International Journal for Electronic Crime Investigation

ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

DOI: https://doi.org/10.54692/ijeci.2024.0804209

Research Article

Vol. 8 issue 4 Oct-Dec 2024

Cyber Security Intrusion Detection Scheme for Malicious Traffic in IoT using Federated Learning

Shahan Yamin Siddiqui[1], Muhammad Farrukh Khan[1], Rabia Tehseen[2], Unaiza Rehman[1], Nusratullah Tauheed[3], and Muhammad Toseef Javaid[3]

[1]Department of Computing, NASTP Institute of Information Technology, Lahore, Pakistan
[2]Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[3]Department of Computer Science, University of South Asia, Cantt Campus, Lahore, Pakistan.
Corresponding Author: drshahan@niit.edu.pk

Received: October 06, 2024; **Accepted:** October 20, 2024; **Published:** December 17, 2024

ABSTRACT

With the rise in cyberattacks, Internet of Things (IoT) devices are increasingly vulnerable to malware, security threats, and suspicious activities. Traditional research has mainly focused on centralized intrusion detection systems in cyber security field. However, these centralized methods often struggle to keep pace with the rapid evolution of digital and mobile technologies and carry the risk of a single point of failure, jeopardizing data security and privacy. To enhance network protection, intrusion detection can benefit from the use of federated learning (FL). FL is a collaborative machine learning approach that allows for model testing without the need to share sensitive local data. Instead, computations are performed directly on distributed end devices, preserving data privacy and addressing concerns related to data ownership, confidentiality, computational efficiency, and storage limitations. Unlike traditional centralized machine learning, FL processes data where it resides, leading to improved security and efficiency. Previous studies on federated learning have examined the challenges posed by non-independent and non-identically distributed data. Implementing FL algorithms in intrusion detection focuses on monitoring routers, detecting intrusions, and analyzing user activity patterns.

By incorporating federated learning into intrusion detection systems, network security can see significant enhancements. Experimental research utilizing network intrusion datasets indicates that the Deep Extreme Learning Machine (DELM), when paired with the CSIDS-FL system model, achieves an impressive accuracy rate of 94.23%, surpassing earlier models and demonstrating the effectiveness of this method.

Keywords: Cyber security, Federated learning; internet of things; intrusion detection; deep extreme learning machine

1. INTRODUCTION

Intrusion detection involves a system or software that keeps an eye on networks for harmful or unauthorized activities. The aim of employing a Federated Learning algorithm for intrusion detection is to effectively monitor routers, identify potential intrusions, and observe network behavior. By incorporating Federated Learning, intrusion detection systems can become much more efficient and secure. This approach specifically targets the management of malicious traffic in IoT environments through Federated Learning-based intrusion detection. It functions by blocking access when an attack is detected and permitting access only when no threats are present. Recently, Federated Learning has proven to be valuable in improving intelligent intrusion detection systems, providing an effective way to handle malicious traffic.

Many intrusion detection systems utilize signature analysis in network traffic to spot recognized threats. These cyber security-based detection systems are often used to identify and estimate malicious activities across cloud-based network endpoints, reducing reliance on integrated IDSs that correlate security data and alert events. Detecting network intrusions is crucial for protecting IT infrastructure from suspicious online traffic. Recently, intrusion detection systems that incorporate Artificial Intelligence (AI) have demonstrated improved precision and efficiency compared to traditional methods. However, AI-based detection can occasionally face challenges with accuracy, as these systems may still generate a confidence score for each data instance, even when there is uncertainty in the detection process.

The effectiveness of traditional machine learning (ML) models largely depends on the computational power and data sets stored on a centralized server. In these standard ML configurations, client data is kept in one place and utilized for both training and testing, leading to the development of extensive ML frameworks. However, this centralized method brings several challenges, such as concerns over computational resources, storage requirements, and the safeguarding of users' private information, which are often overlooked. Recently, Federated

Learning (FL) has emerged as a promising alternative to address these issues. FL allows for the use of artificial intelligence in settings where data privacy and uniformity are essential by moving computation from centralized servers to user devices, ensuring that sensitive information stays protected.

Intrusion detection systems are frequently used to monitor and identify malicious activities at cloud network endpoints, reducing the necessity for fully integrated intrusion detection systems that merge security information with event alerts. When compared to traditional methods, intrusion detection systems that utilize Artificial Intelligence (AI) have shown to be more accurate and efficient. Nevertheless, these AI-driven systems can still encounter challenges regarding detection precision. This is due to the requirement for AI models to deliver a confidence score for each incoming instance, even in cases where there is uncertainty in the evaluation.

The rapid growth of Internet of Things (IoT) technology has resulted in its widespread adoption, greatly improving our daily lives through various household applications. However, the decentralized nature, vast number of connected devices, and ease of access make IoT systems particularly vulnerable to cyberattacks. Additionally, many IoT networks gather, store, and process sensitive personal data, which attracts cybercriminals. Thus, securing IoT networks is a crucial priority for their successful implementation [1].

IoT technology comprises a network of interconnected devices, including sensors, detectors, and communication components. These systems can monitor their environment and transmit data wirelessly, facilitating informed decision-making. In recent years, IoT has gained considerable attention and has become essential for numerous innovative sectors, such as smart cities, industrial automation, and home automation. By 2020, it was estimated that the number of IoT devices in use worldwide would reach into the billions [2].

Recognizing the risks associated with IoT and understanding the current defense strategies is crucial. This includes introducing and categorizing traditional defense methods alongside various types of IoT security threats to equip readers with essential safety knowledge. IoT security presents unique challenges that differ from those of conventional network security. Here are some reasons why IoT security measures are distinct from traditional network security:

IoT devices typically have limited computational power, storage capacity, battery life, and network connectivity, which makes it challenging to implement resource-heavy security solutions.

The decentralized and diverse nature of IoT networks makes centralized security mechanisms less effective and complicates threat management.

IoT systems often operate in dynamic

environments, making them vulnerable to physical tampering and attacks from malicious actors.

IoT devices are connected through the Internet and can be accessed via IP addresses, which introduces new types of Internet-based threats.

The sheer number of IoT devices generates a significant amount of data traffic. With limited bandwidth, these networks can become congested, making them more attractive targets for attackers.

IoT threats can be categorized in two primary ways: one based on the architectural layers of an IoT system and the other based on specific design challenges related to IoT attacks [3].

The IoT architecture is generally structured into three primary layers: the hardware layer, the network layer, and the application layer. The first layer, often referred to as the hardware or perception layer, includes various sensors and devices that gather and transmit data using communication standards such as Bluetooth, Radio Frequency Identification (RFID), and 6LoWPAN. The second layer, the network layer, is tasked with efficiently routing and transmitting data throughout the system, employing communication protocols like Wi-Fi, 5G, GSM, and IPv6. The 3rd layer, known as the application or software layer, is where business logic is applied, and user interfaces are created for end users, facilitating services like traffic monitoring [3].

To gain a clearer understanding of the security threats linked to the Internet of Things (IoT), we will start by defining important terms associated with IoT attacks, followed by an exploration of the challenges that IoT systems encounter.

Spoofing and identity fraud attacks aim to gain unauthorized access to services by stealing login credentials, such as usernames and passwords, through various methods. These credentials can be taken directly from a device, intercepted during transmission, or acquired through cyber-attacks [4]. Common types of spoofing include IP address spoofing, ARP spoofing, and DNS server spoofing. ARP spoofing targets the resolution protocol, which maps IP addresses to Media Access Control (MAC) addresses. Attackers can send fake ARP messages across the LAN allowing them to intercept or modify data. DNS spoofing changes the DNS server address to redirect traffic to an unauthorized server [5].

Routing attacks manipulate routing protocols by falsifying or replaying routing information, which can lead to incorrect data transfer patterns. Sinkhole attacks trick the system into sending excessive traffic to a malicious node by presenting an invalid path as the best route. Selective forwarding attacks involve an attacker transmitting malicious data while blocking legitimate traffic, disrupting the flow of information within the system [6].

Information disclosure attacks focus on unauthorized access to sensitive data. This can happen through eavesdropping

on network traffic or physically compromising a system, such as through port scanning. Sensitive data leakage, like in side-channel attacks, can result from information disclosure [7].

Distributed Denial of Service (DDoS) attacks are executed by a network of compromised devices spread across different locations. A Denial of Service (DoS) attack aims to overwhelm a system with malicious traffic, which consumes resources and disrupts normal operations [8]. DDoS attacks are often conducted using botnets, which are collections of infected devices. These types of attacks are common in IoT and cloud systems, especially in settings like smart cities [9].

Information disclosure attacks can jeopardize user privacy without needing direct access. Hackers can collect sensitive information by examining network traffic and metadata.

Duplicated device attacks involve creating replicas of network devices or system components by stealing their credentials. Attackers can then gain control of the device, inject false information, and disable its functions. This kind of attack can result in network infection without the user's awareness [10].

IoT threats can be classified according to design challenges. Given the complexities of creating IoT systems, it is crucial for developers and industries to take into account the potential risks and implications. Numerous research articles have examined security issues and research opportunities in the IoT [11], focusing on topics like object detection, access control, and the confidentiality of IoT data.

Server-side IoT solutions require the integration of sensing devices, controllers, and routers from various vendors, each possibly running on different versions. To manage this effectively, a strong system is essential to ensure that diverse devices can work together seamlessly [12]. In this kind of environment, the risk of malicious activities such as DoS attacks, spoofing, routing attacks, and Man-in-the-Middle (MITM) attacks is higher than in uniform systems.

In the Internet of Things, it's crucial to establish connectivity among various system components, whether for physical connections or ensuring that services remain available. Peripheral devices, like sensors, need to connect to an IP network via bridging devices. However, this configuration can introduce vulnerabilities, including routing issues and Man-in-the-Middle (MITM) attacks. Furthermore, any changes in service provision must be communicated to connected devices to avoid overwhelming the system with requests for unavailable services, which could lead to a Denial of Service (DoS) attack due to excessive traffic [7].

IoT devices are typically mobile and frequently switch their connections to different network bridges. This mobility can result in service

interruptions or unauthorized connections. As a result, various types of attacks can occur in these dynamic environments, including Denial of Service (DoS), Man-in-the-Middle (MITM), wormhole, and blackhole attacks [8].

In 2017, Google introduced federated machine learning to improve privacy protection. Traditional machine learning methods usually depend on centralized data training, where all information is processed on a single computer, which raises significant privacy concerns for users. In contrast, federated learning employs a distributed approach for data training, enabling end devices in various locations to work together and train a machine learning model without centralizing sensitive data. This method also utilizes virtualized resources to manage complex tasks, ensuring confidentiality by design. Federated learning is particularly relevant in environments where AI, blockchain, cloud-based applications, and the Internet of Things intersect [1, 7].

FL is quickly becoming a popular solution for training machine learning models in distributed environments. Instead of sending training data to a central server, FL allows the model's parameters, such as neural network weights and biases, to be updated collaboratively by numerous internet-connected devices acting as local trainees. This approach is especially suitable for IoT devices with limited power and unreliable network connections, as it eliminates the need to share data with third-party entities, thus preserving privacy [12].

While deep learning and machine learning have made impressive progress in tackling real-world challenges, they still encounter several limitations:

- Training centralized models necessitates that users upload their private data to a central system.
- As the network expands, system performance often declines, which can lead to breakdowns that impact service accuracy and reliability.
- Intrusion Detection Systems (IDS) need rapid detection, but centralized processing can cause delays.
- IoT devices frequently gather sensitive user data, heightening the risk of privacy violations.
- The system incorporates a range of datasets (including text, recordings, live streams, and AR/VR), making data collection in 5G/6G systems both time-consuming and expensive.

Federated Learning (FL) presents a solution to these issues. It addresses multiple challenges by ensuring that private information remains secure while delivering widespread intelligence. FL also facilitates large-scale communication, rapid response times, and energy-efficient solutions, making it well-suited for dynamic, time-sensitive applications. This capability paves the way for advancements in machine learning and

deep learning within intrusion detection systems, necessitating the use of optimization methods to fully leverage these features [13].

2. RELATED WORK

The study highlights the critical need for privacy preservation to safeguard sensitive data transactions from unauthorized access. Intrusion Detection Systems (IDS) play a vital role in protecting cloud computing environments from suspicious activities [10]. Typically positioned as the first line of defense in distributed cloud settings, IDSs tackle both external and internal threats. They continuously monitor and analyze systems for potential security risks and policy breaches, and can be implemented as either hardware devices or software applications. A considerable amount of intrusion detection datasets is derived from network and web logs [14].

In this research, deep learning techniques are utilized to improve the effectiveness of IDSs across different systems. These methods facilitate the automatic extraction of feature representations from large datasets to assess attack activities. With the increasing availability of computing power, algorithms like recurrent and convolutional neural networks (CNNs) are becoming more popular in both qualitative and quantitative learning frameworks for identifying suspicious activities. These models are particularly adept at analyzing sequential data from cloud networks. Although deep learning models typically provide better prediction performance than traditional models (like linear regression), they often face criticism for their "black-box" nature [15].

To tackle privacy issues, Google launched federated machine learning in 2017, enabling data to stay on the client's side while still aiding in the creation of global models [3, 16]. Although there is increasing interest in this field, research specifically focusing on the use of federated learning in intrusion detection systems (IDS) remains scarce. Nonetheless, several studies have explored federated learning and IDS independently. Recent survey papers have outlined and compared different IDS models that utilize deep learning, offering a comprehensive taxonomy based on essential characteristics such as input methods, sensor techniques, output implementations, and performance evaluation approaches.

The main contributions of this study are as follows:

- The proposed model employs federated learning to identify and address malicious activities within an IoT intrusion detection system.
- It safeguards sensitive information in IoT networks while also categorizing different types of network intrusions.
- The federated learning-based model is evaluated against other leading techniques, such as deep auto encoders [17, 12], Self-Organizing Maps [17], Artificial Neural Network-based Intrusion Detection Systems [15], Generative Adversarial Networks [5], deep models utilizing GANs [13], enhanced Backpropagation neural networks [17], and Conditional Variational Autoencoders [18].

3. PROPOSED CYBER SECURITYINTRUSION DETECTION MODEL

This section offers a detailed explanation of the methodology employed in the proposed system. It presents an overview of the Intrusion Detection Scheme designed to identify malicious traffic in IoT networks through the use of Federated Learning.

Federated Learning is a collaborative and distributed approach to data learning, where participants exchange only their locally trained models rather than sharing raw data. Although current FL solutions are becoming increasingly popular, the distribution of traffic data in network intrusion detection systems (NIDS) does not always conform to a single, unified FL framework. Some systems exhibit common characteristics, while others do not.

Federated Learning is a collaborative machine learning approach that enables organizations to train models together while keeping sensitive data private. In this method, once local data training is finished, devices connect to a central server to send their model parameters (weights). The server gathers these parameters, updates the overall model, and then returns the updated weights to the devices. Each device then applies the new model parameters for its specific tasks.

In certain machine learning models, data is stored in the cloud and processed through a database after being trained locally on neural networks. Initially, the data undergoes pre-processing and filtering before being fed into the trained model for anomaly detection. The model functions in two phases: if an attack is detected, access is denied; if no attack is found, access is allowed. This cycle is repeated, with each iteration enhancing the model's accuracy over time [16].

The aim of employing the Federated Learning (FL) algorithm for intrusion detection is to oversee routers, spot potential intruders, and monitor their activities. Federated learning improves the efficiency of intrusion detection systems, which are essential for ensuring network security. This study utilized a dataset from Kaggle because there was no access to real-time data for intrusion detection within the framework of Federated Learning [11, 12].

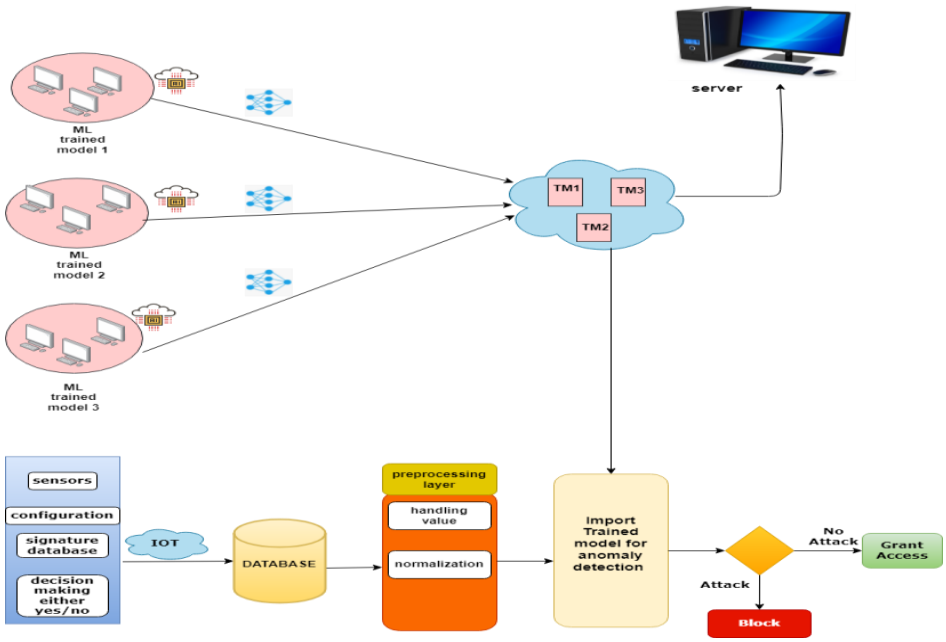

Figure 1: Proposed Cyber Security Intrusion Detection Scheme Model

4. DEEP EXTREME LEARNING MACHINE (DEEP ELM)

Deep Extreme Learning Machine (Deep ELM) is a machine learning approach that improves data analysis by increasing both the number of hidden layers and the size of the neurons. With the rise of GPU capabilities, deep learning has gained significant popularity. The effectiveness and precision of classifier models in deep learning are influenced by the size of the neurons in each layer and the number of hidden layers. Deep Extreme Learning Machines (Deep ELM) serve as a powerful method for efficiently addressing classification problems. The structure of DELM includes an input layer, multiple hidden layers, and an output layer, as illustrated in Figure 2. The input layer takes in various data inputs, and the proposed CSIDS-FL model incorporates 6 hidden layers. Equation (1) illustrates the input layer in the mathematical framework of Deep Extreme Learning Machine (DELM),

$$a_q = m_1 + \sum_{l=1}^{n} (b_{lq} * y_l)$$

while Equation (2) depicts the output following the computations of the first layer.

$$r_q = \frac{1}{1 + e^{-a_q}},$$
$$\text{where } q = 1, 2, 3 \ldots, z.$$

Equation (3) illustrates how information flows from the second layer to the output layer:

$$a_{u^q} = m^i + \sum_{q=1}^{s} \left(v_{qu^{i=1}} * r_q^{i=1} \right)$$

Equation (4) illustrates the activation function utilized in the output layer:

$$r_{u^i} = \frac{1}{1 + e^{-a_u^{i-1}}} \quad \text{where } u = 2,3 \ldots z,$$

$$a_{uq}^{u} = m^i + \sum_{q=1}^{s} \left(v_{qu^i} * r_q^i \right) \text{ Where } i = 1,2,3 \ldots \ldots 6$$

Equation (6) shows how to calculate the error for anomaly detection in the following way:

$$€ = \frac{1}{2} \sum_u (\mathbf{Target_u} - \mathbf{r_u^{i=6}})^2$$

Equation (7) illustrates the ratio of weight changes for the output layer:

$$\Delta \mathbf{b} \propto -\frac{\partial €}{\partial \mathbf{b}}$$

$$\Delta \mathbf{r_{q,u^{i=6}}} = -\epsilon \frac{\partial €}{\partial \mathbf{r_{q,u}^{\,i=6}}}$$

Equation (8) demonstrates how the chain rule is applied in this context:

$$\Delta \mathbf{r_{q,u^{i=6}}} = -\epsilon \frac{\partial €}{\partial \mathbf{r_u}^i} \times \frac{\partial \mathbf{r_u}^i}{\partial \mathbf{a_{u^i}}} \times \frac{\partial \mathbf{a_{u^i}}}{\partial \mathbf{r_{q,u^i}}}$$

By using the chain rule, the adjustments to the weights can be determined as shown in Equation (9), after modifying Equation (8).

$$\Delta r_{q,u^{i=6}} = \epsilon(\text{Target}_u - r_u^i) \times r_{u^i}(1 - r_{u^i}) \times (r_q^i),$$

$$\Delta r_{q,u}^i \epsilon \mu_{u^i} r_q^i,$$

where

$$\Delta \mathbf{b_{l,q^i}} \propto -\left[\sum_u \frac{\partial €}{\partial \mathbf{r_{u^i}}} \times \frac{\partial \mathbf{r_{u^i}}}{\partial \mathbf{a_{u^i}}} \times \frac{\partial \mathbf{a_{u^i}}}{\partial \mathbf{r_{u^i}}}\right] \times \frac{\partial \mathbf{r_{u^i}}}{\partial \mathbf{a_{u^i}}} \times \frac{\partial \mathbf{a_{u^i}}}{\partial \mathbf{b_{l,q^i}}}$$

where

$$\mu_{q^i} = \left[\sum_u \mu_{u^i} \left(x_{q,u^i}\right)\right] \times r_{q^i}(1 - r_{q^i})$$

The weights in Equation (10) represent the enhancements and biases between the output and hidden layers.

$$\mathbf{x}_{\mathbf{q,u^{i=6}}}^{+} = \mathbf{x_{q,u^{i=6}}} + \boldsymbol{\delta}_{\mathbf{e^{i=6}}} \Delta \mathbf{r_{q,u^{i=6}}}.$$

The adjustments in weight and bias between the input and hidden layers are represented in Equation (11).

$$\mathbf{b}_{\mathbf{l,q^i}}^{+} = \mathbf{b_{l,q^i}} + \boldsymbol{\delta}_{\mathbf{e^i}} \Delta \mathbf{b_{l,q^i}}$$

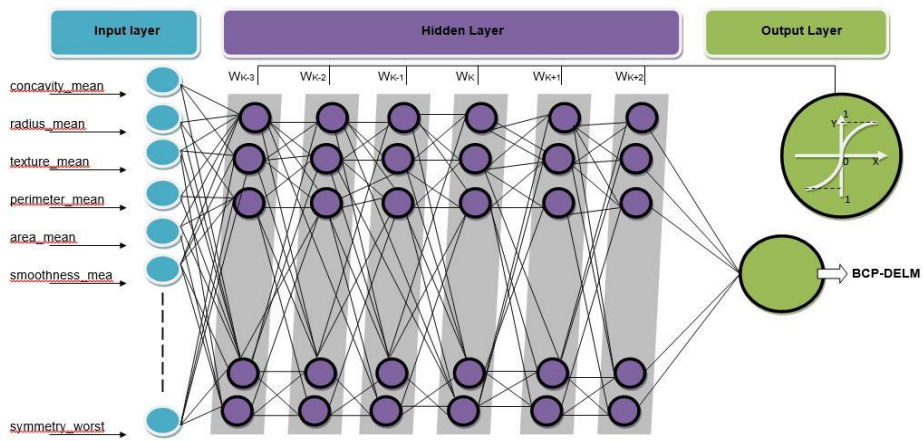

Figure 2: Deep extreme learning machine (Deep ELM) in Proposed Cyber Security Instrusion Detecion Scheme

5. DISCUSSION

The proposed FL based Cyber Security Intrusion Detection Scheme for Malicious Traffic in IoT employs MATLAB 2020a for calculating results and classification. The dataset is categorized into two groups: Normal and Anomaly. A total of 47,736 instances were utilized, with 22,249 classified as Anomaly and 25,487 as Normal. The Deep Extreme Learning Machine (DELM) model for the CSIDS-FL system is executed in two phases: training and validation. In the training phase, 80% of the input instances from each class are chosen, while the remaining 20% are reserved for validation. The model's performance is assessed using various statistical metrics, including Accuracy, Miss Rate and etc.

$$\textbf{Miss Rate} = \frac{\left({}^{O_A}/_{I_N} + {}^{O_N}/_{I_A}\right)}{{}^{O_N}/_{I_N} + {}^{O_A}/_{I_N} + {}^{O_A}/_{I_A} + {}^{O_N}/_{I_A}} \times 100$$

$$\textbf{Accuracy} = \frac{\left({}^{O_N}/_{I_N} + {}^{O_A}/_{I_A}\right)}{{}^{O_N}/_{I_N} + {}^{O_A}/_{I_N} + {}^{O_A}/_{I_A} + {}^{O_N}/_{I_A}} \times 100$$

$$\textbf{Sensitivity} = \frac{{}^{O_N}/_{I_N}}{{}^{O_N}/_{I_N} + {}^{O_N}/_{I_A}} \times 100$$

$$\textbf{Specificity} = \frac{{}^{O_A}/_{I_A}}{{}^{O_A}/_{I_A} + {}^{O_A}/_{I_N}} \times 100$$

The proposed **CSIDS-FL** model for the cyber security Intrusion Detection categorizes the data into two types: Normal and Anomaly.

The input data for the proposed CSIDS-FL system is shown in Table 1. It consists of 47,736 instances, with 80% (38,189 instances) designated for training and the remaining 20% (9,547 instances) set aside for validation. The dataset is divided into two categories: 22,249 instances belong to the Anomaly class, while 25,487 instances are classified as Normal.

Table 2 shows the predictive performance of the proposed DELM model for the CSIDS-FL system during the training phase. A total of 38,189 input instances are divided into two categories: 20,390 for the Normal class and 17,799 for the Anomaly class. In the Normal class, 19,370 instances were correctly predicted, while 1,020 instances were misclassified as Anomaly. For the Anomaly class, 16,900 instances were accurately predicted, and 899 instances were incorrectly classified as Normal.

Table 3 shows the prediction results from the validation phase for the proposed CSIDS-FL system utilizing the DELM model. A total of 9,547 input instances were analyzed, comprising 5,097 instances for the Normal class and 4,452 for the Anomaly class. In the Normal class, 4,801 instances were correctly identified as Normal, while 296 were mistakenly classified as Anomaly. For the Anomaly class, 4195 instances were accurately predicted, with 257 being incorrectly categorized as Normal.

Table 4 presents the performance metrics of the DELM model for the

proposed CSIDS-FL system during both the training and validation phases. In the training phase, the model reached an accuracy of 94.97%, with a miss rate of 2.03%. It demonstrated a sensitivity of 95.56% and a specificity of 94.31%. The positive predictive value was 94.99%, while the negative predictive value stood at 94.95%. During the validation phase, the model achieved an accuracy of 94.23%, with a miss rate of 5.77%. Its sensitivity was 94.92%, and specificity was 93.41%. The positive predictive value was 94.19%, and the negative predictive value was also 94.23%.

We assessed the effectiveness of our method by comparing it with other established algorithms. As illustrated in Table 5, our proposed model shows notable improvements in accuracy while keeping a lower error rate. Specifically, the Deep Extreme Learning Machine (DELM) combined with the CSIDS-FL system outshines other approaches like the Conditional Variational Autoencoder [18, 19] and Generative Adversarial Networks [5]. When compared to other deep learning models, the DELM with the CSIDS-FL system exhibits greater efficiency, whereas the precision of the Conditional Variational Autoencoder [18] is significantly lower. Research in [13] utilizing deep models with Generative Adversarial Networks achieved an accuracy of around 80%. The Self-Organization Map introduced by [17] reached a precision of 75.5%, while the enhanced Backpropagation Neural Network in [18, 20] attained a precision of 93.31%. The Discriminative Multinomial Naïve Bayes model discussed in [8, 19-23] achieved an accuracy of 81.5%, and Generative Adversarial Networks in [5] produced an accuracy of 86.5%. Our DELM with the CSIDS-FL system boasts an accuracy of 94.54%, exceeding previous models and showcasing its superior performance.

There are several approaches that can help address the growing challenges in creating smart and autonomous management systems [24-28]. Common techniques include fuzzy logic, machine learning [29, 31-35, 37-41], soft computing [30, 37], Particle Swarm Optimization (PSO) [33, 42-43], computational intelligence [34, 36], round robin scheduling, equalization methods, and explainable artificial intelligence [44-47]. These methods are frequently used in the development of these sophisticated, intelligent frameworks.

Table 1: Input records for proposed CSIDS-FL model

Input Instances	Anomaly	Normal
Input Data for Training	17799	20390
Input Data for Validation	4450	5097
Total Inputs Records	22249	25487

Table 2: Decision matrix for CSIDS-FL model (Training)

Proposed CSIDS-FL model (Training)			
Input Samples = 38189 (80% Training Instances)		**Output (O_N, O_A)**	
		O_N	**O_A**
Input Instances	**I_N** = 20390	19370	1020
	I_A = 17799	899	16900

Table 3: Decision matrix for CSIDS-FL model (Validation)

Proposed CSIDS-FL Model (Validation)			
Input Samples = 9547 (20% Validation Instances)		**Output (O_N, O_A)**	
		O_N	O_A
Input Instances	I_N = 5097	4801	296
	I_A = 4452	257	4195

Table 4: Performance Evaluation of the proposed CSIDS-FL model

Overall Performance of CSIDS-FL Model (Training and Validation)						
Performance Parameters	Accuracy	Miss Rate	Sensitivity	Specificity	Positive Predictive value	Neg Predictive value
Training	94.97%	5.03%	95.56%	94.31%	94.99%	94.95%
Validation	94.23%	5.77%	94.92%	93.41%	94.19%	94.23%

Table 5: Comparison with literature for the proposed CSIDS-FL model

Literature	**Performance**
Self-Organization Map [17]	75.5%
Artificial Neural Network-based Intrusion Detection System [26]	81.2%
Conditional variational autoencoder [27]	71%
Discriminative multinomial naive Bayes [8]	81.5%
Generative Adversarial Networks [5]	86.5%
Deep models under the GAN: [25]	80%
improved Back Propagation neural network [18]	93.31%
Proposed CSIDS-FL Model	94.23%

6. CONCLUSION

Federated learning allows for the development of a scalable model that can be utilized by multiple institutions, ensuring local data privacy while enabling collective benefits and maintaining data integrity. To analyze traffic patterns for intrusion detection, features associated with malware detection were chosen. The goal was to uphold client confidentiality while implementing an intrusion detection system aimed at identifying malicious traffic. The proposed algorithm emphasizes the management of malicious traffic through intrusion detection in IoT systems using federated learning. The key innovation of this study is the simplification of data exchange between cloud systems, ensuring both security and reliability. Federated machine learning has demonstrated considerable potential in enhancing intelligent cyber security intrusion detection systems for

malicious traffic. The Deep Extreme Learning Machine (DELM) combined with the CSIDS-FL system model achieves an accuracy rate of 94.23%.

Future research could aim to enhance the scalability and resilience of the proposed intrusion detection system (IDS) by integrating advanced machine learning techniques, such as reinforcement learning, to facilitate real-time threat responses. Exploring the application of Federated Learning across various IoT environments with differing data types and network architectures could further improve detection accuracy and adaptability. Moreover, merging edge computing with Federated Learning might reduce latency and boost system performance in dynamic IoT settings.

REFERENCES

[1] J. Zhang, F. Li, and F. Ye, "An ensemble-based network intrusion detection scheme with Bayesian deep learning," *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, pp. 1-6, 2020.

[2] J. Arshad, M. A. Azad, M. M. Abdeltaif, and K. Salah, "An intrusion detection framework for energy constrained IoT devices," *Mechanical Systems and Signal Processing*, vol. 136, p. 106436, 2020.

[3] P. Mishra, E. S. Pilli, V. Varadharajan, and U. Tupakula, "Intrusion detection techniques in cloud environment: A survey," *Journal of Network and Computer Applications*, vol. 77, pp. 18-47, 2017.

[4] H. Bostani and M. Sheikhan, "Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach," *Computer Communications*, vol. 98, pp. 52-71, 2017.

[5] R. Alshinina and K. Elleithy, "A highly accurate machine learning approach for developing wireless sensor network middleware," *2018 Wireless Telecommunications Symposium (WTS)*, pp. 1-7, 2018.

[6] S. Anwar, Z. Inayat, M. F. Zolkipli, J. M. Zain, A. Gani, N. B. Anuar, et al., "Cross-VM cache-based side channel attacks and proposed prevention mechanisms: A survey," *Journal of Network and Computer Applications*, vol. 93, pp. 259-279, 2017

[7] V. Zlomislić, K. Fertalj, and V. Sruk, "Denial of service attacks, defenses and research challenges," *Cluster Computing*, vol. 20, no. 1, pp. 661-671, 2017.

[8] M. Panda, A. Abraham, and M. R. Patra, "Discriminative multinomial naive bayes for network intrusion detection," *2010 Sixth International Conference on Information Assurance and Security*, pp. 5-10, 2010.

[9] M. R. Asghar, G. Dán, D. Miorandi, and I. Chlamtac, "Smart meter data privacy: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2820-2835, 2017.

[10] W. B. Jaballah, M. Conti, G. Filè, M. Mosbah, and A. Zemmari, "Whac-A-Mole: Smart node positioning in clone attack in wireless sensor networks," *Computer Communications*, vol. 119, pp. 66-82, 2018.

[11] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: Centralized, on-device, or federated learning?," *IEEE Network*, vol. 34, no. 6, pp. 310-317, 2020.

[12] S. Savazzi, M. Nicoli, and V. Rampa, "Federated learning with cooperating devices: A consensus approach for massive IoT networks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4641-4654, 2020.

[13] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli, "FED-IIoT: A robust federated malware detection architecture in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8442-8452, 2020.

[14] S. M. Giray and A. G. Polat, "Evaluation and comparison of classification techniques for network intrusion detection," *2013 IEEE 13th International Conference on Data Mining Workshops*, pp. 335-342, 2013.

[15] K. Yadav and B. B. Gupta, "Clustering algorithm to detect adversaries in federated learning," *arXiv Preprint*, arXiv:2102.10799, 2021.

[16] O. Alkadi, N. Moustafa, and B. Turnbull, "A review of intrusion detection and blockchain applications in the cloud: Approaches, challenges and solutions," *IEEE Access*, vol. 8, pp. 104893-104917, 2020.

[17] D. Kwon, K. Natarajan, S. C. Suh, H. Kim, and J. Kim, "An empirical study on network anomaly detection using convolutional neural networks," *ICDCS*, pp. 1595-1598, 2018.

[18] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in IoT," *Sensors*, vol. 17, no. 9, p. 1967, 2017.

[19] S. Y. Siddiqui, A. Athar, M. A. Khan, S. Abbas, Y. Saeed, M. F. Khan, and M. Hussain, "Modelling, simulation and optimization of diagnosis cardiovascular disease using computational intelligence approaches," *Journal of Medical Imaging and Health Informatics*, vol. 10, no. 5, pp. 1005-1022, 2020.

[20] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619-640, 2021.

[21] S. A. Alabady and F. Al-Turjman, "Low complexity parity check code for futuristic wireless networks applications," *IEEE Access*, vol. 6, pp. 18398-18407, 2018.

[22] K. S. Bhosale, M. Nenova, and G. Iliev, "Modified naive bayes intrusion detection system (MNBIDS)," *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, pp. 291-296, 2018.

[23] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," *2018 20th International Conference on Advanced Communication Technology (ICACT)*, pp. 178-183, 2018.

[24] A. Sinha, "Beginners guide to federated learning," 2021.

[25] B. Hitaj, P. Gasti, G. Ateniese, and F. Perez-Cruz, "Passgan: A deep learning approach for password guessing," *International Conference on Applied Cryptography and Network Security*, pp. 217-237, 2019.

[26] F. Cacheda, D. Fernández, F. J. Novoa, and V. Carneiro, "Detección temprana de depresión: Análisis de redes sociales y técnicas de aprendizaje máquina," *Actas del IV Machine Learning Workshop Galicia*, vol. 2, p. 41, 2013.

[27] A. Rehman, A. Athar, M. A. Khan, S. Abbas, A. Fatima, and A. Saeed, "Modelling, simulation, and

optimization of diabetes type II prediction using deep extreme learning machine," *Journal of Ambient Intelligence and Smart Environments*, vol. 12, no. 2, pp. 125-138, 2020.
[28] N. A. Addo, "Master of Science in Information Security and Cryptography."
[29] F. Al-Turjman, "QoS–aware Data Delivery Framework for Safety-inspired Multimedia in Integrated Vehicular-IoT," *RETRACTED*, 2018.
[30] M. A. Khan, S. Abbas, A. Rehman, Y. Saeed, A. Zeb, M. I. Uddin, and A. Ali, "A machine learning approach for blockchain-based smart home networks security," *IEEE Network*, vol. 35, no. 3, pp. 223-229, 2020.
[31] M. A. Khan, S. Abbas, A. Atta, A. Ditta, H. Alquhayz, M. F. Khan, and R. A. Naqvi, "Intelligent cloud based heart disease prediction system empowered with supervised machine learning," *Computers, Materials & Continua*, vol. 65, no. 1, pp. 139-151, 2020.
[32] M. A. Khan, M. Umair, M. A. Saleem, M. N. Ali, and S. Abbas, "CDE using improved opposite based swarm optimization for MIMO systems," *Journal of Intelligent & Fuzzy Systems*, vol. 37, no. 1, pp. 687-692, 2019.
[33] A. Ata, M. A. Khan, S. Abbas, M. S. Khan, and G. Ahmad, "Adaptive IoT empowered smart road traffic congestion control system using supervised machine learning algorithm," *The Computer Journal*, vol. 64, no. 11, pp. 1672-1679, 2021.
[34] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowledge-Based Systems*, vol. 189, p. 105124, 2020.
[35] A. Fatima, M. Adnan Khan, S. Abbas, M. Waqas, L. Anum, and M. Asif, "Evaluation of planet factors of smart city through multi-layer fuzzy logic (MFL)," *The ISC International Journal of Information Security*, vol. 11, no. 3, pp. 51-58, 2019.
[36] S. A. Fatima, N. Hussain, A. Balouch, I. Rustam, M. Saleem, and M. Asif, "IoT enabled smart monitoring of coronavirus empowered with fuzzy inference system," *International Journal of Advance Research, Ideas and Innovations in Technology*, vol. 6, no. 1, pp. 188-194, 2020.
[37] M. Saleem, M. A. Khan, S. Abbas, M. Asif, M. Hassan, and J. A. Malik, "Intelligent FSO link for communication in natural disasters empowered with fuzzy inference system," *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, pp. 1-6, 2019.
[38] T. M. Ghazal, A. U. Rehman, M. Saleem, M. Ahmad, S. Ahmad, and F. Mehmood, "Intelligent Model to Predict Early Liver Disease using Machine Learning Technique," *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*, pp. 1-5, 2022.
[39] M. Saleem, S. Abbas, T. M. Ghazal, M. A. Khan, N. Sahawneh, and M. Ahmad, "Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques," *Egyptian Informatics Journal*, vol. In Press.
[40] M. Asif, S. Abbas, M. A. Khan, A. Fatima, M. A. Khan, and S. W. Lee, "MapReduce based intelligent model for intrusion detection using machine learning technique," *Journal of King Saud University-Computer and Information Sciences*, vol. In Press.
[41] M. Asif, M. A. Khan, S. Abbas,

and M. Saleem, "Analysis of space & time complexity with PSO based synchronous MC-CDMA system," *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pp. 1-5, 2019.

[42] M. M. U. U. A. H. Muhammad and A. M. S. F. M. Saleem, "Intelligent Intrusion Detection System for Apache Web Server Empowered with Machine Learning Approaches," *International Journal of Computational and Innovative Sciences*, vol. 1, no. 1, pp. 1-8, 2022.

[43] R. Akmal and M. Saleem, "A Novel Method to Improve the Round Robin CPU Scheduling Quantum Time Using Arithmetic Mean," *International Journal of Computational and Innovative Sciences*, vol. 1, no. 2, pp. 69-82, 2022.

[44] M. Aslam, "Removal of the Noise & Blurriness Using Global & Local Image Enhancement Equalization Techniques," *International Journal of Computational and Innovative Sciences*, vol. 1, no. 1, 2022.

[45] S. Muneer and M. A. Rasool, "A systematic review: Explainable Artificial Intelligence (XAI) based disease prediction," *International Journal of Advanced Sciences and Computing*, vol. 1, no. 1, pp. 1-6, 2022.

[46] M. Hamza, "Optimizing early detection of diabetes through retinal imaging: A comparative analysis of deep learning and machine learning algorithms," Journal of Computational Informatics & Business, vol. 1, no. 1, pp. 1-12, 2024.

[47] M. H. Zia, A. Hussain, and M.-H. Hamza, "Comparative Analysis of Random Forest and Support Vector Machine Classifiers for unjustified malware detection of Android Devices Data Consuming SMOTE and ROC-AUC Metrics," *2024 Horizons of Information Technology and Engineering (HITE)*, Lahore, Pakistan, pp. 1-4, 2024,