Cluster Computing (2025)28:473
https://doi.org/10.1007/s10586-025-05325-w

Federated learning in intrusion detection: advancements, applications, and future directions

Busra Buyuktanir[1] · Şahsene Altinkaya[2] · Gozde Karatas Baydogmus[1] · Kazim Yildiz[1]

Received: 26 November 2024 / Revised: 12 March 2025 / Accepted: 25 April 2025

Abstract
Federated Learning (FL) has emerged as a promising distributed machine learning approach that addresses confidentiality and integrity concerns in various sectors, including Internet of Things (IoT), healthcare, finance, and cybersecurity. In order to improve privacy protection and detection accuracy in decentralized systems, this study investigates the incorporation of FL into Intrusion Detection Systems (IDS). FL is especially useful in situations where data security and privacy are crucial because it allows for the cooperative training of models without centralizing sensitive data. We examine many FL-based IDS solutions across several domains, emphasizing how well they mitigate data breaches, maintain confidentiality, and enhance intrusion detection capabilities. The use of Generative Adversarial Networks (GANs), artificial immune systems, and hybrid deep learning techniques to maximize IDS performance are among the current developments in FL methodology that are covered in the paper. We also look at issues like the requirement for effective aggregation procedures and non-independent and identically distributed (non-IID) data. Finally, we outline future directions and open research topics to improve the scalability, resilience, and effectiveness of FL-based IDS solutions in practical applications.

Keywords Intrusion detection systems · Federated learning · Deep learning · Machine learning · Literature review

1 Introduction

Lately, the complexity and diversity of cyber threats have challenged the effectiveness of traditional security systems. The increasing complexity of networks is creating many security vulnerabilities.

Actions to undermine the confidentiality, integrity, or availability of a computer system or network, or to circumvent its security measures, are considered intrusions [1]. As a result, there is a continuous search for innovative and more efficient methods in the field of cyber security. To effectively detect different types of attacks, an accurate IDS is essential [2].

Recently, with the advances in the fields of Deep Learning (DL) and Machine Learning (ML), including various unified models, these approaches have been widely pursued in different fields such as cyber security [3], computer vision [4] and healthcare systems [5, 6]. Again, these algorithms can provide autonomous approaches to identifying different types of attacks without human intervention. For example, a paper published in 2020 [2] used deep learning techniques on a large and unstable dataset for intrusion detection. Despite the large size of the dataset, the class distribution is unbalanced. As this imbalance affects the performance of the model, a data sampling method was used. All of this was done on a centralized system. Although IDS operation has been successfully used in approaches, it is usually used for all a

Şahsene Altinkaya, Gozde Karatas Baydogmus, and Kazim Yildiz authors have been contributed equally to this work.

✉ Busra Buyuktanir
busra.buyuktanir@marmara.edu.tr

Şahsene Altinkaya
sahsene.altinkaya@utu.fi

Gozde Karatas Baydogmus
gkaratas@marmara.edu.tr

Kazim Yildiz
kazim.yildiz@marmara.edu.tr

[1] Department of Computer Engineering, Marmara University, Istanbul 34854, Türkiye

[2] Department of Mathematics and Statistics, University of Turku, Turku 20014, Finland

Published online: 04 August 2025

centralized system for processing data collected from users, data privacy [7] issues arise.

Because of these challenges, the concept of FL, which addresses the issues of privacy and learning on the local device, has come to the fore [7]. FL allows devices to connect and learn together without the need to send data to a central server. That is, thanks to machine learning/deep learning (ML/DL) capabilities, multiple devices and servers can be trained without the need for a centralized data repository [8–10]. FL involves selecting a subset of clients to assist in the learning process by distributing the current global model to them. The architecture trains an ML model on each client's data independently. Each client then sends its trained models to a central server, where these models are aggregated to create a new model. The final model is then sent back to the clients [7]. Figure 1 illustrates this process. This iterative process continues until the target level of performance is achieved. In essence, federated learning relies on two core concepts: local updating and global aggregation. This underlines FL's emphasis on data privacy, allowing clients to benefit from other clients' data without transferring it to a central server.

Although FL is being used more and more, the approaches created to deal with its different facets are still disjointed, which calls for a methodical examination. The key challenges such as handling non-IID data, ensuring model security during updates, and optimizing computational efficiency remain unresolved. Motivated by these challenges, the solutions to enhance both security and performance in real-world applications and the evaluation datasets of FL-based intrusion detection techniques are thoroughly reviewed in this work.

The following are the main contributions of this paper:

- We have systematically examined and compared various FL algorithms, including FedAvg, Fed0pt, FedProx, FedAdam, FedAdagrad, and FedYogi, to evaluate their effectiveness in different IDS scenarios. Unlike prior studies that focus on a single or limited selection of FL techniques, our research provides a broader perspective on algorithmic performance across multiple datasets.
- We present an extensive experimental evaluation conducted on well-established cybersecurity datasets such as Edge-IIoTset, CICIDS2017, NSL-KDD, KDD99, and UNSW-NB15. This multi-dataset approach ensures that our findings are robust and generalizable across various real-world settings, setting our study apart from works that rely on limited dataset evaluations.
- While many studies highlight the advantages of FL in IDS, our work goes a step further by critically analyzing key challenges, including handling non-IID data, mitigating the risks posed by untrusted participants, optimizing communication overhead, and improving model scalability. We propose future research directions that address these gaps, guiding the development of more robust and practical FL-based IDS solutions.

The paper is organized into the following sections: Sect. 2 indicates a concise overview of malware and IDSs. Section 3 outlines the general architecture of FL. In Sect. 4, the datasets utilized for intrusion detection are discussed. Section 5 critically examines FL-based IDSs, presenting a detailed description and comparative analysis of selected methodologies. Finally, the paper concludes with a summary and outlines future avenues for research.

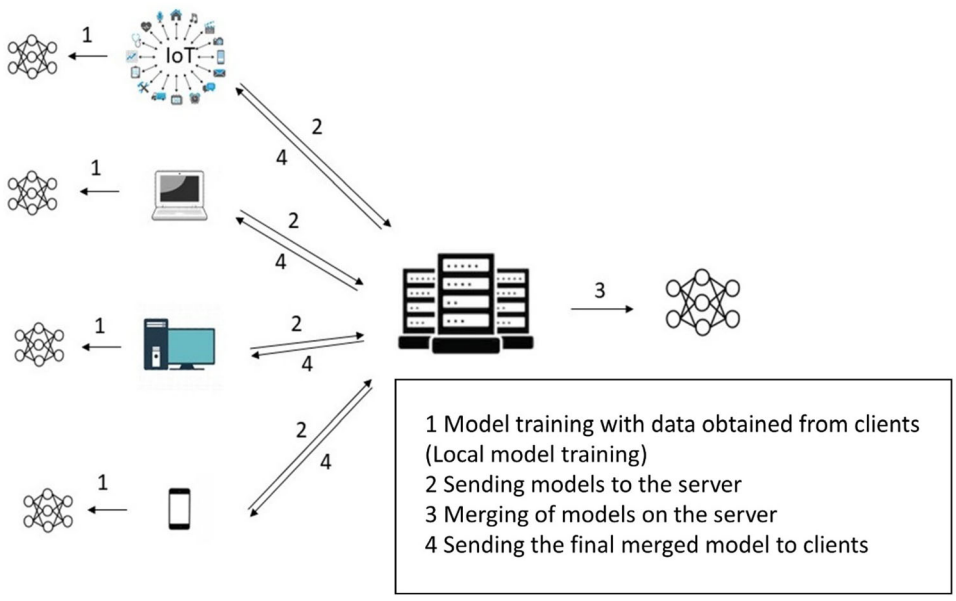

Fig. 1 Architecture of federated learning

2 Malwares and intrusion

2.1 Malware, malware types, and malware analysis

The world is gradually moving into the digital age [11]. The use of computers, smart devices, and the internet is increasing every day, and with these technologies come new concepts such as artificial intelligence and cryptocurrencies. As much as the digital age brings innovation and opportunities, it also brings threats and challenges in the area of cybersecurity. In particular, malware has become more dangerous as reliance on technology has increased. Malware is malicious software used to disrupt the functioning of technology, such as computers, mobile, or IoT devices, by accessing their systems and collecting critical information on these devices. The first examples of malware were created for fun and experimentation. Today, however, it is used to capture and damage sensitive personal or corporate information. According to studies, thousands of new types of malware are being created very quickly as technology advances [12]. Malware can be categorized by function. The names and descriptions of the most common types of malware are given in the table below Table 1 [13].

Malware, some types of which are listed in Table 1, are analyzed to examine and understand them more clearly. This is called malware analysis. With this analysis, it is determined how malware software works, what kind of damage it can cause, how it can be detected, and how it can be neutralized [14]. The types of malware analysis are presented in Fig. 2.

1. Static Analysis: In this type, malicious software is examined without running. The source code, file structures, and file content of the software are analyzed. Calculating the hash value of a malware file or examining the machine code with disassembly tools can be given as examples of static analysis. IDA Pro and PEiD are examples of tools used for this analysis.
2. Dynamic Analysis: This analysis involves running malicious software in a safe environment and examining its behavior. An example of dynamic analysis is running the malware in a virtual machine and examining what files it changes, what system calls it makes, or what network connections it makes. Cuckoo Sandbox, Process Monitor, and Wireshark are examples of tools used for this type of analysis.
3. Memory Analysis: This type examines the activity of malware in memory. Volatility Framework and Rekall are examples of tools used for this type of analysis.
4. Code Analysis: This type of analysis involves a detailed examination of the source code or compiled code of malicious software. Examples of tools used for this type of analysis include Radare2 and Ghidra.
5. Hybrid Analysis: In this type of analysis, static and dynamic methods are combined. Firstly, the basic characteristics of the malware are determined using the static analysis method, and then the behaviors of the malware during operation are examined using the dynamic analysis method. FireEyemis an example of the tools used for this analysis.
6. Artificial Intelligence (AI) and ML-Based Analysis: In this type, AI and ML algorithms are used to detect malicious software. Examples of tools used for this analysis are MalwareBazaar and VirusTotal.

Table 1 Malware types and descriptions

Types	Description
Viruses	Malware that replicates by attaching itself to a program or file, corrupting files, slowing down the system, or making it completely unusable.
Trojans	Software that appears to be useful or harmless but performs harmful operations by gaining unauthorized access to the system in the background.
Worms	Malware that spreads over networks, increases network traffic, and can multiply without user intervention.
Spyware	Malware that monitors the user's activities and collects personal information without the user's knowledge.
Adware	Malware that installs itself on the device without user permission and runs in the background of the system, displaying unwanted ads.
Ransomware	Malware that encrypts personal files on the system, restricting user access, and then requests a ransom to decrypt them.
Rootkits	Malware that takes control of the system and makes it difficult to detect both the malware and its activities.
Botnets	Networks controlled by the attacker, used to send spam messages and perform DDoS attacks.
Cryptojacking	Malware that mines cryptocurrency by using system resources on the compromised computer.
Backdoor	Malicious software that allows attackers to gain unauthorized and secret access to systems or networks.
Keylogger	Malware that records the user's keystrokes and sends the information to the attacker.

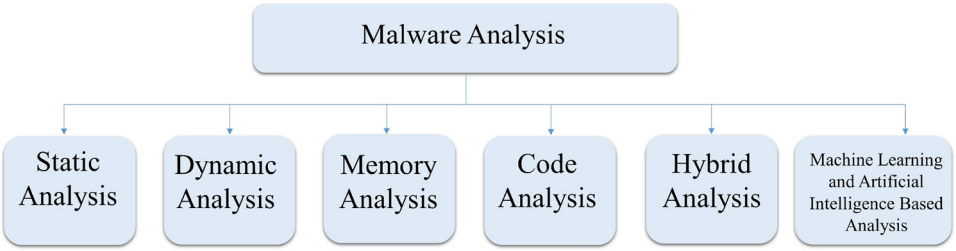

Fig. 2 Types of malware analysis

2.2 Intrusion and intrusion types

Intrusion can be defined as unauthorized access to the systems or networks of technologies such as computers, mobile, and IoT devices. The definition of Intrusion is visualized in Fig. 3. With intrusion, the attacker aims to capture the user's information, change the data, or disrupt the system. As a result of Intrusion, companies, government organizations or individuals may experience serious security risks. On the other hand, this situation can lead to significant financial losses, loss of reputation, or legal problems for the victim.

Intrusions can be categorized into types according to the method, motivation, effects, and objectives of the attacker.

1. According to Attack Methods
   (a) Network-based Attacks: Such attacks are carried out over the network. The goal is to damage the network infrastructure. Examples of this type include Denial of Service (DoS) [15], Distributed Denial of Service (DDoS) [15], Man-in-the-Middle (MITM) [16] and Sniffing attacks [17].
   (b) Application-based Attacks: This type of attack targets applications. The attack detects security vulnerabilities in the application and provides unauthorized access. This can lead to problems such as unauthorized data access, information theft, and system corruption. Instances of this type of attack include SQL Injection [18], Cross-Site Scripting (XSS) [4], and Remote File Inclusion (RFI) [19].
   (c) Social Engineering Attacks: Human psychology is the target of such attacks. Attacks are carried out by manipulating people psychologically. Examples include Phishing [20], Pretexting [21], Baiting [21], and Quid pro quo [21].
2. According to Attack Motivations
   (a) Targeted Attacks: Such attacks target an individual, organization, or system. The attack is intended to cause espionage, information theft, or damage. Examples of this type of attack include Advanced Persistent Threat (APT) [22] and Spear Phishing [23].
   (b) Opportunistic Attacks: In this type of attack, attackers do not select a specific target, but instead target systems with security vulnerabilities. Examples of this type of attack include Worms [13], Viruses [13], and Botnets [24].
3. According to the Effects of the Attack
   (a) Confidentiality Breaching Attacks: In such attacks, confidential and sensitive information is captured by unauthorized persons. Examples include spyware [25] and Data leakage [26].
   (b) Integrity Violating Attacks: The goal is to modify the targeted data or to disrupt the integrity of the data. Examples include file tampering [27] and data manipulation [28].
   (c) Availability Disrupting Attacks: The functioning of systems, services, or networks is disrupted by such attacks. Instances of this type of attack include DoS and DDoS attacks [15].
4. According to the Attacker's Objectives

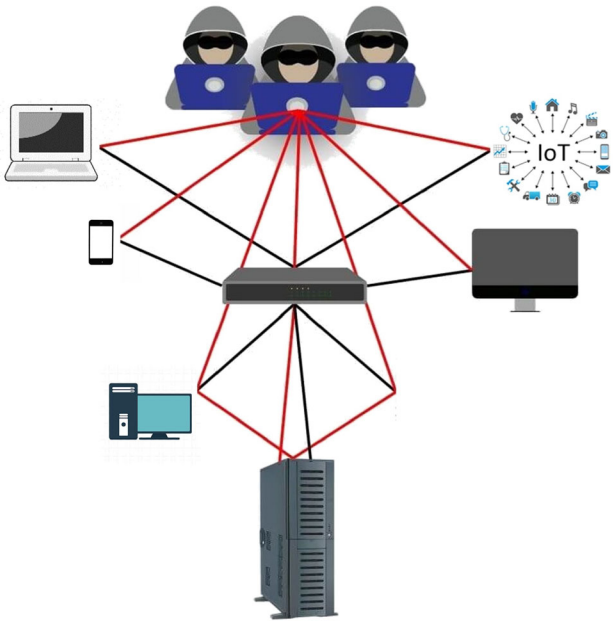

Fig. 3 What is intrusion?

(a) Financially Motivated Attacks: In this type, the goal is money. The attacker may steal the victim's credit card details or commit bank fraud. Ransomware [29] is an example of this type of attack.
(b) Ideologically or Politically Motivated Attacks: Such attacks are designed to achieve ideological or political goals. Attackers may hack websites to deliver social or political messages, or state-sponsored attackers may conduct cyber-attacks to steal confidential information.
(c) Curiosity or Vandalism Motivated Attacks: The purpose of these attacks is not financial or strategic. They are usually harmless attacks carried out to explore systems and gain knowledge and experience. Young attackers known as script kiddies carry out these attacks.

2.3 IDSs

IDSs, first introduced in the 1980 survey "Computer Security Threat Monitoring and Surveillance," are software/hardware security tools used to eliminate threats that may occur during data transmission, prevent unauthorized access, and notify security personnel of attacks [30]. The need for IDS can be explained by several reasons. Firstly, These systems can detect attacks that other security mechanisms cannot prevent. They also provide proactive defense by responding to the analysis phase before the attack occurs. Finally, IDS improves overall security by enabling attack analysis, system repair, and remediation. The advantages of IDS are as follows:

1. Early detection capability
2. Capacity for detailed information collection
3. Ability to provide high-quality evidence

However, IDS also has some weaknesses:

1. Vulnerability to fragmentation of packets and timing-based attacks
2. Confusion of the scan sequence
3. Packet hijacking

An incoming packet may be sent either for regular communication or with malicious intent. For this reason, it is difficult to determine whether incoming packets are being sent for attack purposes. IDS can be classified according to various criteria:

1. Based on Architectural Structure
   (a) Host-Based IDS (HIDS): Operates on servers or individual devices, monitoring their traffic, log files, and transactions [31].
   (b) Network-based IDS (NIDS): Monitors all incoming network traffic, logs the content of each packet, intervenes to block attacks when needed, and produces reports [31].
2. Based on the Type of System They Protect
   (a) Small-Scale Systems: Designed for individual computers or small networks.
   (b) Large-Scale Systems: Designed for corporate networks and large-scale IT infrastructures.
3. Based on Data Processing Time
   (a) Real-Time IDS: Detects and responds to attacks immediately.
   (b) Non-Real-Time IDS: Collects data and analyzes it at intervals to detect attacks.
4. Based on Techniques Used
   (a) Signature-Based IDS: Uses predefined attack signatures to recognize known attack types [32].
   (b) Anomaly-Based IDS: Detects previously unseen attacks by identifying deviations from normal system behavior [32].

2.4 Intrusion detection techniques

Intrusion detection techniques (IDT) are various methods used to identify malicious, unauthorized activity that may be occurring in the network or system. These techniques attempt to detect potential threats by analyzing system behavior and network traffic [33]. IDT can be divided into anomaly-based, behavior-based, signature-based, heuristic-based, ML-based, and hybrid. The writers of [34] presents a survey of various intrusion detection techniques developed and/or researched between 2000 and 2019, including the accuracy of the outputs. With the increasing use of the internet, IDS have become a crucial component of network security. The study highlights the difficulty of identifying the best solution among the numerous available IDS options and provides an inclusive overview of different published solutions in this field. Table 2 shows the types, definitions, advantages, and disadvantages of intrusion detection techniques. Signature-Based Detection is used to quickly detect known threats by utilizing predefined attack signatures; however, it is effective only against known attacks and has limited capacity to detect new or unknown threats. Anomaly-Based Detection aims to identify anomalies by analyzing normal system behavior, allowing the detection of unknown attacks. However, changes in normal behavior can lead to false positives, and it requires high processing power. Hybrid Detection combines both signature-based and anomaly-based methods to provide

Table 2 Intrusion detection techniques

IDT	Description	Advantages	Disadvantages
Signature-based detection [35]	Method of detecting security threats using predefined signatures for known malicious attacks.	1. Quickly detects known threats. 2. Simple method. 3. Low false positive rate since the attack characteristics are predefined.	1. Only effective against known threats. 2. Limited in scope. 3. Ineffective against unknown or emerging threats. 4. Signatures must be constantly updated for new threats.
Anomaly-based detection [36]	Technique that detects unknown security threats by identifying anomalies in normal system behavior.	1. Can detect unknown attacks. 2. Provides comprehensive security by monitoring multiple data sources like system performance, network traffic, and user behavior. 3. Adapts to new threat types.	1. Changes in normal behavior might be seen as anomalies, leading to high false positives. 2. Learning period for normal behavior may miss attacks. 3. High processing power and memory usage required.
Hybrid detection	Systems that detect attacks using a combination of anomaly-based and signature-based techniques.	1. Comprehensive detection. 2. Reduces false positives.	1. Requires high cost and processing power.
Behavior-based detection [37]	Focuses on the behavior of systems or users, identifying deviations from typical activities as potential threats.	1. Detects previously unknown threats. 2. Quickly adapts to evolving threats.	1. High processing power and storage capacity required. 2. Profiling normal behavior may take time, leading to missed threats.
Heuristic-based detection [38]	Detects security threats and malware using characteristic or behavioral analysis.	1. Can detect emerging threats. 2. Can identify potential threats in advance, allowing for preventive actions.	1. Uses more system resources due to complex analysis.
Machine Learning-Based Detection [39]	Uses machine learning algorithms to detect specific events, behaviors, or anomalies.	1. High accuracy detection by learning from large data sets. 2. Real-time anomaly detection. 3. Continuous learning and adaptation.	1. Accuracy depends on sufficient and quality data. 2. Training and maintenance of complex models can be time-consuming and costly.

more comprehensive detection, but it comes with high costs and processing power requirements. Behavior-Based Detection identifies deviations in system or user behavior to detect previously unseen threats, but profiling normal behavior can be time-consuming and requires significant processing power. Heuristic-Based Detection attempts to identify new threats using characteristic analysis, but due to complex analysis, it can consume more system resources. Finally, Machine Learning-Based Detection utilizes machine learning algorithms to detect events, behaviors, or anomalies, but it depends on sufficient and high-quality data for accurate detection, and the training and maintenance of models can be time-consuming and costly.

3 Federated learning

Today, with the fast-paced advancement of Internet-based technologies, users of these technologies are constantly generating data [40]. The generated data are processed and made more meaningful. States, institutions, or organizations that use meaningful data in areas such as security, education, health, finance, etc. can analyze the situation and help predict the future more accurately. Such effective use of data enables acceleration of business processes and increase in revenues [41].

On the other hand, the generated data containing personal information is of great importance in terms of privacy [42]. Data stored and shared on servers raises various security issues to protect data privacy. Countries have made legal arrangements to meet the needs of data privacy and data security. Without the consent of individuals, the processing of personal information is limited by the Personal Data Protection Law (KVKK) and the General Data Protection Regulation (GDPR). Nonetheless, it is not enough to rely only on legal regulations to ensure security [43]. Technological solutions are also needed in practice to ensure real security.

Federated Learning, developed by Google in 2016 to address security concerns by ensuring data privacy, is a new generation artificial intelligence technology [44]. Federated Learning is a technology designed for distributed systems with clients and servers [45]. With this technology, the data generated by the clients is not transmitted to the server. Indeed, a model is trained locally using data generated on each client, and these trained models are then transmitted to the server. The models transmitted to the server are aggregated to form a centralized model. This

centralized model is then distributed to the clients [46]. Figure 4 exhibits how the federated learning architecture works.

Data privacy is ensured by conducting model training at the point of data generation (client) without transmitting the data to the server. Additionally, sending the model rather than client data to the server reduces network traffic and communication costs, while also facilitating low energy consumption and fast communication.

Based on the types of clients and the data they generate, FL is categorized into two distinct types: cross-silo and cross-device [47].

Cross-silo Federated Learning is an architecture where clients are companies or organizations such as banks, hospitals, etc. This architecture is used when the number of clients is limited the data is sensitive and data cannot be easily shared between different clients. Figure 5 exhibits an example of a cross-silo FL architecture whose customers are banks.

Cross-device Federated Learning is an architecture in which clients consist of technological devices such as computers, tablets, phones, wearable devices, etc. In this architecture, the number of clients is more. An example of this architecture can include mobile device applications. Figure 4 shows the cross-device FL architecture.

In FL systems, there are two communication methods centralized design and decentralized design. With centralized design, model parameters trained on clients are transmitted to the server. Updates are functioned on the server and sent back to the clients [48]. The communication between the server and the clients can either be synchronous [49] or asynchronous [50]. With decentralized design, communication takes place between clients and each client can update the central model parameters immediately [51]. Centralized design is widely used. However, decentralized design is also preferred because of the potential risks related to the collection of information on a single server. But, designing a decentralized communication architecture is highly challenging.

The practical success of FL-supported IDS deployments highlights the need for high-quality, unified datasets that reflect real-world cyber threats. Several studies have demonstrated the effectiveness of FL-based IDS in improving detection accuracy while preserving data privacy. For instance, a multi-class IDS for software-defined networking (SDN) achieved an impressive 98.6% detection accuracy using FL with the Edge-IIoTset dataset [52].

Similarly, the FEDGAN-IDS approach, integrating Generative Adversarial Networks (GANs) with FL, significantly enhanced attack detection performance on datasets like KDD99, NSL-KDD, and UNSW-NB15, reaching up to 99% accuracy [53]. Another successful application, the FedAGRU model, secured wireless edge networks by training on non-IID data across multiple datasets, including KDDCUP99, CICIDS2017, and WSN-DS, demonstrating superior robustness and efficiency [54]. Additionally, the GöwFed system combined FL with Gower Uniqueness matrices to develop an adaptive, privacy-preserving IDS framework for decentralized networks [55].

These applications emphasize the critical role of well-structured datasets in FL-based IDS, reinforcing the need for continuous dataset updates and new benchmarks that capture evolving cyber threats and real-world heterogeneity.

3.1 Model development with federated learning

A model training method developed for distributed systems is the Federated Learning Architecture. In this chapter, model training steps for centralized design are explained step by step. These steps continue cyclically at certain

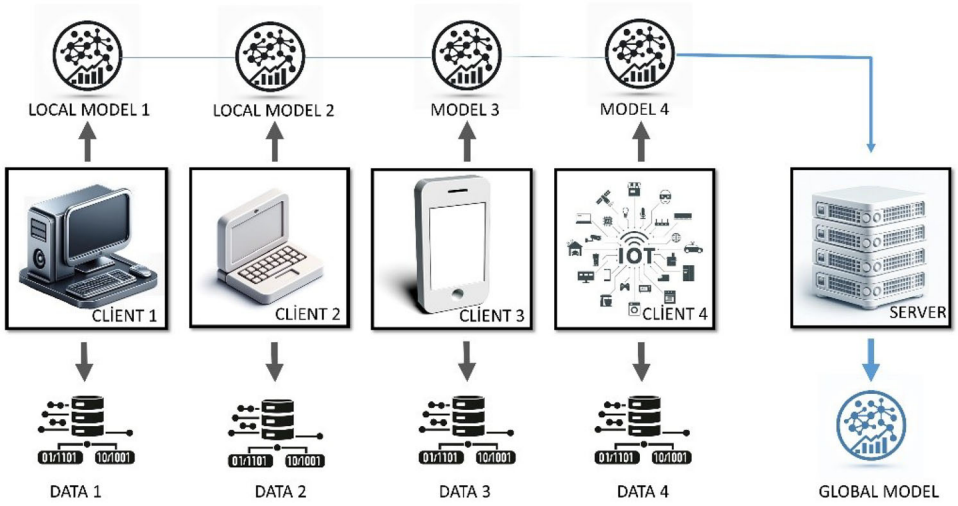

Fig. 4 Federated learning architecture

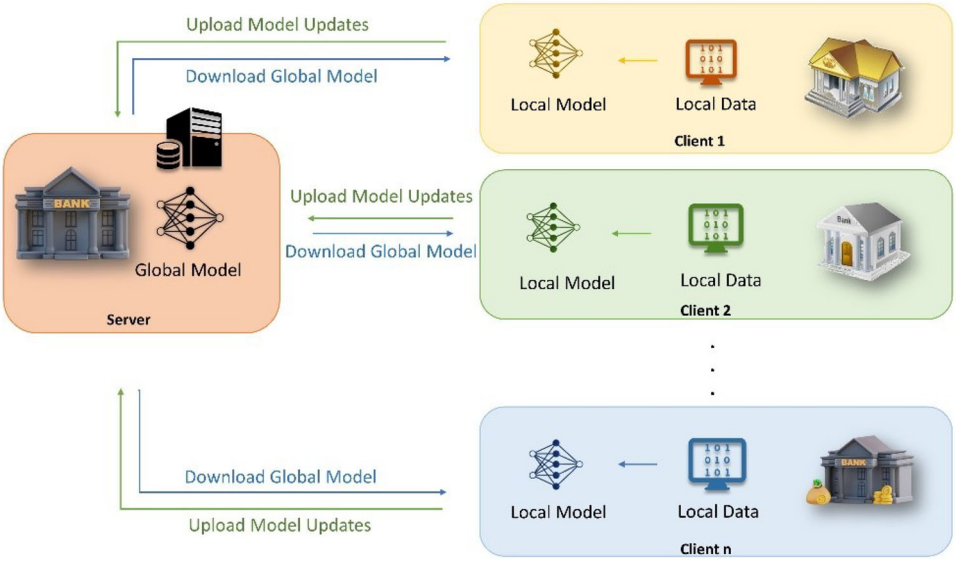

Fig. 5 Cross-silo federated learning architecture whose clients are banks

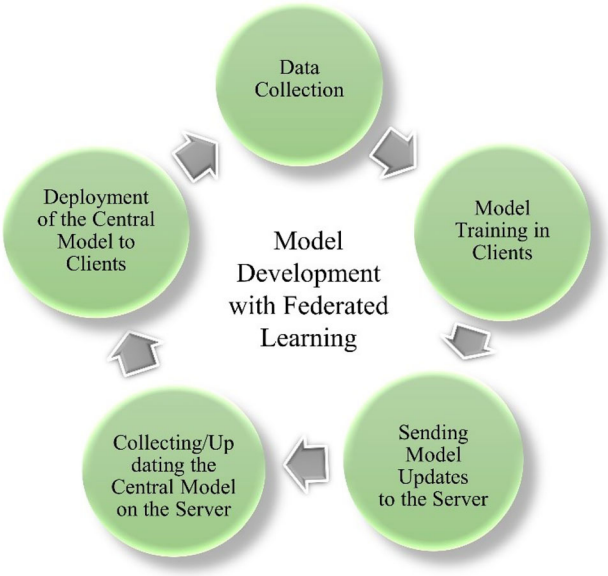

Fig. 6 Model development cycle with federated learning

intervals. In Fig. 6, the loop between the clients and the server is visualized.

- Data Collection
  Each edge/client generates and operates on its local data. The data is not transmitted to a centralized server.
- Training Model on Clients
  Each client trains the model utilizing locally generated data. In subsequent iterations, the model parameters received from the server are updated based on the locally generated data.
- Sending Model Updates to the Server
  The parameters of the trained model, typically in the form of gradients or weights, are transmitted from each client to the server.
- Collecting/Updating the Central Model on the Server
  Model updates from each client are collected on the server. Then these updates are merged and the central model is obtained. In subsequent iterations, updates from each client are merged in this step and then the central model is updated.
- Deployment of the Central Model to Clients
  The updated central model is deployed to clients. The 1st iteration of the model training is completed. Iterations are replicated a certain number of times or until the model reaches a certain performance level. With the updated model, each client begins the next round of training.

3.2 Federated learning types

In the federated learning architecture, different cooperation scenarios occur according to the data to be used for the model to be trained. Solutions suitable for various data partitioning scenarios have been produced for data structure and access limitations. The first of these solutions is Horizontal Federated Learning (HFL), the second is Vertical Federated Learning (VFL) and the third is Federated Transfer Learning (FTL).

Horizontal federated learning: This method represents a federated learning model where the data is divided horizontally between different clients. In this method, datasets with the same set of attributes but with different samples are included. Figure 7 shows an example of an application of HFL. According to the figure, data from several institutions or organizations in different locations are stored on computers. These computers are clients. The independence

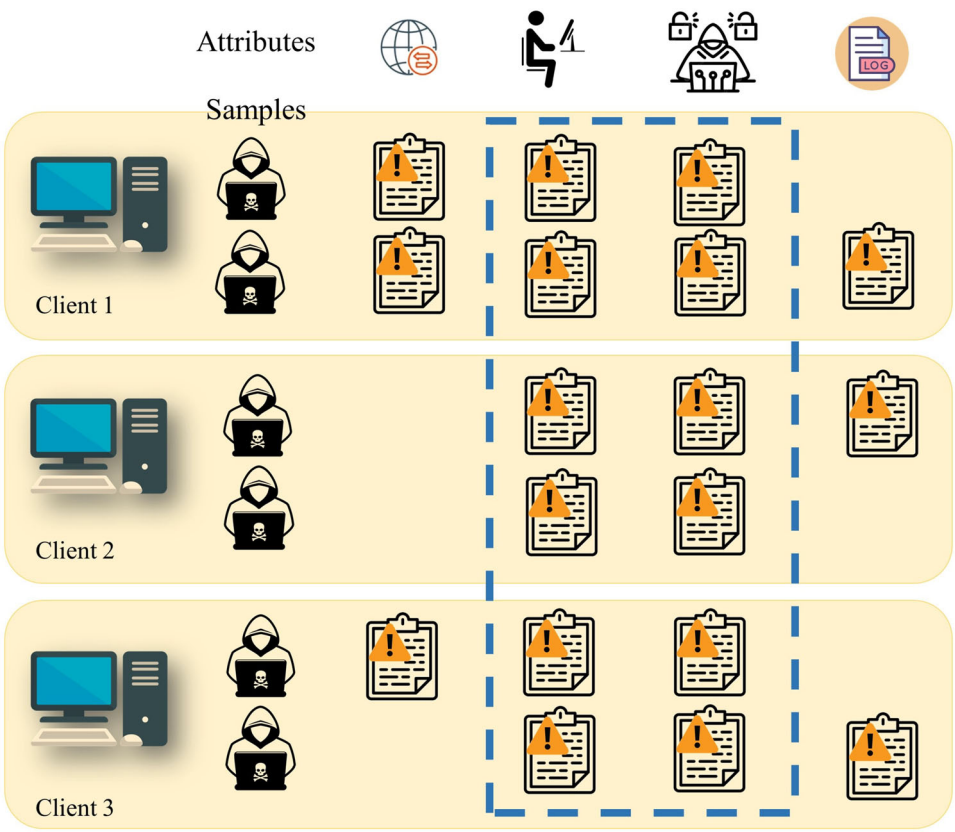

Fig. 7 Example of horizontal federated learning

of the data sets (samples) shared between the clients poses a significant challenge to model training for intrusion detection systems. However, the similarity of attack attributes enables the development of common intrusion detection models and approaches. As in the example shown in Fig. 7, HFL is suitable for model training in cases where the attributes of multiple datasets overlap to a large extent, but the samples overlap less.

Vertical federated learning: This method is an FL where the data is divided vertically between different clients. It covers datasets that have the same samples but contain different characteristics. In Fig. 8, various types of data belonging to the same individuals (samples) are distributed across multiple clients (such as an intrusion detection system (IDS) deployed on a corporate network and another IDS installed on individual user devices). This means that identical individuals interact with the corporate network and their devices. While the corporate IDS collects data on network traffic and potential intrusion attempts within the corporate infrastructure, the user device IDS collects information such as device usage patterns, application behavior, and security events. As a result, the individuals are common to both clients, but the attributes collected from each client are different. The model is trained by combining the characteristics of the same individuals from both IDS clients. As in the example shown in Fig. 8, VFL is suitable for training models in cases where the user attributes of multiple datasets overlap minimally, but the users (samples) overlap significantly.

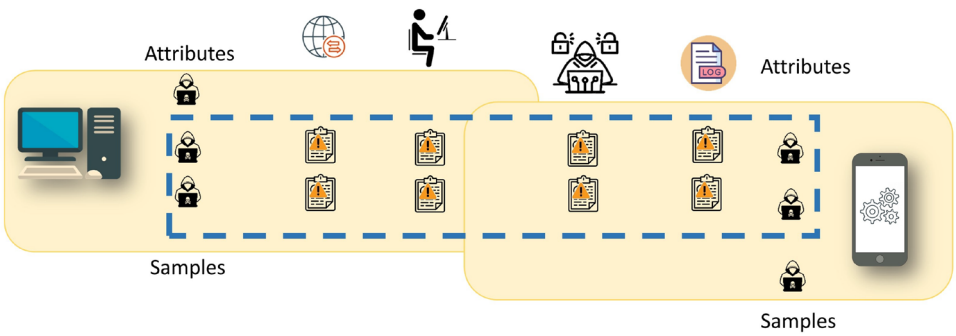

Fig. 8 Example of vertical federated learning

Federated transfer learning: This method is an FL that aims to improve learning by transferring knowledge between clients when both datasets' features and instances are different. It is used for organizations that have limited data or want to improve model performance by using data from different domains. In this method, datasets can be divided both horizontally and vertically. An example application of the FTL is shown in Fig. 9. There are two different network environments in the scenario. The dataset size of one environment is more limited compared to the other. The model parameters developed using the data from the environment with a larger dataset are shared with the environment with a limited dataset. These shared model parameters are fine-tuned using the limited data for optimization. This process enables knowledge transfer without directly sharing any data. In cases where there is little overlap between users and user attributes in two datasets, transfer learning can be used to overcome data or tag deficiencies instead of separating the datasets.

3.3 Privacy in FL

FL is a technique created to ensure data privacy by protecting information confidentiality. In this technique, the data generated on the client is stored here. The trained model parameters are transmitted to the server. Transferring model parameters to the server instead of data protects the confidentiality of the data. However, it is very significant to the confidentiality and security of the model parameters on both the client and the server side, especially for applications using federated learning architecture in sectors such as finance, health, and education. Even if the data remains local, it can be reconfigured with updates transmitted from the server to the client. Various measures are taken to prevent this bad scenario from happening [56].

Cryptographic precautions include privacy-preserving machine learning algorithms such as secure multi-party computation (SMC) [57, 58] and homomorphic encryption [59, 60]. Model parameters are encrypted before transmission. Differential privacy is one of the methods used to provide data security. This is accomplished by introducing noise into the data within extensive datasets [61].

The above methods mentioned in this study are independent of each other and can be used together [62–64].

3.4 Federated learning platforms

Many platforms have emerged for model development with federated learning architecture [51]. The most preferred ones are as follows:

1. TensorFlow federated (TFF): Offering the federated learning features of TensorFlow, this platform enables ML models to run on decentralized data [64].
2. PySyft: PySyft, an open-source FL library, is used for secure and privacy-preserving distributed learning operations [65].
3. Federated AI technology enabler (FATE): This platform provides privacy-preserving data analytics and ML solutions to businesses and researchers [66].
4. PaddleFL (PFL): Developed by Baidu, PaddleFL enables machine learning applications on large data sets using federated learning techniques [51].
5. FedML: FedML is a comprehensive library that allows users to easily deploy and manage federated learning algorithms [67].

4 Datasets

Attackers utilize a range of methods to evade detection by security systems or administrators. To create a robust detection system, it's essential to train the system using a suitable learning method and a top-notch dataset. In academic works, several datasets exist, and this section aims to provide comprehensive insights into these datasets through a comparative analysis.

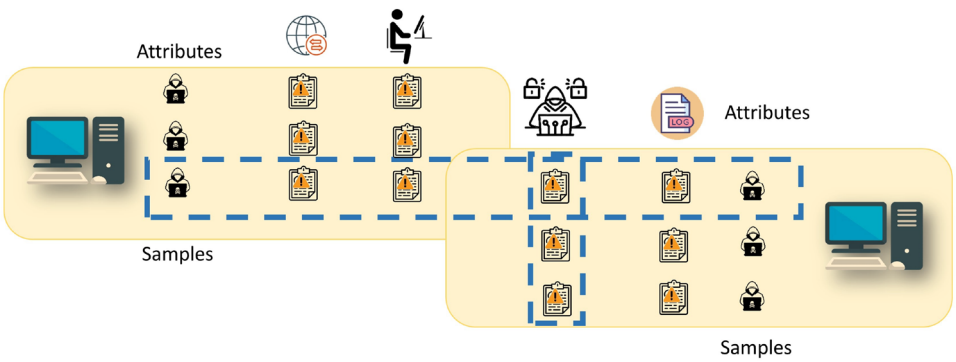

Fig. 9 Example of federated transfer learning

4.1 Common datasets

4.1.1 Kyoto

The dataset was created by Kyoto University between 2006 and 2009 and made available to researchers for intrusion detection [68]. It contains a total of 24 features, of which 14 statistical features were obtained from the KDD Cup99 dataset and 10 additional features for use in the detection system. The dataset was created with attacks captured using honeypots (Windows XP, Nepenthes, Solaris 8 for Intel, Others), darknet (darknet sensors), and other systems (web crawler, e-mail server, and Windows XP). Researchers conducted a thorough investigation of honeypots and darknet sensor data on a large number of real and virtual computers, and they set up various types of darknets, honeypots, and other security mechanisms across all these networks. The activities were carried out both outside and inside the University of Kyoto, and the data collected from the honeypots was used to create a comprehensive report.

As a distinctive property from other datasets, in addition to the normal and attack sessions, unknown datasets were also encountered during the observation period. To distinguish this in the dataset, the value of 1 for normal data, -1 for attack data, and -2 for unknown data is assigned in the "Label" feature. Therefore, only multiclassification can be done on this dataset. The advantages of the Kyoto dataset over other datasets are as follows

- It contains more realistic data.
- It does not include features that contain repetitive data.
- Contains data of existing real networks.

4.1.2 NSL-KDD

Although the KDD Cup99 dataset remains a popular choice [2], the rapid evolution of technology and networking demands new datasets for developing novel models [69]. The limitations of current datasets hinder the achievement of optimal performance by developed models. In response to the deficiencies in the KDD Cup99 dataset, researchers sought to construct a more contemporary and error-free intrusion dataset. This effort led to the creation of the NSL-KDD dataset, which, despite its own shortcomings in accurately representing real network issues, offers several advantages over the KDD Cup99 dataset:

- It does not comprise any records in the training dataset that are not necessary.
- In the testing dataset, there are no instances of duplicate data.
- Both the training and testing datasets include a proportionate and appropriate number of records in each category.

The NSL-KDD dataset includes 41 features, nine of which are foundational, while the remaining 32 arise from the KDD Cup99 and feature map. It retains critical records from the KDD Cup99 and feature map, categorized into four types:

- General features,
- Content features via TCP connections,
- Server-based traffic features with domain information,
- Time-dependent traffic features using "same server" and "same service" features.

Within the NSL-KDD dataset, attacks are categorized into four types: U2R, DoS, R2L, and Probe, each with further subcategories. Additionally, apart from attacks, there's a benign class available as well.

4.1.3 UNSW-NB15

It is a free dataset containing up-to-date real-world attacks that were generated by the IXIA PerfectStorm tool at ACCS' Cyber Range Lab in 2015 [70] and contains 2,540,044 data points with 49 features, including 9 types of attacks, "Backdoors, Fuzzers, Generic, Exploits, Port Scans/Spam, Reconnaissance, Worms, Shellcode, Dos" using 12 algorithms with specific tools. The UNSW-NB15 dataset consists of six groups, which are as follows:

- Additional Generated Features are the features of protocol and flow control.
- Basic Features are features that represent protocol connections.
- Content Features are features of TCP/IP and HTTP protocol connections.
- Flow Features are descriptive features between hosts.
- Labelled Features are the label information of each data. In this feature the data information is kept as it is attack 1 or normal 0.
- Time Features: These are time-dependent transaction features, such as the arrival time of network packets or the round-trip time of the TCP protocol.

4.1.4 CIC-IDS2017

Researchers have intensified their efforts to deliver effective intrusion detection systems (IDSs) as earlier IDSs have demonstrated high accuracy rates. However, only a limited number of IDS models and datasets have gained acceptance for real-world intrusion detection tasks. Addressing this gap, the Canadian Institute of Cybersecurity introduced the CICIDS2017 dataset, which encompasses a wide range

of attack scenarios and fulfills the requirements for real-world intrusion detection [71]. This dataset analyzes network traffic using time labels, protocols, source and destination IP addresses/ports, and various types of attacks. It incorporates contemporary attacks based on actual data from the real world. During the compilation of this dataset, attacks and normal operations were conducted throughout five days' worth of traffic data.

- First day; Benign
- Second day; Benign, SSH-Patator, FTP-Patator,
- Third day; Benign, Heartbleed, DoS Slowhttptest, DoS GoldenEye, DoS slow loris, DoS Hulk
- Forth day; Benign, WebAttack - XSS, WebAttack - Brute Force, Infiltration, WebAttack - SQL Injection,
- Fifth day; Benign, PortScan, Bot, DDoS

were collected and stored in CIC-IDS2017 dataset, which had been designed according to the following standards.

1. A network topology including various network elements and operating systems is used.
2. A real-time attack was made with the help of various tools from 12 different machines.
3. Data must be labeled.
4. System that communicates within the internal LAN with two different networks and Internet communication.
5. All traffic was seamlessly captured and saved to the server.
6. The presence of all existing common protocols in the network is ensured.
7. The most common intrusions were used.
8. Information on the hosts was recorded while the attacks were taking place.
9. It should have many features.
10. Metadata is also made available.

It's crucial to emphasize that an ideal Intrusion Detection System (IDS) should have the ability to detect all types of intrusions. For the creation of an optimal IDS, the complete traffic data collection spanning all days should be amalgamated into a unified dataset.

The dataset comprises 83 features, 3,119,345 data points, and 15 class labels (1 for normal and 14 for attack tags). Furthermore, there are 288,602 data points with missing class labels and 203 data points with incomplete information within the dataset.

4.1.5 CIDDS-001

Coburg Network Intrusion Detection Dataset (CIDDS-001) was produced in 2017 as an example of an IDS dataset, which is a labeled flow-based dataset, and was created for the assessment of AIDSs [72]. The dataset focuses on traffic data from two servers, which are OpenStack and External Servers. A virtual small business environment has been developed to create the CIDDS-001 dataset, which includes typical servers such as clients, e-mail, Web, etc. It provides unidirectional Netflow data and was developed using the Python programming language to simulate human behavior on simulated network virtual machines. For this reason, the features of each user have been adjusted to be realistic by using a configuration file. It contains 13 features for classification and one feature as a label or class.

4.1.6 CSE-CIC-IDS2018

For IDS implementations to develop better cybersecurity systems, researchers are looking for high-quality and up-to-date datasets. Therefore, the researchers at the Canadian Institute for Cybersecurity, who examined the deficiencies in previous datasets due to the need for a more detailed dataset, designed and presented the CSE-CIC-IDS2018 dataset, which includes current attacks and a high number of features from the CIC-IDS2017 dataset. In order to construct the CSE-CIC-IDS2018 dataset systematically, the "Concept of Profile" was used, which implies containing thorough descriptions of intrusions on applications, protocols, and low-level network elements when creating and maintaining the dataset. To produce data items on the network, these profiles may be utilized by agents or by humans, and they can be applied to several network protocols with a variety of topologies and protocols. Additionally, when generating the dataset, the standards that were utilized in the construction of the CIC-IDS2017 dataset were taken into consideration. It also took into account the following properties:

1. It does not contain too much duplicate data.
2. The amount of NaN data is very low.
3. It is presented ready for use in CSV format.
4. Two profiles have been created to generate the dataset which are Profile-B and Profile-M.

- *Profile-B* encapsulates the specific behaviors of users by utilizing different ML and statistical algorithms. It consists of the distribution of packet sizes of users, the number of packets per stream, the size of the stream, and the request time of a protocol.
- *Profile-M* distinctly identifies an attack scenario. In this way, it helps other users to interpret it later.

In addition, 5 different attack methods were used to create the dataset, except for the profiles. These are DoS, Brute Force, Web, Botnet, and Infiltration attacks. It contains 80 features, including label information, and was gathered

from numerous scenarios. These features include the number of packets and bytes sent, the time it takes to send each packet, the length of each packet computed independently in forward and backward order, and information on whether or not there is an attack.

A complete dataset is presented to the users with approximately 16 million records in PCAP and CSV format [52]. The data in the PCAP format is raw data without any feature extraction. Those who will examine the artificial intelligence (AI) methods should use the CSV format dataset, and those who will examine the feature extraction methods should use the PCAP formatted dataset.

Table 3 presents the key characteristics of the most favored and widely used datasets.

Improving existing datasets and developing new datasets adapted for FL-IDS is crucial for enhancing security and detection accuracy in decentralized environments. Existing IDS datasets often suffer from limitations like class imbalance, outdated attack patterns, and a lack of real-world heterogeneity. Addressing these issues through improved data diversity, realistic attack simulations, and integration of multimodal data sources such as host-based logs and endpoint security data will enhance their applicability to FL scenarios.

Beyond improving existing datasets, there is a need for newly designed metrics for FL-based IDS that ensure non-independent and identically distributed (non-IID) data partitions, reflecting real-world deployment challenges. Moreover, considering the significance of FL in these areas, datasets capturing security threats in edge computing and IoT environments are essential. A specialized adversarial robustness dataset simulating threats like model poisoning and Byzantine attacks would also be valuable for assessing FL-IDS resilience. Addressing these gaps will significantly contribute to the development of more effective and privacy-aware IDS solutions in federated environments.

4.2 Latest datasets

4.2.1 CIC-DDoS2019

CIC-DDoS2019 dataset includes modern Distributed Denial of Service (DDoS) attacks and simulates real-world network traffic. This dataset was created for the development and evaluation of new attack detection methods and provides a DDoS taxonomy covering different types of attacks.

DDoS attacks are attacks where the attacker floods the target system with excessive traffic while hiding their identity. The CIC-DDoS2019 dataset categorizes these attacks into two main categories:

- **Reflection-Based DDoS Attacks** The attacker increases traffic towards the target by utilizing third-party legitimate systems.
  - **TCP-based attacks:** MSSQL, SSDP
  - **UDP-based attacks:** CharGen, NTP, TFTP
  - **Attacks using both TCP and UDP:** DNS, LDAP, NETBIOS, SNMP
- **Exploitation-Based DDoS Attacks** The attacker depletes the target system's resources using TCP and UDP protocols.
  - **TCP-based attacks:** SYN Flood
  - **UDP-based attacks:** UDP Flood, UDP-Lag

The dataset includes labelled flow data CSV and raw PCAP files for network traffic analysis. While creating the dataset, it is aimed to create realistic network traffic. For this purpose, 25 user behaviours consisting of Ubuntu and Windows systems were simulated and natural background traffic was created with protocols such as HTTP, HTTPS, FTP, SSH and e-mail protocols.

CIC-DDoS2019 is one of the most comprehensive datasets containing up-to-date and realistic DDoS attacks. It serves as a crucial resource for network security

Table 3 IDS datasets

Dataset	Year	Feature number	Number of data	Class number	Labelled	Balanced	Documented	Public format (csv, pcap, flow)
Kyoto	2006	24			✓	x	✓	-
NSL-KDD	2009	42	4,898,431	5	✓	x	✓	Csv
UNSW-NB15	2015	49	2M packets and flow	10	✓	x	✓	csv
IDS2017	2017	80	2,829,464	7	✓	x	✓	Pcap, csv
CIDDS-001	2017	14	35M flow	6	✓	x	✓	Flow, csv
IDS2018	2018	80	15,822,236	7	✓	x	✓	Pcap, csv

researchers, machine learning experts, and cybersecurity professionals in attack detection and analysis [73, 74].

4.2.2 CIC-MalMem-2022

CIC-MalMem-2022 is a comprehensive and balanced dataset designed to evaluate the detection of obfuscated malware through memory analysis. Simulating real-world scenarios, this dataset specifically addresses challenges encountered in memory-based malware analysis. By enabling the examination of obfuscated malware that operates in memory and is difficult to detect, the dataset provides extensive coverage of various types of malware. It includes a diverse range of malware families categorized into Spyware, Ransomware, and Trojan Horse, offering a valuable resource for assessing and improving memory-based malware detection methods.

The dataset consists of memory dumps obtained using debug mode, ensuring an accurate representation of real-world attack conditions. It maintains a balanced distribution, comprising 50% benign (non-malicious) and 50% malicious memory dumps. With a total of 58,596 records, the dataset contains 29,298 benign and 29,298 malicious samples. This balanced distribution minimizes bias in malware detection system evaluations, ensuring more reliable performance assessments. By encompassing three major malware categories, CIC-MalMem-2022 serves as a crucial benchmark for researchers and security professionals in developing and validating memory-based threat detection systems [75, 76].

5 Federated learning based IDSs

FL, a distributed ML method, has been extensively explored in the literature. For instance, this method is particularly valuable in industries such as IoT [77] and Blockchain [78, 79], mobile network [80, 81], sales [82], finance [83], transportation [84], and healthcare [85, 86] where data privacy, security, and intellectual property concerns prohibit the central aggregation of data [87]. For additional applications in FL, we can refer to [88, 89].

The usage of FL in IDS can mitigate data privacy concerns and minimize the risk of data breaches during transmission. Further, federated IDS have shown promise in improving detection accuracy by leveraging diverse data sources without centralizing the data. For instance, in [90], a multi-class IDS for software-defined networking (SDN) utilizing FL for privacy preservation was suggested. To perform the suggested model, they used a cyber security-based dataset (Edge-IIoTset) which was created for IDS. Further, the model was examined utilizing various evaluation metrics. According to the findings, a high score of around 98.6% was obtained. When they compared the existing literature, they arrived that their suggested model ensures the confidentiality of training data by employing FL and addresses the issue of inaccessible training data resulting from privacy concerns. On the other hand, in order to detect threats among 15 distinct attack types in the Edge-IIoT dataset, Benameur et al. [91] employ CNN-LSTM networks in conjunction with knowledge distillation (KD) and achieve an accuracy of 84.5%. The research paper by Song and Ma [92] proposed a novel approach for intrusion detection in the context of edge-enabled IoT using a federated attention neural network. In a recent paper, an FL mechanism in the context of a privacy-enhanced edge intelligence model Beyond 5G networks was defined. Afterward, an Artificial Immune IDS was developed to oversee and categorize nodes exhibiting anomalies within the edge network, ensuring seamless and secure data transmission according to the necessary standards. The model, analyzed on the datasets CIFAR-10 and KDD-99, outperformed the existing edge security models [53].

In [93], a proposal for an effective IDS utilizing a GAN network with a distributed FL scheme (FEDGAN-IDS) was suggested. The performance of the scheme in regards to accuracy, recall, loss, convergence rate, precision, F1-score, and AUC score was evaluated. By making use of KDD99, NSL-KDD, and UNSW-NB15 datasets, they performed all experiments. Additionally, the outcomes were summarized for both binary and multiclass classification of the scheme. An accuracy of 99% is achieved for binary classification, while 98% accuracy is attained for multiclass classification.

In [94], an ensemble-based method using XGBoost is proposed to improve the detection of network attacks and tested on the KDDCup99 dataset. The proposed method achieved a high success rate with an accuracy of 99.95% and demonstrated a more effective IDS model for detecting unknown attacks. Similarly, the study proposes a novel ensemble-based IDS method using an optimized CatBoost classifier to enhance the security of 5G-enabled embedded and cyber-physical networks. Experiments conducted on the KDDCup99 dataset demonstrate that the proposed approach achieves an accuracy of 99.96%. This research provides valuable insights into strengthening intrusion detection capabilities in modern distributed networks [54]. Moreover, Bhati and Khari [95] proposes an ensemble IDS method based on the voting ensemble technique, using two algorithms: Support Vector Machine (SVC) and ExtraTree. The method was tested on the KDDCup99 dataset and achieved an accuracy of 99.90%. The voting methodology was found to outperform other ensemble techniques by combining both similar and different classifiers to create a stronger model. However, our work surpasses these references by leveraging FL to enhance intrusion detection

without requiring centralized data storage. Unlike traditional ensemble-based methods relying on XGBoost, CatBoost, or voting classifiers, which necessitate direct access to extensive datasets like KDDCup99, our approach preserves data privacy and mitigates risks associated with data breaches. Additionally, FL enables continuous learning across distributed edge devices, ensuring adaptability to evolving cyber threats while maintaining high detection accuracy. Thus, our method not only matches or exceeds the reported accuracy levels but also offers superior scalability, security, and real-world applicability in decentralized network environments.

The FL-enabled IDS approach is a significant subject. Hence, Zhang et al. [96] offered an anomaly-based IDS for Industrial IoT (IIoT) networks utilizing FL. They dealt with training local models with non-IID. They implemented an instance-based transfer learning method utilizing ensemble techniques and introduced a new aggregation algorithm related to a weighted voting mechanism. They used the datasets CICIDS2017 and CICIDS2018. To compare a centralized model in multiclass, they evaluated cloud-based AdaBoost and RF. Their approach demonstrated superior detection performance. In a paper, they aimed to present a thorough assessment of the usage of FL for IDS in IoT based on non-independent and identically distributed (non-IID) data. By utilizing the CIC-ToN-IoT dataset, three scenarios were created and Precision, Recall, F1-score, and FPR metrics were explained. Further, by utilizing the dataset, the FedAvg and Fed+ aggregation functions were evaluated [97]. In [98], for ensuring the security of wireless edge networks, an FL intrusion detection algorithm (FedAGRU) was suggested. The difference from traditional centralized learning methods was that it did not necessitate transmitting the original data to a central server. In this study, three real network datasets (KDDCUP99, CICIDS2017, WSN-DS) to analyze the suggested method not only on IID but also on non-IID data samples were utilized. The simulations determined that FedAGRU demonstrated superior accuracy, robustness, and efficiency. In [99], the application of FL to boost malware detection in IoT networks was explored. Unlike traditional malware detection methods, a decentralized approach where IoT devices collaboratively train a shared model locally, guaranteeing that sensitive data remains on the devices was proposed. Therefore, the method improved the detection accuracy while preserving user privacy. In a scenario within a B5G context, the necessity arose to detect cyberattacks impacting IoT devices, manage sensitive data, handle Non-IID data, and engage with untrusted stakeholders or clients. The findings demonstrated the effectiveness of this approach using real-world IoT datasets, highlighting its potential to provide a scalable and secure solution for malware detection in IoT environments. Similarly, in [100], a novel Fed-IDS was developed. To test the proposed model under IID settings, IoT-ID-20, Edge-IIoT, and 5G-NIDD datasets were considered. Moreover, the WUSTL-IIoT-2021 dataset was considered for evaluation of the model under non-IID data settings. When the findings were compared to FedAvg, they emphasized a marked enhancement in the performance of Fed-IDS.

By incorporating boosting approaches into a federated learning framework, this research makes a substantial contribution to the area with the goal of improving the detection accuracy of cyberattacks in IoT systems. By doing this, it tackles important issues with computational efficiency and data privacy, providing a scalable solution that can be adjusted to the varied and resource-constrained nature of consumer IoT devices. The suggested approach's robustness and practical application are further demonstrated by the use of real-world datasets and thorough evaluation measures [101]. Since the existing models in the literature rely on outdated datasets like NSL-KDD, which fail to capture contemporary IoMT attack vectors and limit their real-world applicability, [86] leveraged a publicly available modern dataset that comprehensively covers a wide range of contemporary IoMT-based attacks.

In 2023, a federate-based IDS for IoT environments was presented. To analyze the effectiveness, the dataset CSE-CIC-IDS2018 was utilized. In contrast to other studies in the literature, this research opted to employ six FL algorithms (FedAvg, FedOpt, FedProx, FedAdam, FedAdagrad, and FedYogi) for training instead of FedAvg. The experiments of evaluation metrics revealed that FL demonstrated superior performance, particularly when dealing with small local datasets, which is often the scenario encountered with IoT devices [102].

A privacy-preserving FL model for IoT intrusion detection was delivered in [55]. To test the effectiveness, they performed comprehensive experiments on the NSL-KDD dataset. The findings indicated that the system outperformed the distributed unaggregated on-device trained models.

In [103], by utilizing a combination of CNN and GRU deep learning techniques, a novel federated deep learning scheme, named DeepFed, for industrial cyber-physical systems (CPSs) was created. The findings demonstrated the effectiveness of DeepFed using real-world industrial datasets, highlighting its potential to bolster cybersecurity in industrial CPS environments. Similarly, the writers of [104] used a method called clustered federated learning, which helps the system learn from different sources of data while keeping information private. The study's experiments showed that this method is effective in identifying and addressing security threats, which is a significant step forward in industrial cybersecurity.Furthermore, Khan et al. [105] presents a unique IDS paradigm known as

federated-simple recurrent units (SRUs) to improve the privacy and the security of IoT-based industrial control systems (ICSs). In order to reduce computing expenses and address the gradient vanishing issue in recurrent networks, the suggested model makes use of an enhanced basic recurrent units design. Experiments are carried out on a real-scale gas pipeline network dataset, a representative example of ICS networks, ensuring a robust and practical evaluation. Numerous modern attack types, including denial-of-service assaults, reconnaissance, sophisticated malicious response injections, and command injection attacks, are included in this dataset. The suggested federated-IDS model has performed better than other cutting-edge benchmark techniques when compared to baseline methods. A clearer comparison would be possible by specifically pointing out any shortcomings in the detection strategy, model effectiveness, or ability to adjust to various attack scenarios. Hence, this paper illustrates how filling these gaps advances the field by increasing robustness against new threats, decreasing processing costs, or boosting accuracy.

In 2023, a new network threat detection system, GöwFed, consisting of the combination of Gower Dissimilarity matrices and Federated averaging has been developed. It uses FL to create a robust IDS that works across multiple decentralized networks. With advanced data encryption and secure model updates, GöwFed aims to be more secure and privacy-preserving than traditional ones. Developed in response to evolving cybersecurity threats, GöwFed offers an adaptive and secure IDS framework. This system contributes to ongoing research in federated learning, providing a practical solution to the limitations of existing IDS technologies [106].

In the research, Al-Marri et al. [9] suggested a novel method that employs mimic learning within FL framework to address the issue of reverse engineering in FL, thereby reducing the risk of compromising users' privacy in IDS. Unlike traditional centralized IDS, this approach trained models locally on user devices, keeping sensitive data secure and mitigating reverse engineering risks. They employed the NSL-KDD dataset to evaluate the suggested method in comparison with current solutions. Techniques such as secure aggregation and differential privacy were also employed to further ensure data privacy and robustness, making this a promising solution for modern IDS.

A hybrid approach consisting of a combination of CNN and Bidirectional Long-Term Short Memory (BiLSTM) was introduced for efficiency. Similar to other studies in the literature, the objective was to assess the efficiency within the framework of FL for IoT. To achieve this, CICIDS2017 and Edge-IIoTset real-word datasets were utilized for evaluation [107].

Amiri-Zarandi et al. [108] established the Social Intrusion Detection System (SIDS), an FL-based scheme for IoT intrusion detection utilizing the Social Internet of Things (SIoT). The presented method utilized the social connections between objects within a system to offer a collaborative mechanism that preserved privacy while detecting intrusions in IoT environments.

To enhance learning efficiency across different attack categories, a novel FL-based intrusion detection method, named MV-FLID, has been developed. This approach leverages multi-view ensemble learning to analyze diverse perspectives of IoT network data. By training in a distributed manner, MV-FLID effectively identifies, categorizes, and safeguards against various attacks [109].

Some researchers have used sophisticated ML models, including autoencoders, to extract significant aspects from client data in order to increase its efficacy. Pope et al. [110], for instance, showed how FedAvg and autoencoders might enhance anomaly identification in IoT networks. An innovative unsupervised DP method utilizing autoencoders for learning from unlabeled data has been introduced. By leveraging FL, this method trains on the unlabeled datasets of edge devices while preserving user privacy. Yadav et al. [111] conducted experiments using the CICIDS 2017 dataset, achieving an impressive 97.75% accuracy in detecting intrusions.

To protect data privacy, an FL-IDS was considered in terms of transportation IoT [112]. The real-time effectiveness of the system was assessed by making use of datasets NSL-KDD and Car-Hacking. When compared to traditional approaches, the system was found to be successful in terms of accuracy and loss.

Designed to facilitate secure and privacy-preserving collaborative IDSs in IoT, Sarhan et al. [113] established a hierarchical blockchain-based FL framework: HBFL. Using a key IoT dataset to evaluate its performance, the study demonstrated that HBFL is a robust and high-performance ML-based IDS. This framework effectively protects and maintains the integrity of IoT networks.

In another study conducted by Sarhan et al. [114], a Cyber Threat Intelligence sharing model utilizing FL was introduced to enable multiple organizations to unite in designing, training, and evaluating a robust ML-based network IDS. To evaluate the model, NF-UNSW-NB15-v2 and NF-BoT-IoT-v2 datasets were analyzed. Further, a centralized model and a localized model were considered as scenarios in the evaluation process. The outcomes demonstrated the efficiency and effectiveness of the model.

The advancement of smart grid technologies have introduced security vulnerabilities in critical infrastructures such as smart meters. Several methods have been presented to enhance security in such systems. For instance, Mirzaee et al. [115] presented a Federated IDS (FIDS) method,

focusing on intrusion detection in 5G smart metering networks.

Communication delay is the primary performance bottleneck in the entire learning framework. Hence, to address the communication delay limitation of FL, an IDS based on FL, Dynamic Weighted Aggregation Federated Learning (DAFL), was established by Li et al. [116]. This approach improved intrusion detection while reducing communication overhead.

Unlike other studies in the literature, Neto et al. [117] examined the statistical obstacles in FL. They developed a selection approach named Federated Score-Based Selection (FedSBS). It involves two steps: choosing participants and using an aggregation algorithm that taps into global momentum. The suggested method was assessed by utilizing the CICIDS2017 dataset. In order to test the method, they used FedDyn, FedAGM, and SlowMo algorithms. Afterward, they assessed the accuracy score, precision score, recall, and F1 score for the test validation sets. By achieving an accuracy of 92% and 82% F1-Score in scenarios where there were no malicious participants, their approach showed superiority.

In recent years, other surveys on FL have been conducted. For example, Agrawal et al. [118] summarized the usage of FL in IDS highlighting the security, privacy, and reliability dimensions. Similarly, in the survey written by Lavaur et al. [119], a fascinating taxonomy concerning the coverage of FL-IDS systems in aspects such as privacy, trust, cybersecurity awareness, and related factors was presented. The recent practical applications of FL were analyzed by Zhang et al. [120]. According to this survey on FL, some applications related to service recommendation and wireless communication were summarized. A current survey in Healthcare Metaverse highlighted that it utilizes FL, allowing individual hospitals to collaborate and learn from shared predictive models. The advantages of FL in the Healthcare Metaverse are examined. Following this, they explored various applications, such as medical diagnosis, infectious disease management, and patient monitoring. To the end, they emphasized the major challenges and potential solutions for implementing FL in the healthcare Metaverse [121]. In order to solve privacy, security, and reputation issues in Healthcare systems, Khan et al. [122] introduced a novel security model for the collecting and transmission of biomedical data. They developed an enhanced SRU network to address fading gradient problems and increase learning by lowering computing costs, and they presented a threat-vector database based on the dynamic behaviors of smart healthcare systems. When compared to current techniques, the model reduces communication overhead and improves feature extraction. However, one drawback that restricts scalability is its reliance on huge datasets. Finally, the recently published paper [123] provided a detailed review and taxonomy of existing FL-based IDS approaches, identified current limitations, and suggested future research directions, highlighting the potential of FL to enhance threat detection while preserving privacy and reducing communication overhead.

Table 4 demonstrates that prior studies have shown the effectiveness of FL for IDSs under different scenarios.

6 Discussion

Federated learning (FL) has great potential to enhance intrusion detection systems (IDS) by addressing confidentiality and integrity concerns. By eliminating the need for centralized data collection and integrating data from various sources, FL can improve detection accuracy while safeguarding privacy. Techniques like secure aggregation and differential privacy, used in conjunction with FL, have proven to be effective in industrial cyber-physical systems and healthcare applications. Specifically, innovations like FedSBS and FEDGAN-IDS achieve high accuracy and F1 scores through aggregation methods like FedAvg and Fed+.These results indicate that FL can successfully learn from decentralized data sources without requiring centralized data storage, ensuring both data privacy and model robustness.

When compared to traditional centralized IDS models, the FL-based approach demonstrated superior scalability and resilience in handling large volumes of distributed data. Traditional methods struggle with privacy concerns and data centralization, whereas FL circumvents these issues by keeping sensitive data local. Moreover, FL's ability to work with non-IID data further enhances its applicability in real-world scenarios, where data distributions often vary across devices and networks.

However, several challenges and research gaps must be addressed before FL can be widely adopted. The variety and quality of data used in FL play a crucial role in the results. Inconsistent data from different devices can negatively impact model performance, underscoring the importance of data preprocessing and standardization. The heterogeneous nature of IoT devices presents additional challenges, and optimized FL methods for these structures are needed. FL claims to be secure and private, however further research is required to address vulnerabilities during model updates and data transfers, especially concerning resilience against malicious actors. Additionally, computational costs, training processes, and real-time application performance must be improved to make FL more efficient and effective.

Addressing the challenges in FL-based IDS requires concrete strategies to enhance model efficiency, scalability,

Table 4 Overview of previous research on intrusion detection using Federated Learning

Year	Reference	Method	Aggregation method	Accuracy	Precision	Recall	F1-Score	Classifier	Dataset	Application
2024	[79]	Blockchain based FL	FedAvg	0.99	_	_	_	CNN	AWID2	IoT
2024	[83]	FL for SCF	FedAvg	0.99 0.99	0.99 0.99	0.99 0.99	_	ANN, IDCNN	Original dataset (CXX)	Supply chain financing
2024	[100]	FIDS	FedAvg					DNN	NSLKDD, Edge-IIoT, ToN IoT	NGNs
2024	[90]	FIDS	FedAvg	0.98	0.98	0.98	0.98	ANN	Edge-IIoT	SDN
2024	[112]	FIDS	FedAvg, FedYogi	0.96 0.93 0.97 0.99	_	_	_	LR, CNN	NSL-KDD, Car-Hacking	Transportation IoT
2024	[107]	FIDS	FedAvg	0.99	_	_	_	CNN-BiLTSM, DNN	CICIDS2017, Edge-IIoT	Zero trust intrusion detection for IoT
2024	[117]	FedSBS	FedAGM, FedDyn, SlowMo, FedAvg	0.92	0.80	_	0.82	_	CICIDS2017	IDSs
2024	[92]	Edge-Detect	FedACNN	0.99	0.90	0.95	0.92	CNN	UNSW2015	IoT
2023	[102]	FIDS	FedAvg, FedProx, FedOpt, FedAdam, FedYogi,FedAdagrad					CNN	CSE-CIC-IDS2018	IoT
2023	[114]	HBFL	FedAvg	0.91 0.93 0.88 0.92	_	_	0.90 0.93 0.88 0.92	DNN, LTSM	NF-UNSW-NB15- v2, NF-BoT-IoT-v2	IoT
2022	[105]	Federated-SRUs	_	0.99	0.99	0.99	0.99	_	a real-scale gas pipeline network	IoT
2024	[122]	Threat-vector database, improved SRU	_	0.99	0.99	0.98	0.99	RNN	ToN_IoT	IoMT
2023	[121]	FL-enabled healthcare Metaverse	_	_	_	_	_	_	_	Healthcare Metaverse
2023	[104]	CFL	FedAvg	0.99	_	_	_	_	_	CPS
2023	[82]	HFL	FedAvg	_	_	_	_	_	A real-world dataset	EVs
2023	[106]	GöwFed	FedAvg	0.93	0.96	0.88	_	(AM)MLP	TON_IOT	NIDS
2023	[116]	DAFL	FedAvg	0.94	0.93	0.92	0.93	CNN	CSE-CIC-IDS2018	NIDS
2023	[108]	SIDS	FedAvg	0.77 0.79	_	_	0.85 0.87	GAN	UNSW-NB15	SIoT
2022	[77]	Edge-IIoTset	FedAvg					DNN	Edge-IIoTset	Cybersecurity for IoT and IIoT
2024	[119]	FIDS	_	_	_	_	_	SLR	_	Intrusion detection and mitigation
2022	[99]	FL for malware detection	FedAvg					_	N-BaIoT	IoT malware detection

Table 4 (continued)

Year	Reference	Method	Aggregation method	Accuracy	Precision	Recall	F1-Score	Classifier	Dataset	Application
2022	[93]	FEDGAN-IDS	FEDGAN-IDS	0.99,	0.99 0.97, 0.99	0.98, 0.96, 0.99	_	ACGAN, CNN	NSL-KDD, KDD-CUPP99 (KDD99), UNSW-NB15.	IDS
2022	[113]	ML-based NIDS	FedAvg	0.99	_	_	_	DNN, LTSM	NF-BoT-IoT-v2	NIDS
2022	[89]	FELIDS	FedAvg	0.93 0.94, 0.94	_	_	_	DNN, CNN, RNN	CSE-CIC-IDS2018, MQTT, InSDN	Agri-IoT
2021	[109]	MV-FLID	FedAvg	_	_	_	_	_	MQTT	IoT
2021	[111]	FDAGMM	Federated Learning SGD	0.97	_	_	_	ANN	CICIDS 2017	IoT
2021	[81]	FedPacket	Federated SVM	_	_	_	_	RNN	NoMoAds, AntShield, in-house	Mobile computing
2021	[53]	Security and privacy-aware IDS	k-means algorithm	0.92	_	_	_	_	CIFAR-10, KDD-99	Artificial IDS
2021	[78]	two-stage IDS	FedAvg	_	_	_	_	CNN	_	Vehicular edge computing
2021	[115]	FIDS	FedAvg	0.99	0.99	0.99	0.99	FDNN	NSL-KDD	5G smart metering network
2020	[103]	DeepFed	AES	0.99	0.98	0.97	0.98	CNN+GRU	A real Industrial CPS	Industrial CPS
2020	[98]	FedAGRU	FedAvg	0.99	_	_	0.97	GRU	KDD99 WSN-DS	Wireless edge networks
2020	[55]	Privacy-preserving FL	FedAvg	0.77	_	_	_	_	_	IoT
2020	[9]	Federated mimic learning	FedAvg	0.98	0.99	0.99	0.99	MLP	NSL-KDD	IDS
2019	[80]	Privacy-aware service placement based FL	PSP	_	_	_	_	_	_	Mobile edge computing
2022	[99]	Coordinate-wise median	FedAvg					MLP, AE	N-Balot	Malware Detection
2024	[100]	Intrusion detection	FedAveg	0.94	0.88	0.96	0.91	DNN	IoT-ID-20, Edge-IIoTset, 5G-NIDD	IDS for Next-Generation Networks

and robustness in real-world deployments. One effective approach is adopting adaptive aggregation methods, such as instance-based transfer learning and weighted voting mechanisms, which improve model accuracy in non-IID data scenarios, as demonstrated in anomaly-based IDS for Industrial IoT (IIoT) networks. Additionally, lightweight FL models adapted for resource-constrained environments, such as Federated-Simple Recurrent Units (FedSRU) and clustered federated learning (CFL), have been proposed to reduce computational overhead while maintaining high detection performance.

To enhance privacy, secure model update techniques like differential privacy and homomorphic encryption can mitigate risks associated with malicious participants in FL networks. Moreover, knowledge distillation (KD) techniques, such as those used in CNN-LSTM networks, allow edge devices to benefit from complex global models without excessive local computation, ensuring scalability in large-scale IoT environments. Finally, communication-efficient FL algorithms, such as Dynamic Weighted Aggregation Federated Learning (DAFL), help reduce network latency and training costs, making FL-based IDS more feasible for real-time applications.

Implementing these strategies will enhance the effectiveness of FL-based IDS, ensuring they remain robust, scalable, and privacy-preserving across various cybersecurity contexts.

7 Conclusion

In this paper, we have presented a comprehensive overview of Federated Learning (FL) based Intrusion Detection Systems (IDSs), highlighting their importance in addressing the escalating security and privacy issues with data, particularly in distributed environments like IoT, healthcare, and industrial networks. FL offers a promising solution for enhancing IDS performance by enabling collaborative learning across multiple decentralized devices while ensuring that sensitive data remains local and protected.

We have examined various approaches and applications of FL in IDS, noting significant advancements such as the incorporation of deep learning techniques, Generative Adversarial Networks (GANs), and hybrid models for improving detection accuracy and robustness. Additionally, FL's ability to manage non-independent and identically distributed (non-IID) data, a common challenge in real-world scenarios, was emphasized, along with the various privacy-preserving techniques integrated into these models.

Furthermore, we have discussed several case studies and research findings that demonstrate the effectiveness of FL-based IDS in diverse fields. These include edge networks, healthcare systems, industrial control systems, and IoT environments, where the proposed methods have shown superior performance in detecting intrusions, reducing computational overhead, and ensuring secure data transmission. The experiments conducted on various datasets, including Edge-IIoTset, CICIDS2017, NSL-KDD, KDD99, and UNSW-NB15, and employing FL algorithms like FedAvg, Fed0pt, FedProx, FedAdam, FedAdagrad, and FedYogi, demonstrate the ability of FL to achieve high accuracy rates, with some methods reaching up to 99.27%. These results showcase the potential of FL to improve the performance of IDSs while maintaining robust privacy protection.

Despite the promising results, there are still open challenges that need to be addressed, such as managing untrusted participants, enhancing the scalability of FL models, and maximizing the trade-off between computing costs and communication. Future studies should focus on creating FL algorithms, exploring advanced privacy-preserving techniques, and addressing the challenges of real-world implementation.

In conclusion, FL-based IDS represents a significant advancement in cybersecurity, providing an efficient and privacy-conscious approach to protecting critical systems and data from evolving threats. Maintaining the security of our increasingly interconnected world will require these systems to be continuously improved.

Author Contributions All authors contributed equally to the conception, design, and writing of this manuscript.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability All data generated or analyzed during this study are either included in this published article or available from the corresponding author upon reasonable request.

Declarations

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of interest The authors declare that they have no conflict of interest, competing financial interests, or personal relationships that could have appeared to influence the work reported in this paper.

Ethical approval and consent to participate This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication All authors have read and approved the final version of the manuscript and agree to its submission to the journal.

Materials availability No specific materials were used in this study, and there are no restrictions on the availability of materials.

References

1. Base, R., Mell, P.: Special publication on intrusion detection systems. NIST Infidel Inc, Scotts Valley, CA (2001)
2. Karatas, G., Demir, O., Sahingoz, O.K.: Increasing the performance of machine learning-based idss on an imbalanced and up-to-date dataset. IEEE Access **8**, 32150–32162 (2020)
3. Miranda-García, A., Rego, A.Z., Pastor-López, I., Sanz, B., Tellaeche, A., Gaviria, J., Bringas, P.G.: Deep learning applications on cybersecurity: A practical approach. Neurocomputing **563**, 126904 (2024)
4. Rodríguez, G.E., Torres, J.G., Flores, P., Benavides, D.E.: Cross-site scripting (xss) attacks and mitigation: A survey. Comput. Netw. **166**, 106960 (2020)
5. Rahman, A., Debnath, T., Kundu, D., Khan, M.S.I., Aishi, A.A., Sazzad, S., Sayduzzaman, M., Band, S.S.: Machine learning and deep learning-based approach in smart healthcare: Recent advances, applications, challenges and opportunities. AIMS Public Health **11**(1), 58–109 (2024)
6. Boussina, A., Shashikumar, S.P., Malhotra, A., Owens, R.L., El-Kareh, R., Longhurst, C.A., Quintero, K., Donahue, A., Chan, T.C., Nemati, S., *et al.*: Impact of a deep learning sepsis prediction model on quality of care and survival. npj Digital Medicine **7**(1), 14 (2024)
7. Buyuktanir, B., Yildiz, K., Ulku, E.E., Buyuktanir, T.: du-cba: Veriden habersiz ve artırımlı sınıflandırmaya dayalı birliktelik kuralları çıkarma mimarisi. Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi **38**(3), 1919–1930 (2023)
8. Nguyen, T.D., Marchal, S., Miettinen, M., Fereidooni, H., Asokan, N., Sadeghi, A.-R.: Dïot: A federated self-learning anomaly detection system for iot. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), pp. 756–767 (2019). IEEE
9. Al-Marri, N.A.A.-A., Ciftler, B.S., Abdallah, M.M.: Federated mimic learning for privacy preserving intrusion detection. In: 2020 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), pp. 1–6 (2021). IEEE
10. Huong, T.T., Bac, T.P., Long, D.M., Thang, B.D., Binh, N.T., Luong, T.D., Phuc, T.K.: Lockedge: Low-complexity cyberattack detection in iot edge computing. IEEE Access **9**, 29696–29710 (2021)
11. Gupta, D., Rani, R.: Big data framework for zero-day malware detection. Cybern. Syst. **49**(2), 103–121 (2018)
12. Ye, Y., Li, T., Jiang, Q., Han, Z., Wan, L.: Intelligent file scoring system for malware detection from the gray list. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1385–1394 (2009)
13. Pachhala, N., Jothilakshmi, S., Battula, B.P.: A comprehensive survey on identification of malware types and malware classification using machine learning techniques. In: 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), pp. 1207–1214 (2021). IEEE
14. Gandotra, E., Bansal, D., Sofat, S.: Malware analysis and classification: A survey. Journal of Information Security **2014** (2014)
15. Anli, Y.A., Ciplak, Z., Sakaliuzun, M., Izgu, S.Z., Yildiz, K.: Ddos detection in electric vehicle charging stations: A deep learning perspective via cicev2023 dataset. Internet Things **28**, 101343 (2024)
16. Conti, M., Dragoni, N., Lesyk, V.: A survey of man in the middle attacks. IEEE Commun. Surv. Tutorials **18**(3), 2027–2051 (2016)
17. Glăvan, D., Răcuciu, C., Moinescu, R., Eftimie, S.: Sniffing attacks on computer networks. Scientific Bulletin" Mircea cel Batran" Naval Academy **23**(1), 202–207 (2020)
18. Anley, C.: Advanced sql injection in sql server applications. Unknown Journal (2002)
19. Robledo, H.F.G.: Types of hosts on a remote file inclusion (rfi) botnet. In: 2008 Electronics, Robotics and Automotive Mechanics Conference (CERMA'08), pp. 105–109 (2008). IEEE
20. Khonji, M., Iraqi, Y., Jones, A.: Phishing detection: a literature survey. IEEE Commun. Surv. Tutorials **15**(4), 2091–2121 (2013)
21. Kamruzzaman, A., Thakur, K., Ismat, S., Ali, M.L., Huang, K., Thakur, H.N.: Social engineering incidents and preventions. In: 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0494–0498 (2023). IEEE
22. Alshamrani, A., Myneni, S., Chowdhary, A., Huang, D.: A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. IEEE Commun. Surv. Tutorials **21**(2), 1851–1877 (2019)
23. Caputo, D.D., Pfleeger, S.L., Freeman, J.D., Johnson, M.E.: Going spear phishing: Exploring embedded training and awareness. IEEE Secur. Privacy **12**(1), 28–38 (2013)
24. Silva, S.S., Silva, R.M., Pinto, R.C., Salles, R.M.: Botnets: A survey. Comput. Netw. **57**(2), 378–403 (2013)
25. Stafford, T.F., Urbaczewski, A.: Spyware: The ghost in the machine. Commun. Assoc. Inf. Syst. **14**(1), 49 (2004)
26. Cheng, L., Liu, F., Yao, D.: Enterprise data breach: causes, challenges, prevention, and future directions. Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. **7**(5), 1211 (2017)
27. Wagner, J., Rasin, A., Heart, K., Malik, T., Furst, J., Grier, J.: Detecting database file tampering through page carving. In: 21st International Conference on Extending Database Technology (2018)
28. Becker, G.T.: Robust fuzzy extractors and helper data manipulation attacks revisited: Theory versus practice. IEEE Trans. Dependable Secure Comput. **16**(5), 783–795 (2017)
29. Beaman, C., Barkworth, A., Akande, T.D., Hakak, S., Khan, M.K.: Ransomware: Recent advances, analysis, challenges and future research directions. Comput. Secur **111**, 102490 (2021)
30. Karatas, G., Demir, O., Sahingoz, O.K.: Deep learning in intrusion detection systems. In: 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), pp. 113–116 (2018). IEEE
31. Yadav, J., Dhull, D., et al.: Comparison between hids and nids. Int J Manag IT Eng **3**(2), 316–324 (2013)
32. Atasever, S., Özçelık, İ., Sağiroğlu, Ş.: An overview of machine learning based approaches in ddos detection. In: 2020 28th

Signal Processing and Communications Applications Conference (SIU), pp. 1–4 (2020). IEEE
33. Verwoerd, T., Hunt, R.: Intrusion detection techniques and approaches. Comput. Commun. **25**(15), 1356–1365 (2002)
34. Bhati, N.S., Khari, M., García-Díaz, V., Verdú, E.: A review on intrusion detection systems and techniques. Int. J. Uncertain. Fuzziness Knowlege-Based Syst. **28**(Supp02), 65–91 (2020)
35. Hubballi, N., Suryanarayanan, V.: False alarm minimization techniques in signature-based intrusion detection systems: A survey. Comput. Commun. **49**, 1–17 (2014)
36. Jyothsna, V., Prasad, R., Prasad, K.M.: A review of anomaly-based intrusion detection systems. Int. J. Comput. Appl. **28**(7), 26–35 (2011)
37. Galal, H.S., Mahdy, Y.B., Atiea, M.A.: Behavior-based features model for malware detection. J. Comput. Virol. Hacking Tech. **12**, 59–67 (2016)
38. Bazrafshan, Z., Hashemi, H., Fard, S.M.H., Hamzeh, A.: A survey on heuristic malware detection techniques. In: The 5th Conference on Information and Knowledge Technology, pp. 113–120 (2013). IEEE
39. Singh, J., Singh, J.: A survey on machine learning-based malware detection in executable files. J. Syst. Archit. **112**, 101861 (2021)
40. Lee, J., Bagheri, B., Kao, H.-A.: A cyber-physical systems architecture for industry 40-based manufacturing systems. Manuf. Lett. **3**, 18–23 (2015)
41. Atalay, M., Çelik, E.: Büyük veri analizinde yapay zekâ ve makine öğrenmesi uygulamalari-artificial intelligence and machine learning applications in big data analysis. Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi **9**(22), 155–172 (2017)
42. Zhou, K., Liu, T., Zhou, L.: Industry 4.0: Towards future industrial opportunities and challenges. In: 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 2147–2152 (2015). IEEE
43. Gökçay, B., Arda, B.: Kişisel sağlık verilerinin korunması kapsamında sağlık araştırmalarında etik bakış. Türk Kardiyoloji Derneği Arşivi **47**(3) (2019)
44. Jiang, J.C., Kantarci, B., Oktug, S., Soyata, T.: Federated learning in smart city sensing: Challenges and opportunities. Sensors **20**(21), 6230 (2020)
45. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., *et al.*: Advances and open problems in federated learning. Foundations and trends® in machine learning **14**(1–2), 1–210 (2021)
46. Süzen, A.A., Kayaalp, K.: Büyük verilerde gizlilik tabanli yaklaşim: Federe öğrenme. International Journal of 3d Printing Technologies and Digital Industry **3**(3), 297–304 (2019)
47. Huang, C., Huang, J., Liu, X.: Cross-silo federated learning: Challenges and opportunities. arXiv preprint arXiv:2206.12949 (2022)
48. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečnỳ, J., Mazzocchi, S., McMahan, B., et al.: Towards federated learning at scale: System design. Proc. Mach. Learn. Res. **1**, 374–388 (2019)
49. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics, pp. 1273–1282 (2017). PMLR
50. Sprague, M.R., Jalalirad, A., Scavuzzo, M., Capota, C., Neun, M., Do, L., Kopp, M.: Asynchronous federated learning for geospatial applications. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 21–28 (2018). Springer
51. Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., Liu, X., He, B.: A survey on federated learning systems: Vision, hype and reality for data privacy and protection. IEEE Trans. Knowl. Data Eng. **35**(4), 3347–3366 (2021)
52. Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E.S.: A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Commun. Surv. Tutorials **21**(1), 686–728 (2018)
53. Kumar, K.S., Nair, S.A.H., Roy, D.G., Rajalingam, B., Kumar, R.S.: Security and privacy-aware artificial intrusion detection system using federated machine learning. Comput. Electr. Eng. **96**, 107440 (2021)
54. Bhati, N.S., Khari, M.: Empowering intrusion detection in 5g embedded and cyber-physical networks. Int. J. Embed. Syst. **16**(5–6), 401–412 (2023)
55. Rahman, S.A., Tout, H., Talhi, C., Mourad, A.: Internet of things intrusion detection: Centralized, on-device, or federated learning? IEEE Netw. **34**(6), 310–317 (2020)
56. Wagner, I., Eckhoff, D.: Technical privacy metrics: a systematic survey. ACM Comput. Surv. **51**(3), 1–38 (2018)
57. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K.: Practical secure aggregation for federated learning on user-held data. arXiv preprint arXiv:1611.04482 (2016)
58. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K.: Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1175–1191 (2017)
59. Aono, Y., Hayashi, T., Wang, L., Moriai, S., et al.: Privacy-preserving deep learning via additively homomorphic encryption. IEEE Trans. Inf. Forensics Secur. **13**(5), 1333–1345 (2017)
60. Hardy, S., Henecka, W., Ivey-Law, H., Nock, R., Patrini, G., Smith, G., Thorne, B.: Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. arXiv preprint arXiv:1711.10677 (2017)
61. Dwork, C.: Differential privacy. In: International Colloquium on Automata, Languages, and Programming, pp. 1–12 (2006). Springer
62. Goryczka, S., Xiong, L.: A comprehensive comparison of multiparty secure additions with differential privacy. IEEE Trans. Dependable Secure Comput. **14**(5), 463–477 (2015)
63. Kaissis, G., Ziller, A., Passerat-Palmbach, J., Ryffel, T., Usynin, D., Trask, A., Lima, I., Jr., Mancuso, J., Jungmann, F., Steinborn, M.-M., et al.: End-to-end privacy preserving deep learning on multi-institutional medical imaging. Nat. Mach. Intell. **3**(6), 473–484 (2021)
64. Xu, R., Baracaldo, N., Zhou, Y., Anwar, A., Ludwig, H.: Hybridalpha: An efficient approach for privacy-preserving federated learning. In: Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security, pp. 13–23 (2019)
65. Solanki, T., Rai, B.K., Sharma, S.: Federated learning using tensor flow. In: Federated Learning for IoT Applications, pp. 157–167. Springer, ??? (2022)
66. Ziller, A., Trask, A., Lopardo, A., Szymkow, B., Wagner, B., Bluemke, E., Nounahon, J.-M., Passerat-Palmbach, J., Prakash, K., Rose, N., et al.: Pysyft: A library for easy federated learning. Federated learning systems: Towards next-generation AI, 111–139 (2021)
67. Liu, Y., Fan, T., Chen, T., Xu, Q., Yang, Q.: Fate: An industrial grade platform for collaborative learning with data protection. J. Mach. Learn. Res. **22**(226), 1–6 (2021)
68. Protić, D.D.: Review of kdd cup'99, nsl-kdd and kyoto 2006+ datasets. Vojnotehnički glasnik **66**(3), 580–596 (2018)
69. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the kdd cup 99 data set. In: 2009 IEEE Symposium

on Computational Intelligence for Security and Defense Applications, pp. 1–6 (2009). IEEE
70. Moustafa, N., Slay, J.: Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS), pp. 1–6 (2015). IEEE
71. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSp **1**, 108–116 (2018)
72. Ring, M., Wunderlich, S., Grüdl, D., Landes, D., Hotho, A.: Flow-based benchmark data sets for intrusion detection. In: Proceedings of the 16th European Conference on Cyber Warfare and Security. ACPI, pp. 361–369 (2017)
73. Sharafaldin, I., Lashkari, A.H., Hakak, S., Ghorbani, A.A.: Developing realistic distributed denial of service (ddos) attack dataset and taxonomy. In: 2019 International Carnahan Conference on Security Technology (ICCST), pp. 1–8 (2019). IEEE
74. Cil, A.E., Yildiz, K., Buldu, A.: Detection of ddos attacks with feed forward based deep neural network model. Expert Syst. Appl. **169**, 114520 (2021)
75. Carrier, T.: Detecting obfuscated malware using memory feature engineering (2021)
76. Shafi, S., Tariq, N., Khan, F.A., Ali, A.: Federated learning for enhanced malware threat detection to secure smart power grids. In: International Conference on Ubiquitous Computing and Ambient Intelligence, pp. 692–703 (2024). Springer
77. Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H.: Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning. IEEE Access **10**, 40281–40306 (2022)
78. Liu, H., Zhang, S., Zhang, P., Zhou, X., Shao, X., Pu, G., Zhang, Y.: Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing. IEEE Trans. Veh. Technol. **70**(6), 6073–6084 (2021)
79. Sun, N., Wang, W., Tong, Y., Liu, K.: Blockchain based federated learning for intrusion detection for internet of things. Front. Comput. Sci. **18**(5), 185328 (2024)
80. Qian, Y., Hu, L., Chen, J., Guan, X., Hassan, M.M., Alelaiwi, A.: Privacy-aware service placement for mobile edge computing via federated learning. Inform. Sci. **505**, 562–570 (2019)
81. Bakopoulou, E., Tillman, B., Markopoulou, A.: Fedpacket: A federated learning approach to mobile packet classification. IEEE Trans. Mobile Comput. **21**(10), 3609–3628 (2021)
82. Ramapuram Campus, C.: Federated learning approach for analyzing electric vehicle sales in the indian automobile market. Unknown Journal (Unknown year)
83. Kong, L., Zheng, G., Brintrup, A.: A federated machine learning approach for order-level risk prediction in supply chain financing. Int. J. Prod. Econ. **268**, 109095 (2024)
84. Hussain, N., Rani, P., Chouhan, H., Gaur, U.S.: Cyber security and privacy of connected and automated vehicles (cavs)-based federated learning: challenges, opportunities, and open issues. Federated learning for IoT applications, 169–183 (2022)
85. Nguyen, D.C., Pham, Q.-V., Pathirana, P.N., Ding, M., Seneviratne, A., Lin, Z., Dobre, O., Hwang, W.-J.: Federated learning for smart healthcare: A survey. ACM Comput. Surv. (Csur) **55**(3), 1–37 (2022)
86. Khan, I.A., Razzak, I., Pi, D., Khan, N., Hussain, Y., Li, B., Kousar, T.: Fed-inforce-fusion: A federated reinforcement-based fusion model for security and privacy protection of iomt networks against cyber-attacks. Inform. Fusion **101**, 102002 (2024)
87. Singh, P., Singh, M.K., Singh, R., Singh, N.: Federated learning: Challenges, methods, and future directions. In: Federated Learning for IoT Applications, pp. 199–214. Springer (2022)
88. Li, L., Fan, Y., Tse, M., Lin, K.-Y.: A review of applications in federated learning. Comput. Indus. Eng. **149**, 106854 (2020)
89. Friha, O., Ferrag, M.A., Shu, L., Maglaras, L., Choo, K.-K.R., Nafaa, M.: Felids: Federated learning-based intrusion detection system for agricultural internet of things. J. Parallel Distrib. Comput. **165**, 17–31 (2022)
90. Raza, M., Saeed, M.J., Riaz, M.B., Sattar, M.A.: Federated learning for privacy preserving intrusion detection in software defined networks. IEEE Access (2024)
91. Benameur, R., Dahane, A., Souihi, S., Mellouk, A.: A novel federated learning based intrusion detection system for iot networks. In: ICC 2024-IEEE International Conference on Communications, pp. 2402–2407 (2024). IEEE
92. Song, X., Ma, Q.: Intrusion detection using federated attention neural network for edge enabled internet of things. J. Grid Comput. **22**(1), 1–17 (2024)
93. Tabassum, A., Erbad, A., Lebda, W., Mohamed, A., Guizani, M.: Fedgan-ids: Privacy-preserving ids using gan and federated learning. Comput. Commun. **192**, 299–310 (2022)
94. Bhati, B.S., Chugh, G., Al-Turjman, F., Bhati, N.S.: An improved ensemble based intrusion detection technique using xgboost. Trans. Emerg. Telecommun. Technol. **32**(6), 4076 (2021)
95. Bhati, N.S., Khari, M.: A new ensemble based approach for intrusion detection system using voting. J. Intell. Fuzzy Syst. **42**(2), 969–979 (2022)
96. Zhang, J., Luo, C., Carpenter, M., Min, G.: Federated learning for distributed iiot intrusion detection using transfer approaches. IEEE Trans. Indus. Inform. **19**(7), 8159–8169 (2022)
97. Campos, E.M., Saura, P.F., González-Vidal, A., Hernández-Ramos, J.L., Bernabe, J.B., Baldini, G., Skarmeta, A.: Evaluating federated learning for intrusion detection in internet of things: Review and challenges. Comput. Netw. **203**, 108661 (2022)
98. Chen, Z., Lv, N., Liu, P., Fang, Y., Chen, K., Pan, W.: Intrusion detection for wireless edge networks based on federated learning. IEEE Access **8**, 217463–217472 (2020)
99. Rey, V., Sánchez, P.M.S., Celdrán, A.H., Bovet, G.: Federated learning for malware detection in iot devices. Comput. Netw. **204**, 108693 (2022)
100. Singh, G., Sood, K., Rajalakshmi, P., Nguyen, D.D.N., Xiang, Y.: Intrusion detection scheme for next generation networks. IEEE Transactions on Network and Service Management (2024)
101. Khan, I.A., Pi, D., Kamal, S., Alsuhaibani, M., Alshammari, B.M.: Federated-boosting: A distributed and dynamic boosting-powered cyber-attack detection scheme for security and privacy of consumer iot. IEEE Transactions on Consumer Electronics (2024)
102. Hamdi, N.: Federated learning-based intrusion detection system for internet of things. Int. J. Inform. Secur. **22**(6), 1937–1948 (2023)
103. Li, B., Wu, Y., Song, J., Lu, R., Li, T., Zhao, L.: Deepfed: Federated deep learning for intrusion detection in industrial cyber-physical systems. IEEE Trans. Industr. Inform. **17**(8), 5615–5624 (2020)
104. Jayagopal, V., Elangovan, M., Singaram, S.S., Shanmugam, K.B., Subramaniam, B., Bhukya, S.: Intrusion detection system in industrial cyber-physical system using clustered federated learning. SN Comput. Sci. **4**(5), 452 (2023)
105. Khan, I.A., Pi, D., Abbas, M.Z., Zia, U., Hussain, Y., Soliman, H.: Federated-srus: A federated-simple-recurrent-units-based ids for accurate detection of cyber attacks against iot-augmented industrial control systems. IEEE Internet Things J. **10**(10), 8467–8476 (2022)

106. Belenguer, A., Pascual, J.A., Navaridas, J.: Göwfed: A novel federated network intrusion detection system. J. Netw. Comput. Appl. **217**, 103653 (2023)
107. Javeed, D., Saeed, M.S., Adil, M., Kumar, P., Jolfaei, A.: A federated learning-based zero trust intrusion detection system for internet of things. Ad Hoc Networks, 103540 (2024)
108. Amiri-Zarandi, M., Dara, R.A., Lin, X.: Sids: A federated learning approach for intrusion detection in iot using social internet of things. Comput. Netw. **236**, 110005 (2023)
109. Attota, D.C., Mothukuri, V., Parizi, R.M., Pouriyeh, S.: An ensemble multi-view federated learning intrusion detection for iot. IEEE Access **9**, 117734–117745 (2021)
110. Pope, J., Spyridopoulos, T., Kumar, V., Raimondo, F., Gunner, S., Oikonomou, G., Pasquier, T., McConville, R., Carnelli, P., Sanchez-Mompo, A., *et al.*: Intrusion detection at the iot edge using federated learning. In: Security and Privacy in Smart Environments, pp. 98–119. Springer (2024)
111. Yadav, K., Gupta, B.B., Hsu, C.-H., Chui, K.T.: Unsupervised federated learning based iot intrusion detection. In: 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), pp. 298–301 (2021). IEEE
112. Bhavsar, M., Bekele, Y., Roy, K., Kelly, J., Limbrick, D.: Fl-ids: Federated learning-based intrusion detection system using edge devices for transportation iot. IEEE Access (2024)
113. Sarhan, M., Lo, W.W., Layeghy, S., Portmann, M.: Hbfl: A hierarchical blockchain-based federated learning framework for collaborative iot intrusion detection. Comput. Electr. Eng. **103**, 108379 (2022)
114. Sarhan, M., Layeghy, S., Moustafa, N., Portmann, M.: Cyber threat intelligence sharing scheme based on federated learning for network intrusion detection. J. Netw. Syst. Manag. **31**(1), 3 (2023)
115. Mirzaee, P.H., Shojafar, M., Pooranian, Z., Asefy, P., Cruickshank, H., Tafazolli, R.: Fids: A federated intrusion detection system for 5g smart metering network. In: 2021 17th International Conference on Mobility, Sensing and Networking (MSN), pp. 215–222 (2021). IEEE
116. Li, J., Tong, X., Liu, J., Cheng, L.: An efficient federated learning system for network intrusion detection. IEEE Systems Journal (2023)
117. Neto, H.N.C., Hribar, J., Dusparic, I., Fernandes, N.C., Mattos, D.M.: Fedsbs: Federated-learning participant-selection method for intrusion detection systems. Comput. Netw. **244**, 110351 (2024)
118. Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., Bhattacharya, S., Maddikunta, P.K.R., Gadekallu, T.R.: Federated learning for intrusion detection system: Concepts, challenges and future directions. Comput. Commun. **195**, 346–361 (2022)
119. Lavaur, L., Pahl, M.-O., Busnel, Y., Autrel, F.: The evolution of federated learning-based intrusion detection and mitigation: a survey. IEEE Trans. Netw. Serv. Manag. **19**(3), 2309–2332 (2022)
120. Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., Gao, Y.: A survey on federated learning. Knowl.-Based Syst **216**, 106775 (2021)
121. Bashir, A.K., Victor, N., Bhattacharya, S., Huynh-The, T., Chengoden, R., Yenduri, G., Maddikunta, P.K.R., Pham, Q.-V., Gadekallu, T.R., Liyanage, M.: Federated learning for the healthcare metaverse: Concepts, applications, challenges, and future directions. IEEE Internet of Things Journal (2023)
122. Khan, I.A., Razzak, I., Pi, D., Zia, U., Kamal, S., Hussain, Y.: A novel collaborative sru network with dynamic behaviour aggregation, reduced communication overhead and explainable features. IEEE J. Biomed. Health Inform. **28**(6), 3228–3235 (2024)
123. Belenguer, A., Pascual, J.A., Navaridas, J.: A review of federated learning applications in intrusion detection systems. Computer Networks, 111023 (2025)

Busra Buyuktanir Research Assistant Busra was born in Uşak in 1992. She completed her secondary education at Hatay Hüseyin Özbuğday Anatolian High School in 2010 and received her B.Sc. degree in Computer Engineering from Kırıkkale University in 2014. She earned her M.Sc. in Computer Engineering from Marmara University in 2021. Between 2019 and 2021, she worked as a research assistant in the Department of Software Engineering at Istanbul Altınbaş University. Since 2021, she has been serving as a research assistant in the Department of Computer Engineering at the Faculty of Technology, Marmara University, where she is also pursuing her Ph.D. studies. Her research interests include machine learning, machine unlearning, and federated learning.

Şahsene Altinkaya has been a Postdoctoral Researcher at the Department of Mathematics and Statistics, University of Turku, since 2024. Previously, she was an Associate Professor at Istanbul Beykent University, Türkiye. She received her Ph.D. degree in Mathematics from Bursa Uludağ University, Türkiye, in 2019. Her research interests include Geometric Function Theory, Special Functions, and Machine Learning.

Gozde Karatas Baydogmus was born in İstanbul, Türkiye, in 1991. She received the bachelor's degree from the Mathematics and Computer Science Department, Istanbul Kultur University, in 2009, the M.S. degree from the Computer Engineering Department, Istanbul Kultur University, in 2013, and the Ph.D. degree from the Computer Engineering Department, Marmara University. In 2015, she completed the master's thesis on NoSql Database Testing in Istanbul Kultur University and the Ph.D. thesis on Intrusion Detection Systems in Marmara University. She worked as a Research Assistant with the Department of Mathematics and Computer Science, Istanbul Kültür University. She is currently working as

an Assistant Professor with the Computer Engineering Department, Marmara University. During the master's studies, she worked on distributed databases. She continues to work in the field of computer security. Her research interests include computer networks and security, machine learning, deep learning, cryptography, python programming, and statistics.

Assoc. Prof. Dr. Kazim Yildiz was born in Istanbul in 1985. He received his B.Sc. degree in Computer-Control Education from Marmara University in 2007. He completed his M.Sc. in 2010 and his Ph.D. in 2014 at the same university. He began his academic career in 2009 as a research assistant in the Department of Electronics and Computer Education at Marmara University. Between October 2015 and October 2016, he was a postdoctoral researcher at the Georgia Institute of Technology, School of Electrical and Computer Engineering, in the United States. In 2017, he obtained a B.Sc. degree in Computer Engineering from Istanbul University. Since 2015, he has been working as a faculty member in the Department of Computer Engineering at Marmara University. He was awarded the title of Associate Professor in Computer Science and Engineering in 2020. His research interests include artificial intelligence, deep learning, machine learning, and computer vision. He also serves as a cosupervisor at Marmara University Vision Lab.