Mayur Girish Taunk, 2022, 10:1
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

# Federated Learning Architectures for Privacy-Preserving Intrusion Detection in IoT Networks

**Mayur Girish Taunk,**
**Jigarkumar Ambalal Patel .**
Department of Computer Engineering
Government Polytechnic Bhuj.
Bhuj, India

**Abstract – The rapid proliferation of Internet of Things (IoT) devices and smart infrastructure has led to an exponential surge in network traffic, rendering traditional security perimeters increasingly vulnerable. Intrusion Detection Systems (IDS) serve as a critical frontline defense; however, conventional centralized Machine Learning (ML) models are struggling to reconcile high-volume data processing with stringent privacy regulations such as the General Data Protection Regulation (GDPR). Federated Learning (FL) has emerged as a pivotal decentralized paradigm, allowing edge devices and organizations to cooperatively train global models while keeping raw data local, thereby ensuring privacy and reducing model-offloading bandwidth consumption. This review provides a comprehensive analysis of the evolution of FL-based IDS, focusing on its implementation within Industrial Control Systems (ICS) and smart manufacturing environments. We systematically examine the primary technical hurdles facing these architectures, specifically focusing on statistical heterogeneity (non-IID data), communication overhead in resource-constrained networks, and vulnerability to adversarial machine learning attacks such as poisoning and evasion, Furthermore, we discuss specialized integrations with Information-Centric Networking (ICN) and the efficacy of deep learning architectures, such as Long Short-Term Memory (LSTM), in enhancing detection accuracy. The paper concludes by identifying future research avenues, including the need for enhanced model interpretability and robustness against adaptive adversarial threats**

**Keywords – Federated Learning, Intrusion Detection Systems (IDS), IoT Security, Deep Learning, Non-IID Data, Adversarial Machine Learning, Anomaly Detection, Privacy-Preserving.**

## I. INTRODUCTION: THE EVOLUTION OF NETWORK SECURITY IN THE IOT ERA

The global digital landscape has undergone a radical transformation over the last decade, characterized by an **exponential increase in the size and complexity of the Internet** [1]. This growth is largely driven by the predominant usage of mobile phones, wearable devices, and autonomous vehicles, which function as distributed network nodes generating massive amounts of data daily [2]. It is estimated that the number of connected devices will reach **75 billion worldwide by 2025** [2]. While this expansion offers significant opportunities for smart infrastructure, it simultaneously exposes network systems to diverse security vulnerabilities and sophisticated intrusions [1].

### 1. The Evolving Threat Landscape in IoT and ICS

As network infrastructures become more heterogeneous, they become increasingly susceptible to attempts to compromise the **confidentiality, integrity, or availability** of data[3]. Typical modern threats include password cracking, Man-in-the-Middle (MitM) attacks like ARP spoofing, and high-volume Denial of Service (DoS) attacks [4]. Specifically, **Industrial Control Systems (ICS)** and smart manufacturing environments are now top targets for cyber-physical attacks, which can have devastating consequences for national facilities like power grids and gas pipelines [5]. These environments require real-time analysis to respond to attacks before they cause physical damage [5].

### 2. The Role and Limitations of Intrusion Detection Systems (IDS)

Intrusion Detection Systems (IDS) serve as the **first line of defense** in network security, constantly monitoring traffic for malicious indicators [1]. Historically, these systems relied on **signature-based detection**, which uses databases of known attack patterns [3]. However, this traditional method is increasingly ineffective against **zero-day attacks**—novel threats for which no signatures yet exist [3]. This has shifted the research focus toward **anomaly-based detection** powered by Machine Learning (ML) and Deep Learning (DL), which can identify deviations from "normal" network behavior to flag previously unseen threats[1].

### 3. The Decline of Centralized Learning Paradigms

While DL and ML have achieved high classification accuracy in intrusion detection, their traditional **centralized training architecture** faces significant hurdles[3]. Centralized learning requires all raw data to be offloaded and stored on a single server, which creates single points of failure and massive communication bottlenecks in resource-constrained environments [5]. Furthermore, the emergence of stringent privacy regulations, such as the **General Data Protection Regulation (GDPR)**, has made the explicit sharing of sensitive raw data between organizations or devices legally and ethically challenging [6].

### 4. The Emergence of Federated Learning

To address these data privacy and bandwidth concerns, **Federated Learning (FL)** has emerged as a pivotal decentralized paradigm [6]. FL allows multiple clients to collaboratively train a shared global model while keeping their **raw data locally stored** on their own devices [7]. By transmitting only model parameters or gradients to a central orchestrator rather than raw datasets, FL ensures that user privacy is preserved and network latency is reduced [7]. This review systematically explores the integration of FL into IDS, analyzing its potential to build robust, universal detection models through collaborative **Cyber Threat Intelligence (CTI)** sharing without compromising the data integrity of individual organizations [8].

.

Table I: Comparative Analysis of Intrusion Detection Methodologies

| **Detection Method** | **Core Mechanism** | **Primary Strengths** | **Primary Weaknesses** | **Zero-Day Attack Capability** |
|---|---|---|---|---|
| Signature-Based IDS (SIDS) | Matches network traffic patterns against a predefined database of known threat signatures. | Highly effective and accurate at identifying recognized, historically documented attacks | Requires constant database updates and is fundamentally rigid against uncatalogued threats. | **Incapable**. |
| Anomaly-Based IDS (AIDS) | Establishes a baseline profile of "normal" network behavior and flags statistical deviations as potential intrusions. | Vital for identifying novel threats and adapting to dynamic, evolving attack vectors | Often suffers from a high False Alarm Rate (FAR) due to the fluid nature of legitimate network environments. | **Highly Capable**. |

*Note:* FAR = False Alarm Rate. The evaluation of zero-day capability highlights the fundamental constraint of traditional architectures that rely on pre-existing threat databases versus the adaptive nature of baseline variance models

## II. TAXONOMY OF IDS AND DEEP LEARNING INTEGRATION

The architectural design of an **Intrusion Detection System (IDS)** is fundamentally governed by its deployment environment and its underlying detection methodology. As cyber threats become more sophisticated, the integration of **Deep Learning (DL)** has become essential to move beyond rigid, rule-based systems toward intelligent, self-learning frameworks[1].

### 1. Deployment-Based Classification

Intrusion detection systems are primarily categorized based on where they are situated within a digital ecosystem. A **Host Intrusion Detection System (HIDS)** is installed on specific endpoints to monitor internal system activities, such as log file modifications and user actions [3]. In contrast, a **Network Intrusion Detection System (NIDS)** is positioned at strategic points throughout a network to scrutinize all incoming and outgoing traffic between nodes [3]. While HIDS offers deep visibility into host-level breaches, it imposes a significant processing overhead on

individual devices, whereas NIDS provides a broader defensive perimeter for the entire network [3] .

## 2. Signature-Based vs. Anomaly-Based Detection

Traditional security relies on **Signature-based Intrusion Detection (SIDS)**, which identifies malicious activity by matching patterns against a database of known threats[1] . While highly effective for recognized attacks, SIDS is inherently incapable of detecting **zero-day exploits** because no predefined signature exists for them [3] . This limitation has catalyzed the development of **Anomaly-based Intrusion Detection (AIDS)**, or behavior-based IDS. AIDS establishes a profile of "normal" network behavior and flags any statistically significant deviation as a potential intrusion [1]. Although AIDS is vital for identifying novel threats, it often suffers from a high **False Alarm Rate (FAR)** due to the difficulty of defining precise boundaries in dynamic network environments [3]. To further elucidate the operational differences, security trade-offs, and zero-day detection capabilities between traditional pattern-matching and behavior-based monitoring, **Table 1** presents a comparative analysis of these foundational IDS methodologies.

Table 2: Deep Learning Architectures in Intrusion Detection

| **Architecture Type** | Primary Function | Key Advantage in IDS |
|---|---|---|
| AE | Unsupervised feature extraction and dimensionality reduction. | Unsupervised feature extraction and dimensionality reduction |
| CNN | Extracts spatial features from data originally designed for image processing | Converts raw network traffic into two-dimensional representations, improving classification accuracy |
| VAE | Reproduces input data using symmetric encoder and decoder architectures | symmetric encoder and decoder architectures. Proven to be particularly effective for detecting cyberattacks within Industrial Control Systems (ICS) |
| LSTM | Processes sequential and contextual data using specialized "cell states" and gating mechanisms | Remembers long-term dependencies across entire network sessions while avoiding the vanishing gradient problem. |

*Note:* AE = Autoencoder; CNN = Convolutional Neural Network; VAE = Variational Autoencoder; LSTM = Long Short-Term Memory. The selection of architecture depends heavily on whether the IDS prioritizes spatial feature extraction or sequential session context.

## 3. Deep Learning Architectures in IDS

The integration of DL has revolutionized AIDS by allowing systems to automatically extract high-level features from raw network packets [3]. To address the complexities of modern network traffic, various deep neural architectures have been deployed to enhance detection performance, the core functions and advantages of which are summarized in **Table 2**.

- **Autoencoders (AE):** These unsupervised neural networks are frequently used for **feature extraction** and dimensionality reduction, compressing high-dimensional traffic data into a manageable latent space [4].
- **Convolutional Neural Networks (CNN):** Originally designed for image processing, CNNs are utilised in NIDS to extract **spatial features** by converting raw network traffic into two-dimensional representations for classification [3].
- **Variational Autoencoders (VAE):** VAEs use a symmetric architecture to reproduce input data through encoders and decoders, which is particularly effective for detecting cyberattacks in industrial control systems[5].

## 4. Handling Sequential and Contextual Data

Network traffic is inherently sequential, where the meaning of a single command is often dependent on the entire session's context [7]. **Recurrent Neural Networks (RNN)** and their variants, such as **Long Short-Term Memory (LSTM)**, are particularly suited for this task [7]. LSTM frameworks utilize a specialized "cell state" and gating mechanisms to remember long-term dependencies while avoiding the vanishing gradient problem [7]. By modelling the bidirectional

association between network commands, architectures like **Bi-directional LSTM (BiLSTM)** can better capture the semantic dependencies of user behavior, significantly improving prediction accuracy [7]

## III. FEDERATED LEARNING ARCHITECTURES FOR COLLABORATIVE SECURITY

**Federated Learning (FL)** is a decentralized training paradigm designed to move model computations to the data rather than offloading raw data to a central location [2]. This approach allows multiple participants—such as edge devices or independent organizations—to cooperatively train a shared global model while ensuring that **sensitive raw data never leaves the local environment** [8]. This methodology is particularly effective for large-scale distributed systems where communication bandwidth is a bottleneck and data privacy are a legal necessity [6].

### 1. The Federated Learning Lifecycle and Phases

An FL training process is typically an iterative loop coordinated by a central server [1]. Each round follows a specific sequence of operations [2]:

1. **Client Selection:** The central server identifies a subset of active clients that meet eligibility requirements, such as being idle or connected to an unmetered network [2].
2. **Broadcast:** The server disseminates the current global model weights and the training program to the selected participants [8].
3. **Local Training (Client Computation):** Each device runs a local optimisation algorithm, such as **Stochastic Gradient Descent (SGD)**, on its private dataset to update the model parameters [1].
4. **Aggregation:** The server collects the updated model parameters from the clients. It then combines these updates, most commonly using the **Federated Averaging (FedAvg)** algorithm, which computes a weighted average based on the number of samples each client contributed [1].
5. **Global Model Update:** The server applies the aggregated results to update the global model, which is then used as the starting point for the next training round [9].

### 2. Taxonomy of Federated Learning Architectures

Depending on how the data is distributed among clients in terms of samples and feature characteristics, FL architectures are categorized into three primary types [6]:

- **Horizontal Federated Learning (HFL):** This is used when different clients have datasets that share the same feature space but differ in the specific observations or samples [1]. An example is multiple regional hospitals using identical medical record formats but serving distinct patient populations [9].
- **Vertical Federated Learning (VFL):** Also known as feature-based FL, this applies when clients share the same sample IDs but have different feature sets [1]. This occurs when two different businesses (e.g., a bank and an e-commerce site) have different types of data about the same set of users [9].
- **Federated Transfer Learning (FTL):** FTL is employed when datasets differ in both the sample space and the feature space [1]. It utilises knowledge from a source domain to improve model performance in a target domain where data may be scarce [2].

### 3. Frameworks and Implementation Platforms

To implement these complex architectures, various open-source frameworks have emerged to handle communication and security [1]. **TensorFlow Federated (TFF)** is a popular platform that provides distributed computation capabilities for decentralized data [9]. Other important tools include **PySyft**, which integrates with PyTorch to provide privacy-preserving deep learning, **LEAF**, which serves as a standard benchmark for FL settings, and **FATE**, which supports the secure implementation of federated machine learning in industrial environments [1].

## IV. TECHNICAL CHALLENGES: NON-IID DATA AND COMMUNICATION EFFICIENCY

While **Federated Learning (FL)** offers a robust framework for privacy-preserving security, its implementation in real-world **Intrusion Detection Systems (IDS)** is hindered by significant technical hurdles. The two most prominent challenges are the statistical heterogeneity of decentralized data and the massive communication overhead required to maintain synchronized global models [1].

### 1. Statistical Heterogeneity (Non-IID Data)

The foundational assumption of many Machine Learning (ML) algorithms is that data is **Independent and Identically Distributed (IID)**. However, in a federated network of IoT devices or industrial plants, this assumption rarely holds [6]. Each node generates

data based on its unique environment, usage patterns, and specific attack exposures, leading to **Non-IID data distribution** [6].

The sources identify several critical types of data skew that complicate IDS training [6]:

- **Feature Distribution Skew:** The same type of network traffic may look different across devices due to varied hardware or protocols [6].
- **Label Distribution Skew:** Some nodes may experience frequent Denial-of-Service (DoS) attacks, while others only see benign traffic, creating an imbalance that biases the global model [6].
- **Quantity Skew:** Different clients contribute vastly different volumes of data, which can lead to the global model being dominated by a few high-volume nodes [6] .

These skews cause "client drift," where local updates pull the global model in divergent directions, significantly reducing **convergence speed and detection accuracy** [6]. In the context of IDS, this variance often results in a high **False Alarm Rate (FAR)**, as the global model fails to find a stable boundary for "normal" behaviour across heterogeneous environments [1].

**2. Communication Overhead and Bandwidth Bottlenecks**

The iterative nature of FL requires constant exchange of model parameters (weights and gradients) between the central server and potentially millions of clients [1]. This creates a **communication bottleneck**, especially in resource-constrained IoT environments where bandwidth is limited and expensive [2].

Key factors contributing to this overhead include [1]:

- **Model Size:** Modern Deep Learning models have millions of parameters. Transmitting these "bulky" weights over low-power protocols like **LPWAN** or **6LoWPAN** is often infeasible [1].
- **Server Accumulation:** Even if only 1% of a million-device network participates in a round, the server must handle concurrent updates from 10,000 clients, often leading to a **single point of failure** or extreme latency [1].
- **Network Unreliability:** IoT devices are prone to losing connectivity, leading to "stragglers"—clients that fail to report updates on time, slowing down the entire training round [2].

**3. Emerging Solutions for Efficiency and Robustness**

To mitigate these challenges, researchers are exploring several optimization strategies:

- **Lightweight Architectures:** Algorithms like **Binarized Neural Networks (BNN)** convert floating-point weights into binary formats (0s and 1s), reducing the size of transmitted parameters and allowing for bit-wise operations that speed up packet processing [1].
- **Asynchronous Federated Learning (AFL):** AFL allows the central server to update the global model as soon as individual updates arrive, rather than waiting for all clients to finish, which helps eliminate system lag and prevents server bottlenecks [1].
- **Clustering and Personalization:** To solve Non-IID issues, **Hierarchical Clustering** groups clients with similar data distributions together. This allows for the creation of multiple specialized global models that better fit the specific traffic patterns of different "neighbourhoods" in the network [1], [6].
- **Data Augmentation:** The **FAug algorithm** uses Generative Adversarial Networks (GANs) to supplement missing attack labels on local devices, helping to balance imbalanced datasets without compromising raw data privacy [6] .

## V. ADVERSARIAL MACHINE LEARNING AND ROBUSTNESS IN FL

Despite its privacy-preserving advantages, Federated Learning (FL) introduces new attack surfaces that can be exploited to compromise the integrity of Intrusion Detection Systems (IDS). Adversarial Machine Learning (AML) involves the creation of "adversarial examples"—small, often imperceivable perturbations to input data that cause Deep Learning models to misclassify malicious traffic as benign [4] .

**1. Taxonomy of Adversarial Attacks**

To systematically categorize the vulnerabilities introduced by decentralized architectures, adversarial threats against Federated Learning systems are generally classified based on the attacker's system knowledge and the execution phase of the exploit. **Table 3** outlines the primary categories of Adversarial Machine Learning (AML) attacks targeting IDS

frameworks, detailing their core mechanisms and targeted impacts:

Table 3: Taxonomy of Adversarial Attacks in Federated Learning

| Attack Category | Execution Phase | Attacker Knowledge | Core Mechanism & Targeted Impact |
|---|---|---|---|
| White-Box Attack | Training / Testing | Complete | The attacker possesses full knowledge of the model's parameters and architecture, utilizing gradient-based methods to calculate the most effective and damaging data perturbations. |
| Black-Box Attack | Testing | Limited / None | The attacker treats the target model as an opaque oracle, querying it to observe input-output pairings and often training a "substitute model" to craft transferable adversarial examples |
| Poisoning Attack | Training | Variable | A malicious or compromised client injects "dirty" data or manipulated gradients during the local training phase, aiming to slowly corrupt the global model and induce targeted blind spots. |
| Evasion Attack | Testing | Variable | Malicious network packets are subtly modified with imperceptible perturbations to bypass a fully deployed IDS without altering the underlying detection model itself. |

- **White-Box Attacks:** The attacker has complete knowledge of the model's parameters and architecture, allowing them to use gradient-based methods to find the most effective perturbations [4].
- **Black-Box Attacks:** The attacker treats the model as an oracle, querying it to observe outputs and potentially training a "substitute model" to craft transferable adversarial examples [4].
- **Poisoning vs. Evasion:** Poisoning attacks occur during the **training phase**, where a malicious client injects "dirty" data or manipulated gradients to corrupt the global model [1]. Evasion attacks occur during the **testing phase**, where malicious packets are modified to bypass a deployed IDS without altering the model itself [4].

**2. FL-Specific Vulnerabilities: Poisoning and Privacy Leaks**

In an FL environment, the decentralized nature of training allows malicious participants to conduct **federated poisoning attacks**. Because the central server cannot inspect raw local data, a single compromised client can transmit poisonous model updates that eventually cause the global model to collapse or perform poorly against specific intrusion types [1]. Furthermore, FL is susceptible to **Inference Attacks**, where a curious server or an external eavesdropper analyses the gradients sent by clients to reverse-engineer and reconstruct sensitive local training data [2].

**3. Robustness and Defensive Strategies**

To ensure the reliability of FL-based IDS, several defensive mechanisms have been proposed:

- **Information Encryption:** Techniques such as **Homomorphic Encryption (HE)** and **Secure Multi-Party Computing (SMPC)** allow the server to aggregate model updates without ever seeing the plaintext parameters, preventing privacy leaks [2].
- **Differential Privacy (DP):** By adding calculated Gaussian noise to the local gradients before transmission, DP masks the contribution of individual data points, making it significantly harder for attackers to perform model inversion [9].
- **Robustness Optimisation: Adversarial Training** involves including adversarial examples in the training dataset to help the model learn the boundaries of "malicious" perturbations [4]. Additionally, emerging solutions like **Digital Twins** can simulate network behaviour to predict vulnerabilities and ensure that only intended updates are integrated into the global model [1].
- **Blockchain Integration:** Utilising **Blockchain** as a decentralised ledger for model updates ensures that all participants are verified and that updates are traceable, which effectively filters out untrustworthy data from malicious nodes [2].

## VI. SPECIALIZED APPLICATIONS AND FUTURE DIRECTIONS

The integration of **Federated Learning (FL)** and **Intrusion Detection Systems (IDS)** has moved beyond theoretical frameworks into specialized domains that require high-speed analysis and strict data privacy. This section reviews key applications in modern infrastructure and outlines the critical research avenues necessary to mature these technologies.

### 1. Industrial IoT (IIoT) Security

In the era of Industry 4.0, Industrial IoT (IIoT) devices are prime targets for cyber-physical attacks that can disrupt national infrastructure [5] . FL-based IDS are uniquely suited for these environments because they allow decentralized anomaly detection at the network edge, near the data source [5]. By utilizing hybrid models such as Variational Autoencoders (VAE) combined with Long Short-Term Memory (LSTM), systems can effectively monitor time-series data from sensors to identify zero-day attacks without offloading sensitive industrial secrets to a central cloud [5]. Research indicates these architectures can achieve high detection performance while significantly reducing bandwidth consumption [5].

### 2. Information-Centric Networking (ICN)

Traditional IP-based addressing is increasingly insufficient for the billions of connected IoT nodes [2]. **Information-Centric Networking (ICN)** offers a content-based naming paradigm that improves data retrieval efficiency and security [2]. Integrating FL with ICN-IoT enables in-network caching and decentralized intelligence, which are essential for the upcoming **6G networks** [2]. These future networks will require a convergence of **Communication, Computing, and Caching (3C)**, where FL provides a secure distributed platform to optimize network capacity and user experience [2].

### 3. Specialized Domain Applications

Beyond industrial settings, FL-based IDS are gaining traction in several niche areas:

- **Healthcare:** FL allows multiple medical institutions to train robust models for **medical image processing** and disease analysis without violating patient confidentiality or regulations like GDPR [9].
- **Building Energy Management:** Adaptive FL systems are used for community-level **energy load forecasting** and anomaly prediction, helping to identify unexpected consumption patterns while keeping household usage data local [10].
- **Smart Cities and Vehicles:** FL supports collaborative security in **Autonomous Vehicles** and smart traffic systems, where real-time coordination is required between moving nodes and roadside units [9].

### 4. Open Issues and Future Challenges

To achieve widespread deployment, several open issues must be addressed:

- **Model Interpretability:** Current Deep Learning models often function as "black boxes" [1]. Future research must focus on **Explainable AI (XAI)** to help network operators understand the context and reasoning behind flagged intrusions [4].
- **Adversarial Robustness:** Most existing defences are evaluated against feature-level attacks, which are often impractical [4]. There is a critical need for NIDS that are resilient against **packet-level adversarial modifications** that maintain the malicious behaviour of the traffic [4].
- **Emerging Defensive Technologies:** The use of **Digital Twins** to simulate and predict vulnerabilities before deployment is a promising avenue [1]. Furthermore, integrating **Blockchain** as a decentralized ledger for FL updates can ensure that all training participants are verified and updates are traceable [2].
- **Standardization:** There is a lack of universal formats and updated benchmark datasets for heterogeneous networks; future work should leverage **Transfer Learning** to adapt models across different organizational environments efficiently [4].

## VII. CONCLUSION

The integration of Federated Learning (FL) into Intrusion Detection Systems (IDS) represents a paradigm shift in network security, effectively addressing the "data silo" problem and the growing constraints of privacy regulations like GDPR [6], [2]. This review has demonstrated that while FL enables collaborative Cyber Threat Intelligence (CTI) sharing without compromising raw data, it introduces complex technical challenges, most notably statistical heterogeneity (Non-IID data) and significant communication overhead [1], [8].

Furthermore, the vulnerability of these decentralized systems to Adversarial Machine Learning—including sophisticated poisoning and evasion attacks—

necessitates the development of more robust, context-aware defensive mechanisms [4]. As we move toward 6G networks and the convergence of communication, computing, and caching (3C), the role of FL-based IDS will become even more critical in securing specialized domains such as Smart Manufacturing, Healthcare, and Autonomous Vehicles [5], [2]. Ultimately, the future of intelligent intrusion detection lies in achieving a balance between high detection accuracy, communication efficiency, and verifiable privacy preservation [1].

## REFERENCES

1. V. Mothukuri et al., "A survey on security and privacy of federated learning," Future Generation Computer Systems, vol. 115, pp. 619–640, Jan 2021.
2. B. Nour et al., "A survey of Internet of Things communication using ICN: A use case perspective," Computer Communications, vol. 142, pp. 95–123, 2019.
3. Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Transactions on Emerging Telecommunications Technologies, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.
4. O. Ibitoye et al., "The threat of adversarial attacks on machine learning in network security—A survey," 2019, arXiv:1911.02621.
5. T. T. Huong et al., "Detecting cyberattacks using anomaly detection in industrial control systems: A Federated Learning approach," Comput. Ind., vol. 132, Nov. 2021, doi: 10.1016/j.compind.2021.103509.
6. H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-IID data: A survey," Neurocomputing, vol. 465, pp. 371–390, Dec 2021.
7. R. Zhao, Y. Yin, Y. Shi, and Z. Xue, "Intelligent intrusion detection based on federated learning aided long short-term memory," Physical Communication, vol. 42, Oct. 2020, doi: 10.1016/j.phycom.2020.101157.
8. D. Preuveneers et al., "Chained anomaly detection models for federated learning: an intrusion detection case study," Applied Sciences, vol. 8, no. 12, p. 2663, 2018.
9. Q. Yang et al., "Federated machine learning: Concept and applications," ACM Transactions on Intelligent Systems and Technology, vol. 10, no. 2, pp. 1–19, 2019.
10. M. N. Fekri et al., "Distributed load forecasting using smart meter data: federated learning with recurrent neural networks," International Journal of Electrical Power & Energy Systems, vol. 137, 2021.