in contrast, is a simpler aggregation method that computes a weighted average of model updates received from clients. Although computationally efficient, its performance degrades when client data distributions are non-IID, as noted in McMahan et al. [8]. To enhance its effectiveness, some researchers have incorporated advanced ML models, such as autoencoders, to extract meaningful features from client data. For example, Pope et al. demonstrated how FedAvg combined with autoencoders could improve anomaly detection in IoT networks [32].

The centralized ML approach, which aggregates all data on a central server, serves as a baseline in this analysis. While achieving high accuracy, it fails to address privacy concerns and scalability issues inherent to distributed systems.

TABLE 1. Binary classification accuracy of different methods.

Method	Accuracy (%)
Proposed Framework	98.26
FedAvg	97.32
FedAdam	98.75
FedAvg with Autoencoder	97.96
Centralized ML	98.09

The proposed framework achieves competitive accuracy (98.26%), outperforming FedAvg and FedAvg with Autoencoder while being close to the performance of FedAdam and centralized ML. However, the key advantage of the proposed framework lies in its efficiency and resource optimization. While FedAdam achieves slightly higher accuracy (98.75%), it comes at the cost of significant computational overhead and memory usage. As shown in Figure 8, the average memory consumption of FedAdam is 1.71 GB, and it requires 200 seconds for execution. In contrast, the proposed framework consumes only 1.69 GB of memory and completes execution in 180 seconds. This reduction in resource utilization makes it a more viable solution for deployment in real-world applications, particularly in resource-constrained environments such as underwater sensor
networks.

The balance between accuracy and efficiency demonstrated by the proposed framework addresses a critical challenge in FL systems: achieving high model performance without imposing excessive demands on computational and communication resources. This is particularly relevant for dynamic and distributed networks, where latency, bandwidth, and energy constraints significantly impact the feasibility of ML solutions.

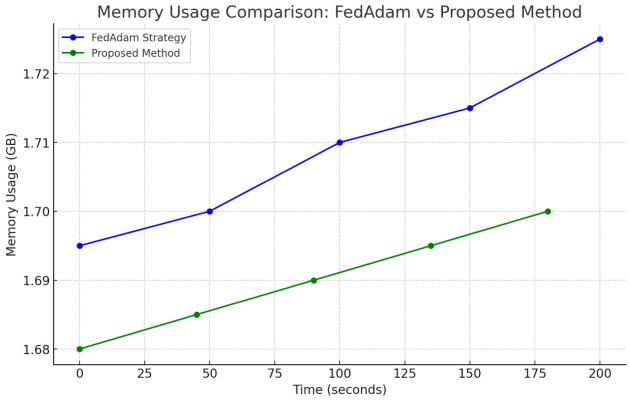

FIGURE 8. Memory comparison between proposed and FedAdam.

B. ZERO DAY ATTACK

Next, we will test to see if our FL approach is going to work for detecting zero-day attacks. A zero-day attack exploits a previously unknown vulnerability in software or hardware, which the developers have had no opportunity to identify or patch. This makes zero-day attacks particularly dangerous, as traditional security defenses are ineffective against threats that target undiscovered weaknesses. Attackers can leverage zero-day vulnerabilities to gain unauthorized access, steal sensitive data, or disrupt services. Since the vulnerability is unknown, there are no existing patches or signature-based defenses, requiring organizations to rely on anomaly detection and proactive monitoring to identify unusual patterns of behavior [33].

To evaluate whether the FL framework can effectively detect zero-day attacks, we set up an experiment with three clients, leveraging the SOLIDS Lab Underwater Dataset to simulate real-world conditions in an underwater drone network. In this scenario, each client dataset consists solely of benign traffic rows, with no exposure to any known attack patterns during training. This setup mimics real-world conditions where the system has not yet encountered the specific type of attack being introduced. The goal is to determine whether the federated model can generalize well enough to identify previously unseen threats.

After training the model using these benign-only datasets on each client, we then evaluate its performance on a test dataset containing both benign and attack rows. By introducing attack traffic only during the testing phase, we aim to simulate a zero-day attack situation. If the model achieves high accuracy on this mixed test dataset, it indicates that the FL framework has successfully learned underlying patterns of normal behavior and can flag deviations as potential threats, even if those deviations represent previously unknown attack types.

We employ the same ANN model and training process as in previous experiments to maintain consistency in the evaluation. The FL framework uses each client's updates

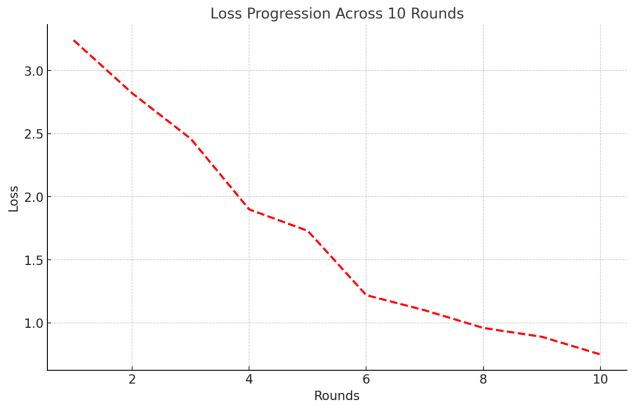

FIGURE 9. Server loss for zero day attack.

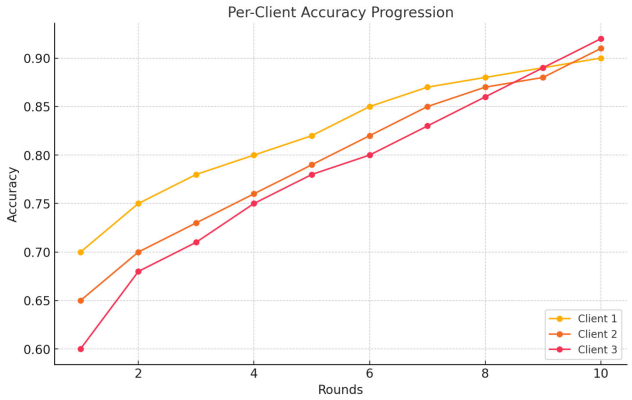

FIGURE 10. Client accuracy for zero day attack.

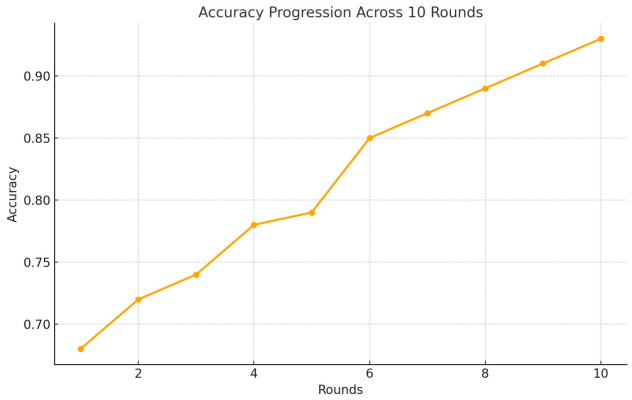

FIGURE 11. Server accuracy in FL classification for zero day attack.

to build a global model capable of detecting anomalies and potential zero-day threats. A high detection accuracy would demonstrate the framework's effectiveness in identifying zero-day attacks, reinforcing the robustness of FL in real-world intrusion detection scenarios.

These graphs illustrate the training results for the federated learning framework when tested for zero-day attacks. The first graph, Fig. 8, displays the server loss progression over 10 rounds. The loss metric indicates how well the model is minimizing prediction errors during training. As seen in the graph, the server loss decreases steadily from around 3.0 to approximately 1.0 over the course of 10 rounds, indicating that the global model is continuously improving and converging as it aggregates updates from the clients. This steady decline in loss suggests that the server is effectively synthesizing the local models' updates to enhance its understanding of normal traffic behavior.

The second graph, Fig. 9, illustrates the accuracy progression for each client over the same 10 rounds. Each client (Client 1, Client 2, and Client 3) starts with an initial accuracy of around 0.60 to 0.70 and progressively improves as training continues. By the end of 10 rounds, all clients achieve accuracies between 0.85 and 0.90, indicating that each local model is successfully learning to identify patterns of benign traffic. This improvement in accuracy across all clients highlights the model's capability to generalize well, even when trained on benign-only datasets. Figure 10 shows the server accuracy progression over the rounds, which is an aggregation of the three clients.

Overall, the training results demonstrate that the federated learning framework effectively reduces server loss and improves client accuracy, leading to a final global accuracy of 93.53%. This high accuracy on the mixed test dataset indicates that the federated model has successfully learned the baseline behavior of benign traffic and can identify anomalous patterns indicative of zero-day attacks. These findings reinforce the effectiveness of the federated learning approach in detecting unknown threats without prior exposure to attack data.

In the context of the Zero Day attack, we applied the same comparative methodology to evaluate the performance of our proposed framework against existing approaches, as presented in Table 2. The accuracy trends observed are consistent with those seen in the binary classification scenario. FedAdam achieves the highest accuracy (94.36%), followed by the proposed framework (93.53%) and FedAvg with Autoencoder (93.12%). FedAvg and Centralized ML trail behind with accuracies of 91.42% and 92.82%, respectively.

TABLE 2. Zero day accuracy of different methods.

Method	Accuracy (%)
Proposed Framework	93.53
FedAvg	91.42
FedAdam	94.36
FedAvg with Autoencoder	93.12
Centralized ML	92.82

Despite the slightly smaller dataset used in this scenario, the relative performance of the methods remains consistent. This demonstrates that the proposed framework maintains its competitive accuracy while being resource-efficient. Figure 12 highlights the memory usage comparison between

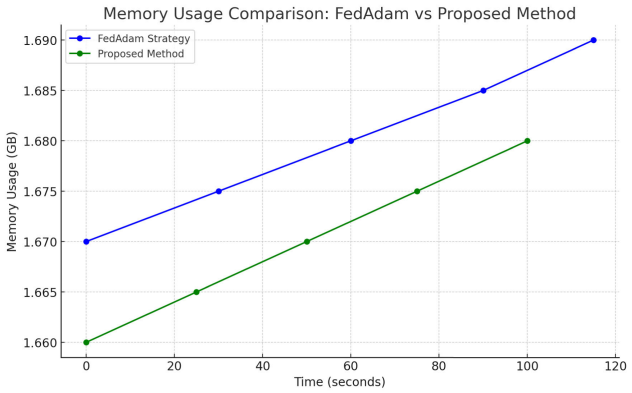

FIGURE 12. Memory comparison for zero day attack scenario.

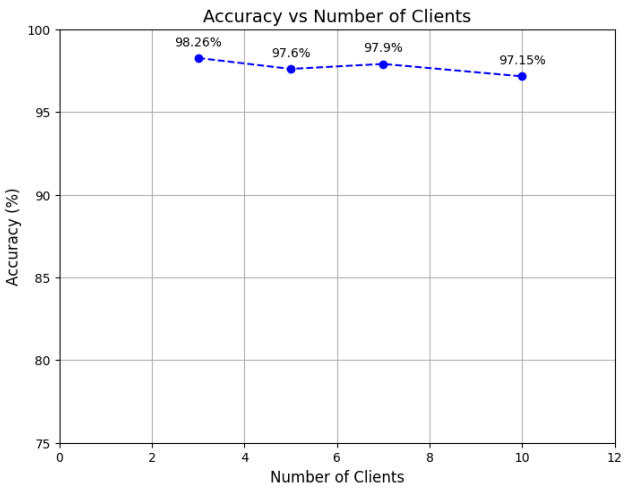

FIGURE 13. Number of clients vs accuracy in FL.

FedAdam and the proposed method. Similar to the binary classification experiment, FedAdam requires more memory and computational time due to its reliance on second-order gradient information for optimizing global updates. However, because of the smaller dataset size in this scenario, the absolute time and memory usage are lower compared to the previous experiment. For instance, FedAdam consumes 1.69 GB of memory and takes approximately 120 seconds, while the proposed method consumes only 1.665 GB and completes execution in 100 seconds.

This consistent pattern underscores the scalability and adaptability of the proposed framework. The reduced resource demands make it particularly suitable for real-world deployment in scenarios where system resources are constrained, such as IoT and underwater sensor networks. While FedAdam continues to excel in terms of accuracy due to its sophisticated optimization strategy, its higher computational overhead highlights the trade-offs that practitioners must consider in resource-sensitive environments.

C. MULTIPLE CLIENTS

To test the scalability of the FL framework, we conducted experiments by partitioning the SOLIDS Lab Underwater Dataset into different numbers of clients: 3, 5, 7, and 10. The purpose of these experiments was to observe how the model's accuracy and performance behave as the number of participating clients increases. By distributing the dataset across a growing number of clients, we aim to simulate more extensive and diverse underwater network scenarios, where multiple autonomous underwater vehicles (AUVs) or sensors contribute to the FL process.

This evaluation is crucial for determining whether the FL framework can maintain consistent performance as the system scales. A scalable framework should be able to handle increasing numbers of clients without significant loss of accuracy or convergence issues. By analyzing the behavior across varying numbers of clients, we can assess the robustness and adaptability of the FL approach in real-world distributed network environments.

The graph in Fig. 11 depicts the relationship between the number of clients and the accuracy of the federated learning framework. The accuracy starts at 98.26% with 3 clients and shows a slight decrease as the number of clients increases, stabilizing around 97.15% with 10 clients. This trend suggests that while the accuracy remains relatively consistent, there is a marginal decline as the client count rises.

This drop in accuracy can be attributed to the division of the dataset among an increasing number of clients, which results in each client having access to a smaller subset of the total data. In a real-world context, such as underwater drone networks, individual drones may collect and train on limited data due to constraints like their geographical range, operational conditions, or varying missions. Despite this reduction in data per client, the relatively consistent accuracy suggests that the FL framework can still function effectively even when data is spread across multiple nodes with limited local datasets. This highlights the framework's capacity to aggregate insights from smaller, distributed datasets without a significant drop in overall performances. This mirrors practical applications, where maintaining consistency and effectiveness in distributed learning is crucial despite potential data scarcity at each individual node.

V. CONCLUSION

This research focused on implementing a FL framework for enhancing data security and cyber intrusion detection in the context of the IoUT. The study leveraged the SOLIDS Lab Underwater Dataset and CIC IDS 2017 dataset to simulate real-world conditions faced by distributed underwater networks, such as limited bandwidth, high latency, and the need for secure and decentralized data processing.

By using the FL framework, we demonstrated that individual underwater nodes (drones) could locally train models on benign data and contribute to a global model without directly sharing sensitive information. The results showed

that this approach effectively detected both known and zero-day attacks, achieving an accuracy of 93.53% for zero-day attack scenarios. This indicates the model's capability to learn normal traffic behavior and identify anomalies without explicit exposure to attack data during training.

We also tested the framework's scalability by distributing the dataset across varying numbers of clients (3, 5, 7, and 10). The model maintained a relatively stable accuracy between 97.15% and 98.26% as the number of clients increased, reflecting the framework's ability to aggregate distributed insights from smaller datasets while accommodating potential non-IID data distributions. The use of the FedProx strategy was instrumental in addressing challenges posed by client heterogeneity and ensuring effective model convergence.

When compared to existing methods such as FedAdam, FedAvg, and centralized ML, the proposed framework achieved a favorable balance between accuracy and computational efficiency. While FedAdam demonstrated slightly higher accuracy, it incurred significantly greater computational and memory costs, which could hinder its applicability in resource-constrained environments such as underwater networks. FedAvg and its variations, including the FedAvg with Autoencoder method, provided reasonable accuracy but were less effective in handling heterogeneous data. In contrast, the proposed framework consistently delivered competitive accuracy while optimizing resource usage, making it a more practical and scalable solution for real-world IoUT deployments.

Overall, this study demonstrated the feasibility of employing FL in IoUT networks, providing a decentralized solution for real-time anomaly detection that balances privacy, accuracy, and scalability. The findings underscore the framework's potential to improve the resilience and security of underwater networks, offering a promising direction for future advancements in IoUT systems.

VI. FUTURE RESEARCH

"Building on the successful implementation of FL for intrusion detection in the IoUT, further research is required to address critical challenges related to adversarial robustness and real-time operational security. These aspects are vital to ensuring the resilience and reliability of FL frameworks in dynamic and potentially hostile underwater environments.

Adversarial Robustness in FL: The decentralized nature of FL is inherently vulnerable to adversarial threats, including model poisoning, backdoor attacks, and gradient inversion. Model poisoning occurs when malicious nodes contribute corrupted updates to disrupt global model performance, while backdoor attacks aim to insert hidden triggers that only activate under specific conditions. Gradient inversion, on the other hand, exploits shared gradients to infer sensitive local data, posing a significant privacy risk.

Future research should prioritize the integration of Byzantine-resilient aggregation methods, which are designed to detect and mitigate the impact of malicious updates during the aggregation process [34].Techniques such as Krum and Multi-Krum, which selectively aggregate updates based on their statistical properties, have shown promise in reducing the influence of adversarial nodes [35]. Additionally, implementing Differential Privacy (DP) can enhance privacy by adding calibrated noise to shared gradients, ensuring that individual data points remain indistinguishable from aggregated contributions [36].

Federated Adversarial Training (FAT) is another promising avenue. By simulating adversarial attacks during the training process, FAT can enhance the model's robustness against adversarial inputs and ensure reliable anomaly detection in diverse and hostile environments. Future studies should also explore trust-based mechanisms that evaluate the reliability of participating nodes, dynamically excluding those exhibiting adversarial behavior from contributing to the global model [30].

Real-Time Threat Response Mechanisms: While this study demonstrates the effectiveness of anomaly detection, real-time response capabilities are essential for mitigating the impact of detected threats and ensuring the operational security of underwater networks. Underwater drones often operate in isolated and unpredictable conditions, making autonomous and timely threat response mechanisms crucial for maintaining network functionality and resilience.

Future work should focus on the development of dynamic quarantine protocols that can isolate compromised nodes from the network without disrupting the overall communication flow. By combining real-time anomaly detection with autonomous recovery protocols, drones can adaptively reconfigure network routes to maintain connectivity and prevent cascading failures. Collaborative threat mitigation, leveraging FL's decentralized nature, would allow drones to exchange threat intelligence and preemptively adapt to emerging threats across the network.

ACKNOWLEDGMENT

The authors express their deepest gratitude to Dr. Issa Traoré and the National Research Council Canada (NRC) for their substantial support of their research endeavors. The funding provided by NRC and the invaluable data supplied by their research teams have been instrumental in advancing their work on enhancing data security and cyber intrusion detection in underwater drone networks. They also extend their appreciation to the University of Victoria for its academic and infrastructural support, fostering a collaborative environment for the successful completion of this project.

REFERENCES

[1] S. Wati, N. Rakesh, and P. N. Astya, "Data communication issues in underwater sensor network," in *Proc. Int. Conf. Comput., Commun., Intell. Syst. (ICCCIS)*, Greater Noida, India, Oct. 2019, pp. 150–155.

[2] J. Pei, W. Liu, L. Wang, C. Liu, A. K. Bashir, and Y. Wang, "Fed-IoUT: Opportunities and challenges of federated learning in the Internet of Underwater things," *IEEE Internet Things Mag.*, vol. 6, no. 1, pp. 108–112, 2023.

[3] N. Victor, C. Rajeswari, M. Alazab, S. Bhattacharya, S. Magnusson, P. K. R. Maddikunta, K. Ramana, and T. R. Gadekallu, "Federated learning for IoUT: Concepts, applications, challenges and opportunities," 2022, *arXiv:2207.13976*.

[4] K. Y. Islam, I. Ahmad, D. Habibi, M. I. A. Zahed, and J. Kamruzzaman, "Green underwater wireless communications using hybrid optical-acoustic technologies," *IEEE Access*, vol. 9, pp. 85109–85123, 2021.

[5] Z. Zhang, C. Wen, Y. Song, and G. Feng, "Adaptive quantized output feedback control of nonlinear systems with mismatched uncertainties and sensor failures," *IEEE Trans. Autom. Control*, vol. 68, no. 12, pp. 8216–8223, Dec. 2023.

[6] H. Zhang, L. Zhang, and X. Zhao, "Dynamic guaranteed cost event-triggered-based anti-disturbance control for T-S fuzzy wind-turbine systems," *IEEE Trans. Fuzzy Syst.*, vol. 31, no. 10, p. 112, Oct. 2023.

[7] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, and S. Zhao, "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, nos. 1–2, pp. 1–210, 2021.

[8] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist. (AISTATS)*, Ft. Lauderdale, FL, USA, Apr. 2017, pp. 1273–1282.

[9] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[10] H. Zhu, Z. Zhou, and Z. Shao, "Federated learning for collaborative intrusion detection in edge computing: Opportunities and challenges," *IEEE Network*, vol. 34, no. 6, pp. 272–279, Jun. 2020.

[11] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," 2018, *arXiv:1811.03604*.

[12] X. L. Nguyen, T. H. Nguyen, and D. H. Tran, "Federated learning for enhanced data security and cyber intrusion detection in distributed networks of underwater drones," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Jul. 2021, pp. 1–6.

[13] H. Fatorachian and H. Kazemi, "AI-enhanced fault-tolerant control and security in transportation and logistics systems: Addressing physical and cyber threats," *Complex Eng. Syst.*, vol. 4, no. 3, p. 17, Sep. 2024.

[14] S. Yan, M. Shen, and X. Sun, "Fusion-based event-triggered h infinity state estimation of networked autonomous surface vehicles with measurement outliers and cyber-attacks," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 4901–4913, Jul. 2024.

[15] A. Sharma, S. Kalra, and K. Yadav, "Federated learning in smart home intrusion detection systems," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 1135–1142, Jun. 2020.

[16] H. Nguyen, S. Marchal, K. Miettinen, N. Asokan, and A. Sadeghi, "FL-based intrusion detection in industrial IoT: A case study," *IEEE Trans. Ind. Inf.*, vol. 15, no. 5, pp. 3040–3049, May 2019.

[17] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 2938–2948.

[18] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, 2019.

[19] L. Liu, J. Zhang, S. Song, S. Yu, and H. Chen, "Federated transfer learning for smart home IDS," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1234–1245, Feb. 2021.

[20] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[21] K. T. T. Tran, P. Pandey, R. Gill, and N. M. Noronha, "FedProx: Federated learning with heterogeneous data," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–8.

[22] Q. Li, Z. Wen, and B. He, "FedNova: Robust federated learning with normalized averaging," in *Proc. 27th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2021, pp. 1–6.

[23] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv:1806.00582*.

[24] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[25] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, Madeira, Portugal, 2018, pp. 108–116, doi: 10.5220/0006639801080116.

[26] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," in *Proc. 3rd Int. Conf. Inf. Syst. Secur. Privacy*, Porto, Portugal, 2017, pp. 253–262, doi: 10.5220/0006105602530262.

[27] M. Kim. (2024). *DDoS Attack Prediction Datasets*. Accessed: Aug. 10, 2024. [Online]. Available: https://drive.google.com/drive/u/0/folders/1IjFET0QOOZ4foEG-qzBLzTQKag6cUdiy

[28] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016, doi: 10.1109/COMST.2015.2494502.

[29] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K. H. Li, T. Parcollet, P. P. B. de Gusmão, and N. D. Lane, "Flower: A friendly federated learning research framework," 2020, *arXiv:2007.14390*.

[30] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konecný, S. Mazzocchi, B. McMahan, T. Van Overveldt, D. Petrou, D. Ramage, and J. Roselander, "Towards federated learning at scale: System design," in *Proc. 2nd SysML Conf.*, Apr. 2019, pp. 1–11.

[31] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Kone, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," in *Proc. 9th Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–9.

[32] J. Pope, T. Spyridopoulos, V. Kumar, F. Raimondo, S. Gunner, G. Oikonomou, and A. Khan, "Intrusion detection at the IoT edge using federated learning," in *Security and Privacy in Smart Environments*. Cham, Switzerland: Springer, 2024, pp. 98–119.

[33] S. Pal, "A comprehensive study of zero-day attacks and mitigation techniques," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 4, pp. 1058–1067, Sep. 2011, doi: 10.1109/TETC.2019.2916205-448.

[34] P. Blanchard, E. M. E. Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–8.

[35] P. Pillutla, T. M. H. Ng, and Z. Wang, "Robust aggregation for federated learning," *IEEE Trans. Signal Process.*, vol. 69, pp. 500–515, 2021.

[36] N. Papernot, M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," 2016, *arXiv:1610.05755*.

[37] C. Kapoor, N. K. Popli, A. Sharma, and R. Gupta, "Evaluation of concrete characteristics using smart machine learning techniquesa review," in *Proc. Can. Soc. Civil Eng. Annu. Conf.* Cham, Switzerland: Springer, May 2022, pp. 1279–1294.

[38] R. P. Singh, B. Singh, and N. K. Popli, "Temporal analysis of oceanographic data: Insights into environmental variability and trends," in *Proc. IEEE Pacific Rim Conf. Commun., Comput. Signal Process. (PACRIM)*, Sep. 2024, pp. 1–9.

[39] N. K. Popli, A. Girdhar, V. Chauhan, and Y. Goyal, "Employing decentralized technologies for machine-learning detection of advanced malware," *J. Advance Res. Dyn. Control Syst.*, vol. 10, no. 12, pp. 1292–1299, May 2018. [Online]. Available: https://iitmjp.ac.in/wp-content/uploads/2017/06/IT-Conference-2015.pdf

MANSAHAJ SINGH POPLI (Student Member, IEEE) is currently pursuing the bachelor's degree in software engineering with the University of Victoria. His research interests include federated learning, intrusion detection using machine learning, cybersecurity and threat modeling, and machine learning in IoT and IoUT environments.

RUDRA PRATAP SINGH (Student Member, IEEE) is currently pursuing the M.A.Sc. degree with the University of Victoria. He is also a Graduate Research Assistant with the University of Victoria. His work combines academic rigor with practical innovation, addressing key challenges in smart connected systems, and cybersecurity. He has published work in the areas of monitoring systems and oceanographic applications, contributing to advancements in the integration of IoT technologies for marine research. His research interests include the Internet of Things (IoT), embedded systems, and cybersecurity, with an emphasis on designing secure and efficient solutions for real-world challenges.

NAVNEET KAUR POPLI (Senior Member, IEEE) received the B.Sc. degree in instrumentation from the University of Delhi, India, in 2000, the M.C.A. degree in computer applications from Guru Gobind Singh Indraprastha University, Delhi, India, in 2003, and the Ph.D. degree in computer and information systems security from Tilak Maharashtra Vidyapeeth, Pune, India, in 2019. She is currently an Associate Professor with the Department of Computer Science, University of Victoria, Victoria, BC, Canada. She has over 20 years of experience in cybersecurity, distributive computing, software development, scalability and testing, and artificial intelligence. Her research interests include strategies for the development and security of large intelligent systems, smart cities, the IoT devices, and cyber-physical systems. She is a registered Professional Engineer (P.Eng.) with Engineers and Geoscientists British Columbia (EGBC).

MOHAMMAD MAMUN (Senior Member, IEEE) received the M.S. degree in information and communication system security from the Royal Institute of Technology (KTH), Sweden, and the Ph.D. degree from Japan Advanced Institute of Technology (JAIST), in 2014. He is currently a Senior Research Officer with the Cybersecurity Team, National Research Council (NRC), Canada. He is also an Adjunct Professor with the University of Victoria and the University of Windsor. Prior to joining NRC, he held several research and development positions, including a Research Associate at CIC-UNB and the Analytics Manager at The Learning Bar Inc. He is leading several NRC G&C projects. He is the author of more than 45 cybersecurity-related articles, including a patent on cybersecurity risk assessment. His research interests include human-centric cybersecurity, entity behavioral fingerprinting, insider threat detection, applied cryptography, applied machine learning, network security, and the IoT security and privacy. In 2022, he received the NRC Rising Star Award. He received the Outstanding Performance Award for the Ph.D. degree.