International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(18s) 2026 pp. 463-483
https://doi.org/10.70917/ijcisim-2026-4882
Article

Adaptive Intrusion Detection in WSNs Using Federated Deep Learning and Homomorphic Encryption

N. Karthick[1], K. Ranjith Singh[2]

[1]Department of Computer Science, Karpagam Academy of Higher Education (Deemed to be University), Coimbatore, Tamil Nadu, India. nkmcapgp@gmail.com
[2]Department of Information Technology, Karpagam Academy of Higher Education, (Deemed to be University), Coimbatore, Tamil Nadu, India. ranjithsingh.koppaiyan@kahedu.edu.in

Abstract: Wireless Sensor Networks (WSNs) play a crucial role in various applications, but their vulnerability to malicious nodes and data breaches hinders their full potential. Traditional security methods often struggle to keep pace with evolving attack patterns and can introduce privacy concerns. This research proposes a novel framework for anomaly detection in WSNs that leverages federated deep learning and prioritizes real-time adaptation and data privacy. Sensor nodes collaboratively train adaptive deep learning models to identify anomalies in real-time, enabling continuous learning and response to evolving threats. Partial Homomorphic Encryption (PHE) safeguards sensitive data throughout the network, ensuring data confidentiality. The trade-off between security and computational cost associated with PHE is acknowledged. The effectiveness of the proposed system FedShield-PHE will be evaluated through simulations, comparing its performance to existing methods across various metrics including detection accuracy, network overhead, and energy consumption. This research offers a promising path forward for securing WSNs by enabling distributed, privacy-preserving anomaly detection with real-time adaptation capabilities.
Keywords: Blockchain, Deep Learning, Partial Homomorphic Encryption (PHE), Intrusion Detection, WSN, Anomaly Detection

1. INTRODUCTION

WSN plays a vital role in a wide range of mission-critical applications, including military surveillance, disaster response, environmental monitoring, and healthcare systems. However, their decentralized architecture, wireless communication, and physical exposure make them particularly susceptible to a variety of cyber threats. Intrusion Detection Systems are crucial for safeguarding WSNs, as they provide a secondary line of defense by identifying malicious activities that bypass conventional security protocols. An effective IDS in WSNs must balance strong detection capabilities with minimal resource consumption, ensuring real-time protection without degrading network performance. This makes the development of lightweight, adaptive, and privacy-preserving IDS frameworks a pressing requirement for secure WSN deployment.

Adaptive intrusion detection in Wireless Sensor Networks addresses the critical need for robust security measures in these persistent, resource-constrained networks. Wireless Sensor Networks consist of numerous tiny sensor nodes deployed in diverse environments to collect and transmit data wirelessly. Due to their distributed nature and typically unattended operation, WSNs are prone to various security threats such as node compromise, data interception, and unauthorized access (Godala, S., et. al., 2020). Traditional intrusion detection systems designed for conventional networks often fail to meet the unique challenges posed by WSNs, including limited computational resources, constrained energy supplies, and dynamic network conditions. Therefore, adaptive intrusion detection systems tailored for WSNs are essential to continuously monitor network activities, detect anomalies or intrusions promptly, robust and respond effectively to mitigate potential threats.

The integration of Federated Deep Learning and Partial Homomorphic Encryption offers a promising solution to the critical challenges posed by intrusion detection in Wireless Sensor Networks. FDL (Agrawal, S., et. al., 2022) addresses privacy concerns by training models collaboratively across multiple nodes without centralizing sensitive data, while PHE ensures the confidentiality of information during model aggregation. This synergistic approach enables real-time detection and response capabilities, making it indispensable for safeguarding WSNs against sophisticated cyber threats. By leveraging local computational power and minimizing communication overhead, FDL optimizes resource efficiency in WSNs. Additionally, its adaptive learning capabilities allow for continuous model updates based on local data, enabling effective countermeasures against evolving threats. PHE complements FDL by facilitating secure model aggregation, ensuring the confidentiality and integrity of information during the process. Together, these technologies empower WSNs to operate efficiently and securely in diverse and dynamic environments.

This paper explores the innovative application of FDL and PHE in WSN intrusion detection delving into the technical aspects of these technologies and their potential benefits in enhancing WSN security. The aim to create a robust and adaptable intrusion detection system that can effectively address the unique challenges of WSNs by combining FDL and PHE. In conclusion, leveraging federated deep learning and homomorphic encryption for adaptive intrusion detection in WSN offers a robust solution to enhance security while preserving data privacy. This approach enables real-time threat detection, continual adaptation to new threats, and efficient utilization of resources in dynamic WSN environments, marking a significant advancement in safeguarding these critical systems against evolving cybersecurity risks.

2. LITERATURE REVIEW

Godala, S., et. al., 2020 conducted a detailed study on intrusion detection system in WSNs. They examined the architecture of WSN and categorizes of WSN attacks and explored different IDS mechanisms like signature-based, anomaly-based, specification-based and hybrid approaches. Additionally, they discussed the performance metrics of these mechanisms and highlighted the energy efficiency advantages of cluster-based WSN architectures. Gavel, S., et. al., 2021 proposed a novel technique by combining multi-varying kernel density estimation with distributed computing to detect long-lasting intrusive attacks caused by compromised nodes in the network. They analyzed individual probabilities and computed global Probability Density Functions (PDFs) to estimate and detect intrusions. Pearson's divergence (PE) is used for in-network detection with a low False Positive Rate (FPR) and approximation of PE divergence is achieved using distributed computing techniques. An entropy-based centralized computing method is also proposed. The robustness of the approaches is compared based on accuracy and FPR using real-world datasets. Lu, X., et. al., 2020 proposed an intrusion detection model for WSNs using the improved particle swarm optimization (IPSO) algorithm combined with a back-propagation neural network (BPNN), referred to as IPSO-BPNN. The IPSO algorithm optimized the initial parameters of BPNN, preventing local optima. They tested their model with the NSL-KDD and UNSW-NB15 datasets and demonstrated faster convergence, higher detection accuracy, and a lower false positive rate compared to BPNN and BPNN optimized with PSO. Abhale, A.B., et. al., 2023 proposed a method to enhance intrusion detection in Wireless Sensor Networks using recursive feature elimination and resampling. They tested machine learning algorithms such as Random Forest, Support Vector Machine, K Nearest Neighbor, and Decision Tree on the NSL KDD dataset. This approach reduced the feature set to improve speed and accuracy. Recursive Feature Elimination and Resampling techniques were used for feature selection. Their results demonstrated significant improvements in performance, speed, and accuracy. Agarkar, A.A., et. al., 2020 proposed a post-quantum security solution for data aggregation in WSNs using lattice cryptography (R-LWE). They addressed the vulnerabilities of traditional methods like Elliptic Curve Cryptography against quantum attacks. Their solution provided confidentiality, integrity, and authenticity during communication. It also demonstrated better performance and was more lightweight compared to ECC-based and symmetric homomorphic encryption schemes.

Alzubi, O.A., 2022 proposed a deep learning-based Frechet and Dirichlet model for intrusion detection in IWSNs. This model enhanced the data security by detecting intrusions through network activity analysis. They introduced FHD-DS model and improved the intrusion detection rate (IDR) with minimal detection time. The FHDT

feature extraction method identified key traffic features, and the Dirichlet distribution ensured secure data transmission. Experimental results demonstrated that the FHD-DS model achieved higher IDR with reduced detection time on the KDD Cup 99 dataset. Rajasoundaran, S., et. al., 2024 proposed a secure intrusion detection system for UWSNs using LSTM networks and GANs to enhance monitoring and channel assessment. Intelligent agents on sensor nodes integrated LSTM-MAC principles, rule-based monitoring, and secure algorithms like SHA-3 and Two Fish to ensure real-time intrusion detection and secure communication. This system achieved 5–10% improved performance over existing methods in metrics like attack prevention and network reliability. Alkhdour, T., et. al., 2024 proposed a hybrid-deep learning protocol integrating federated learning and fog computing to enhance healthcare systems. Federated learning ensured data privacy by training models on devices, while fog computing processes data near the source, reducing latency and improving security compared to cloud systems. In this framework, they employed homomorphic distributed protocols to safeguard electronic medical records, offering a secure, efficient, and low-latency solution for next-generation healthcare. Goswami, C., et. al., 2023 proposed a DCNN-based approach for disease prediction in WSN data to optimize Henry's gas solubility. Their method leveraged DCNNs to extract spatial and temporal patterns from sensor data which improved prediction accuracy. Patient health data collected by wireless sensors were encrypted using homomorphic encryption to ensure privacy. Experimental results showed that the DCNN approach outperformed traditional methods in disease prediction. They highlighted the potential of DCNNs for robust disease prediction in WSN applications. Giachanatzis Grammatikopoulos, A., 2021 explored the use of Federated Learning (FL) to enhance intrusion detection in WSNs. This study focused on designing FL-based architectures to improve detection accuracy while preserving data privacy, addressing challenges like resource constraints and communication overhead. Techniques like differential privacy and secure aggregation were utilized to ensure secure model updates while demonstrating FL's potential to advance cybersecurity in distributed WSN environments.

Babenko, L., et. al., 2020 reviewed methods to secure data in wireless sensor networks by focusing on cryptographic protection techniques. They discussed the challenges posed by memory, processor speed, and power consumption limitations in these networks which impact the selection of encryption algorithms. They investigated the use of homomorphic encryption, particularly partial homomorphic encryption through the Paye Cryptosystem, to safeguard data transmitted by sensor nodes and improve network routing, particularly for sensors monitoring environmental parameters. Zeng, R., et. al., 2011 proposed a distributed fault/intrusion-tolerant data storage scheme based on network coding and homomorphic fingerprinting for WSNs. This scheme is used for network coding to encode and distribute data fragments for high availability. Homomorphic fingerprinting enabled the fast detection of incorrect fragments and initiated data maintenance. This approach addressed data integrity and availability in unreliable WSNs. Theoretical analysis and simulations demonstrated the scheme's effectiveness and efficiency. Bhushan, B., et. al., 2020 proposed a secure location-based aggregator node selection scheme for WSNs to balance energy consumption and enhance security. This scheme used ECC with non-pairing homomorphic encryption, generating a 176-bit key from node ID, ECC key, and cluster head (CH) distance. Homomorphic encryption enabled encrypted data aggregation and reduced energy use of CH. Simulations demonstrated improved network lifetime and robust attack countermeasures compared to existing schemes. Khedr, A.M., et. al., 2024 proposed a model called Time Synchronized Multivariate Regressive Convolution Deep Neural Network (TSMR-CDNN) for detecting sinkhole attacks in WSNs. The model used reverse time synchronization to reduce detection delays and analyzed multivariate data with the Broken-stick regression method. It optimized weight parameters using a steepest gradient function to lower false positive rates and improved accuracy. Experimental results showed a 5% increase in detection accuracy, 3% improvement in precision, recall, and F-measure, and a 10% reduction in detection time. Sadia, H., et. al., 2024 introduced an Intrusion Detection System for Wi-Fi-based Wireless Sensor Networks by employing a convolutional neural network. They applied feature selection to reduce the dimensionality of data from 154 features to 13 key features and used standard scaling for preprocessing. This model achieved 97% accuracy, loss of 0.14, and a minimal False Alarm Rate, surpassing other models such as DNN and LSTM in detecting cyber threats.

Ifzarne, S., et. al., 2021 (a) proposed a Cluster-based Semi-Homomorphic Encryption Aggregated Data (CSHEAD) scheme for secure data aggregation in WSNs. This scheme addressed the privacy protection challenges with reduced control overhead and higher attack detection accuracy. They measured network metrics such as control

packet overhead, delay, throughput, and packet delivery ratio alongside attack detection accuracy. Ifzarne, S., et. al., 2021 (b) proposed a secure data collection scheme for WSN to protect data during aggregation. They used homomorphic encryption and compressed sensing to maintain data confidentiality while reducing network load. Compressed sensing reduced data size during sampling, with data recovery performed at the resource-rich sink. Encryption was applied at each node with minimal effort, allowing operations on encrypted data without decryption. Kantzavelou, I., et. al., 2023 proposed a game-theory-based approach for detecting intrusive activities from insiders in WSN. The approach involved modeling a three-player non-cooperative game to analyze the interactions between an insider and the IDS within the WSN. By solving the game, the researchers were able to identify the Nash equilibria, which helped determine optimal strategies for the players. The analysis of these equilibria provided useful recommendations for strategies that assisted in identifying and isolating compromised nodes in the WSN. This approach was tested on a WSN environment, where the proposed strategies successfully enhanced node identification and isolation. Ali, M., et. al., 2022 proposed a Secure Information Sharing Protocol (SEIS) for mobile wireless networks using homomorphic encryption and Elliptic Curve Cryptography (ECC) curve along with discrete logarithm. The protocol optimized data aggregation security with minimal communication overhead and efficient key trace communication. AES operations improve performance by reducing computational time and extending network lifespan. Elliptic curve point multiplication is executed in 2.623 seconds on a 1.50 GHz processor to ensure secure and efficient wireless device communication. Aljohani, A.A., et. al., 2023 proposed a Trust-Bat-Adaptive Homomorphic Crypto Routing Protocol for secured healthcare data sharing through wireless networks in mobile computing. In their protocol, they employed 5G technology for data transfer, homomorphic encryption for secure communication, and a multi-gradient spider monkey optimization technique for efficient data processing. Performance evaluation demonstrated improved security, energy efficiency, and reduced memory usage compared to traditional methods.

Zhang, Z., et. al., 2021 proposed a Homomorphic Fingerprinting-based Detection and Location of Malicious Nodes (HFDLMN) scheme for WSNs. This scheme divided data into packets, then it is transmitted them through multiple paths, and verified the packets at the base station to identify malicious nodes. After detecting the malicious nodes, a location algorithm was used to pinpoint their positions, while valid packets were combined to reconstruct the original data. This approach eliminated the need for trust models or monitoring nodes. Qi, X., et. al., 2020 proposed a privacy-preserving data aggregation scheme for WSNs using an asymmetric key encryption scheme based on elliptic curve cryptography. The scheme optimized key generation and management, employed privacy homomorphism for end-to-end encryption, and integrated a rotation-based MAC generation algorithm for hop-by-hop verification. Experimental results demonstrated lower energy consumption and higher security levels, effectively addressing the trade-off between energy efficiency and security in WSNs. Halidoddi, G., et. al., 2022 proposed a Multi-Objective Trust-Based Bat Optimization Algorithm (MOTBOA) by combining an Enhanced Homomorphic Cryptosystem (EHC) for secure data transmission in WSN. This MOTBOA method ensured secure clustering and routing, while EHC enhanced encryption and decryption during data communication. This MOTBOA-EHC method was evaluated against existing methods like TDSR, MCSO, and SQEER, achieving a Packet Delivery Ratio (PDR) of 92% at 100 rounds, which outperformed the benchmarks. Peng, C., et. al., 2021 developed a secure data aggregation scheme for WSNs using a Chinese remainder theorem conversion method to encode multidimensional data and support linear homomorphic encryption. This scheme enabled diverse aggregation functions, ensured data security, and resisted false data injection attacks. It demonstrated superior performance in high-dimensional scenarios, supporting twice the dimensions per ciphertext of existing methods while reducing computation and communication costs. Saravanaselvan, A., et. al., 2024 proposed a Feed Forward Back Propagation Neural Network (FFBPNN) optimized with the Woodpecker Mating Algorithm for secure and efficient dynamic cluster-based routing in WSN. The FFBPNN-WMA-ECHC-WSN method was implemented using the NS2 tool and evaluated against metrics such as delay, throughput, network lifetime, and energy consumption. This method achieved a significant improvement and reduced delays by up to 99.01% and throughput increased by up to 97.25%, outperforming existing models including EHCERA-SDT-WSN and DSA-ECC-PSO-SDT-WSN. Li, L., et. al., 2022 proposed a secure and efficient scheme for wireless sensor networks, combining compressed sensing, deep learning, and encryption techniques. Their system ensured high-

performance data collection, authenticated nodes using lightweight hash algorithms, and protected data with homomorphic encryption. It also evaluated and excluded compromised nodes, demonstrating effectiveness in image data acquisition with 99.4% accuracy. Khan, Z.A., et. al., 2023 proposed a system that utilized four DL models, GRU, LSTM, CNN, and ANN along with a Real-Time Message Content Validation (RMCV) scheme based on blockchain for detecting and removing malicious nodes. The system also employed a decentralized blockchain network on cluster heads and base stations, using a proof-of-authority consensus protocol for efficient node registration. Results showed improved accuracy, throughput, delay and node lifetime compared to traditional routing protocols such as LEACH and HMGEAR. Additionally, the blockchain network was verified to be secure against vulnerabilities as analyzed by Oyente.

Even though there have been many improvements in systems that detect attacks in WSNs, several important challenges remain unresolved. Many existing IDS models depend on centralized learning approaches that require the transmission of raw data from sensor nodes to a central server. This centralized approach introduces serious concerns related to data privacy and security, especially in sensitive environments such as military, healthcare, and industrial monitoring. Although federated learning has emerged as a decentralized solution to preserve data privacy, most implementations lack effective integration with privacy-preserving techniques during the model aggregation process. Furthermore, traditional IDS frameworks often rely on static models that are unable to adapt in real-time to new and evolving cyber threats. Deep learning techniques like convolutional neural networks, long short-term memory networks, and hybrid models have improved detection capabilities, but they tend to be computationally intensive and are not well suited for the limited processing power and energy resources of WSN nodes. Additionally, while some studies have explored the use of homomorphic encryption to secure data, the specific combination of PHE with federated deep learning for adaptive intrusion detection in WSNs remains underexplored. There is also a lack of comprehensive evaluation methodologies that assess detection accuracy, energy efficiency, network overhead, and privacy protection in an integrated manner, which is essential for real-world implementation in resource-constrained WSN environments.

While existing research explores individual aspects like federated deep learning for privacy and PHE for secure aggregation, a comprehensive framework integrating these with real-time adaptation in resource-constrained WSNs is lacking. Existing studies often overlook the dynamic nature of threats and the need for continuous model updates. Furthermore, the compromises between security and computational overhead associated with PHE in the context of federated learning for WSNs require further investigation. The objective of this work is to develop a robust and secure anomaly detection system for sensor networks by leveraging advanced techniques such as partial homomorphic encryption, federated deep learning, and blockchain-based trust management. The FedShield-PHE aims to ensure real-time adaptation and privacy-preserving data processing by enabling local model training on sensor nodes while facilitating collaborative learning across the network. Additionally, the system seeks to enhance network security through encrypted data aggregation and secure routing decisions based on blockchain trust scores, ultimately improving the overall performance and reliability of the sensor network. This research introduces a federated deep learning framework designed for real-time intrusion detection in resource-constrained WSNs. By combining adaptive learning and privacy-preserving encryption, the system enables secure, efficient, and scalable threat detection across sensor node. The contribution of this research work includes:

1. **Using Federated Learning:** This work introduces a new intrusion detection system for WSNs that uses federated learning to detect attacks without sending raw data to a central server by keeping the data private.
2. **Real-Time Learning and Adaptation:** The system can keep learning and updating itself across sensor nodes, helping it quickly detects new or changing types of attacks.
3. **Data Privacy with Encryption:** It uses Partial Homomorphic Encryption to keep model updates secure, so the original sensor data stays safe and encrypted during training.
4. **Designed for Low Power Devices**: The system is made lightweight using model compression and efficient communication methods, so it uses less energy and works well on low-power sensor nodes.

The paper is organized as follows: Section 1 introduces the concept of traditional intrusion detection systems and the integration of FDL and PHE. Section 2 surveys related work and identifies research gaps. Section 3 describes the methodology, including feature extraction, partial homomorphic encryption, federated learning, and anomaly detection. Section 4 presents experimental results, dataset description, and performance analysis. Section 5 concludes with key findings of the proposed work and future directions.

3. METHODOLOGY

The FedShield-PHE is tailored for detecting anomalies within a network of sensor nodes. Sensor nodes autonomously collect and preprocess data, employing Partial Homomorphic Encryption for secure data encryption. Subsequently, features are extracted and utilized to locally train a deep learning model on each node, enabling real-time adaptation. Federated deep learning is leveraged to enhance the model's performance and safeguard privacy by enabling collaborative model training without exposing raw data. Models are aggregated at base station and it monitors network performance and securely stores data. Figure 1 illustrates the comprehensive architecture of the proposed methodology.

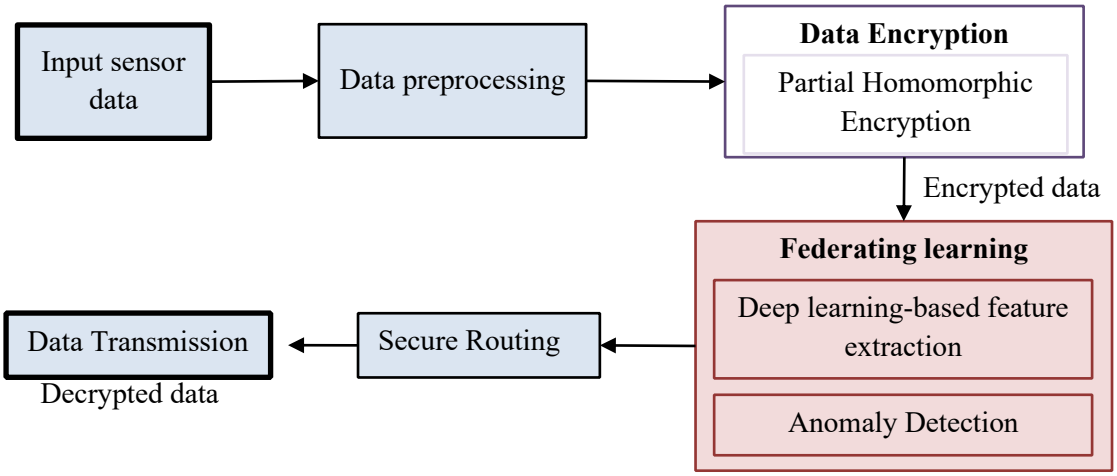

Figure 1: Architecture of the proposed work

3.1 Sensor Nodes

Sensor nodes serve a dual purpose by functioning both as data collectors and as intelligent processors capable of local analysis. The system continuously monitors network traffic patterns, transmission rates, and energy consumption behaviours to identify any anomalies that could indicate a potential intrusion. In sensor nodes, these features help in identifying unusual activities that could indicate network intrusions. Traffic features like duration, protocol type, service, and flag provide insights into how data flows through the network. Communication features such as source bytes and destination bytes track the volume of data exchanged between nodes. Behavioural features like land, logged in, count, and same service count help detect repetitive or suspicious patterns. Resource-based features, including attack type and difficulty level, assist in classifying the type of intrusion and understanding its detection complexity.

The extracted feature vector from each observation consists of 12 features, forming a vector in a shape of 1×12. If a sensor node collects N such observations within a given time window, the resulting dataset will have a size of N×12, where each row represents an observation and each column corresponds to a feature. This structure ensures that the dataset is well-organized for preprocessing and machine learning operations. To ensure the reliability of collected data, each node performs lightweight preprocessing techniques including noise filtering, normalization, and data transformation. This early-stage processing minimizes the influence of environmental disturbances and hardware inconsistencies, allowing the system to operate with cleaner, more consistent inputs. Once preprocessed, the data is

fed into a locally deployed lightweight deep learning model. These models are designed to adapt dynamically, learning from real-time network behaviors to identify potential anomalies more effectively over time. Unlike traditional centralized learning systems that send raw data to a remote server, the proposed framework adopts FL, a privacy-preserving strategy in which only model updates are transmitted while raw data remains on the device.

To secure the communication of these updates, PHE is applied at each sensor node before transmission. This ensures that even if information is intercepted during transit, its contents remain unintelligible to unauthorized entities. The encrypted updates are then sent to a central FL server, typically hosted at the base station which acts as a secure aggregator. Without accessing any raw data, the server combines the encrypted contributions received from all nodes to build a refined global model. The encryption module embedded within each sensor node facilitates the homomorphic encryption and decryption processes efficiently and is optimized for resource-constrained environments. This layered design enables real-time, privacy-preserving anomaly detection while maintaining scalability and energy efficiency across distributed wireless sensor networks.

3.2 Data Preprocessing

Once the feature vectors are collected, they undergo preprocessing to enhance data quality before encryption and training. In this system, preprocessing is applied directly to the raw sensor data collected from deployed WSN nodes to ensure that the anomaly detection model receives clean, reliable input.

3.2.1 Data cleaning

Kalman filtering and duplicate removal are the two key techniques are employed, to handle noisy fluctuations and duplicates commonly observed in real-time sensor data. For instance, consider a sequence of temperature readings like [30.0, 31.5, 29.2]. These values may fluctuate due to environmental noise or sensor sensitivity. The Kalman filter uses a predictive-corrective model to smooth the data, yielding refined values such as [30.0, 30.4, 30.2]. This stabilizes the input without overreacting to random deviations, enhancing the accuracy of anomaly detection. In addition, duplicate data entries often occur due to retransmissions in unstable network conditions. If the system receives readings like: $[10{:}00{:}01 \rightarrow 30.0°C, 10{:}00{:}02 \rightarrow 30.1°C, 10{:}00{:}02 \rightarrow 30.1°C, 10{:}00{:}03 \rightarrow 30.2°C]$, the second 10:00:02 the entry would be considered as a duplicate. Such redundant entries are identified by comparing timestamps and values, and are removed to prevent skewing the training process. This not only reduces unnecessary processing but also conserves energy and bandwidth in resource-constrained sensor nodes. By applying Kalman filtering to smooth noisy signals and performing duplicate removal to eliminate redundant data, the system ensures that only high-quality data is used for local model training. This enhances detection accuracy while supporting the real-time and privacy-preserving goals of our federated learning-based anomaly detection framework.

3.2.2 Data normalization

After eliminating noise and duplicates, data normalization is included as an additional layer of preprocessing at the sensor node level. While cleaning techniques help ensure that the data is accurate and complete, normalization maintains numerical consistency across features, which supports the effective functioning of machine learning models. In WSNs, sensor nodes typically collect heterogeneous parameters such as packet size (bytes), energy consumption (volts), and transmission delay (milliseconds). These features often differ in scale and units, which can introduce bias into the learning process, particularly for models that rely on distance-based calculations or gradient-based optimization. Without normalization, a feature like energy consumption with a larger numerical range could create a disproportionately large influence on the model and overshadow features such as transmission delay or packet count with smaller ranges. To address this, min–max normalization is applied to rescale each feature f_x into a fixed range, typically $[0,1]$ using:

$$f_x' = \frac{f_x - f_{x_{min}}}{f_{x_{max}} - f_{x_{min}}}$$

where f_x' is the normalized value, $f_{x_{min}}$ is the minimum observed value of the feature, and $f_{x_{max}}$ is the maximum observed value. For instance, an energy value of 0.1V within a range of 0.1V to 5V is mapped to 0, while 5V becomes 1, with intermediate values scaled proportionally. This transformation helps ensure that all features contribute more equally during training. However, since min-max normalization is sensitive to outliers, its application is monitored carefully, especially in datasets that contain rare but impactful events. Overall, normalization supports improved interpretability, training stability, and model convergence within the federated anomaly detection framework deployed in resource-constrained WSN environments.

3.3 Encrypt Data using PHE

After preprocessing, the cleaned and normalized data is encrypted using PHE to ensure privacy-preserving transmission. PHE is a cryptographic technique for enhancing data security and privacy, particularly in applications of network security. By utilizing PHE, data from both legitimate and potentially malicious nodes can be encrypted before transmission, ensuring that sensitive information remains confidential throughout the processing stages. This encryption method enables computations to be carried out directly on ciphertexts, producing encrypted results that, once decrypted, are identical to the outcomes of performing the same operations on the original plaintext data. This property is essential for maintaining privacy while still enabling useful insights to be extracted. In network security, especially within resource-constrained WSN environments, sensor nodes collect data that must be securely transmitted. PHE enables these nodes to encrypt their data before sending it, thereby protecting the data from interception or tampering. The process begins with data collection from the sensor node. Data gathered at the sensor nodes is then preprocessed to ensure the quality and consistency. The extracted feature vector from each observation is represented as

$$f_x = [f_1, f_2, f_3, \dots \dots, f_{12}]$$

where, each f_i represents one of the 12 extracted features. For N observations at a node, the dataset becomes $f_x \in R^{N\times 12}$. The preprocessed data f_x' then encrypted using PHE, implemented via the Paillier cryptosystem, which enables secure computation on encrypted data.

Step 1: Key Generation

Two large prime numbers p and q are chosen, and their product $n = p.q$ forms part of the public key. A generator $g \epsilon Z_{n^2}^*$ is selected for encryption computations, g is the set of all integers less than n^2 that are coprime to n^2 is used. In the Paillier cryptosystem for encrypting WSN sensor features, the private key consists of λ and μ. λ and μ is computed using the below equation.

$$\lambda = lcm(p-1, q-1)$$

$$\mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} mod\ n$$

where λ is first calculated and then used to compute μ. The function $L(u) = (u-1)/n$ is applied in $L(g^{\lambda} \bmod n^2)$ to extract a value proportional to the plaintext during decryption. The public key $(pk = (n, g))$ is used to encrypts feature values, while the private key $(sk = (\lambda, \mu))$ is kept by the trusted authority.

Step 2: Encryption

After preprocessing, each feature f_i' from the feature vector f_x' is directly encrypted using the Paillier cryptosystem. The encryption formula is

$$c_{ij} = E_{pk}(f_i') = g^{f_i'}.r^n \bmod n^2$$

where E_{pk} denotes the encryption function using the public key $pk = (n, g)$ and $r \in Z_n^*$ is a randomly chosen value for each encryption. The resulting encrypted features c_{ij} are then securely transmitted and used for federated learning model training. Now, consider a preprocessed feature f_1' with a corresponding random number r_{i1}. The encrypted feature is calculated as: $c_{i1} = E_{pk}(f_1') = g^{f_1'}.r_{i1}^n \ mod \ n^2$. All the feature values are stored in c_{ij}.

Step 3: Aggregation

Finally, the aggregated encrypted result is computed using the homomorphic property: $c_{agg} = c_{i1}.mod\ n^2$, where the aggregation is expressed as $c_{agg1} = c_{ij}.mod\ n^2$. This aggregated ciphertext corresponds to the encryption of the sum of features: $c_{agg2} = E_{pk}(f_1' + f_2' + \cdots + f_n')$. where, c_{agg1} will perform as product of cipher text and c_{agg2} is the encryption form of sum of plain text. By combining these values, will perform the decryption.

The ability to perform such operations is particularly beneficial in IoT environments, where sensor nodes are often limited in computational power and energy resources. By encrypting data at the source, these nodes ensure that their data remains secure during transmission. The central server or processing unit can then perform necessary computations on the encrypted data, such as aggregating sensor readings, without compromising privacy. This approach not only enhances security by preventing unauthorized access to sensitive information but also preserves privacy by ensuring that individual data points are not exposed during processing. Consequently, PHE provides a robust framework for secure and efficient data handling in network security applications, making it an ideal solution for scenarios requiring privacy-preserving computations.

Algorithm: PHE_Encryption()

Input: Preprocessed feature vector $f_x' = [f_1', f_2', \dots, f_{12}']$

Step 1: Key Generation

1. Select two large prime numbers p and q.
2. Compute modulus: $n = p.q$, $n^2 = n\,.n$
3. Compute:

$$\lambda = lcm(p-1, q-1)$$

4. Select a generator $g \in Z_{n^2}^*$ (commonly $g = n + 1$).
5. Compute:

$\mu = (L(g^\lambda \ mod n^2))^{-1} \ mod \ n,$ where $L(u) = \frac{u-1}{n}$

6. Define keys:
    - Public key: $pk = (n, g)$
    - Private key: $sk = (\lambda, \mu)$

Step 2: Encryption

For each normalized feature $f_i' \in f_x'$

1. Select random number $r_{ij} \in Z_n^*$
2. Encrypt using public key:

$$c_{ij} = E_{pk}(f_i') = g^{f_i'}.(r_{ij})^n \ mod \ n^2$$

3. Store ciphertext $c_{ij.}$

Output: Encrypted feature vector c_{ij}

3.4 Federated Learning

In the federated learning framework, each sensor node collects its own network data, such as packets, delays, and energy usage, which is then preprocessed through techniques like noise removal and normalization before extracting relevant features, including traffic patterns, behavioral indicators, and resource metrics. This refined data is encrypted using Partial Homomorphic Encryption. Instead of transmitting raw data, the node will transmit encrypted data to ensure confidentiality. Sensor node generates a trained local model and sends them to the cluster head. CH collects all the local model from the sensor nodes and perform local aggregation and update the model parameter. The CH plays a main role by first identifying and discarding malicious nodes through anomaly detection, then aggregating the encrypted updates from legitimate nodes into a cluster-level model using homomorphic properties, allowing secure aggregation without decryption. These cluster-level models from multiple CHs are further sent to the base station, where they are aggregated into a global deep learning model. The improved global model is redistributed back to all sensor nodes, which use it as the new baseline for further local training, enabling continuous refinement and adaptive learning against evolving threats. Throughout this process, no raw data is ever exchanged, malicious nodes are effectively isolated, and encrypted updates ensure privacy and security, making the system robust for intrusion detection in wireless sensor networks.

Federated learning framework (shown in Figure 2) consist of five components: i) Base station initializes a global deep learning model with random weights and distributes it to all sensor nodes, ii) Each sensor node trains the model using its local dataset. Lightweight deep learning ensures efficiency in resource-constrained nodes while maintaining robust anomaly detection, iii) After local training, each node generates model updates**,** iv) Transmitted to the cluster head which identifies and aggregates updates into a cluster-level model using homomorphic properties, ensuring secure aggregation without exposing individual updates, v) Cluster-level models from multiple Cluster heads are sent to the base station, where they are aggregated to update the global deep learning model. The refined model is redistributed to all sensor nodes for the next training round, enabling continuous refinement and adaptive learning.

Federated learning process begins with initialization, where the base station creates a global deep learning model with random weights and distributes it to all sensor nodes. During the local training phase, each sensor node trains this model using its own dataset consists of traffic, communication, behavioral, and resource-based attributes. Lightweight deep learning ensures that anomaly detection is performed efficiently despite the limited resources of sensor nodes. After local training, each node generates the updated model. The updated models are validated and used for further model aggregation**.** These local models are then transmitted to the cluster head. At the CH, anomaly detection techniques are applied to filter out malicious contributions, and the cluster-level aggregation step securely combines the updated models using homomorphic properties. The resulting cluster-level model is then forwarded to the base station for global aggregation, where updates from multiple clusters are integrated into the global deep learning model. The refined global model is redistributed to all nodes, providing a stronger baseline for the next training round and enabling adaptive learning against evolving threats.

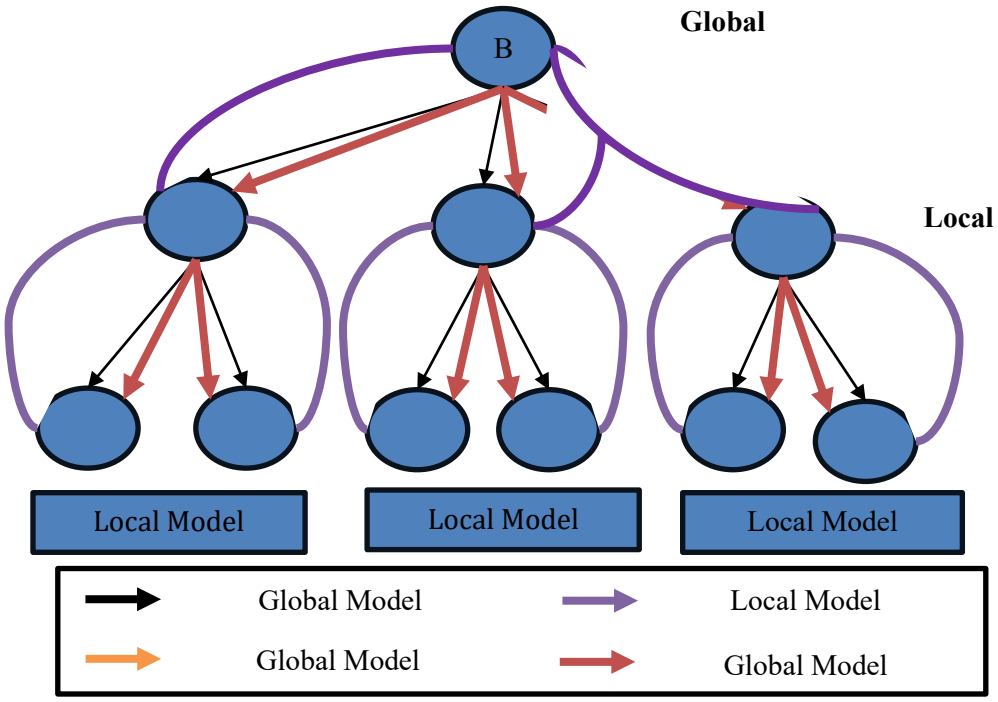

Figure 2: Hierarchical Federated Learning Architecture

3.5 Deep Learning-Based Feature Extraction

The deep learning-based feature extraction process utilizes a lightweight model designed to efficiently capture complex patterns from sensor node data. The model starts with an input layer that accepts a 12-dimensional feature vector composed of traffic-based, communication-based, behavioral, and resource-based attributes. These features represent various aspects of network activity and node behavior, providing a rich dataset for analysis. The input data is then passed through a one-dimensional convolutional layer (Conv1D) with 32 filters and a kernel size of 3. This layer uses ReLU activation to introduce non-linearity and extract meaningful local patterns across the feature sequence. The convolution operation slides over the input sequence, producing 32 distinct feature maps that highlight different characteristics of the data. Following this, a max pooling layer reduces the dimensionality by downsampling the feature maps, retaining the most significant information while reducing computational complexity. After pooling, the multi-dimensional output is flattened into a one-dimensional vector, enabling the dense layers to process the extracted features collectively. The flattened vector is fed into two fully connected dense layers. The first dense layer contains 64 neurons with ReLU activation, followed by a dropout layer with a rate of 0.2 to prevent overfitting and improve generalization. The second dense layer has 32 neurons, also with ReLU activation and a dropout of 0.2, further refining the feature representation while enhancing robustness against noise.

Finally, the model concludes with an output layer consisting of two neurons and a softmax activation function, which produces probabilities for binary classification between legitimate and malicious nodes. The neuron with the higher probability determines the classification outcome. This deep learning-based feature extraction approach effectively transforms raw sensor data into a compact and discriminative representation, enabling accurate and efficient anomaly detection in wireless sensor networks. Table 1 outlines the layers of the deep learning model used for feature extraction. It starts with an input layer, followed by a Conv1D and MaxPooling layer to extract and reduce features. The flattened output is passed through two dense layers with dropout to prevent overfitting, and a final softmax layer classifies nodes as legitimate or malicious. This design ensures effective feature extraction with efficient computation.

Table 1: Layer Structure of the Deep Learning-Based Feature Extraction Model

Layer Type	Details	Output Shape	Activation
Input Layer	12-dimensional feature vector	(12, 1)	-
Conv1D Layer	32 filters, kernel size = 3	(10, 32)	ReLU
MaxPooling1D Layer	Pool size = 2	(5, 32)	-
Flatten Layer	Flattening the pooled feature maps	(160,)	-
Dense Layer 1	64 neurons	(64,)	ReLU
Dropout Layer 1	Dropout rate = 0.2	(64,)	-
Dense Layer 2	32 neurons	(32,)	ReLU
Dropout Layer 2	Dropout rate = 0.2	(32,)	-
Dense Layer	2 neurons	(2,)	Softmax

3.6 Anomaly Detection

Malicious node detection is the process of identifying compromised or harmful nodes within a Wireless Sensor Network. These nodes are introduced by attackers or become faulty due to external manipulations, leading to data corruption, packet dropping, or network disruption. The proposed federated deep learning model classifies nodes into legitimate nodes (LN) and four types of malicious nodes ($MN-1, MN-2, MN-3, MN-4$) based on their behavior.

The WSN is set up with X sensor nodes ($S = \{s_1, s_2, \dots, s_x\}$), including Cluster Heads (CHs) for data aggregation and a Base Station (BS) for central decision-making. Nodes are grouped into clusters $C = \{c_1, c_2, \dots, c_k\}$, where k represents the number of clusters. Communication and routing within the network are facilitated using the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol. The selection of CH follows a probability function:

$$T(s_i) = \begin{cases} \dfrac{p}{\left(1 - p * \left(r \bmod \dfrac{1}{p}\right)\right)} & if s_i \in CH(t-1)\ p \\ & otherwise \end{cases}$$

where, p be the desired percentage of Cluster Heads, r is the current round number and $CH(t-1)$ is a set of cluster heads from the previous round.

Each CH aggregates data from its member nodes before forwarding it to the Base Station (BS). Sensor nodes collect environmental data and send it to CHs. The CHs process and aggregate this data. The Federated Deep Learning Model is applied to extract features from node communication behavior. Convolutional layers help to identify spatial patterns in data transmission, activation function introduces non-linearity, and pooling layers reduce dimensionality. The processed data is passed through fully connected layers and a Softmax classifier for final classification. The output layer assigns probability values to five classes (LN, MN-1, MN-2, MN-3, MN-4), and the node is classified based on the highest probability class.

If a node is classified as legitimate node, it continues normal operations. If a node is classified as malicious node, the base station marks it as malicious, the node is isolated or blocked from further communication, and security mechanisms such as re-keying, authentication, or blacklist updates are triggered. Malicious node detection model ensures secure data transmission by classifying and isolating harmful nodes using a federated deep learning approach. By leveraging the LEACH protocol for routing and deep learning for classification, the system efficiently detects and mitigates security threats in WSN. This research identifies malicious nodes in wireless sensor networks using federated

deep learning. Attack types such as backdoors, DoS, fuzzers, and shellcode are detected using UNSW-NB15 benchmark dataset, while smurf, neptune, backdos, and spy attacks are classified using KDDCUP99 dataset. The results of a comparative analysis indicate that integrating federated learning with LEACH protocols provides superior performance in malicious node detection compared to a traditional deep learning approach. The network's performance was assessed both before and after the deployment of this detection mechanism, specifically measuring throughput, energy consumption, and delay.

3.7 Secure Routing

Secure routing in WSN ensures that data is transmitted efficiently while preventing attacks and unauthorized access. The system employs the Low-Energy Adaptive Clustering Hierarchy protocol which optimizes energy consumption and enhances security by organizing sensor nodes into clusters. Each cluster has a cluster head responsible for aggregating data from its member nodes and securely transmitting it to the base station. The dynamic selection of CHs helps balance energy consumption and reduces the risk of targeted attacks. To prevent malicious activities such as blackhole and grayhole attacks**,** authentication protocols are used to verify node identities before allowing them to participate in data transmission. Additionally, secure routing algorithms dynamically adjust paths to avoid compromised nodes and ensure uninterrupted communication. By integrating anomaly detection mechanisms and encryption-based authentication, secure routing in WSNs enhances network stability, efficiency, and resilience against cyber threats.

3.8 Secure Data Transaction

To maintain secure data transactions in WSNs, end-to-end encryption ensures that transmitted data remains confidential and protected from interception or tampering. Cryptographic techniques such as homomorphic encryption allow computations on encrypted data without exposing its contents, enhancing privacy and security. Additionally, integrity verification mechanisms, including cryptographic hash functions, detect any modifications in data during transmission, ensuring reliability. The FedShield-PHE uses partial homomorphic encryption to secure data exchanges between sensor nodes, cluster heads, and the base station. This encryption method enables data aggregation while preserving privacy, reducing the risk of unauthorized access. Furthermore, digital signatures and secure key exchange protocols authenticate nodes, preventing malicious entities from injecting false data into the network. By combining encryption, integrity verification, and anomaly detection, secure data transactions in WSNs ensure trusted communication and prevent data manipulation.

3.9 Data Decryption

After the encrypted data is transmitted and aggregated, the final stage involves decrypting the ciphertext to recover the original preprocessed feature values. Decryption is performed only at the trusted authority, typically the Base Station (BS), which holds the private key of the Paillier cryptosystem. This ensures that confidentiality is maintained throughout the transmission process, as no intermediate node has access to the plaintext data. The Paillier decryption algorithm leverages the private key $sk = (\lambda, \mu)$ to transform the ciphertexts back into their corresponding plaintext values. For a given ciphertext $c \in Z_{n^2}^*$, the decryption is computed as:

$$f_i' = D_{sk}(c) = L\left(c^{\lambda} mod\ n^2\right).\mu\ mod\ n$$

where $L(u) = \frac{u-1}{n}$, f_i' represents the recovered normalized feature value. By applying this process to all ciphertexts c_{ij}. The complete feature vector $f_x' = [f_1', f_2', \ldots, f_{12}']$ is reconstructed for each observation. When aggregation has been performed on encrypted data, the decrypted result corresponds to the sum of the underlying plaintext features. This property enables the BS to securely perform anomaly detection and network monitoring without exposing sensitive intermediate values.

Algorithm: Decryption()

Input: Encrypted feature vector c_{ij}, Private key $sk = (\lambda, \mu)$.

Output: Decrypted feature vector f_x'.

1. For each ciphertext c_{ij}:
    a. Compute $u = c_{ij}^{\lambda} mod\ n^2$
    b. Compute $L(u) = \frac{u-1}{n}$.
    c. Recover plaintext: $f_i' = (L(u) \cdot \mu) mod n$.
2. Reconstruct feature vector $f_x' = [f_1', f_2', \ldots, f_{12}']$

4. EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Dataset Description

The UNSW-NB15 dataset, developed by Moustafa and colleagues in 2015, comprises 49 features extracted using Argus and Bro-IDS tools. The dataset includes two primary categories: normal traffic and attack traffic. The attack records are further divided into nine distinct families based on the nature of the attacks: analysis, backdoors, DoS, exploits, fuzzers, generic, reconnaissance, shellcode, and worms.

Similarly, the KDDCUP99 dataset, created by Stolfo and colleagues in 2000, classifies connections as either normal or belonging to a specific attack category. This dataset identifies four main types of attacks: DoS, R2L, U2R, and probing. Each connection record is approximately 100 bytes in size. The training data includes normal traffic and 24 different attack types, including back dos, buffer_overflow, ftp_write, guess_passwd, imap, ipsweep, land, loadmodule, multihop, neptune, nmap, perl, phf, pod, portsweep, rootkit, satan, smurf, spy, teardrop, warezclient, and warezmaster. These connection records are analyzed through three distinct feature sets: basic features, which capture fundamental details of individual TCP connections; content features, which utilize domain knowledge to analyze the content within a connection; and traffic features, which are calculated based on network traffic patterns observed over a two-second window.

In Wireless Sensor Networks, feature extraction is important in intrusion detection systems, especially when machine learning models are employed. Due to the limited computational capacity and energy constraints of sensor nodes, the extraction process must be lightweight yet informative. The objective is to transform raw network traffic into a compact set of meaningful features that highlight the differences between normal and anomalous behaviors. In our proposed framework, this extraction is done locally at the sensor node level, which minimizes communication overhead and upholds data privacy, as raw traffic data is never transmitted. Features derived include basic attributes like IP addresses, port numbers, and protocol types, as well as behavioral indicators such as packet size, connection duration, and transmission frequency. In addition, traffic pattern-based features are analyzed to identify anomalies that might only be visible through time-based or aggregated observations.

To ensure compatibility and thorough evaluation, we employed two widely used intrusion detection datasets: UNSW-NB15 and KDDCup99. The UNSW-NB15 dataset offers 49 well-defined features obtained from real-world traffic, representing various attack categories like DoS, shellcode, and reconnaissance. KDDCup99 categorizes features into basic, content-based, and traffic-based groups, providing a deeper insight into connection-level characteristics and attack types such as R2L, U2R, and probing. The selected features from both datasets align with the types extracted by our sensor nodes, allowing for realistic simulation and training of our models. Moreover, we apply dimensionality reduction techniques to retain only the most significant features, thereby reducing computational load without compromising detection accuracy. Overall, the feature extraction process plays a critical role in enabling privacy-preserving, real-time anomaly detection and supports the efficiency and scalability of the federated deep learning approach.

4.2 Results and Performance Analysis

This research adopted a multi-layered strategy for intrusion detection in WSNs. It employed PHE to ensure data privacy during both transmission and processing. Federated Deep Learning enabled collaborative model training across sensor nodes while safeguarding the confidentiality of raw data. By extracting critical features from sensor data and utilizing a deep learning model for anomaly detection, the proposed model integrated these advanced techniques to strengthen network security and enhance resilience against malicious threats. Our previous system, referred to as "Federated Learning with LEACH" was designed for anomaly detection in a network of sensor nodes. Sensor data was collected and preprocessed locally by the nodes. Features were extracted and used to train a deep learning model on each node, enabling real-time adaptation. To improve the model and preserve privacy, federated deep learning was employed. It allowed collaborative training of the model without sharing raw data. Encrypted data was aggregated at the base station to ensure confidentiality, while routing decisions were guided by trust scores stored on a blockchain. Network performance was monitored, and data was securely stored by the base station. Work of Khan, Z.A., et. al., 2023 proposed a model called Secure LEACH-DL Blockchain and it is referred as LEACH while comparing the results. It integrated four deep learning techniques, GRU, LSTM, CNN, and ANN with a real time message content validation scheme based on blockchain to detect and mitigate malicious nodes in WSNs. It used the WSN-DS 2016 dataset to train the DL models, while blockchain was deployed on cluster heads and the base station to overcome single point failures. Legitimate nodes were registered using a proof-of-authority (PoA) consensus protocol for efficient computation. Routing was performed using LEACH and HMGEAR protocols, and the results with DL and RMCV enhancements were evaluated against the original protocols. Additionally, the Oyente tool performed security analysis, confirming the resilience of the blockchain-based smart contract. The performance of these existing works is compared with the performance of the proposed work. The training and testing loss, along with training and testing accuracy for the UNSW-NB15 dataset, are calculated by repeating epochs until optimal values are achieved. The performance metrics for federated learning have been evaluated and are presented in Table 2.

Table 2: Comparison of Training and Testing loss, Training and Testing accuracy from the UNSW-NB15 dataset

Epochs	Training Loss	Testing Loss	Training accuracy	Testing accuracy
1	0.127	0.121	0.927	0.981
2	0.162	0.147	0.931	0.873
3	0.182	0.094	0.896	0.931
4	0.141	0.172	0.958	0.913
5	0.186	0.169	0.912	0.852
6	0.159	0.181	0.984	0.933
7	0.112	0.111	0.954	0.921
8	0.109	0.089	0.999	0.961
9	0.127	0.082	0.869	0.892
10	0.167	0.102	0.942	0.925
Overall	**0.1472**	**0.1268**	**0.9372**	**0.9182**

The average training and testing accuracies are 0.9372 and 0.9182, respectively. The average training and testing losses identified during the epochs are 0.1472 and 0.1268, respectively. Figure 3 illustrates the results obtained from the epochs, showcasing both training and testing accuracy. These results provide insights into the dataset's performance, helping to identify areas for improvement and optimization in the UNSW-NB15 dataset.

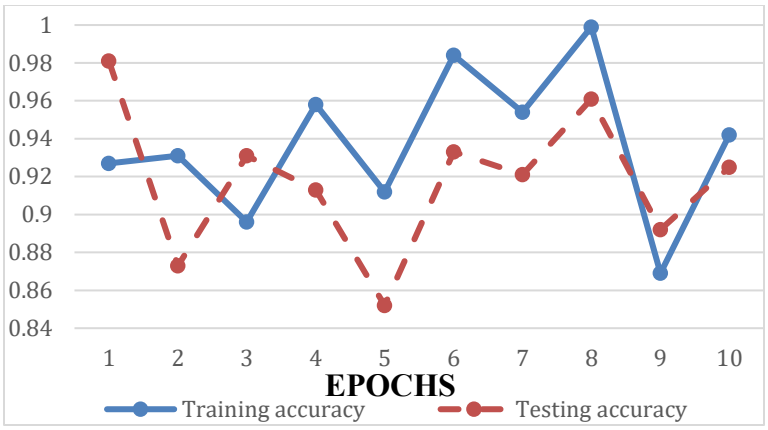

Figure 3: Visualization of the training and testing accuracies obtained over epochs from the UNSW-NB15 dataset

Figure 4 displays the training and testing losses recorded over multiple epochs, along with the average loss value calculated during these periods. Analyzing these results provides valuable insights into the UNSW-NB15 dataset, helping to pinpoint areas for improvement. Therefore, a careful review of these findings is essential for optimizing the performance of the UNSW-NB15 dataset.

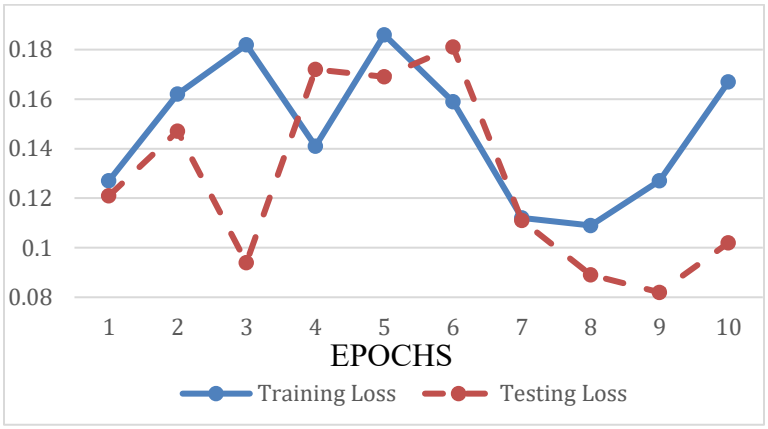

Figure 4: Visualization of the training and testing losses obtained over epochs from the UNSW-NB15 dataset

Figure 5 illustrates the Receiver Operating Characteristic (ROC) curves for a federated deep learning classifier applied to the UNSW-NB15 dataset, showcasing the model's performance across five classes (class 0 to class 4). The x-axis represents the FPR, while the y-axis represents the True Positive Rate (TPR). Class 0 achieves a TPR of 0.675 at an FPR of 0.01, increases to 0.856 at an FPR of 0.026, and eventually reaches 1 at an FPR of 1. Similarly, class 1 shows slightly higher TPR values at low FPRs, with a TPR of 0.756 at an FPR of 0.025 and 0.857 at an FPR of 0.037. Comparable trends are observed for classes 2, 3, and 4, indicating strong classifier performance across all categories. The overlapping nature of the curves suggests relatively consistent performance across the different classes.

Figure 5: Evaluation of a Federated Deep Learning Classifier's Performance Using Sensitivity and Specificity on the UNSW-NB15 Dataset

Figure 6 illustrates the ROC curves for a federated deep learning classifier applied to the KDDCUP99 dataset, showcasing the model's performance across five classes (class 0 to class 4). The x-axis represents the FPR, while the y-axis represents the True Positive Rate (TPR). Class 0 achieves a TPR of 0.678 at an FPR of 0.015, increases to 0.754 at an FPR of 0.025, and eventually reaches 1 at an FPR of 1. Similarly, class 1 shows a TPR of 0.675 at an FPR of 0.022, increases to 0.857 at an FPR of 0.04, and reaches 0.975 at an FPR of 0.088. Comparable trends are observed for classes 2, 3, and 4, indicating strong classifier performance across all categories. The overlapping nature of the curves and the gradual increase in TPR with FPR across all classes suggest consistent and robust performance of the classifier on the KDDCUP99 dataset.

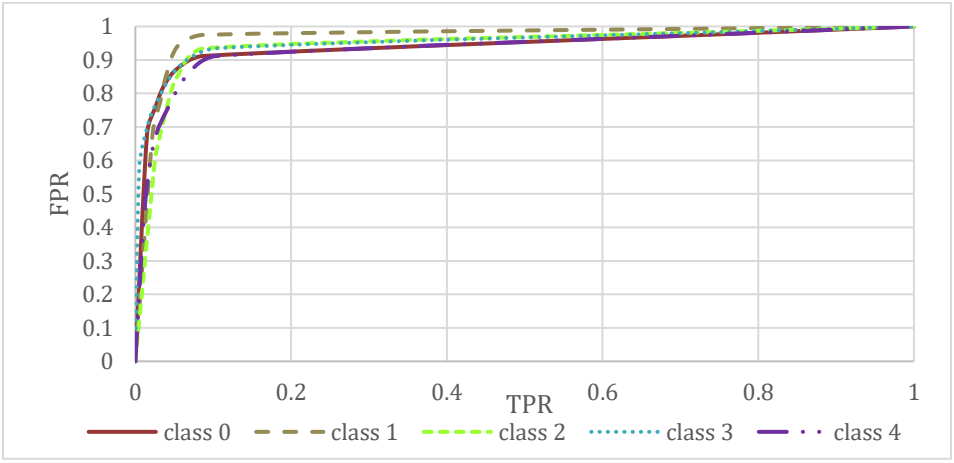

Figure 6: Evaluation of Federated Deep Learning Classifier Performance Using Sensitivity and Specificity on the KDDCUP99 Dataset

The training and testing loss, along with training and testing accuracy, for the KDDCUP99 dataset are computed by iterating epochs until optimal values are achieved. The performance of federated learning is evaluated and presented in Table 3 for further comparative analysis.

Table 3: Comparison of Training and Testing Loss and Accuracy from the KDDCUP99 Dataset

Epochs	Training Loss	Testing Loss	Training accuracy	Testing accuracy
1	0.125	0.113	0.912	0.901

2	0.148	0.143	0.935	0.924
3	0.119	0.109	0.926	0.953
4	0.137	0.127	0.943	0.942
5	0.154	0.151	0.954	0.921
6	0.163	0.107	0.961	0.918
7	0.171	0.122	0.929	0.935
8	0.182	0.101	0.987	0.966
9	0.196	0.132	0.948	0.911
10	0.141	0.102	0.938	0.937
Overall	**0.1536**	**0.1207**	**0.9433**	**0.9308**

The overall average training and testing accuracies are 0.9433 and 0.9308, respectively. The overall average training and testing losses over the epochs are 0.1536 and 0.1207, respectively. Figure 7 illustrates these results, including the training and testing accuracy trends, enabling informed decisions about the dataset's performance. Analyzing these findings can help identify areas for improvement and optimize the performance of the KDDCUP99 dataset.

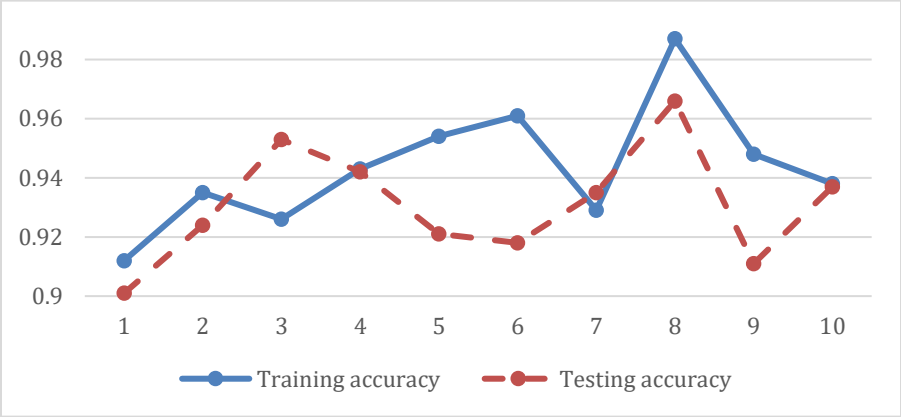

Figure 7: Visualization of the training and testing accuracies obtained over epochs from the KDDCUP99 dataset

Figure 8 shows the training and testing loss over multiple epochs, along with the average loss value calculated for these epochs. Analyzing these results provides valuable insights into the UNSW-NB15 dataset and helps identify areas for improvement. Careful review of these findings is crucial to optimizing the performance of the KDDCUP99 dataset. Additionally, Figure 9 illustrates the network delay caused by the presence of malicious nodes.

Figure 8: Visualization of the training and testing losses obtained over epochs from the KDDCUP99 dataset

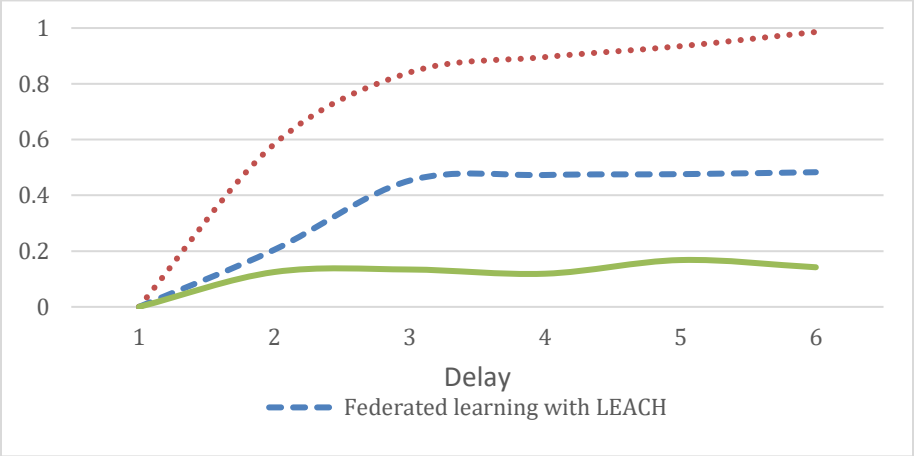

Figure 9: Delay with LEACH, Federated Learning with LEACH and Proposed method (FedShield-PHE)

When using LEACH for routing, malicious nodes are not detected, leading to longer data transmission times. Federated Learning with LEACH, however, detects and removes malicious nodes, allowing routing to be performed only with legitimate nodes, but the proposed method even performs better, thus minimizing delay. Figure 9 shows that Proposed method is more accurate in detecting malicious nodes compared to LEACH and Federated Learning with LEACH. Figure 10 illustrates the total network throughput in terms of rounds, comparing LEACH, Federated Learning with LEACH and the proposed method.

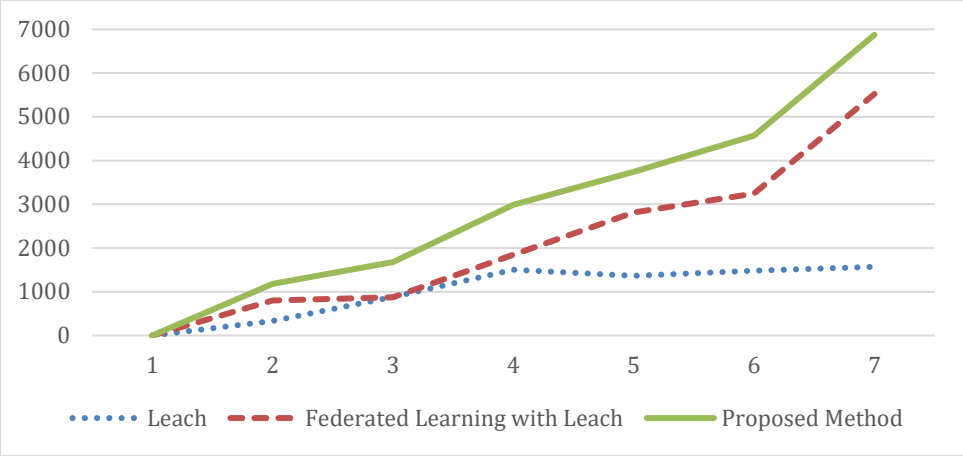

Figure 10: Throughput with LEACH, Federated Learning with LEACH, and the Proposed Method (FedShield-PHE)

In the LEACH protocol, data packets typically achieve peak throughput in the absence of malicious nodes. Integrating federated learning with LEACH enhances this by identifying and removing malicious nodes, thereby restricting routing to legitimate nodes and potentially reducing overall throughput. The FedShield-PHE improves upon this approach by optimizing detection and elimination processes, ensuring higher throughput while effectively handling malicious nodes. This blend of security and performance enhances the robustness and efficiency of the proposed system.

5. CONCLUSION

This paper presented an adaptive intrusion detection framework for WSNs by combining federated learning with PHE (FedShield-PHE). PHE excels at maintaining data privacy and integrity while enabling efficient computation on encrypted data. This method ensures robust security and privacy across distributed networks, effectively addressing dynamic threats and fostering facilitating enhancements in network security practices. The framework employs Conv1D-based feature extraction with lightweight dense layers to achieve accurate and efficient learning on resource-constrained nodes, while cluster heads filter malicious updates and the base station refines the global model for continuous adaptation against evolving threats. Experimental results on UNSW-NB15 and KDDCup99 datasets show that the proposed framework achieves high detection accuracy (over 97%), with significant reductions in false alarm rate, communication delay, and energy consumption. The model also demonstrates improved throughput, ensuring that the network remains efficient while maintaining robust security. Compared to existing intrusion detection methods, the framework consistently delivers higher reliability and faster detection. Moreover, the results confirm its adaptability to evolving attack patterns and its scalability for deployment in larger WSNs. These outcomes validate the practicality of deploying FDL with PHE in real-world scenarios, providing both strong security guarantees and efficient system performance. Future work will focus on enhancing the framework with advanced architectures such as RNNs or GNNs and refining lightweight encryption to further improve scalability and real-time performance.

REFERENCES

1. Godala, S. and Vaddella, R.P.V., 2020. A study on intrusion detection system in wireless sensor networks. International Journal of Communication Networks and Information Security, 12(1), pp.127-141.
2. Gavel, S., Raghuvanshi, A.S. and Tiwari, S., 2021. A novel density estimation based intrusion detection technique with Pearson's divergence for wireless sensor networks. ISA transactions, 111, pp.180-191.
3. Lu, X., Han, D., Duan, L. and Tian, Q., 2020. Intrusion detection of wireless sensor networks based on IPSO algorithm and BP neural network. International Journal of Computational Science and Engineering, 22(2-3), pp.221-232.
4. Abhale, A.B. and Avulapalli, J.R., 2023. Enhancing intrusion detection recursive feature elimination with resampling in WSN. International Journal of System Assurance Engineering and Management, 14(6), pp.2642-2660.

5. Agarkar, A.A., Karyakarte, M. and Agrawal, H., 2020, May. Post quantum security solution for data aggregation in wireless sensor networks. In 2020 IEEE wireless communications and networking conference (WCNC) (pp. 1-8). IEEE.
6. Alzubi, O.A., 2022. A deep learning-based frechet and dirichlet model for intrusion detection in IWSN. Journal of Intelligent & Fuzzy Systems, 42(2), pp.873-883.
7. Rajasoundaran, S., Kumar, S.S., Selvi, M., Thangaramya, K. and Arputharaj, K., 2024. Secure and optimized intrusion detection scheme using LSTM-MAC principles for underwater wireless sensor networks. Wireless Networks, 30(1), pp.209-231.
8. Alkhdour, T., Ali, A., Almaiah, M., Tin, T.T., Al-Shareeda, M.A., Alali, R., Aldahyani, T. And Lutfi, A., 2024. Transforming Healthcare With Federated Learning: Securing Fog Systems For The Next Generation. Journal of Theoretical and Applied Information Technology, 102(9).
9. Goswami, C., SenthilRagavan, V.K., Ramesh, J.V.N., Balajee, J., Ronald Doni, A., Saravanan, T.R. and Siva Shankar, S., 2023. Deep convolutional neural network-based Henry gas solubility optimization for disease prediction in data from wireless sensor network. Soft Computing, pp.1-12.
10. Giachanatzis Grammatikopoulos, A., 2021. Intrusion detection systems with the use of federated learning.
11. Babenko, L. and Tolomanenko, E., 2020, October. Threats to wireless sensor networks and approaches to use homomorphic encryption to secure its data. In Journal of Physics: Conference Series (Vol. 1624, No. 3, p. 032055). IOP Publishing.
12. Zeng, R., Jiang, Y., Lin, C., Fan, Y. and Shen, X., 2011. A distributed fault/intrusion-tolerant sensor data storage scheme based on network coding and homomorphic fingerprinting. IEEE Transactions on Parallel and Distributed Systems, 23(10), pp.1819-1830.
13. Bhushan, B. and Sahoo, G., 2020. Secure location-based aggregator node selection scheme in wireless sensor networks. In Proceedings of ICETIT 2019: Emerging Trends in Information Technology (pp. 21-35). Springer International Publishing.
14. Khedr, A.M., Raj, P.P. and Rani, S.S., 2024. Time Synchronized Multivariate Regressive Convolution Deep Neural Network Model for Sinkhole Attack Detection in WSN. Wireless Personal Communications, 134(1), pp.361-382.
15. Sadia, H., Farhan, S., Haq, Y.U., Sana, R., Mahmood, T., Bahaj, S.A.O. and Rehman, A., 2024. Intrusion Detection System for Wireless Sensor Networks: A Machine Learning based Approach. IEEE Access.
16. Ifzarne, S., Hafidi, I. and Idrissi, N., 2021. A Novel Secure Data Aggregation Scheme Based on Semi-Homomorphic Encryption in WSNs. J. Commun., 16(8), pp.323-330.
17. Ifzarne, S., Hafidi, I. and Idrissi, N., 2021. Secure data collection for wireless sensor network. In Emerging Trends in ICT for Sustainable Development: The Proceedings of NICE2020 International Conference (pp. 241-248). Springer International Publishing.
18. Kantzavelou, I., Tzikopoulos, P.F. and Katsikas, S.K., 2023, May. Detecting intrusive activities from insiders in a wireless sensor network using game theory. In Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments (pp. 1-8).
19. Ali, M., Nalgozhina, N., Tasmagambetov, O. and Seitkulov, Y.N., 2022. Homomorphic Encryption with Enhanced Elliptic Curve for Mobile Wireless Network Security. In *DTESI*.
20. Aljohani, A.A., Tripathi, R.K., Bhardwaj, R., Kaushal, R.K., Kumar, N., Gupta, S.K. and Hamato, G.G., 2023. Secured Healthcare Data Sharing Through Wireless Networks for Mobile Computing Using Trust-Bat-Adaptive Homomorphic Crypto Routing Protocol.
21. Zhang, Z., Yang, Y., Yang, W., Wu, F., Li, P. and Xiong, X., 2021. Detection and location of malicious nodes based on homomorphic fingerprinting in wireless sensor networks. *Security and Communication Networks*, *2021*(1), p.9082570.
22. Qi, X., Liu, X., Yu, J. and Zhang, Q., 2020. A privacy data aggregation scheme for wireless sensor networks. Procedia Computer Science, 174, pp.578-583.
23. Halidoddi, G. and Pandu, R., 2022. Secured Data Transmission Using Multi-Objective Trust Based Bat Optimization Algorithm and Enhanced Homomorphic Cryptosystem for WSN. International Journal of Intelligent Engineering & Systems, 15(1).
24. Peng, C., Luo, M., Vijayakumar, P., He, D., Said, O. and Tolba, A., 2021. Multifunctional and multidimensional secure data aggregation scheme in wsns. IEEE Internet of Things Journal, 9(4), pp.2657-2668.
25. Saravanaselvan, A. and Paramasivan, B., 2024. FFBP Neural Network Optimized with Woodpecker Mating Algorithm for Dynamic Cluster-based Secure Routing in WSN. IETE Journal of Research, pp.1-10.
26. Li, L., Li, S., Peng, H. and Bi, J., 2022. An efficient secure data transmission and node authentication scheme for wireless sensing networks. Journal of Systems Architecture, 133, p.102760.
27. Khan, Z.A., Amjad, S., Ahmed, F., Almasoud, A.M., Imran, M. and Javaid, N., 2023. A blockchain-based deep-learning-driven architecture for quality routing in wireless sensor networks. *IEEE Access*, *11*, pp.31036-31051.