Federated Learning based framework for improving Intrusion Detection System in IIOT

N. Balakumar[1], Dr Antony Selvadoss Thanamani[2], P. Karthiga[3], Dr A. Kanagaraj[4], S. Sathiyapriya[5], A. Shubha[6]

[1,3,5,6] Ph.D Research Scholar, Department of Computer Science, Nallamuthu Gounder Mahalingam College
Pollachi, Tamil Nadu - 642001, India.

[2] Associate Professor & Head, Department of Computer Science, Nallamuthu Gounder Mahalingam College
Pollachi, Tamil Nadu - 642001, India.

[4] Assistant Professor, Department of Computer Science, Nallamuthu Gounder Mahalingam College
Pollachi, Tamil Nadu - 642001, India.

msgtobalakumar@gmail.com, selvdoss@gmail.com, a.kanagaraj@gmail.com

ABSTRACT

The rapid development of the Internet and smart devices increases network traffic, which in turn makes the infrastructure increasingly complex and heterogeneous. Distributed networks that generate massive volumes of data daily include mobile phones, wearable technology, and autonomous vehicles. Intrusion detection systems considerably improve the security and privacy of such devices. Intrusion detection for such paradigms is a non-trivial challenge that has gained extra significance because of the rapid expansion in the number and variety of security threats for such systems. However, due to the unique characteristics of such systems, such as battery power, bandwidth and CPU overheads, and network dynamics, intrusion detection for the Industrial Internet of Things (IIoT) is a challenge that necessitates taking into account the trade-off between detection accuracy and performance overheads. The decentralized learning technique known as federated learning (FL), which trains models locally and communicates the parameters to a centralized server, is a suitable illustration. The current study aims to provide a comprehensive and in-depth investigation of the use of FL-based intrusion detection systems for IIoT.

Keywords: Industrial Internet of Things, Differential Privacy, Federated Learning, Intrusion Detected Systems, Internet of Things.

1. INTRODUCTION

Over the past few years, a wide range of businesses, including the medical field, the auto industry, the home appliance industry, etc., have seen a rapid uptake of the Internet of Things (IoT) paradigm. Our way of life has been significantly impacted by the introduction of IoT technologies. The idea behind the Industrial IoT (IIoT) was developed expressly for use in contemporary industry. The current industrial internet of things refers to the application of the standard IoT in numerous industrial activities and organizations. The IIoT includes a wide variety of actuators, sensors, control systems, interfaces for communication and integration, cutting-edge security systems, networks for cars and home appliances, etc. [3, 16]. Each node in the IIoT can connect to the Internet. The usage of IIoT in modern enterprises has significantly increased the capabilities of various sectors, including manufacturing facilities, asset management systems, complex logistics systems, etc. Additionally, the

IIoT enables a wide range of applications, devices, and services to connect the physical and digital worlds. Most IIoT nodes have the ability to collect, process, and transport data.

Each node in the IIoT can connect to the Internet. The usage of IIoT in modern enterprises has significantly increased the capabilities of various sectors, including manufacturing facilities, asset management systems, complex logistics systems, etc. Additionally, the IIoT enables a wide range of applications, devices, and services to connect the physical and digital worlds. Most IIoT nodes have the ability to collect, process, and transport data. Intrusion Detection Systems (IDSs) with hybrid bases integrate anomaly- and signature-based IDSs. One of the drawbacks of conventional IDSs is their high false positive rate and limited detection accuracy. Additionally, they are unable to block events like zero-day assaults and are unable to recognize novel types of invasions [1]. Researchers have investigated the use of Artificial Intelligence (AI), and more specifically, the application of Machine Learning (ML) based approaches for IDS, to enhance the performance of conventional IDSs.

Federated learning, also known as collaborative learning, is a machine learning technique that employs a number of dispersed edge devices or servers that maintain local data samples in order to train an algorithm without sending the data samples [4]. This strategy contrasts from conventional centralized machine learning techniques, where all local datasets are uploaded to a single server, and more usual decentralized approaches, which frequently assume that local data samples are uniformly dispersed. Federated learning allows multiple participants to create a similar, trustworthy machine learning model without sharing data, enabling the resolution of important issues such data privacy, security, access rights, and the availability of diverse data [12]. Defense, telecommunications, internet of things, and pharmaceutical industries are just a few of the sectors where it has applications.

2. INTRUSION DETECTION

The process of identifying and blocking malicious activity directed at computer systems and networking resources is known as an intrusion detection system (IDS) [11]. These are software used to keep an eye on network activity in order to spot malicious users or occurrences when necessary. A new sensor input is examined by an IDS before it is used as input for action. Organizations are more vulnerable to a variety of cyber risks as a result of the falling costs of information processing and Internet connectivity.

2.1 Anomaly Detection

When using an anomaly detection approach, the system defines the anticipated network behaviour beforehand. Techniques such as statistical methodologies, association rules, and neural networks are used to construct the profile of normal behaviour. Any noteworthy departures from this expected conduct are reported as potential assaults. The capacity to recognize novel assaults for which signatures have not yet been identified is the main benefit of anomaly-based detection. Yet, in reality, this is challenging to accomplish since it is challenging to gather precise and thorough profiles of typical behaviour. Because of this, an anomaly detection system may generate too many false alerts, and sorting through the resulting data can be very labour and time-intensive.

2.2 Recent trends in IDS based anomaly detection

The technique of identifying illegal traffic on a network or a device is known as intrusion detection [1]. An IDS is a piece of hardware or software that keeps track of network traffic and spots unwanted access that breaks security rules. On the basis of the system's installation, IDS may be divided into three groups: host-based IDS (HIDS), network-based IDS (NIDS), and hybrid IDS. On a single host, HIDS are implemented. In HIDS, threats are identified as coming from a single computer system, and the critical operating system files are examined. Hence, these attacks are typically simple to identify, with the exception of some in-filtered malware that is challenging to identify. A NIDS is installed on routers or switches in a network and uses different computer connections to identify harmful information. In contrast, hybrid IDS can be installed both on hosts and over a network [5].

2.3 Simulators for Intrusion Detection

There are two ways to simulate intrusion traffic: one involves simulating the intrusions in actual environments, and the other involves simulating the intrusions in a test environment. In the first method, real machines in a much smaller scaled network communicate with each other. Each of the machines portrays an assailant and victim in a unique way. The second method uses simulation software to build a network environment on possibly a single computer [11]. Each network component, including traffic, is simulated. The most well-known network simulation programmes are PCP, NS2, and OPNET.

2.4 Misuse Intrusion

Misuse intrusion involves the detection of intrusions by the observation of network activity in an effort to find precise matches of recognized attack patterns known as signatures or rules. Sometimes, "signature-based detection" is used to describe this kind of detection. One common misuse detection method used in commercial IDS is the classification of attacks for each pattern of events that resembles a particular signature. Nonetheless, there are more advanced techniques known as state based analysis that can use a single signature to find a number of attacks. This method has the limitation of only being able to identify incursions that match a predefined signature or set of rules.

3. RECENT INTRUSION DETECTION SYSTEM APPROACHES

The placement of an intrusion detection system is significant because it affects the level of visibility it can provide to the actions within the monitored system, just like with modern computing systems [13]. For example, a network-based IDS is only able to keep track of network traffic that is directed at or originated from the monitored host; as a result, it is unable to keep track of any process subversion or privilege escalation on the monitored host. With recent research demonstrating the inadequacy of safety for IoT devices, this is becoming more and more crucial for IoT systems.

The dynamic nature of the system with the participating nodes adhering to an adhoc pattern is one of the key characteristics of IoT systems. The goal of detecting an assault as quickly as possible to prevent infection spreading to new devices makes the time frame of detection even more crucial. The majority of current approaches are developed and evaluated in isolated environments, which has two negative effects: (i) The implementation and evaluation do not accurately reflect an IoT environment, and (ii) experimentation results carried out in isolated, simulated environments, such as Matlab or Contiki, do not accurately simulate the challenges encountered within a real-world IoT environment. As a result, the bulk of current techniques are only capable of offline detection, which limits an IDS's capacity to quickly defend against harmful threats.

How a detection system performed its detecting functions is specified in the system architecture of an IDS. The system architecture has an impact on user privacy in addition to performance and detection accuracy. Due to the lack of sufficient data and the stealthy nature of the attacker, the standalone detection system, which mostly acts at the local machine or device, is vulnerable to longer detection times. A collaborative architecture, on the other hand, makes use of data from several sources, including network devices or IoT devices in the same or distinct organizations [11]. Although it can increase detection accuracy, it also poses a privacy risk for data that is exchanged between organizations.

The resources that can be used by activities that need a lot of computing power, such intrusion detection, are typically limited in an IoT device. As a result, one of the crucial criteria is the performance overhead brought on by an IDS, which may be assessed through the IDS's energy or CPU usage [7]. The rate at which an IDS may successfully identify a malicious effort is known as detection performance. It is one of the key characteristics of an IDS because its efficiency can be directly correlated with it. In contrast to modern systems, an IoT system often has a much higher number of devices. Scalability of the IDS is a crucial factor because of the considerable number of devices involved.

4. INTRUSION DETECTION SSYSTEM TYPES

There are five different types of intrusion detection systems [13]: host intrusion detection systems (HIDS), network intrusion detection systems (NIDS), protocol-based intrusion detection systems (PIDS), application protocol-based intrusion detection systems (APIDS), and hybrid intrusion detection systems.

Host Intrusion Detection System (HIDS): In order to detect odd activity, HIDS keeps track of system events and keeps track of device artifacts, operations, and memory areas. Due to the employment of an anti-tampering mechanism, HIDS's adaptation is reliable despite problems with tampering assaults. In isolation, a host-based IDS is not the best option. It has serious drawbacks like high resource usage that impairs host performance. Such attacks might go undetected unless they manage to get into the target.

Network Intrusion Detection System (NIDS) : In order to examine traffic from all linked networks, NIDS are deployed at a preset location throughout the network. It analyses all the traffic that moves through the sub-net and locates an intrusion by comparing it to a database of abnormalities [9]. NIDS monitors the traffic while remaining unnoticed while looking for irregularities.

Protocol-based Intrusion Detection System (PIDS) : The protocol-based intrusion detection system (PIDS) consists of a system or agent that is permanently installed at the server's front end that controls and interprets the protocol used by users and devices to communicate with the server. It continuously monitors the HTTPS protocol stream and accepts the associated HTTP protocol in an effort to secure the web server [8].

Application Protocol-based Intrusion Detection System (APIDS) : A system or agent known as an application Protocol-based intrusion detection system (APIDS) typically lives within a server cluster. By observing and analyzing communication on application-specific protocols, it detects intrusions.

Hybrid Intrusion Detection System: A hybrid intrusion detection system is created by combining two or more intrusion detection system methodologies. The host agent or system data is merged with network data in the hybrid intrusion detection system to create a comprehensive picture of the network system [5]. In compared to conventional intrusion detection systems, the hybrid intrusion detection system is more effective.

5. INTRUSION DETECTION METHODS TYPES

IDS mainly relies on two approaches: signature-based detection and anomaly-based detection;

5.1 Signature-based intrusion detection

The amount of bytes, number of 1s, or number of 0s in the network traffic are only a few examples of the specific patterns that a signature-based IDS uses to identify attacks. Additionally, it identifies based on the malware's already-known harmful instruction sequence. Signatures are the patterns that the IDS has identified. The ultimate goal is to locate the breach using a library of recognized signatures from prior attempts. The main problem with signature-based intrusion detection is that it will miss new attacks that aren't already in the library if an attacker utilizes them.

5.2 Anomaly-based intrusion detection

The difficulties raised by signature-based intrusion detection are overcome by anomaly-based intrusion detection. The user's actions are what are used to detect intrusions [13]. The system's behaviour is modelled normally utilizing statistical methods and other techniques. An anomaly is when actual behaviour differs from what was expected. Anomaly-based intrusion detection does have some restrictions, though. For instance, it fails to recognize encrypted packets, creating a vulnerability. Also, it is quite difficult to create a normal model for huge amounts of dynamic data, which causes false alarms. In anomaly-based IDS, machine learning is used to build a reliable activity model that is compared to anything arriving and is labelled suspicious if it is not found in the model [4]. As these models can be trained based on the applications and hardware configurations, the machine learning-based technique has a better generic property than the signature-based IDS.

6. PRIVACY-PRESERVING FEDERATED LEARNING

The adoption of FL can address many IIoT privacy-specific difficulties, yet FL has some privacy flaws. For instance, using information from IIoT devices' local models, the global model automatically updates [3]. By using construction assaults, the adversary can launch an attack and obtain user information. Furthermore, the adversary can learn what kind of information is being shared by utilizing an inference attack by the bad user. Blood samples, the type of disease, and other pertinent information may be included. Differential privacy (DP) is a novel idea designed for these federated situations and is capable of offering a quantitative measure of data anonymization. Diverse strategies, such as distributed stochastic gradient descent and local and Meta differential privacy methods, are being investigated. These methods essentially introduce noise to protect user-data privacy during federated training. Although FL places a lot of emphasis on privacy, it is important to keep in mind that there is a trade-off between privacy protection settings and the convergence of the ML models during training. Better convergence results in less privacy. The different types of attacks are as follows;

Information Leakage: Information leakage is an application flaw when sensitive data, such as environment or user-specific information, or technical information about the online application, is exposed. An attacker may use sensitive data to take advantage of the target web application, its hosting network, or its users.

Poisoning attack: The adversary in this kind of attack modified the local model update parameters to lower the aggregator accuracy. The training data is affected in a data poisoning attack, whereas model parameters are attacked in a model poisoning assault.

Byzantine attack: In this attack, the adversary or malicious node that participates in FL communicates the fictitious model parameters with the nearby nodes. Moreover, the attacker may shorten the model's convergence time and accuracy by disseminating bogus data.

Privacy data leakage attack: The exposure of sensitive, confidential, or protected data to an unreliable environment is known as a data breach or data leak. Data leaks can happen as a result of hacker attacks, insider attacks by people who are presently or have previously worked for a business, or unintentional data loss or exposure.

Inference based attack: An inference attack happens when a person can guess more substantial knowledge about a database without actually accessing it [18]. Attackers use information at one security level to piece together facts that should be protected at a higher security level. Data mining is the main strategy used in this attack. The adversary uses data mining techniques during this assault to evaluate the data and extract some relevant information from it.

Differential privacy is the most well-known method that is frequently used to improve privacy preservation in numerous contexts [2]. Many researchers began developing differential privacy-based FL systems for the Industrial Internet of Things and many other important fields as a result of this property to prevent privacy leakage.

6.1 Privacy-Preserving FL-Enabled IDS for IIoT

According to the problem context, the coordinator chooses a subset of clients for the FL training process and provides them the parameters of a global model in a series of rounds. Each client then updates these parameters with their own data they have collected and sends them back to the coordinator. The entities combine the parameters they get from the clients using a certain aggregation method [6,15]. FL's key benefit is that parties don't have to exchange data in order to train a specific model.

6.1.1 DP-Enabled FL

The overall architecture of DP-enabled FL strategy for IIoT intrusion detection is shown in Fig. 1. A client is a device that serves as the final destination for local training. Each client is responsible for using its local data to train the global model given by the aggregator and producing a model update. Clients also possess the logic required to use the corresponding DP method [14]. The aggregator is also the primary service that accepts model updates from clients and creates an aggregated model that is provided back to clients for each training round.

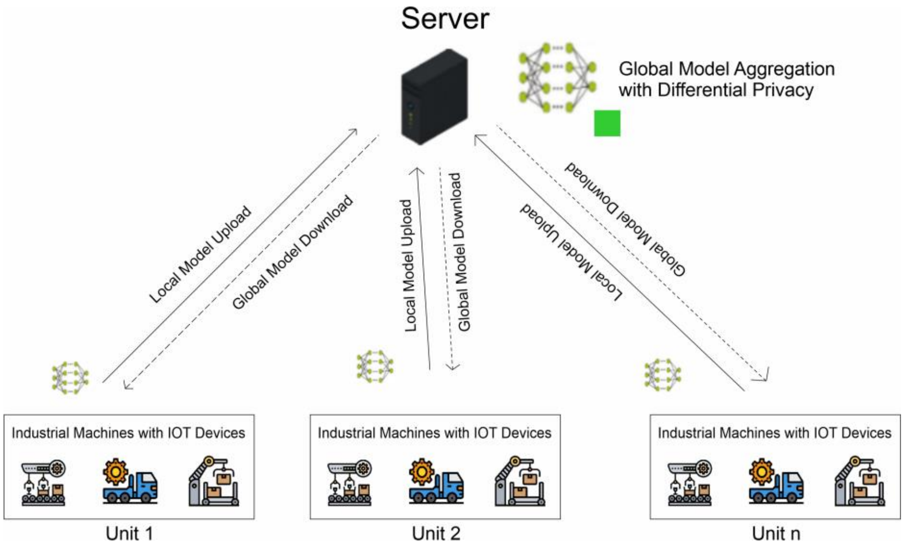
Server
Global Model Aggregation with Differential Privacy
Local Model Upload
Global Model Download
Local Model Upload
Global Model Download
Local Model Upload
Global Model Download
Industrial Machines with IOT Devices
Unit 1
Industrial Machines with IOT Devices
Unit 2
Industrial Machines with IOT Devices
Unit n

Figure 1: FL with DP architecture

6.1.2 Perturbation DP Mechanisms

The progression of anonymization techniques was from k-anonymity through l-diversity to t-closeness. Differential privacy aims to prevent any client model's output from revealing enough information about a specific person to enable identification. This implies that intruders cannot identify any information by adding a randomized value of noise. This means that no client model will allow an attacker to safely assume sensitive information. With two adjacent databases with just one unique member of difference, ß and ß′, differential privacy may be explicitly specified. Data is protected utilizing the noise addition technique in differential privacy by being perturbed using pre-defined techniques. In most differential privacy techniques, three noise addition processes are utilised. They are referred to as the Laplace mechanism (LM), the Gaussian mechanism, and the Exponential mechanism (EM). The real level of additional noise is directly proportional to the global sensitivity and privacy budget. In some cases, another concept known as privacy bound also contributes to noise addition. Unless a precise bound is necessary, the privacy bound is sometimes referred to as the privacy budget.

Laplace Mechanism: The noise is computed using the Laplacian function in the Laplace mechanism, and each coordinate of data is disturbed using the derived Laplacian noise from the LM distribution. The scale of noise supplied is determined by the sensitivity of the differential privacy function. The randomized procedure Å meets the ε-differential privacy parameter in a given dataset ß, function $\mathfrak{R}$, and global sensitivity δf_s, if the estimated noise value matches the real value of the Laplace distribution; that is, noise ~ Lap($\delta f_s / \varepsilon$). In most cases, LM is utilised to represent numerical output results.

$$\text{Å} = \mathfrak{R}(\mathfrak{R}) + Lap(\delta f_s / \varepsilon)$$

Exponential Mechanism: Exponential mechanism is a way for implementing differential privacy in the situation of non-numerical outputs. The exponential mechanism was created expressly for situations where the best reaction was required.

If, in a given dataset ß, $l \in \text{Ł}$ signifies a potential response, and a scoring function $u : \text{ß} \times \text{Ł} \rightarrow \text{Ł}$; and a randomized algorithm Å picks a probability based answer, then the supplied randomised method Å will meet the ε-differential privacy.

$$\text{Å}(\text{ß}, u) = l : |\mathbb{P}_{\mathbb{R}}\ [\mathrm{l} \in \text{Ł}]\ \infty \exp(\varepsilon u\ (\text{ß}, l)/)\ /\ 2\Delta u$$

In the preceding equation, u represents the sensitivity of the exponential scoring function. The value of u changes depending on the user's needs.

Gaussian Mechanism: Another important building component that is now being employed in the construction of differential privacy algorithms is the Gaussian mechanism. Noise in the Gaussian mechanism is estimated using a Gaussian distribution, similar to the Laplace mechanism. If the value of ε in a query function f is between 0 and 1, the outcome for Gaussian perturbation σ will be as follows:

$$\sigma = \frac{\Delta_2 f}{\varepsilon}\sqrt{2log(1.25\ /\ \varepsilon)}$$

6.2 Privacy Implications of Intrusion Detection Systems

Many devices and end users are protected from hostile actors by intrusion detection systems. While preventing the attacker from utilizing the victims' personal information for malicious purposes, the performance of these systems frequently depends on the type of data and architectural arrangement they use. The issues of security and privacy are introduced by the use of data for intrusion detection.

6.2.1 Type of Data

The network data that intrusion detection systems utilize to identify hostile actors can be used to classify them [1]. Correct data type identification has an impact on user privacy in addition to system performance. For instance, it is simple to identify the owner of the IP-address if the detection system uses it to examine the activity of a device. The challenge in this regard is two-fold: (1) Determining the data type that offers the best detection rate performance, and (2) Make sure network users' privacy is protected. For the purpose of identifying hostile actors, the current intrusion detection systems use two types of data. i.e., (1) application layer data logs, and (2) network traces data. The first form of data is data that was created at the application level and is typically connected to a particular kind of data set. This kind of information can be used to identify malicious and non-malicious devices by their fingerprints and can reveal details about the device architecture. The second data type, which includes far more information about the behaviour of the devices, is the IP traces of the network traffic [11]. The content or payload of information sent between devices, such as a temperature reading, a web page, or meta data, is another data type that can be employed.

6.2.2 System Architecture

One of two operating modes for an IoT intrusion detection system is either (1) independent or (2) collaborative. The traffic patterns seen locally within the network domain or Internet service provider are the basis for the stand-alone detection systems. Within a network of service providers, these systems operate independently. The stand-alone systems are easily defeated by stealth tactics and clever attackers who control attack traffic to one domain while concurrently attacking a huge number of other domains because they have no knowledge of how their users behave in other domains [18]. In order to assist the creation of a cooperative network, an effective intrusion detection system is expected to take into account the collective behaviour of nodes across several domains.

The two main categories of collaborative solutions are (1) Centralized, where alert data from domain collaborators is reported to a central system that categorizes traffic sender behaviour by examining traffic patterns across multiple domains, or (2) distributed or decentralized settings, where alert data from each service provider is shared and processed entirely decentralized without a central coordinator..

The major challenge towards the design of a collaborative IDS is regarding privacy protections for the data used for detection.

7. EMPLOYING FEDERATED LEARNING SYSTEMS FOR IDS SYSTEM

An artificial neural network called Long Short-Term Memory (LSTM) is employed in deep learning and artificial intelligence [10]. LSTM has feedback connections in contrast to traditional feed forward neural networks. Such a recurrent neural network (RNN) may analyze whole data sequences in addition to single data points (such as photos) (such as speech or video). Because of this feature, LSTM networks are perfect for handling and forecasting data. A Recurrent Neural Network (RNN) architecture type is a Gated Recurrent Unit (GRU). A GRU can analyze sequential data, including time series, voice, and spoken language, just like other RNNs. How the network manages information flow across time distinguishes a GRU from other RNN architectures, such as the Long Short-Term Memory (LSTM) network.

An artificial neural network called an autoencoder is utilised to provide effective codings for unlabeled input. It picks up two skills: encoding, which modifies the input data, and decoding, which reconstructs the input data from the encoded representation [17]. For dimensionality reduction, typically, the autoencoder learns an effective representation (encoding) for a set of data. There are variations that try to make the learnt representations take on beneficial features. Examples include Variational autoencoders, which have applications as generative models, and regularized autoencoders (Sparse, Denoising, and Contractive), which are efficient in learning representations for later classification tasks [17]. Autoencoders are used to solve a variety of issues, such as word meaning acquisition, feature detection, anomaly detection, and facial recognition. Moreover, autoencoders are generative models that have the ability to generate new data at random that is similar to the input data.

8. CHALLENGES

FL for IDS is vulnerable to excessive latency, false alarms, poisoning assaults, etc. starting with the broadcast of global models and continuing with the transmission of learned models. Hence, the main difficulties and problems with using FL for IDS are covered in this section.

8.1 Communication Overhead

The cost of communication every cycle of training is the main restriction for any FL application. Federated training necessitates the transmission of model parameters from the main server to a minimum of n clients, followed by the transmission of learned models from each client back to the server. Depending on the network bandwidth, the server traffic, packet transmission loss, and communication time for the parameters might all vary significantly. Also, the utilization of various devices in a network implies that each one's capacity for computation varies. Given all of these factors, it becomes sense that any federated network will have a low overall throughput in real-world applications. The need for effectiveness and speed becomes much more important in circumstances where the same technologies are used in intrusion detection applications. Consider a network with millions of continuously collecting, producing, and transmitting devices as an example. In this scenario, the server must manage a traffic load of at least 10,000 clients each broadcast, even if a client ratio of 1% is assumed. In most circumstances, it must not only wait for each client to communicate its parameters, but it must also aggregate all incoming clients at once [6]. In any case, the resulting server system eventually causes the federated architecture to bottleneck.

8.2 Federated Poisoning Attacks

The scattered nature of the data present in client edge devices is what gives federated architectures their supremacy [16]. The data in question is still at danger even though this attribute safeguards the privacy of the data while it is in transit and prevents its gathering in a single location. The labels of the data are simply editable in a client device. Poisoning attacks are what they are known as. The local and global models can be altered by a client to give forecasts that are tainted with poison if they pose as a benign participant. The worst-case scenario is that the global model either fails to train or exhibits a misleading performance when compared to the accuracy of the data it was trained on.

8.3 High False Alarms through the use of non-IID data

It is impossible to predict the characteristics of moving traffic by looking at a few samples. Network traffic has various qualities, many of which may be significant in some circumstances but may be insignificant in others, even though its properties are distinct. The difficulty of classification is increased by the complexity of intrusion detection data itself. As a result of the availability of data, FL is favorable; yet, the variance in the vast data itself may lead to inappropriate local and global model training, leading to a high number of false alarms. The size, kind, ambiguity, and complexity of the data differ in each edge device. As a result, locally trained models are lopsided. These high variance models would aggregate to an under-fitting of the overall model. The system may not be affected if one of the local models fails, but given the reality of real-time scenarios, the likelihood of numerous client failures is high.

8.4 Resource Management in Low Power IoT Devices

FL could make use of low-power IoT edge devices. In practice, the bulk of these IoT devices lack the processing power to train DL models in parallel and continuously. These IoT devices are so low-power that they hardly ever support pre-programmed algorithms or even adequate security measures. The majority of devices in this category might not be able to train the DL model locally, even though this makes them even more dependent on IDS. When used for training, they can quickly run out of energy or train slowly, which causes a significant amount of server delay. A complete FL edge device approach would also include communication, model training, data storage, and elicitation of data in addition to its regular operations [16]. In the worst-case scenarios, many devices could malfunction and cause a server stalemate. FL must be implemented in low-power devices using reliable energy-efficient algorithms.

8.5 Vulnerabilities in Intrusion Detection Setups

In order to protect a single workstation or a computer network, intrusion detection systems are often used. An FL design primarily uses DL algorithms to categorize or group the anomalous characteristics of the network packets in order to provide a warning. As a result, the IDS is exposed to several different vulnerabilities. These flaws may be devastating and dangerous to the integrity and confidentiality of the system that is being protected.

8.6 Precise Deployment of IDS

There are a huge amount of parameters and hyper-parameters used in deep learning models [14]. Some of these parameters and hyper-parameters can be trained, while others can be customized by the user. These user-defined parameters include initial weights, the learning rate, the number and size of hidden layers, the number of epochs per client model, etc. While others are training parameters, some of these parameters are model parameters. Each client in a federated learning scenario has their own model, making it impossible to fine-tune any of these parameters. The customer data is heterogeneous and unpredictable, thus using the same settings for each model would not produce the best outcome [12]. However, the data in the case of intrusion detection systems is quite unpredictable depending on the nature of assaults, their source, their intent, and a number of other aspects.

8.7 IoT Security Challenges

The term "Internet of Things" (IoT) refers to the interconnectedness of physical objects, such as vehicles, home appliances, and other goods, which enable these things to communicate and exchange data. These objects are embedded with electronics, software, sensors, and connectivity. The Internet of Things (IoT) idea entails expanding Internet connectivity to a variety of gadgets and common objects in addition to traditional devices like desktop and laptop computers, smartphones, and tablets. Offering enhanced device, system, and service connectivity that extends beyond machine-to-machine communications and encompasses a range of protocols, domains, and applications is the ultimate goal of the Internet of Things.

9. FUTURE DIRECTIONS

The future scope of development in the field is examined following a critique of FL-IDS systems and their problems in the preceding sections. Directions and potential solutions are offered for each of those problems. Several cutting-edge solutions appear practical and appropriate to the field of IDS when taking into account the most recent research and technology developments on FL systems.

9.1 Communication Efficient Federated IDS

The expense of transmitting large model parameters and the delay in packet transmissions are important aspects of a federated architecture's communication overhead. The final global model would get larger the more complicated the application. For every round, moving them to and from every customer uses a lot of energy [7]. Other compression and encryption standards can be used to get around this restriction. More crucially, using lightweight DL technology can help to increase prediction accuracy while also reducing the size of parameters that need to be sent.

9.2 Encryption Standards for Federated Learning

FL's main focus is privacy. One of FL's features is the transmission of data-related information through weights and their subsequent updating in the server model. The user's private information is protected because this transmission of weights incorporates rather than contains the user data. Yet, if a third party were to extract these weights, they might reflect sensitive data aspects that would be harmful. To resolve these problems, weight encryption is useful. The compression of the weights might be aided by specific lossy and lossless encryption functions. Although there may be some information loss, the bandwidth and transfer speed have greatly increased, enhancing the deep learning model's performance and overall throughput [10].

9.3 Edge Computing in FL based IDS

Our smartphones, laptops, and other low-power IoT gadgets are also considered edge devices. Each has a distinct processing capacity, which is reflected in the computation support they provide for federated systems. In these low power edge devices, effective resource management is desired to prevent this [8].

9.4 Implementation of FL through secure channels

Systems with intrusion detection are set up to shield the targeted systems from online threats. While the overall system performing better, the likelihood of intrusions increases when such IDS systems are deployed utilizing FL. This is because as the network grew, several weak points were created. This means that no protection method is 100% effective, and because FL is integrated, the system is at risk from an entire network made up of the server and many clients rather than just one network channel or virtual machine. To solve these problems, collaborative learning must be used to facilitate safe communication and business transactions.

9.5 Optimization of FL and IDS parameters

Deep learning models with a variety of trainable and user-defined parameters are used frequently by Federated Learning. Moreover, these values are very susceptible to intrusion detection systems [13]. A direct correlation exists between superior outputs and optimized training when these parameters are optimized. This aids in lowering the overall amount of space that the parameters occupy, increasing the bandwidth and lowering the cost of communication.

9.6 Efficient handling of non-IID data

Based on the usage of the customers' data and Federated Learning's experience, each client receives a personalized prognosis. In the real-time scenario, user-generated data is not homogeneous and is vulnerable to significant noise. Particularly in the case of IDS, the usage, timing, intent, and many other factors strongly affect the qualities and characteristics of the intercepted packets. Hence, it is impossible to assume that the data are consistent, identical, or unrelated to one another. These characteristics of the data sometimes lead to irregularities and problems throughout the client models' training and aggregation phases. In order to achieve adequate training and fault tolerance, processing of the non-IID data thus becomes a crucial component of FL frameworks.

10. CONCLUSION

In the current world, devices like mobile phones, smart home and office gadgets, wearable technology, and driverless cars all generate tremendous volumes of data every day. Such data, as well as the related equipment, must have their privacy secured. Machine learning-based intrusion detection systems (IDS) fill this need. IDS systems based on federated learning train high-quality centralized models while maintaining user privacy, however current frameworks still struggle to achieve these requirements since they must store and transfer data to a centralized server. In-depth examination of FL-based IIoT IDS systems is provided in the paper, with a focus on security, privacy, and dependability. The report also addresses the difficulties and shortcomings in FL implementations relating to high latency, false alarms, poisoning attacks, and other problems. How these issues adversely affect various aspects of IDS is also covered. The report concludes by outlining potential paths for future research and making suggestions for potential fixes to the issues with Federated Learning-based IDS implementations in IIoT.

REFERENCES

[1] A. Kenyon, L. Deka et al., "Are public intrusion datasets fit for purpose characterising the state of the art in intrusion event datasets", Elsevier, Computers & Security 99 (2020), 102022.

[2] Anwer Mustafa Hilal, Jaber S. Alzahrani et al., "Intelligent Deep Learning Model for Privacy Preserving IIoT on 6G Environment", Computers, Materials & Continua, https://doi.org/10.32604/cmc.2022.024794

[3] Attila Franko, Gergely Hollosi et al., "Applied Machine Learning for IIoT and Smart Production—Methods to Improve Production Quality, Safety and Sustainability", MDPI, Sensors, 2022, 22, 9148, https://doi.org/10.3390/s22239148.

[4] E. G. Dada, J. S. Bassi et al., "An Investigation into the Effectiveness of Machine Learning Techniques for Intrusion Detection", Arid Zone Journal of Engineering, Technology and Environment, December, 2017; Vol. 13(6):764-778.

[5] Hakan Can Altunay, Zafer Albayrak, "A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks", Engineering Science and Technology, an International Journal 38 (2023) 101322, https://doi.org/10.1016/j.jestch.2022.101322.

[6] Hongbin Fan, Changbing Huang et al., "Federated Learning-Based Privacy-Preserving Data Aggregation Scheme for IIoT", IEEEAccess, Volume 11, 2023, https://10.1109/ACCESS.2022.3226245.

[7] Junaid Arshad, Muhammad Ajmal Azad et al., "A Review of Performance, Energy and Privacy of Intrusion Detection Systems for IoT", MDPI, Electronics 2020, 9, 629; https://doi:10.3390/electronics9040629.

[8] Khaled Ali Abuhasel, Mohammad Ayoub Khan, "A Secure Industrial Internet of Things (IIoT) Framework for Resource Management in Smart Manufacturing", IEEEAccess, volume 8, 2020, pp.117354-117364, https://doi:10.1109/access.2020.3004711.

[9] Khalid Albulayhi, Abdallah A. Smadi et al., "IoT Intrusion Detection Taxonomy, Reference Architecture, and Analyses", MDPI, Sensors 2021, 21, 6432. https://doi.org/10.3390/s21196432.

[10] Kuruva Lakshmanna, R. Kavitha et al., "Deep Learning-Based Privacy-Preserving Data Transmission Scheme for Clustered IIoT Environment", Hindawi, Computational Intelligence and Neuroscience, Volume 2022, Article ID 8927830, 11 pages, https://doi.org/10.1155/2022/8927830.

[11] Md Mamunur Rashid, Shahriar Usman Khan et al., "A Federated Learning-Based Approach for Improving Intrusion Detection in Industrial Internet of Things Networks", MDPI, Network 2023, 3, 158–179, https://doi.org/10.3390/network3010008.

[12] Nasr Abosata, Saba Al-Rubaye et al., "Customised Intrusion Detection for an Industrial IoT Heterogeneous Network Based on Machine Learning Algorithms Called FTL-CID", MDPI, Sensors 2023, 23, 321. https://doi.org/10.3390/s23010321.

[13] Pedro Ruzafa-Alcazar, Pablo Fernandez-Saura et al., "Intrusion Detection Based on Privacy-Preserving Federated Learning for the Industrial IoT", IEEE Transactions on Industrial Informatics, Vol. 19, No. 2, February 2023, https://doi.org/10.1109/TII.2021.3126728.

[14] Priyanka Verma, John G. Breslin et al., "FLDID: Federated Learning Enabled Deep Intrusion Detection in Smart Manufacturing Industries", MDPI, Sensors 2022, 22, 8974, https://doi.org/10.3390/s22228974.

[15] Swarna Priya R.M., Praveen Kumar Reddy Maddikunta et al., "An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture", Computer Communications 160 (2020) 139–149, https://doi.org/10.1016/j.comcom.2020.05.048.

[16] Taimur Hafeez, Lina Xu et al., "Edge Intelligence for Data Handling and Predictive Maintenance in IIoT", volume 9, 2021, pp.49355-49371, https://10.1109/ACCESS.2021.3069137.

[17] Wumei Zhang, Yongzhen Zha, "Intrusion Detection Model for Industrial Internet of Things Based on Improved Autoencoder", Hindawi, Computational Intelligence and Neuroscience, Volume 2022, Article ID 1406214, 8 pages, https://doi.org/10.1155/2022/1406214.

[18] Xiuzhang Yang, Guojun Peng et al., "An Enhanced Intrusion Detection System for IoT Networks Based on Deep Learning and Knowledge Graph", Hindawi, Security and Communication Networks, Volume 2022, Article ID 4748528, 21 pages, https://doi.org/10.1155/2022/4748528.