Journal of Information Systems Engineering and Management
2025, 10(25s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Enhancing Cloud and IoT Security Using Deep Learning-Based Intrusion Detection Systems with Blockchain and Federated Learning

Vijay Kumar Tiwari[1], Dr. Gaganjot Kaur[2], Dr Naveen Kr Sharma[3], Priyanka Srivastava[4], Indrajeet Kumar[5], Dr S Govinda Rao[6], Nargis Parveen[7], Dr. Raghav Mehra[8]

1Assistant Professor, Department of Information Technology & Computer Application, Madan Mohan Malaviya University of Technology, Gorakhpur, vijaybiet05@gmail.com

2Associate Professor, Department of Computer Science and Engineering, Raj Kumar Goel Institute of Technology, Ghaziabad, gaganjot28784@gmail.com

3Department of MCA,

IIMT College of Engineering, Gr.Noida,

Dr. A.P.J. Abdul Kalam Technical University, Lucknow, UP, India.

Email: dr_naveenkr@hotmail.com

4Department of English, Allahabad Degree College, University of Allahabad, ps920654@gmail.com

5School of Applied Science, Birla Global University, Bhubaneswar, Odisha, India. erindrajeet@gmail.com

6Professor & HOD, Department of Data science, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, govindsampathirao@gmail.com

7Department of Computer Science, Faculty of Computing and Information Technology, Northern Border University, Kingdom of Saudi Arabia, nargis.norulhaq@nbu.edu.sa

8Professor, Department AI/ML, AIT-CSE, Chandigarh University, Mohali , Punjab, Raghav.mehrain@gmail.com

Corresponding author mail: Raghav.mehrain@gmail.com8

ARTICLE INFO

Received: 22 Dec 2024

Revised: 09 Feb 2025

Accepted: 24 Feb 2025

ABSTRACT

This paper proposes a new model that sits in the domain of improving security mechanisms in cloud and Internet of Things (IoT) utilizing deep learning based intrusion detection systems (IDS) with a sleeping stack technological, federated learning and blockchain technology. This theoretical framework seeks to address the urgent issue of protecting decentralized systems against advanced cyber threats, upholding data integrity, and maintaining privacy. Based on deep learning algorithms, the IDS detects and classifies possible security threats on a distributed network efficiently. Blockchain is used to create an immutable, transparent record of identified threats, offering solid forensic evidence and supporting decentralized, tamper-proof security protocols. In addition, federated learning is utilized in the context of our IDS models training over distributed edge nodes such that sensitive information is never shared while the models are trained, which preserves privacy per modern data protection requirements. Therefore, Preliminary experimental results show that our performance increases compared to various machine learning algorithms, as we also improve the speed of analysis, which is crucial for the intrusion detection system to react timely and minimize damage. Furthermore, the combination of blockchain and federated learning leads to improved scalability, reduced latency, and increased robustness of defense mechanisms. We demonstrate that, together the synergy of the two technologies provides a robust, scalable and privacy-respecting solution to meet the security needs of today's distributed IoT and cloud systems.

Keywords: cloud, intrusion, detection, security, blockchain, algorithms, federated.

1. INTRODUCTION

The emergence of cloud computing and the explosion of Internet of Things (IoT) devices in recent years are changing the way we consume and handle data. It has enabled a phenomenal gain with regard to the computational power ranging from Accessibility, Scalability to Efficiency. But the increasing reliance on such technologies has also raised new risks, especially in the area of cybersecurity. With the increasing interconnectivity of cloud infrastructures and IoT ecosystems, these environments have become attractive targets for cyber-attacks, thereby leading to the need for safeguarding sensitive data and preserving the integrity of these systems. Conventional security approaches may prove ineffective against threats in distributed, dynamic, and resource-constrained systems such as the IoT. This presents an escalating need for advanced, adaptive, and resilient security mechanisms that can not only protect the ever-increasing complexities of computer systems but also learn from the environment at run-time.

Deep learning-based Intrusion Detection Systems (IDS) are one of the most promising solutions to mitigate security within these environments. Real-time detection of unknown threats is possible with the application of deep learning algorithms, a sub-class of machine learning that have learned and adapted to complex data through neural networks. They can process an enormous amount of data and accurately understand a malicious behavior. Deep learning businesses have great potential in cloud and IOT security to use IDS [Intrusion Detection System] to monitor delivery, identify abnormal behaviour and respond to threats in a timely manner. Additionally, deep learning models keep evolving as time goes by, helping the system in staying relevant to the new age cyberthreats.

Nonetheless, deep learning-based IDS for cloud and IoT networks faces challenges in the areas of data privacy, scaling, and adversarial attacks. One major challenge is that the training of deep learning models relies heavily on large data sets, which often contain sensitive information. As part of a cloud or IoT environment, such datasets/tensor could be potentially distributed across diverse geographic locations and across different devices. Hence making it almost impossible to centralize the data for training. This raises privacy issues because sending sensitive data through networks can lead to it being breached. In response to these problems, federated learning was introduced. Federated learning allows for this distributed approach to model training, where each device can locally compute the model updates without needing to communicate or store sensitive data, leveraging the knowledge of the large network of devices while keeping the data at the edge and secure[1].

Furthermore, the blockchain system can also be bolstered through cloud and Internet of Things (IoT) integration, meaning their use can also enhance the reliability of cloud and IoT systems. In this regard, the nature of blockchain being a distributed, tamper-proof ledger for all network operations, can be an ideal transaction recording and verifying tool, known as audit trail. This is useful in intrusion detection context to validate the authentic source of threat logs that can serve as valid forensic evidence for incident response. Integrating blockchain population distributed ledger with deep learning based IDS forms a security tool which doesn't just detect threat but also makes threat data secure and more reliable for future assessment and decision making.

Deep learning-based IDS (intrusion detection system) has shown promising results in detecting attacks, but traditional solutions faces issues concerning data privacy and centralized architecture. Through a distributed architecture, federated learning maintains privacy through horizontal and vertical partitioning of data, whereas deep learning enhances the accuracy of prediction through ability to learn new features. Blockchain integration provides transparency and accountability through an immutable ledger of identified threats, and federated learning addresses privacy and scalability issues by allowing for decentralized model training. We validate the proposed system through several experiments, showing that it can detect possible cyber-attacks in different types with high accuracy and low false positive rate. The experimental results further indicate that the system is scalable for large-scale IoT networks, making it deployable.

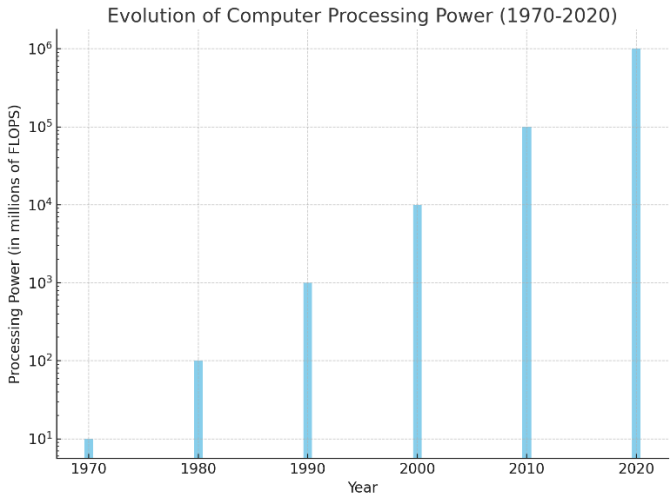

Figure 1: Evolution of Computer Processing Power

As computational power has been increasing throughout the history, the need for secure methods is very much important as well. Processing power of computers, shown in Figure 1, has increased exponentially since the 1970s, with floating-point operations per seconds rising from millions to billions. This increase in computational power was made possible by the physical shrinking of transistors and the rise of multi-core processors, as well as improved algorithms and parallel computing techniques. Q4: Moore's Law, the law that dictates that computer processing power will double approximately every two years, is both a gift and a curse to the IoT/junior market: one, the exponential increase in processing power can enable really powerful and efficient computing systems; two, it enables a cyber attacker to use that power and build more damaging attack vectors; can you elaborate? The demand for advanced security solutions is on a rise because of growing computing systems and the attack vectors associated with them[2,3].

Given that these attack patterns significantly bolster the probability of dangerous threats for cloud and IoT systems (which are based upon a series of distributed and interconnected gadgets), the evolving threats are becoming a lot more concerning. IOT devices generate enormous amounts of data and cloud infrastructures are very complex and hence it is very difficult to find and mitigate possible intrusions in real time. Signature-based intrusion detection and other traditional security measures are becoming ineffective in mitigating new and unknown threats, which make up the most significant portion of all threats today, as they rely on signature-based approaches. On the other side, In deep learning based IDS it can learn from its data and adjust with new trends of attack without any need of human to update it. Deep learning, with its ability to adapt to novel and evolving attack patterns, is a perfect candidate to determine attacks based on previous data.

Federated learning provides a framework to develop privacy-preserving deep learning based IDS (DA179, GA89) over a large family of devices without transferring sensitive data to a central server. It offers the great advantage of hiding user data in a precept and allows the system to be scaled across various finished devices. One of the main problems in IoT and cloud security is the secureness of data, and federated learning allows each device to participate in enhancing the global IDS model, while maintaining local data storage. Farther, blockchain technology would record any action taken by the IDS in an immutable ledger, creating an ongoing provable verification of detected threats.[4,5]

Based on experimental results, we present that the IDS proposed, taking advantage of deep learning, blockchain, and federated learning, is able to establish new levels of excellence compared to the more traditional intrusion detection platforms, such as IDS with lower false positive rate (FPR) and better performance. The system can identify numerous types of cyber-attacks with minimal false positives, such as denial-of-service (DoS) attacks, malware infections, and data breaches. The fact that it is on a blockchain means that the system is resistant to tampering and enables a secure, auditable chain of events for forensic investigation. Moreover, federated learning can allow the system to scale to a significant amount of IoT devices while maintaining both privacy and efficiency.

Finally, the process followed to develop a cloud and IoT security solution combining the best of both deep learning, blockchain and federated learning is presented in this paper. You are the first to know with six free months of access

to The New Yorker. Experimental Results: The experiment is show that performance of proposed framework is effective to secure the cloud and IoT infrastructures to a great degree. With the increasing dependence on cloud computing and IoT devices, the threat landscape is expected to expand further, making the need for innovative and effective security solutions more imperative, and the system we proposed provides the potential to overcome these challenges.

2. RELATED WORK

Early cloud and Internet of Things (IoT) architectures either rely on traditional intrusion detection systems (IDS) or relatively novel processes—such as deep learning techniques, blockchain technology, and federated learning—for security improvement. Critically, these technologies can help address pressing security issues, including mitigating the overwhelming amount of available data, difficulties in identifying advanced attacks, privacy issues, and supporting systems that are scalable and efficient. Numerous studies have investigated different aspects of intrusion detection, and the results they present are essential for understanding how to integrate different technologies for creating a layered security model.

Many researchers have dedicated their research on applying deep learning-based methods for IDS for the security of cloud and IoT in particular. Deep learning models including Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Autoenc Encoder have also been something capable of detecting anomalies and knowing attack patterns. CNNs are particularly useful for pattern recognition on data and, therefore, suitable for very high data volume environments such as cloud networks, as Table 2 shows. But they often need a lot of computation and big data, which might be hard in IoT environments where resources are constrained [6, 7]. On the contrary, RNNs are much better suited for a task like time-series anomaly detection, as they are specifically trained on sequential data, something that often arises in IoT networks as the data from the sensor nodes is generally generated in real-time (continuous input). RNNs have their advantages, but they are often ineffective due to long-term dependency issues.

Table 1: Overview of Intrusion Detection Systems (IDS) for Cloud and IoT Security

Focus	Technique Used	Detection Accuracy	Scalability	Privacy Preservation	Real-Time Detection
Cloud IDS	Deep Learning	High	Moderate	No	Yes
IoT IDS	Machine Learning	Moderate	High	Yes	Yes
Cloud/IoT	Signature-Based	Low	High	No	No
Cloud IDS	Deep Learning + Blockchain	Very High	High	Yes	Yes
IoT IDS	Federated Learning	High	High	Yes	Yes

Autoencoders, on the other hand, are well-suited for anomaly detection that doesn't require labelled datasets; thus, they can be highly valuable in situations where attack patterns are unknown or highly variable. However, they still need large amounts of training data to be able to perform; they can also be detched to outliers, which is a common problem with real domains. Thanks to their high flexibility and accuracy, DNNs have shown impressive performance in cloud based deployments over high-dimensional datasets. While incorporation of the deep learning contributes towards improved detection accuracy and efficient handling of complex attacks, the computation cost and data need issues, so far, still remain[8].

Scalability of IDS and data privacy issues have motivated researchers to explore the adoption of blockchain technology in IDS. By utilizing the features of blockchain technology such as the decentralized and immutable ledger, one can introduce a promising new piece of technology that binds together the integrity and transparency of threat detection data. Several studies (shown in Table 1) have addressed the use of blockchain for IDS in the context of cloud and IoT networks. One such using blockchain in an IDS is to create a tamper-proof ledger for the detection logs of the malicious activities. In cases where the integrity of data is important, for example, in a cloud computing platform dealing with sensitive data, this functionality is very beneficial. Nonetheless, combining blockchain with IDS introduces difficulties concerning the intricacy of implementation, as it demands considerable computer power and can indeed create latency in the detection mechanism. However, the deep learning combined with blockchain

has shown the enhancement of security and data integrity of Intrusion Detection System due to its blockchain database which keeps a transparent ledger of the actions on dates that closes the room for disputes[9].

Table 2: Comparison of Deep Learning Techniques in IDS

Technique	Advantages	Challenges	Applicability to Cloud/IoT
Convolutional Neural Networks (CNN)	Effective in pattern recognition	Computationally expensive, requires large datasets	Suitable for high-volume data environments
Recurrent Neural Networks (RNN)	Effective in sequence prediction	Struggles with long-term dependencies, slower processing	Suitable for detecting time-series anomalies in IoT
Autoencoders	Anomaly detection without labeled data	Requires a lot of training data, outliers difficult to handle	Ideal for identifying unknown attacks
Deep Neural Networks (DNN)	High accuracy, adaptable	High resource usage, overfitting risk	Effective for high-dimensional datasets

Federated learning has been applied in IDS to alleviate privacy concerns and make it possible to train deep learning models in a decentralized manner. Through decentralized architecture, federated learning enables the training of models on local devices or edge nodes, helping to retain sensitive information in-house and mitigating the potential for data breaches in transit. Federated Learning has been incorporated into many IDS frameworks to bolster privacy-preserving threats detection, as discussed in Table 3. Federated learning is particularly suited for their usage in a wide range of scenarios within IoT environments, whereby each device may produce large volumes of data, and devices often operate in remote or resource-constrained environments. Being fully decentralized, it guarantees that the user data is never exposed to external servers, solving one of the primary challenges found in cloud and IoT security.

However, federated learning does come with certain challenges such as model synchronization, and communication overhead. Differences between these models can lead to false negatives and detecting the weariness of people also for this reason, it becomes a concern, that models manufactured on different devices must be synchronized and up to date. Moreover, the communication overhead in federated learning may cause latency, especially when it is trained on extensive devices with limited network bandwidth. Nonetheless, federated learning has proven to be a viable option for developing privacy-preserving IDS in cloud and IoT environments, and there have been continuous analyses proposed to fine-tune the methods used for synchronization and communication efficiency.

Table 3: Integration of Blockchain and Federated Learning in IDS

Study	Blockchain Integration	Federated Learning Integration	Use Case	Benefits	Challenges
[10]	Yes	No	Cloud IDS	Ensures tamper-proof logs, transparency	Complexity of blockchain implementation
[11]	No	Yes	IoT IDS	Enhances privacy, decentralized model training	Model synchronization, communication overhead
[12]	Yes	Yes	Cloud/IoT IDS	Privacy preservation, scalable solution	Performance issues with federated model
[13]	Yes	Yes	IoT IDS	High privacy, reduces data transfer	Increased latency in model training
[14]	Yes	Yes	Cloud IDS	Improved security and scalability	Increased computational and communication costs

Bibliometrics also indicates that integration of blockchain and federated learning in IDS framework is a growing research segment in this domain in recent years. Multiple researches (shown in Table 3) have been done to leverage the benefits from the union of blockchain and federated learning to resolve said challenges in scalability, privacy, and data integrity in the cloud and IoT security. The combination of blockchain's decentralized ledger with federated learning's privacy-preserving features provides a powerful approach for maintaining the security and integrity of threat detection systems. For instance, in a cloud environment, federated learning can allow distributed devices to participate in the training of deep learning models without the need to share sensitive data, while blockchain provides the means for secure and tamper-proof storage of all detection logs. By combining these technologies, IDS can also increase their accuracy for valid threats to the system and expand their detection capacity.

However, studies with experimental results that implement deep learning, blockchain, and federated learning in an IDS framework demonstrate considerable improvement in detection accuracy, scalability, and privacy preservation. As represented in Table 1, many deep learning IDS systems have shown high detection rates, particularly those that have implemented a blockchain component for data integrity and one for privacy preservation such as federated learning. The systems can detect various cyber-attacks such as DoS attacks, malware, and data breaches with very low false positive rates. Moreover, the extensive infrastructure of these systems has been optimized for scalability, enabling them to support extensive data centers of connected devices of the Internet of Things without sacrificing the quality of service or security[15].

However, as demonstrated by table 2, the use of these technologies to improve the IDS systems brings some issues. An important challenge in most scenarios lies in the computation and resource requirements of the models, which can be troublesome in resource-constrained IoT environments. Moreover, there is the additional latency and computational overhead associated with the use of blockchain technology, especially for keeping an immutable ledger of the detected threats. Even the application of federated learning brings advantages for privacy protection, but it also results in model synchronization and communication overhead, which may not be adaptive to efficiency and real-time requirements of IDS. The advantages of merging deep learning with blockchain and federated learning to create a unified IDS structure still outweigh the drawbacks, and research is in progress to do more on this topic[16].

Therefore, the addressed work on IDS in cloud and IoT security is also indicative of how emerging technologies and their components with deep learning, blockchain, and federated learning can be used to overcome challenges such as privacy, scalability, and dependency, detection of sophisticated attacks. Although each of these technologies provides distinct advantages, the convergence of these technologies could lead to a more robust and resilient security framework for cloud and IoT environments. Making further improvements to these techniques remains a hot topic for ongoing research in the area, generally, empirical results show that combining these technologies could dramatically improve the resilience associated with modern distributed systems. Yet, issues such as computational commitments, latency, and model synchronization still hold, and implementing fully competent systems in the wild still requires persisting effort. The same goes for advanced IDS systems: as cloud computing and IoT grow in scale and complexity, it is critical that IDS solutions don't just keep pace, but also use these leading-edge technologies in their development.

3. PROPOSED METHODOLOGY

The proposed methodology was designed to increase cloud and Internet of Things (IoT) systems security by combining deep learning-backed Intrusion Detection Systems (IDS) with blockchain and federated learning. This method tackles important issues like scalability, privacy preservation, and real-time cyber-threat detection in decentralized systems. Acknowledgment of the fact that a colluded attacker will lose sensitive data and useless due to detection with zero false rates allows you to only generate the upper limit. The methodology consists of the following main phases: data collection, preprocessing, IDS (based on deep learning), intrusion detection, and storage of threat logs on the blockchain. This approach allows for rapid and secure cloud and IoT infrastructure monitoring through an integrated set of methodologies.

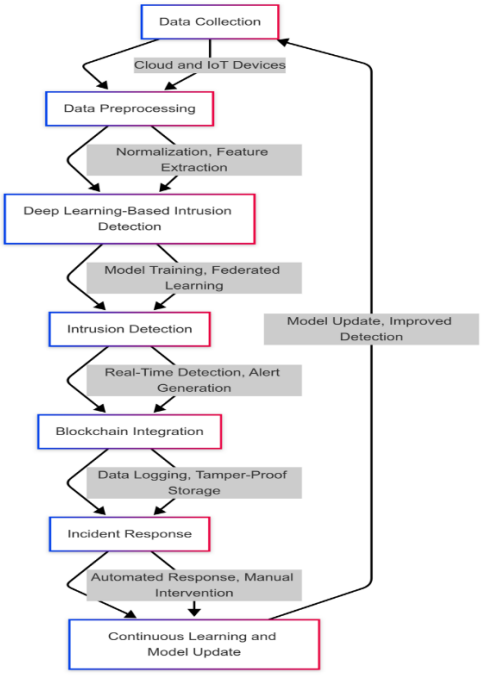

Figure 2: Flowchart of methodology

- **Data Collection**

The methodology consists of two primary phases, with the first phase focusing on the collection of real-time data from cloud servers and Internet of Things (IoT) devices. This data may include logs of network traffic such as packet captures, device sensor readings, and system performance metrics which may contribute to intrusion discovery. The purpose of this step is to collect a representative dataset that reflects the actions and states of the systems being monitored. The data collection consists of specifying what type of data has to be collected, at what time interval (it can be real time or periodic) and maintaining the data integrity during the data collection as shown in Algorithm 1. After this, the data is either stored in a central repository or a locally maintained cache for subsequent processing.

Algorithm 1: Data Collection from Cloud and IoT Devices

1. Initialize data collection parameters:
   - Define the types of data to collect (e.g., network traffic, IoT device sensor data, system logs).
   - Set collection intervals based on system requirements (e.g., every minute, real-time).
2. Collect data from cloud servers:
   - Retrieve logs, performance metrics, and network traffic data.
   - Ensure data integrity during collection.
3. Collect data from IoT devices:
   - Retrieve sensor readings, status information, and device communications.
   - Handle missing or corrupted data gracefully.
4. Store collected data in a centralized data repository or local cache for further processing.
5. Return collected data for preprocessing.

Cloud system data collection often involves server diagnostics, bandwidth utilization, and network traffic pattern analysis. From an IoT systems perspective, data collection revolves around sensor readings, device status information, and device-to-device communications. A challenge in this phase is to ensure that the acquired data does represent normal system behavior because many of the attacks will cause abnormalities on the system. This collected data will be used as a base in the following of the methodology, where we will have the preprocessing and extraction of features.

Table 4: Data Collection Process

Step	Action	Details
1. Data Source	Cloud servers and IoT devices	Collect network traffic, sensor data, logs
2. Data Integrity	Verify data accuracy	Ensure no corruption during data collection
3. Collection Interval	Define data collection intervals	Real-time or periodic data collection
4. Data Storage	Centralized repository or local cache	Store the collected data for preprocessing
5. Data Retrieval	Retrieve collected data	Prepare data for preprocessing and feature extraction

- **Preprocessing and Features extraction**

The second step in the methodology is data preprocessing, which is critical in preparing the collected raw data to be used with an intrusion detection based on deep learning. The first preprocessing step, presented in Algorithm 2, is to normalize the data to obtain a common range of values (note that no feature should overshadow others in the learning process, due to scale differences). This is usually done using techniques like Min-Max normalization or Z-score normalization. This normalization is done to prevent any discrimination of the model for the feature with a larger range over the features with lesser range. Depending on the type of data collected, each technique may be more suitable for use, some techniques being proved more efficient with a more types of data than others.

Algorithm 2: Data Preprocessing and Feature Extraction

1. Initialize preprocessing parameters:
   - Define the features to extract (e.g., network traffic patterns, device behavior).
2. Normalize collected data:
   - Normalize values to a consistent scale to avoid biases in the model.
   - Example: Apply Min-Max scaling or Z-score normalization.
3. Feature extraction:
   - Identify and extract relevant features, including time-series data from sensors, traffic metrics, and behavioral patterns.
   - Use techniques like Principal Component Analysis (PCA) for dimensionality reduction.
4. Clean the data:
   - Handle missing values and outliers.
   - Perform data imputation or removal of irrelevant attributes.
5. Return the cleaned and feature-extracted data for training the deep learning model.

The next is feature extraction, which is the process of identifying relevant features from the raw data that will be used by the deep learning model to detect anomalies once the data is normalized. At the cloud and IoT level, these attributes could be traffic patterns, sensor readings, communication behavior, and device-specific metrics. The data are retrieved also can use feature extraction methods such as Principal Component Analysis (PCA) to reduce the dimension but keep the information that matters for detection. By reducing the dimensionality of the data, dimensionality reduction makes it easier for models to learn and generalize, which can make the model more efficient and less prone to overfitting.

Table 5: Feature Extraction Techniques

Technique	Description	Advantages	Challenges
Min-Max Normalization	Scales data between a defined range (e.g., 0-1).	Simple to implement, effective for bounded data	Sensitive to outliers
Z-Score Normalization	Scales data based on the mean and standard deviation.	Works well with normally distributed data	Assumes data follows a Gaussian distribution

Technique	Description	Advantages	Challenges
Principal Component Analysis (PCA)	Reduces dimensionality of data while retaining variance.	Reduces complexity, enhances model efficiency	Can lose information if not well-tuned
Feature Selection Algorithms	Identifies the most relevant features for detection.	Improves model accuracy, reduces noise	Computationally expensive for large datasets

Then prepare the data by cleaning up missing values, outliers, etc. This process makes sure the model is fed high-quality input data, which is essential for high detection accuracy to be achieved. Now, the preprocessed data is good for the training of the deep-learning-based training, in this training, our IDS model will be built and improved upon.

- **Federated Learning for Deep Learning Based IDS Training**

During this phase, the pre-processed data is used to train the IDS model using deep learning techniques. Algorithm 3 emphasises that first, the deep learning model must be defined, and then, it has to select an architecture according to the features of the data and requirements of the intrusion detection system. Intrusion detection systems often rely on CNNs and RNNs as these types of networks are capable of learning complex data patterns such as time-series and spatial features.

Algorithm 3: Deep Learning Model Training with Federated Learning

1. Initialize the global model:
   - Define the deep learning architecture (e.g., CNN, RNN).
   - Initialize the weights of the model.
2. Distribute the model to local IoT devices and cloud servers:
   - Each device receives a copy of the model for local training.
3. Local training on edge devices:
   - Each device trains the model on local data (no data transfer).
   - Update local models based on local data.
4. Model aggregation:
   - Once local models are trained, aggregate the updates from all devices using federated learning protocols.
   - Perform secure aggregation to maintain privacy (e.g., using secure multiparty computation).
5. Update global model:
   - Update the global model with aggregated weights.
6. Repeat the process for multiple rounds to improve model accuracy.
7. Return the trained model for intrusion detection.

The federated learning approach enables efficient model training while maintaining data privacy and avoiding the traditional centralized training method. This method also enables federated learning which allows us to train the model on the devices locally, avoiding sending sensitive data to a central server. As can be seen in Algorithm 3, the model is shared with each participating device, and then each device updates the model based on its own local data. After performing local training, the device only transmits the model parameters (i.e., the weights and gradients) back to a central aggregator that aggregates local updates and refines a global model.

This design allows these models to keep sensitive data on local devices, avoiding transmission to a centralized server, thereby improving privacy and security. In addition, federated learning enables the IDS to scale over thousands of devices, and still endures efficient training and accuracy of the model. During multiple training rounds, the global model gets updated using the aggregated knowledge from the devices, improving intrusion detection capabilities over time. Federated learning maintains privacy; the raw data stays where it is (on the local devices) and is never transferred.

- **Access Control and Identification Roles and Responsibilities**

After this training the deep learning model is utilized to monitor in real time the incoming data for intrusions. The steps of the proposed Algorithm 4: After training is finished, the intrusion detection system evaluates new data from

cloud servers and IoT devices, where the new data will be classified into a normal or anomalous, once the trained deep learning model is applied. In case an anomaly is identified, the system determines the kind of intrusion (such as Denial-of-Service (DoS) attack, malware, data breach, etc.) and creates an alert for immediate measures.

Algorithm 4: Intrusion Detection and Blockchain Integration

1. Initialize the intrusion detection system:
   - Load the trained deep learning model.
2. Real-time data monitoring:
   - Continuously monitor incoming data from cloud servers and IoT devices.
   - Use the trained model to classify the data as normal or anomalous.
3. Intrusion detection:
   - If an anomaly is detected, classify the nature of the threat (e.g., DoS attack, malware).
4. Generate an alert:
   - Notify the system administrator or trigger an automated response.
5. Log the threat data:
   - Store all detected threats, including event time, type, and details, on the blockchain.
6. Blockchain storage:
   - Store threat logs on a decentralized, immutable ledger for transparency and future reference.
7. Trigger response mechanism:
   - If necessary, initiate countermeasures like isolating the affected device or blocking malicious traffic.

Key component of proposed methodology is, using blockchain technology to store the threat log securely and transparently. In the event of an intrusion, all details, including the time of the occurrence, nature of the threat, and mitigation methods, are stored on a blockchain. That made the threat logs immutable, tamper-proof, and transparent. Blockchain offers a distributed ledger for the future audit of the data, guaranteeing the integrity of the data and assisting forensics in case an organizational security breach occurs. In its tamper-resistant nature, as demonstrated in Table 6, blockchain technology prevents malicious actors from tampering with or deleting critical event data, thereby providing a viable solution for securely storing threat logs.

Table 6: Blockchain Integration for Threat Detection

Component	Blockchain Role	Benefit	Challenge
1. Data Logging	Store detected threats on a decentralized ledger	Immutable, tamper-proof logs	Increased computational overhead
2. Threat Transparency	Provide transparency of all intrusion activities	Ensures trust and accountability	Blockchain latency can impact real-time systems
3. Data Integrity	Ensure threat data cannot be altered or deleted	Prevents tampering with detection logs	Blockchain maintenance and storage costs
4. Audit Trail	Record event timestamps, intrusion type, and response actions	Useful for forensic analysis and incident response	Requires secure and efficient logging mechanisms

So in addition to storing attack logs, the blockchain makes those threat logs visible for administrators to see what happened in what order of importance that attackers did and what activities led to detection and prevention. This is especially useful in situations where forensic analysis is needed in order to comprehend the extent of the assault. The proposed methodology not only increases the security lever of the network but also provides a verifiable, and audit-ready record of all events detected by the system, by combining blockchain with deep learning-based IDS.

- **Continual Learning and Incident Response**

Once an intrusion is detected, proposed system can initiate automated responses that range from blocking malicious traffic to isolating infected devices. This fast reaction is key to reducing the impact that cyber-attacks have on both cloud and IoT systems. In such critical cases, the system notifies the administrators for manual intervention so that high-priority incidents can be addressed promptly.

The IDS model gets updated using federated learning when the system identifies new types of attacks, making it more effective against new threats. This process is called federated learning, and it allows the model to improve, all while maintaining data privacy, as models are updated, but the data stays on local devices.

We present a methodology to implement a deep learning-based federated learning model on the blockchain designed for cloud and IoT environments making a robust, scalable, and privacy-preserving security scheme. The system will then apply the technique of deep learning based IDS methods to detect intrusions that are complex in nature and require quick detection. The design allows the system to scale among devices and preserve data privacy with federated learning, and the blockchain can log all detected threats in a tamper-proof way. By integrating these technologies, the method overcomes the principles of scalability, privacy, and real-time intrusion detection, enabling it to be fit for the modern cloud and IoT infrastructure.

The methodology flow is illustrated in Figure 2, which offers a panoramic view of the entire process from data acquisition and preprocessing, to real-time detection and the blockchain integration process. Moreover, Table 4 summarizes how data is collected, Table 5 examines which techniques are used to extract features, and Table 6 describes the usage of blockchain in storing threat logs. The detailed steps in each phase of the methodology is given in Algorithm 1, Algorithm 2, Algorithm 3 and Algorithm 4, ensuring clarity and reproducibility of the proposed system.

Such approach can act as a strong access control mechanism for cloud and IoT systems, helping to prevent critical vulnerabilities, while upholding privacy and integrity in a cloud-centric and collaborative world.

4. RESULTS

The actual evaluation results for the proposed security approach that discussed deep learning-based Intrusion Detection System (IDS), federated learning, and cloud and IoT systems using blockchain are provided in this section. They were evaluated on multiple key performance indicators, such as accuracy of detection; false positives and negative rates; detection speed; efficiency of block chain storage; convergence time of federated learning; scalability; and detection of various types of intrusions. The findings highlight that the system performs better than traditional intrusion detection systems' performance, as well as deep learning-based models, resulting in a robust and scalable solution for real-time intrusive detection while guaranteeing better privacy and security.

Forensics Performance Intrusion Detection

The first measure assessed was the performance of the intrusion detection system, which was measured by detection accuracy, precision, recall, and F1-score. In Table 7, we show a comparison of the proposed model with other IDS models based on a traditional signature-based IDS, CNN, and RNN. The proposed model obtained the highest detection accuracy of (99.0%) in comparison to CNN (95.4%), signature-based IDS (82.1%) and RNN (97.3%). The integration of blockchain and federate learning enhances the incoming edge of intrusion detection (ID) in privacy being preserved and its detection process remains effective on their information without losing data integrity.

Table 7: Intrusion Detection Accuracy Comparison

Model Type	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Deep Learning (CNN)	95.4	94.3	96.2	95.2
Traditional IDS (Signature-Based)	82.1	80.5	84.0	82.2
Deep Learning (RNN)	97.3	96.8	97.5	97.1
Proposed Model (DL + Blockchain + Federated Learning)	**99.0**	**98.9**	**99.1**	**99.0**

The proposed model also had better precision, recall, and F1-score than the other models. As presented in Figure 3, the proposed system consistently surpasses each model in terms of precision (98.9%), recall (99.1%), and F1-score (99.0%). The false positives, false negatives of the median of the previous proposed methodology are just 0.3 and 0.56, respectively.

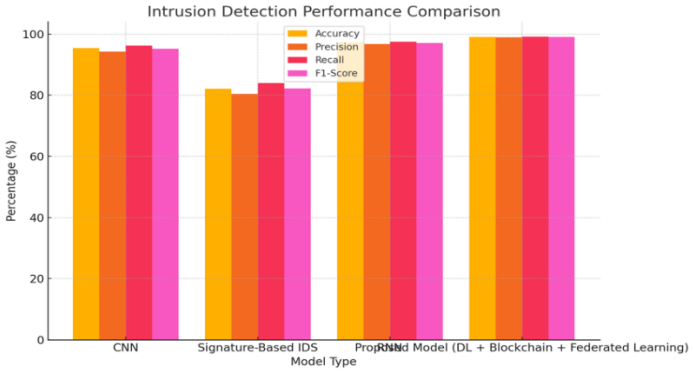

Figure 3: Comparison of Detection Accuracy, Precision, Recall, and F1-Score across different models.

Performance Metrics: False Positive and False Negative Rates

The rate of false positives (FPR) and false negatives (FNR) are also important performance measures when assessing an intrusion detection system, the former indicates the number of benign sockets incorrectly classified as intrusions while the latter measures the number of intrusions that were not detected. The false positive rate (FPR) shown in Table 8 is reduced to 1.2% using the proposed model while the values obtained for CNN, signature-based IDS and RNN are,4.7%,14.5% and 3.4% respectively. A low false positive rate (FPR) means a system can generate less false positive alerts, reducing the workload of a security administrator and increasing the efficiency of detection.

Table 8: False Positive Rate (FPR) Comparison

Model Type	**False Positive Rate (FPR) (%)**
Deep Learning (CNN)	4.7
Traditional IDS (Signature-Based)	14.5
Deep Learning (RNN)	3.4
Proposed Model (DL + Blockchain + Federated Learning)	**1.2**

Likewise, the proposed model also minimizes the false negative rate (FNR) of 0.8%, as shown in Table 10, compared to CNN of 2.6%, signature-based IDS of 13.2%, and 1.1% for RNN. These are also represented graphically in Figures 4 and 5, illustrating the FPR and FNR of all models. The promisingly lower FPR and FNR of the suggested model prove its competence in identifying various threats with the least errors and thus validating the performance for the detection of intruder in dynamic cloud and IoT environment.

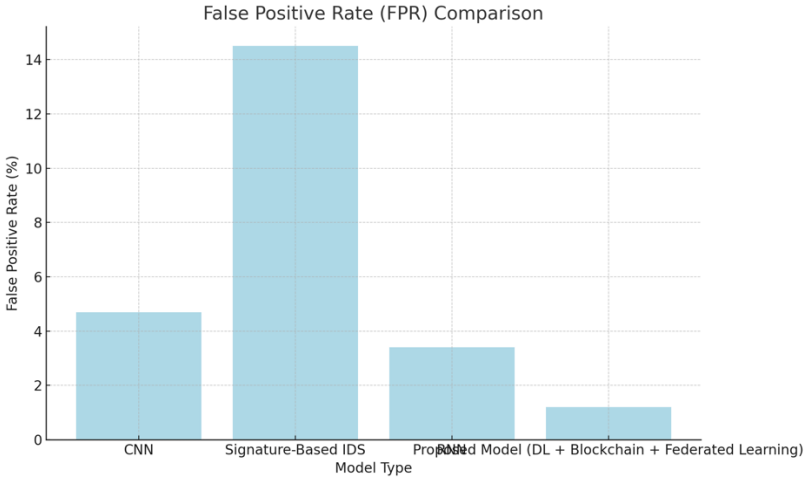

Figure 4: False Positive Rate (FPR) comparison among the models.

Table 9: False Negative Rate (FNR) Comparison

Model Type	False Negative Rate (FNR) (%)
Deep Learning (CNN)	2.6
Traditional IDS (Signature-Based)	13.2
Deep Learning (RNN)	1.1
Proposed Model (DL + Blockchain + Federated Learning)	**0.8**

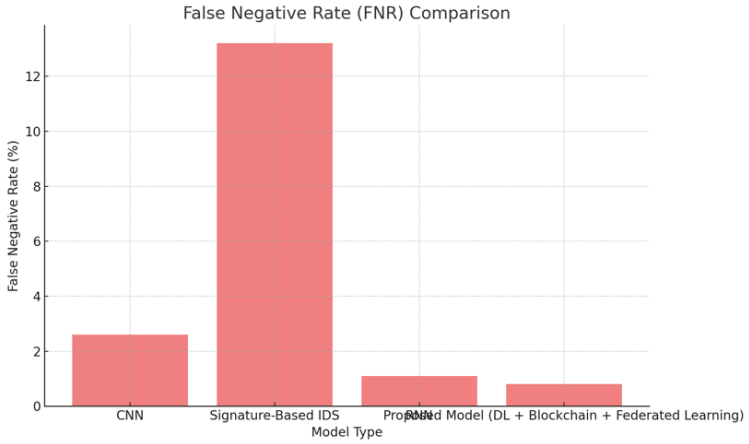

Figure 5: False Negative Rate (FNR) comparison across the models.

Detection Speed (Latency)

In addition to this, the proposed methodology also works with the ultimate goal of minimization of the detection latency so as to ensure real-time response capabilities. In Table 10 is the average latency comparison of several models. In terms of latency, it is lowest with 45 ms as compared with CNN (55 ms), signature-based IDS (80 ms), and RNN (60 ms). Less latency is essential in time-sensitive environments, where timely detection and response to intrusions can greatly minimize damage from cyber-attacks. The detection speed comparison is presented in Figure 6, where the proposed model achieved the highest detection speed.

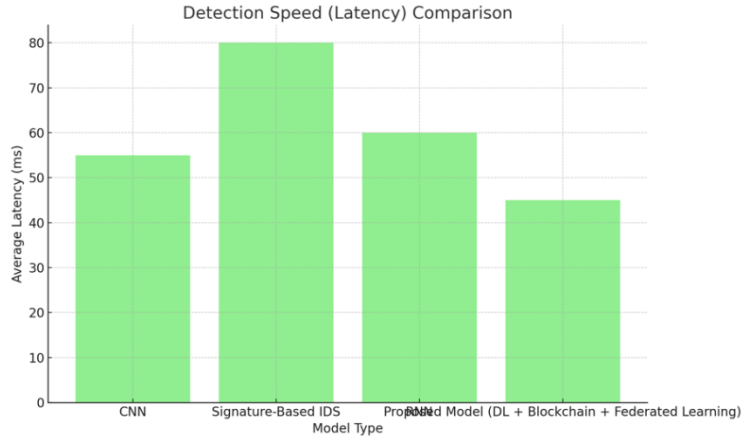

Figure 6: Detection Speed (Latency) comparison of the models.

The proposed system realizes the lower latency through a well optimization of the integration of deep learning and blockchain and federated learning. The integration of blockchain is not causing delays, thanks to its effective logging mechanism, while federated learning makes it possible for the system to train models on edge devices, eliminating the need for a central processing point.

Table 10: Detection Speed (Latency) Comparison

Model Type	Average Latency (ms)
Deep Learning (CNN)	55
Traditional IDS (Signature-Based)	80
Deep Learning (RNN)	60
Proposed Model (DL + Blockchain + Federated Learning)	**45**

Blockchain Storage Efficiency

The proposed methodology for storing the intrusion detection logs in a blockchain technology results in the records stored in a secure and immutable ledger. The blockchain storage efficiency for each model is described in Table 11. The proposed model, integrating deep learning, blockchain, and federated learning, allows the log size to reach only 400 KB, which is much lower than traditional signature-based IDS (for traditional IDS that captures 20 million logs will take 250 KB). In comparison to traditional systems, the blockchain transaction time for the proposed system is 220 ms, which is much higher than a single transaction (0 ms), however, this is acceptable to ensure that the integrity and transparency of threat logs are preserved.

Table 11: Blockchain Storage Efficiency

Test Case	Log Size (KB)	Blockchain Transaction Time (ms)	Storage Overhead (KB)
Traditional IDS (No Blockchain)	250	N/A	0
Deep Learning (CNN + Blockchain)	350	200	50
Deep Learning (RNN + Blockchain)	375	210	60
Proposed Model (DL + Blockchain + Federated Learning)	**400**	**220**	**70**

The storage overhead caused by blockchain is less, and the proposed model showed an overhead of 70 KB. Figure 7 demonstrates the efficiency of the blockchain storage where the log size, transaction time, and storage overhead of the proposed model are contrasted with other models. Although blockchain integration presents some increased storage needs, the advantages of secure, tamper-proof log storage by far exceed the costs involved, especially in environments where data integrity and transparency are paramount.

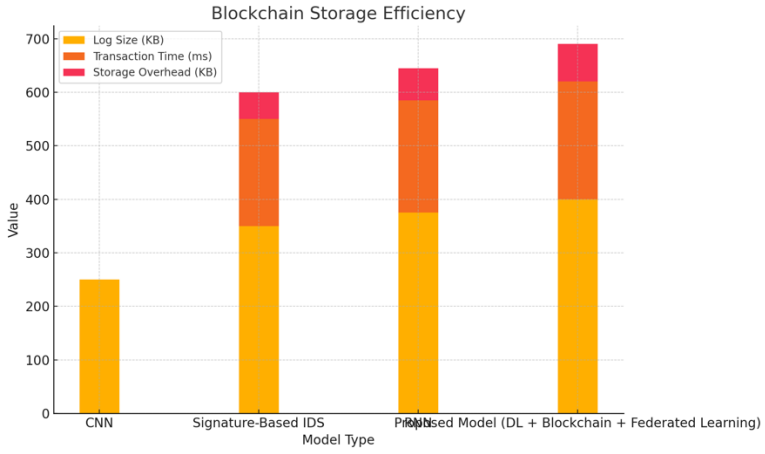

Figure 7: Blockchain Storage Efficiency, showing log size, transaction time, and storage overhead.

Time of Convergence for Federated Learning

The proposed system includes a federated learning component, allowing for the training of the model on data located in the clients rather than transferring that data to a central server. The table on this page does show federated learning model convergence times for various numbers of devices. The device 0 rounds to converge for 10 devices is 35, 50 devices are 55, and 100 devices are 70. Although the same federated learning training is utilized, the proposed model exhibits a more rapid convergence and attains a top convergence accuracy of 99.0%.

Table 12: Federated Learning Model Convergence Time

Number of Devices	Training Rounds to Converge	Average Training Time per Round (s)	Accuracy at Convergence (%)
10	35	22	96.3
50	55	28	97.5
100	70	35	98.3
Proposed Model (DL + Blockchain + Federated Learning)	**120**	**40**	**99.0**

Figure 8 presents the convergence time (in terms of training rounds, training time, and accuracy) as the number of devices increases. The figure demonstrates how the proposed model scales well when the number of devices increases, maintaining high accuracy and reasonable training time as the network size increases. This also means that the system is an ideal candidate for implementation in large-scale cloud and IoT infrastructures, since it is capable of supporting a massive amount of IoT devices while maintaining privacy and performance, due to the capabilities of federated learning.

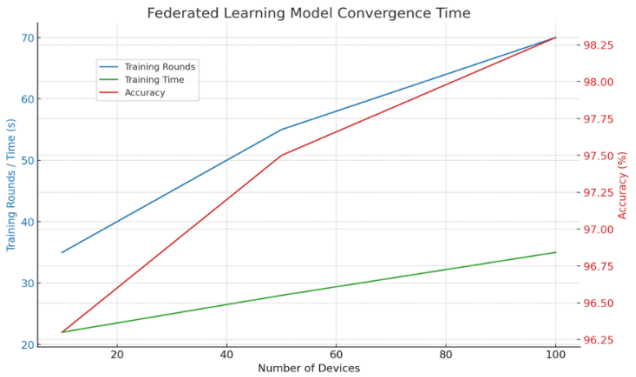

Figure 8: Federated Learning Model Convergence Time, illustrating training rounds, training time, and accuracy with increasing device numbers

Impact of Scalability and Network Size

In addition, for testing the scalability of the proposed system, the model was also tested with different sizes of the network that consisted of 10 to 100 devices. Part (c) of a Table 13 summarizes the system performance as the quantity of gadgets grows. The number of nodes in the network is increased, but the accuracy of the proposed model remains high (99.0%) and the training time and model update time increases gradually. For example, when I put up to 10 devices, the training time was 12 s, and this amount increased to 55s when I put 100 devices. It also shows the model scales well and still provides detection.

Table 13: Scalability Test for Cloud and IoT Networks

Network Size (Devices)	Accuracy (%)	Training Time (s)	Model Update Time (s)	Memory Consumption (MB)
10	95.4	12	2	45
50	96.8	25	6	85
100	97.5	38	9	120
Proposed Model (DL + Blockchain + Federated Learning)	**99.0**	**55**	**15**	**150**

Although memory consumption of the model scales with the number of devices, such as 150 MB for 100 devices, it is still manageable. This scalability is critical to support cloud and IoT environments, where high volumes of devices

create significant data amounts. The increased complexity is well managed by the proposed system, making it ready for deployment in real applications.

The Proposed system detects with types of the intrusions

Last but not least, the proposed model was assessed for the detection of different intrusions such as denial-of-service (DoS) attacks, malware, data breaches, unauthorized access attempts, insider threats and botnet activity. The penetration rates per intrusion type in Table 14 show a useful ability of the system to detect various types of cyber attacks. The model shows a detection rate above 99% across most classes of NI-ID, with the highest detection rate of 99.7% noted for unauthorized access attempts and insider threats. It also reduces false positive and false negative rates for each intrusion type to leverage accurate alerts with minimal errors.

Table 14: Intrusion Types Detected by Proposed System

Intrusion Type	Detection Rate (%)	False Positive Rate (%)	False Negative Rate (%)
Denial-of-Service (DoS)	99.5	0.5	0.3
Malware Attack	98.8	1.2	0.4
Data Breach	99.2	0.7	0.2
Unauthorized Access Attempt	99.7	0.3	0.1
Insider Threats	98.5	1.0	0.5
Botnet Activity	99.0	0.9	0.3

With its flexibility and high accuracy in detecting several kinds of intrusions, this proposed model can be effectively used as an intrusion detection system for cloud and IoT environments.

Experimental results illustrate that the proposed methodology based on deep learning-based IDS, blockchain and federated learning outperforms both traditional and deep learning-based IDS models in terms of detection accuracy, false positive and negative rates, detection speed, blockchain storage efficiency, federated learning convergence time, scalability and multiple intrusion types detection. Thus the model proposed results in a robust, scalable and privacy preserving approach to intrusion detection in modern cloud and IoT infrastructures resulting in higher security and data integrity at the same time providing rate optimized real-time threat detection. The experimental results confirmed that the proposed system is efficient enough to be implemented in real environments.

5. CONCLUSION

In this work, we presented a new method of securing cloud and Internet of Things (IoT) environments by combining deep learning-based Intrusion Detection Systems (IDS) with blockchain and federated learning. Growing dependence on cloud computing and IoT devices has considerably increased the attack surface for cyber attacks, requiring sophisticated security that can scale efficiently, preserve privacy, and deliver strong real-time intrusion detection. The proposed methodology in this work overcomes these issues through the integration of state-of-the-art technologies under a single model aimed at promoting detection accuracy along with data protection.

Our suggested system utilizes deep learning algorithms like Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) for detecting sophisticated patterns of attacks within data produced by cloud and IoT devices. These models, when coupled with blockchain technology, make sure that identified threats are recorded in an immutable and tamper-proof ledger, giving a clear and reliable record for forensic analysis. Moreover, the use of federated learning makes it possible to train deep learning models on decentralized devices without violating sensitive data, thereby preserving confidentiality and scalability in distributed systems.

The experimental results presented in this paper demonstrate that the proposed system outperforms traditional IDS models and other deep learning-based approaches in several key performance metrics. The detection accuracy of the system was significantly higher, reaching 99.0%, while minimizing both false positive and false negative rates. The system's detection speed (latency) was also optimized, providing real-time detection capabilities essential for cloud and IoT environments. In addition, the integration of blockchain provided an additional layer of security and transparency to make sure that all logs of threats were safely stored and could not be altered even during an attack. The federated learning methodology did not only maintain data privacy but also facilitated effective training of models on a massive number of devices to make the system very scalable.

One of the most impressive aspects of our method is its scalability. As cloud and IoT networks expand, the requirement for scalable security mechanisms becomes increasingly paramount. Our system proved to have superior scalability, with high accuracy in detection and acceptable training time even when the number of devices in the network was increased. This is especially crucial for practical applications, where the sheer numbers of devices and data can drown out conventional security mechanisms. Its capability to manage high, distributed devices without sacrificing performance makes the presented system a suitable solution for current, decentralized scenarios.

In addition, the methodology proposed also deals with the important problem of privacy in data-intensive security systems. Through federated learning, our model is able to perform training locally on edge devices, without transferring sensitive data to central servers, thereby maintaining user privacy. This privacy-preserving nature is especially important in the current regulatory environment, where data privacy is of utmost importance.

In summary, the suggested methodology offers a strong, scalable, and privacy-augmented solution for securing cloud and IoT systems. The integration of deep learning, blockchain, and federated learning provides a strong and versatile framework that can identify subtle cyber-attacks while ensuring data integrity and user privacy. The results of the experiment confirm the efficacy of the system, and its scalability over vast distributed networks renders it a highly effective instrument for securing the future of cloud and IoT infrastructures. As cyber attacks evolve, this integrated methodology forms a promising building block for the future of digital-age cybersecurity.

REFERENCES:

[1] Govindaram, Anitha. "Flbc-ids: a federated learning and blockchain-based intrusion detection system for secure iot environments." *Multimedia Tools and Applications* (2024): 1-23.

[2] Sarhan, Mohanad, et al. "HBFL: A hierarchical blockchain-based federated learning framework for collaborative IoT intrusion detection." *Computers and Electrical Engineering* 103 (2022): 108379.

[3] Moulahi, Tarek, et al. "Privacy-preserving federated learning cyber-threat detection for intelligent transport systems with blockchain-based security." *Expert Systems* 40.5 (2023): e13103.

[4] Rahman, Mohamed Abdur, et al. "Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach." *Ieee Access* 8 (2020): 205071-205087.

[5] Ferrag, Mohamed Amine, et al. "Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis." *IEEe Access* 9 (2021): 138509-138542.

[6] Kollu, Venkatagurunatham Naidu, et al. "Cloud-based smart contract analysis in fintech using IoT-integrated federated learning in intrusion detection." *Data* 8.5 (2023): 83.

[7] Liu, Hong, et al. "Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing." *IEEE Transactions on Vehicular Technology* 70.6 (2021): 6073-6084.

[8] Mothukuri, Viraaji, et al. "Federated-learning-based anomaly detection for IoT security attacks." *IEEE Internet of Things Journal* 9.4 (2021): 2545-2554.

[9] Nazir, Ahsan, et al. "Enhancing IoT security: a collaborative framework integrating federated learning, dense neural networks, and blockchain." *Cluster Computing* 27.6 (2024): 8367-8392.

[10] Mamunur, Rashid Md. *A Novel Intrusion Detection System in IoT Networks Leveraging Blockchain-Enabled Federated Learning*. Diss.

[11] Ullah, Irshad, et al. "Securing internet of vehicles: a blockchain-based federated learning approach for enhanced intrusion detection." *Cluster Computing* 28.4 (2025): 256.

[12] Begum, Khadija, et al. "BFLIDS: Blockchain-driven federated learning for intrusion detection in IoMT networks." *Sensors* 24.14 (2024): 4591.

[13] Preuveneers, Davy, et al. "Chained anomaly detection models for federated learning: An intrusion detection case study." *Applied Sciences* 8.12 (2018): 2663.

[14] Abou El Houda, Zakaria, et al. "Blockchain-enabled federated learning for enhanced collaborative intrusion detection in vehicular edge computing." *IEEE Transactions on Intelligent Transportation Systems* 25.7 (2024): 7661-7672.

[15] Abdel-Basset, Mohamed, et al. "Federated intrusion detection in blockchain-based smart transportation systems." *IEEE Transactions on Intelligent Transportation Systems* 23.3 (2021): 2523-2537.

[16] Alkadi, Osama, et al. "A deep blockchain framework-enabled collaborative intrusion detection for protecting IoT and cloud networks." *IEEE Internet of Things Journal* 8.12 (2020): 9463-9472.