Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Secure WSN-IoT using end-to-end elliptic curve and homomorphic encryption

Mohammad Ibrahim Adawy *

Department of Data Systems and Networks, Faculty of Information Technology, World Islamic Sciences and Education University, Amman, Jordan

ARTICLE INFO

Article history:
Received 8 June 2025
Received in revised form
1 October 2025
Accepted 4 December 2025

Keywords:
Internet of Things
Wireless sensor networks
Data aggregation
Homomorphic encryption
Message authentication

ABSTRACT

The security of the Internet of Things (IoT) has become a major research concern, particularly in the perception layer where wireless sensor networks (WSNs) operate and generate large amounts of data. Without encryption, transmitted data in WSN-IoT networks is vulnerable to security attacks. Conventional methods that decrypt, aggregate, and re-encrypt data consume excessive energy at the aggregator and increase end-to-end delay. To address these issues, this study proposes a secure data aggregation scheme based on end-to-end elliptic curve and homomorphic encryption (EEECHE). The scheme ensures data confidentiality from nodes to the server while minimizing energy consumption and latency. It also applies a message authentication code (MAC) to verify data authenticity and detect false data efficiently. Experimental results show that the proposed scheme achieves stronger security with lower energy usage and end-to-end delay compared to cluster-based semi-homomorphic encryption aggregated data (CSHEAD) and cluster-based secure data aggregation (CSDA).

1. Introduction

The Internet of Things (IoT) exists as a group of wireless devices on the Internet in the world, such as TVs, smartphones, sensor nodes, and actuators, that seamlessly connect (Zijie et al., 2023). A wireless sensor network (WSN) is defined as a group of dispersed sensor nodes that observe and collect information about the physical conditions of the surroundings, such as temperature, pressure, movement, humidity, etc. (Kumar et al., 2023; Adawy et al., 2023). Therefore, sensor nodes can quickly consume their energy in collecting and transmitting data (Mocanu et al., 2015; Randhawa and Jain, 2017). As illustrated in Fig. 1, the sensor nodes consider IoT devices that play a significant role in gathering environmental data and transmitting redundant data to gateway devices, resulting in a huge volume of data and significant energy consumption (Qu and Li, 2022). A gateway is an aggregator in a wireless sensor network-IoT (WSN-IoT) that receives data from the sensor nodes, aggregates it to remove redundant data (Adawy et al., 2023; Siddiqui et al., 2015), and then sends the results to a server. Data aggregation is a significant method for conserving the energy of sensor nodes (Al-Baz and El-Sayed, 2018). The server performs data querying, storage, and extraction operations (Gulati et al., 2022). Cloud computing is an important component of the IoT, as it is used to store and examine large amounts of data (Huang et al., 2018).

There are many applications in WSN-IoT, such as road condition monitoring in megacities, multi-robot coordination, automated sensing, robot planning and robot navigation, building structure monitoring, electromagnetic field monitoring, and forest fire detection (Ifzarne et al., 2021). However, sensor nodes are vulnerable to security breaches (Lavanya et al., 2022) by attackers, resulting in incorrect data transmission in WSN-IoT. Furthermore, data aggregation poses new security threats. For example, a compromised sensor node may send misleading and misleading data to the aggregator, negatively impacting the aggregated data. Consequently, an attacker may violate the privacy and integrity of data in WSN-IoT by attacking many aggregators close to the server (Kumar et al., 2023).

Furthermore, the study in Qu and Li (2022) allowed sensor nodes to convert their data into curve points using the Elliptic Curve Cryptography (ECC) algorithm. The study in Ifzarne et al. (2021) focused on Homomorphic Encryption (HE), which

* Corresponding Author.
Email Address: mohammad.adawi@wise.edu.jo
https://doi.org/10.21833/ijaas.2025.12.023
Corresponding author's ORCID profile:
https://orcid.org/0009-0006-7983-3110

allows the Cluster Head (CH) in the network to apply a specific aggregation function, such as this addition, to encrypted data without the need for decryption during data transmission from the sensor nodes to the Base Station (BS). The proposed scheme focuses on combining ECC with HE methods. The proposed scheme allows the sensor node to use the ECC method to convert sensor data into curve points and then encrypt them using the private and public keys. It also allows the aggregator to use the HE method to aggregate the received encrypted data without decryption, with minimal energy depletion and reduced end-to-end latency. Because decrypting, aggregating, and then encrypting all the encrypted data consumes most of the aggregator's energy and upsurges the end-to-end latency (Kumar et al., 2023). Furthermore, in the proposed scheme, the private and public keys are important in data encryption, as each sensor node receives unique private and public keys from the server in each round, which are changed in the next round. The private key is the distance between the sensor node and its aggregator, while the public key is the multiplication of prime numbers by the generation point. The proposed scheme also uses Message Authentication Code (MAC) to verify the accuracy of the aggregated data, allowing forensic detection as quickly as possible.

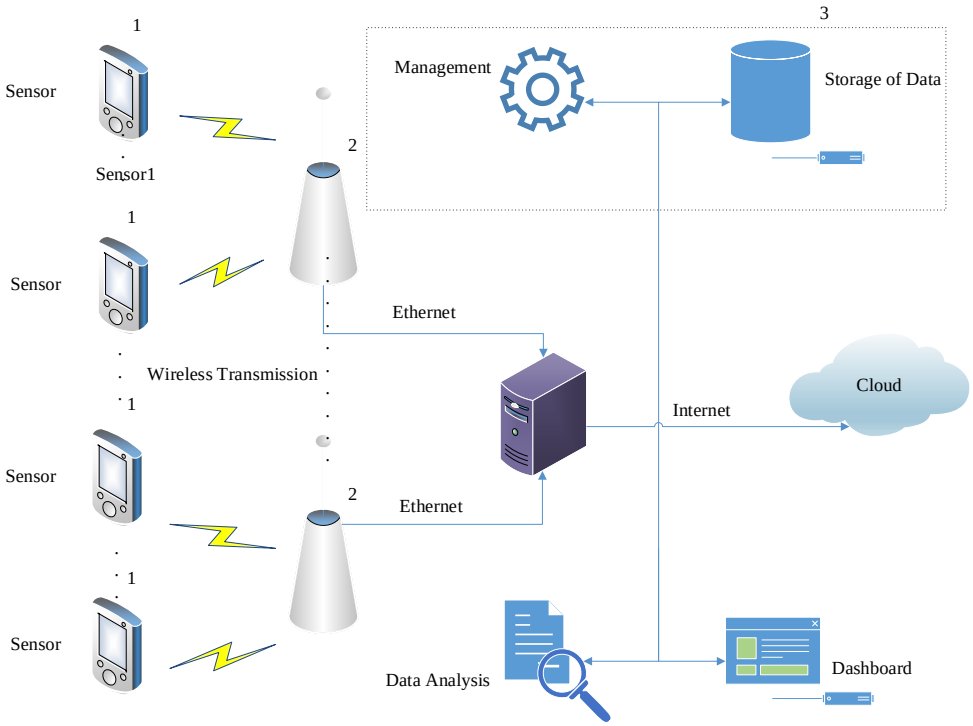


Fig. 1: WSN-IoT architecture (Zijie et al., 2023)

However, our contribution to this work was a motivated attempt to advance the discussion on secure data aggregation in WSN-IoT networks:

- Simulate a WSN-IoT network using the OMNeT++ tool.
- Generate encrypted data traffic using elliptic curve cryptography.
- Implement data aggregation to encrypted data without decrypting it using Homomorphic cryptography.
- Ensure the confidentiality of aggregated data.
- Conserve sensor nodes' energy and reduce end-to-end latency
- Ensure the integrity of the aggregated data.

This paper is structured into several sections as follows: related works are presented in Section 2, background is introduced in Section 3, the WSN-IoT model is described in Section 4, the methodology of the proposed scheme is presented in Section 5, the simulation result and performance metrics are explained in Section 6, and then the paper is completed in Section 7.

2. Background

2.1. Elliptic curve cryptography

The Elliptic Curve cryptosystem was offered in 1985 (Gentry and Halevi, 2011). ECC is an asymmetric cryptography method depending on the formulas of elliptic curves over the determined fields. Elliptic curve executions for microprocessors can be established in Szczechowiak and Collier (2009). They used a mixture of x, y positions and a non-contiguous shape, which reduced the number of point additions required. The elliptic curve cryptosystem describes two forms of determined fields. The first form is the initial field Fp, where p is a prime number; the second form is binary fields F_{2^m}. The public elliptic curve equation is:

$$E: y^2 + ax + b \bmod p \quad (1)$$

where, a, b, x, y ϵ $G_F(p)$ and $4a^3+27b^2 \neq 0 \bmod p$. Any reading such as mi can be converted to the point (x, y) on the elliptic curve by using the function Convert (m_i) where mi ϵ $G_F(q)$.

$$Convert(m_i) = (x_i, y_i)\ \epsilon\ E \quad (2)$$
$$M_i = Convert(m_i) \quad (3)$$

Thus, each value is converted to the curve point M, where M is a multiple of m of the generator point G, presenting these two forms of ECC cryptosystem, which depend on the public key in the other rest sections.

The initial method: Firstly, electing the private key as an arbitrary number k and computing the public key Pk:

$$Pk = k * G \quad (4)$$

The value m_i is converted to the curve point M_i. After selection of the random number S, we can obtain the encryption value:

$$C_1 = S * G \quad (5)$$
$$C_2 = S * Pk + M_i \quad (6)$$
$$Enc(m_i) = [\,C_1, C_2]. \quad (7)$$

To decrypt Enc(m_i), we use the private key Sk:

$$Sk = k * C_1. \quad (8)$$

Secondly, point M_i on Elliptic Curve E is calculated:

$$M_i = C_2 - C_1. \quad (9)$$

The second method:

$$C_1 = S * G \quad (10)$$
$$C_2 = Sk * Pk \quad (11)$$
$$C_3 = m_i * C_2. \quad (12)$$

To decrypt Enc(m_i), we use the private key Sk:

$$M_i = C_2^{-1} * C_3 = Sk * pk^{-1} * m_i * Sk * pk. \quad (13)$$

Encryption of aggregated data:

$$E(m_1 + m_2 + m_3 \ldots + m_n) = E(m_1) \oplus E(m_2) \oplus E(m_3) \ldots \oplus E(m_n) \quad (14)$$

Decryption of aggregated data:

$$m_1 + m_2 + m_3 \ldots + m_n = Dec(E(m_1) \oplus E(m_2) \oplus E(m_3) \ldots \oplus E(m_n)) \quad (15)$$

where, + is the addition operator applied to a value from a general encryption method. This operation is called an additive homomorphic cryptosystem. However, additive encryption is ineffective and too expensive for wireless sensor networks (WSNs) because it relies on binary linear coupling (Zhou et al., 2014).

3. Literature review

Many studies have explored the benefits and security measures associated with data aggregation. These studies concentrated on how to present data aggregation security approaches using cryptosystems in WSN. Kumar et al. (2023) proposed an End-to-End Homomorphic Encryption (EEHE)-based secure data aggregation protocol for IoT-based WSNs that secures end-to-end security, allows the implementation of data aggregation operations, and detects wormhole attacks through the data aggregation method. The proposed protocol has not been fully demonstrated experimentally, as the authors state they will show the lower energy requirements and fewer false positives in the next section. Also, the proposed protocol can fail if there is a malicious node with a fake ID that is able to mislead the routing protocol.

Zhou et al. (2014) provided that SEDA-ECC is a secure data aggregation scheme that splits the aggregation tree into three disjoint subtrees to ensure data integrity while providing privacy homomorphic encryption to protect data privacy. However, the paper points to the need for security analysis against node-level attacks and suggests that the energy consumption of SEDA-ECC could potentially be improved with more advanced sensor devices. The proposed scheme can only handle integer-based calculations and has issues with divisibility or negative differences. Also, the current full homomorphic encryption schemes have limitations in terms of large public keys, large Ciphertext expansion, and computationally intensive calculations.

Ugus et al. (2009) presented an enhanced execution of the Elliptic Curve ElGamal (EC-ElGamal) cryptography for additive homomorphic encryption on a MicaZ mote to enable secure and efficient data aggregation in the TinyPEDS framework for WSNs. The number of precomputed points used can be further optimized to improve performance. Also, the Interleave method may be more effective than the precomputation method used in prior work, suggesting further research into optimizing the point multiplication algorithm.

Fang et al. (2019) offered a new energy-efficient data aggregation security approach named Cluster-based Secure Data Aggregation (CSDA) that uses data slicing and cluster-based privacy preservation to reduce communication overhead and energy depletion while preserving data aggregation precision and privacy protection. The algorithm has two limitations: firstly, evaluating CSDA against other privacy-preserving approaches in real-world WSN deployments. Secondly, providing data integrity protection in addition to privacy preservation. Further, Hong et al. (2016) proposed an Elliptic Curve Cryptography (ECC) based homomorphic encryption approach for secure combined calculation in cloud computing to reduce computation and communication costs compared to traditional encryption algorithms.

Ifzarne et al. (2021) offered a new data aggregation security approach for WSNs based on semi-homomorphic cryptography. The approach shows that this CSHEAD scheme excels an existing CSDA approach in terms of reduced communication overhead, improved network performance metrics, and more accurate attack detection. Information integrity was not addressed and is left as a future research direction. Also, the proposed approach does not take into consideration injection of data attacks, where an adversary may insert new or duplicate data in the aggregation process.

Li et al. (2015) proposed a secure data aggregation scheme named FESA that uses fully homomorphic encryption (FHE) to accomplish end-to-end data privacy and integrity in large-scale WSN and detect fake data injection attacks as early as possible. The proposed approach produces an aggregated data security method, but has more energy consumption when measuring total energy depletion of the sensor nodes and an increase in End-to-End latency.

Elhoseny et al. (2016) proposed a security framework based on Elliptic Curve Cryptography (ECC) and homomorphic encryption to protect data transmission in wireless sensor networks (WSNs). Their method uses a diversity-based clustering approach and applies a genetic algorithm to improve network lifetime. Due to the heterogeneous nature of WSNs and the limited resources of sensor nodes, efficient use of resources remains a major challenge. In particular, designing a secure and optimal network architecture is difficult. Therefore, a trade-off must be maintained between energy consumption during data transmission and the level of security.

Kumar et al. (2015) proposed a secure and efficient data aggregation mechanism for WSNs using mobile agents and homomorphic encryption to reduce energy consumption and increase network lifetime. The proposed approach only supports single mathematics or logic aggregation processes. Tree-based secure aggregation lacks redundancy, making it vulnerable to message loss due to node failure. Bajpai and Yadav (2024) proposed a novel secure data aggregation approach for battlefield surveillance using WSNs, which involves four phases: cluster head selection, key establishment and data encryption using elliptic curve cryptography, data aggregation and verification using HMAC, and decryption at the base station. Implement the proposed scheme to increase the total energy consumption of the sensor nodes. However, Table 1 covers a review analysis of the related work for data aggregation security approaches implemented in WSN-IoT.

Table 1: Data aggregation security approaches in WSN-IoT

Reference	Security scheme	Objectives	Limitations
Kumar et al. (2023)	Proposed an End-to-End Homomorphic Encryption (EEHE)-based secure data aggregation protocol for IoT-based WSNs.	The scheme encrypts data using HE method and allows the implementation of data aggregation, and detects wormhole attacks through the data aggregation method.	The proposed protocol has not been fully demonstrated experimentally, as the authors state they will show the lesser energy requirements and fewer false positives in the next section. The proposed approach can fail if there is a malignant node with a false ID that is able to mislead the routing protocol.
Zhou et al. (2014)	Provides SEDA-ECC is a data aggregation security scheme.	Splits the aggregation tree into 3 disjoint subtrees to ensure data integrity while providing privacy homomorphic encryption to protect data privacy.	The proposed scheme can only handle integer-based calculations and has issues with divisibility or negative differences. The schemes have limitations in terms of large public keys, large Ciphertext expansion, and computationally intensive calculations.
Fang et al. (2019)	Offerings Cluster-based Secure Data Aggregation (CSDA) scheme.	Uses data slicing and cluster-based privacy preservation to reduce communication overhead and energy depletion.	Evaluating CSDA against other privacy-preserving approaches in real-world WSN deployments. Providing data integrity protection in addition to privacy preservation.
Ifzarne et al. (2021)	Offering a new data aggregation security approach for WSNs based on semi-homomorphic cryptography.	The approach is used to encrypt and aggregate data using semi-homomorphic cryptography to secure data aggregation.	The scheme does not take into consideration injection of data attacks, where an attacker may insert new or duplicate data in the aggregation method.
Li et al. (2015)	Proposes a secure data aggregation scheme named FESA.	The approach uses fully HE to achieve end-to-end data confidentiality and integrity in large-scale WSN, and to detect fake data injection attacks as early as possible	The approach has more energy consumption when measuring total energy depletion of the sensor nodes and an increase in End-to-End latency.
Elhoseny et al. (2016)	Offering an Elliptic Curve Cryptography (ECC) and homomorphic cryptography to secure data transmission in WSNs with diver clustering.	The approach encrypts data transmission using ECC and improves the network lifetime using a genetic algorithm-based clustering method.	The diver nature of WSNs and the limitation of sensor nodes' resources make searching for resources limited for all nodes. Searching for a secure, optimal network architecture is an ongoing challenge. There is a balance between energy consumption in the data transmission nodes and security that needs to be balanced.
Kumar et al. (2015)	Proposes an efficient data aggregation security approach for WSNs using mobile agents and homomorphic encryption	The proposed approach only supports single mathematics or logic aggregation processes to diminish energy depletion and increase the lifetime of the network.	Tree-based secure aggregation lacks redundancy, making it vulnerable to message loss due to node failure.
Bajpai and Yadav (2024)	Proposes a novel secure data aggregation approach for battlefield surveillance using WSNs.	The scheme contains four phases and uses ECC to encrypt data and HMAC to validate data to secure data aggregation.	The proposed scheme increases the total energy consumption of the sensor nodes.

4. Proposed design scheme

In this section, the proposed scheme is offered as a novel scheme for securing data aggregation based on elliptic curve and homomorphic encryption methods. The proposed scheme is compared to a recent Cluster-based Semi-Homomorphic Encryption Aggregated Data (CSHEAD). The primary goal of

CSHEAD is to protect the aggregated data and detect active attacks. Also, it takes into consideration the overhead communication as the main factor for practical use:

4.1. Network model

Fig. 2 reproduces the proposed schemes in our system model from Pu et al. (2019) (three entities are available, including server, aggregator, and sensor nodes). We primarily concentrate on how to transfer sensory data from sensor nodes to the aggregator effectively and confidential manner. Fig. 2 illustrates the topology of a two-level gateway network in IoT. We assume that the server is directly connected to an aggregator device connected to N sensor nodes.

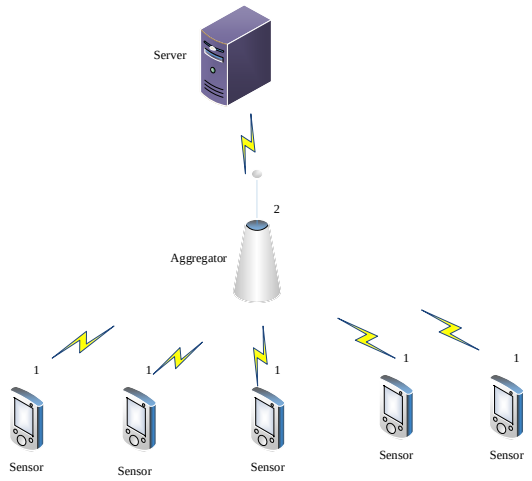


Fig. 2: WSN in IoT (Pu et al., 2019)

4.1.1. Server

A server is a trusted and efficient device capable of providing storage space for IoT devices to store aggregated data, which can be accessed by authorized users. Furthermore, the server also processes and assesses the aggregated data to manage implemented applications in WSN-IoT and guarantees their smooth operation, as in Pu et al. (2019).

4.1.2. Aggregator

The aggregator focuses on data collection and transmission. The primary task of the aggregator is to perform aggregation on the data received from the transmitting nodes and then send the aggregation result to the server. In addition, the aggregator has a MAC address, an IP address, and most wireless network protocols.

4.1.3. Sensor node

Sensor nodes can provide accurate results with some computation and storage. The sensor nodes are randomly located in a certain area. In addition, the sensor nodes collect data from the environment, which is duplicated data. Each sensor node transmits its data to an aggregator, which aggregates it and sends the result to a server. Each sensor node has the same transmission range as that of the aggregator.

4.1.4. Aggregator model

In this paper, we examine how sensor nodes sense temperature readings, encrypt them, and transmit them to an aggregator. The aggregator collects the received encrypted data without decrypting it and sends the aggregated results to a server. The server decrypts the aggregated data and verifies the authenticity of the sensor nodes.

4.1.5. Attack model

This paper also addresses the ability of an attacker to capture data and perform injection attacks. An attacker might insert fake data when capturing a data packet, enabling them to send forged data to the aggregation node. An attacker might also send duplicate encrypted data to the aggregation node.

4.2. Proposed scheme

The proposed scheme for End-to-End Elliptic Curve Cryptography with Homomorphic Encryption (EEECHE) in WSN-IoT network consists of a set of stages, as shown in Fig. 3, where each phase has its actions dependent on the previous stage.

4.2.1. WSN-IoT building

The OMNeT++ network tool is used to build a WSN-IoT network. The WSN-IoT network consists of 60 randomly distributed sensor nodes (yellow), three aggregators (blue), and one server. Fig. 4 shows that sensor nodes gather data about environmental parameters such as temperature, light intensity, and humidity. They then encrypt and send it to their connected aggregator device.

In the WSN-IoT network, the sensor node starts to communicate with its aggregator device. It then sends its location, node ID, aggregator ID, and aggregator location to the server.

4.2.2. Key distribution

The server generates a set of public and private keys based on the elliptic curve method. It then sends one unique public key to each aggregator device in the network, which then transmits it to every node connected to it. The public and private keys are changed in each round, and so on. The public key consists of 140 bits, while the private key consists of 16 bits. Key distribution will use the aggregator to route the public key from the server to the nodes:

- Generation point G^J on the Elliptic curve has 128 bits
- Two prime numbers q_1 and q_2 have 16 bits
- Distance between node and aggregator dist (S_i, Aggregator^J) has 16 bits.

Each aggregator J receives one unique public key Pk^J from the server:

$$pk^J = q_1 * q_2 * G^J = q_1 * q_2 * (X_0, Y_0). \quad (16)$$

The distance between node S_i and aggregator J is given by:

$$dist(S_i, Aggregator^J) = \sqrt{\left(X_{S_i} - X_{Aggregator^J}\right)^2 + \left(Y_{S_i} - Y_{Aggregator^J}\right)^2} \quad (17)$$

where, 0≤ dist (S_i, Aggregator^J) ≤ Tr. Tr is transmission range. The private key Sk^i for each sensor node is:

$$Sk^i = dist(S_i, Aggregator^J) \quad (18)$$

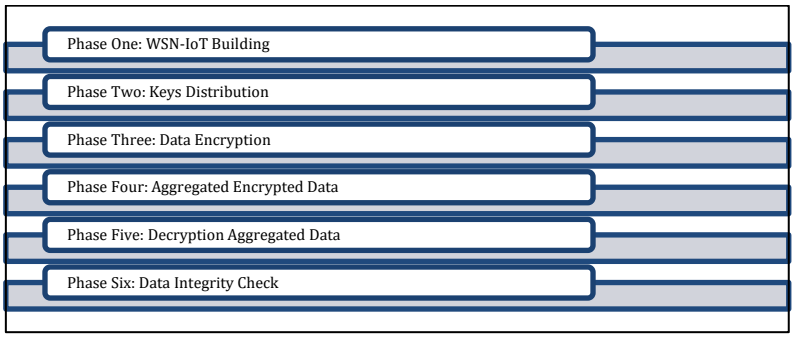


Fig. 3: Proposed methodology

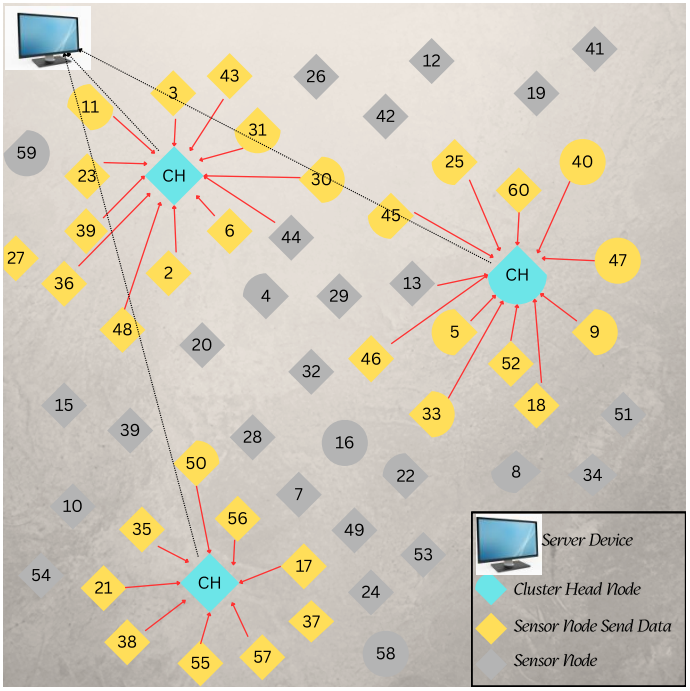


Fig. 4: WSN-IoT building using OMNeT++ simulator

4.2.3. Encryption data using elliptic cryptography

We define the following readings:

- m_i^J is the sensor temperature reading for sensor node Si that connects with aggregator node J.
- M_i^J is the elliptic curve point that represents m_i^J. where i refers to the sensor node number, and J refers to the aggregator number.
- E is the elliptic curve method as in Eq. 1:
- We convert each m_i^J to M_i^J:

1. We convert each m_i^J to M_i^J:

$$Convert(m_i^J) = M_i^J \quad (19)$$

2. We calculate the first part of the encrypted data:

$$C_{i1}^J = Sk^i * G^J \quad (20)$$

3. We calculate the second part of the encrypted data:

$$C_{i2}^J = M_i^J + Sk^i * Pk^J \quad (21)$$

4. We find the encrypted data:

$$Enc(M_i^J) = [C_{i1}^J, C_{i2}^J] \quad (22)$$

4.2.4. Aggregation of encrypted data using homomorphic encryption

$Aggr\_Data^J$ refers to aggregated data generated by the J aggregator. The aggregate method is the summation of all the received encrypted data:

$$Aggr\_Data^J = [(\sum_{i=1}^{n}(dist\ (S_i, Aggregator^J) * G^J)\ , dist\ (S_i, Aggregator^J) * Pk^J)\ )] \quad (23)$$

4.2.5. Decryption aggregated data

In the decryption stage, the server may decrypt the aggregated data as follows:

$$Dec(Aggr\_Data^J) = (\sum_{i=1}^{n}(M_i^J + Sk^i\ *\ Pk^J) - q_1.q_2 * \sum_{i=1}^{n}(Sk^i * G^J)) \quad (24)$$

4.2.6. Data integrity check

A message authentication code (MAC) can simultaneously verify the integrity and authenticity of message data. A MAC can be calculated using a homogeneous elliptic curve-based algorithm:

$$Aggr_{MAC^J} = \sum_{i=1}^{n}(M_i^J + Sk^i\ *\ Pk^J) \quad (25)$$

To validate the authenticity of the aggregated data by calculating the sum of the MAC values of the encrypted data after decryption, the server pre-defines the generator nodes G^J, q_1, and q_2. Based on the public key Pk^J and the secret keys Sk^i assigned to the aggregator, where i = {1, 2, 3, … n}, the server knows the distance between the sensor node and its aggregator. Therefore, the sum of the private keys multiplied by the public key pointing to the MAC server is easily calculated. Furthermore, the server accepts the aggregated data from the aggregator J when the MAC server value is equal to $Aggr\_MAC^J$ or rejects it when they are not equal.

Proposed Scheme: EEECHE

01: **Input**: WSN-IoT network, Elliptic curve model E_P, and group of points on the elliptic curve E_P, server has information about locations for sensor nodes, their connected aggregator nodes, and aggregator nodes' locations.
Output: Secure aggregated data after the integrity test.

02: **Phase 1** Keys-Distribution Keys: The server generates and distributes q_1, q_2, G^J:
compute (Pk^J, Sk^i) using parameters: q_1, q_2, G^J, dist (Si, $Aggregator^J$)
where q1, q2 are prime numbers and $1 \le q_1.q_2 \le (p-1)$, G^J is generation point on elliptic curve E_P is assigned to aggregator J. dist (Si, $Aggregator^J$) is distance between node i and aggregator J where $0 \le$ dist (S_i, $Aggregator^J$) $\le Tr$

03: **Phase 2** Encryption Sensory Data: each sensor node S_i encrypts its data based on elliptic curve E_P:
convert data m_i^J to M_i^J, where i denotes to the sensor node number, J denotes to aggregator node number and M_i is the point on the elliptic curve E_P
compute C_{i1}, where C_{i1} is point on elliptic curve E_P
compute C_{i2}, where C_{i2} point on elliptic curve E_P
compute MAC^i
Enc(M_i^J)= [C_{i1}, Ci2]
Also, the node transmits data packet to aggregator, where data packet contains: || Enc(M_i^J) || MAC^i || node id||

04: **Phase 4** Aggregated Encryption Data: each aggregator node J using Homomorphic cryptograph to aggregate encryption data:
compute $Aggr\_Data^J$, where J denotes to the aggregator node number.
compute $Aggr\_MAC^J$
Also, aggregator node sends the aggregated packet to the server, where aggregation packet contains:
|| $Aggr\_Data^J$ || $Aggr\_MAC^J$ || nodes ids that send data||

05: **Phase 5** Decryption of Aggregated packet: the server decrypt aggregated packet based on private key Sk^i:
compute Dec($Aggr\_Data^J$)

06 **Phase 6** Data Integrity Check: the server checks the integrity of aggregated packet
compute MAC^{Server}
check if MAC^{Server}== $Aggr\_MAC^J$ then
Aggregated packet is secured and accepted
Else
Aggregated packet is not secured and rejected
End if

4.3. Case study

We offer a case study to demonstrate how the proposed scheme works. There is a group of sensor nodes connected with the aggregator nodes that receive their data packets as follows:

- Aggregator 0 receives 8 data packets from nodes 0, 5, 8, 11, 12, 14, 18, and 19.
- Aggregator 1 receives 3 data packets from nodes 28, 31, and 39.
- Aggregator 2 receives 6 data packets from nodes 58, 41, 42, 43, 47, and 49.

4.3.1. Aggregator 0 has (X0, Y0)0 and Pk0

- Node 0 generates m_0^0 and convert it to M_0^0 and encrypt it into C_{01}^0 =Sk^0*(X_0, Y_0)^0 and C_{52}^0=M_0^0 + Sk^0* Pk^0
- Node 5 generates m_5^0 and convert it to M_5^0 and encrypt it into C_{51}^0 = Sk^5*(X_0, Y_0)^0 and C_{52}^0=M_5^0 + Sk^5* Pk^0
- Node 8 generates m_8^0 and convert it to M_8^0 and encrypt it into C_{81}^0=Sk^8*(X_0, Y_0)^0 and C_{182}^0=M_8^0+ Sk^8* Pk^0
- Node 11 generates m_{11}^0 and convert it to $M_{11}^0$and encrypt it into C_{111}^0=Sk^{11}*(X_0, Y_0)^0 and C_{112}^0=M_{11}^0+ Sk^{11}* Pk^0
- Node 12 generates m_{12}^0 and convert it to M_{12}^0 and encrypt it into C_{121}^0=Sk^{12}*(X_0, Y_0)^0 and C_{122}^0=M_{12}^0+ Sk^{12}* Pk^0
- Node 14 generates m_{14}^0 and convert it to M_{14}^0 and encrypt it into C_{141}^0=Sk^{14}*(X_0, Y_0)^0 and C_{142}^0=M_{14}^0+ Sk^{14}* Pk^0
- Node 18 generates m_{18}^0 and convert it to M_{18}^0 and encrypt it into C_{181}^0=Sk^{18}*(X_0, Y_0)^0 and C_{182}^0=M_{18}^0+ Sk^{18}* Pk^0

- Node 19 generates m_{19}^0 and convert it to M_{19}^0 and encrypt it into C_{191}^0=Sk^{19}*(X$_0$, Y$_0$)^0 C_{192}^0=M_{19}^0+ Sk^{19}* Pk^0

4.3.2. Aggregator 1 has (X0, Y0)1 and Pk1

- Node 28 generates m_{28}^1 and convert it to M_{28}^1 and encrypt it into C_{281}^1= Sk^{28}*(X$_0$, Y$_0$)^1 and C_{282}^1=M_{28}^1 + Sk^{28}* Pk^1
- Node 31 generates m_{31}^1 and convert it to M_{31}^1 and encrypt it into C_{311}^1=Sk^{28}*(X$_0$,Y$_0$)^1 and C_{312}^1=M_{31}^1 + Sk^{28}* Pk^1
- Node 39 generates m_{39}^1 and convert it to M_{39}^1 and encrypt it into C_{391}^1=Sk^{28}*(X$_0$, Y$_0$)^1 and C_{392}^1=M_{39}^1 + Sk^{28}* Pk^1

4.3.3. Aggregator 2 has (X0, Y0)2 and Pk2

- Node 41 generates m_{41}^2 and convert it to M_{41}^2 and encrypt it into C_{411}^2=Sk^{41}*(X$_0$, Y$_0$)^2 and C_{411}^2=M_{41}^2 + Sk^{41}* Pk^2
- Node 42 generates m_{42}^2 and convert it to M_{42}^2 and encrypt it into C_{421}^2=Sk^{42}*(X$_0$, Y$_0$)^2 and C_{421}^2=M_{42}^2 + Sk^{42}* Pk^2
- Node 43 generates m_{43}^2 and convert it to M_{43}^2 and encrypt it into C_{431}^2 Sk^{43}*(X$_0$, Y$_0$)^2 and C_{431}^2=M_{43}^2 + Sk^{43}* Pk^2
- Node 47 generates m_{47}^2 and convert it to M_{47}^2 and encrypt it into C_{471}^2=Sk^{47}*(X$_0$, Y$_0$)^2 and C_{471}^2=M_{47}^2 + Sk^{47}* Pk^2
- Node 49 generates m_{49}^2 and convert it to M_{49}^2 and encrypt it into C_{491}^2=Sk^{49}*(X$_0$, Y$_0$)^2 and C_{491}^2=M_{49}^2 + Sk^{49}* Pk^2
- Node 58 generates m_{58}^2 and convert to M_{58}^2 and encrypt it into C_{581}^2=Sk^{58}*(X$_0$, Y$_0$)^2 and C_{581}^2=M_{49}^2 + Sk^{58}* Pk^2

The encrypted data is sent to the aggregators. The aggregated data generated by aggregator 0 is $[\sum_{i=1}^{n} C_{i1}^0, \sum_{i=1}^{n} C_{i2}^0]$. Similarly, the aggregated data generated by aggregators 1 and 2 are $[\sum_{i=1}^{n} C_{i1}^1, \sum_{i=1}^{n} C_{i2}^1]$ and $[\sum_{i=1}^{n} C_{i1}^2, \sum_{i=1}^{n} C_{i2}^2]$ respectively. Each aggregator sends the aggregated data to the server. The decryption of aggregated data is sent by aggregator 0 as $[\sum_{i=1}^{n} C_{i2}^0 - \sum_{i=1}^{n} C_{i1}^0]$. Similarly, the decryption of aggregated data is sent by aggregators 1 and 2 as $[\sum_{i=1}^{n} C_{i2}^1 - \sum_{i=1}^{n} C_{i1}^1]$ and $[\sum_{i=1}^{n} C_{i2}^2 - \sum_{i=1}^{n} C_{i1}^2]$ respectively.

5. Results and discussion

To measure the performance metrics of this scheme, the use of OMNeT++ is simulated, and the performance is analyzed under different conditions. In the simulation experiment, we suppose that the number of sensor nodes is 60 and they are randomly distributed in the (100*100) m^2 area. The sensing range is 20 m. The number of aggregator devices in the network is 3. The initial energy for the aggregator device is 1000 joules, and the initial energy for the sensor node that transmits data is 500 joules. The energy consumption for sending one bit is 1*10^{-6} joule, energy consumption to aggregate one bit is 1*10^{-7} joule and energy consumption to amplify data is 1*10^{-8} joule. However, we can conduct many simulation experiments with different values of the above-mentioned parameters. Also, the simulation experiment run on that network topology has different transmission ranges.

Fig. 5 shows the total number of sensor nodes transmitting data to aggregators in the network. We observe that as the transmission range increases, the total number of nodes transmitting data to aggregators increases significantly. To interpret this experiment, each node has a unique location (x, y) randomly selected from the simulation. Communication between a node and an aggregator depends on the transmission range and the distance between them, which is calculated based on their locations.

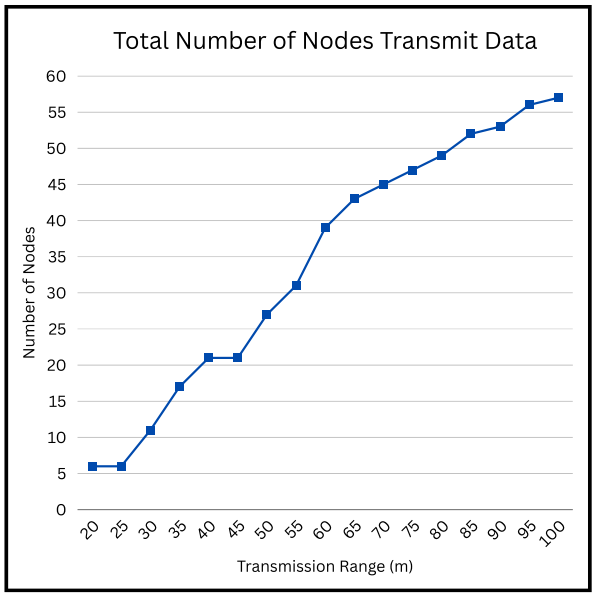


Fig. 5: Total number of nodes transmitting data to aggregator nodes

In a simulation experiment, we also measured the total energy consumption and remaining energy of three aggregators. Energy consumption is defined as the amount of energy spent by a sensor node in the WSN. In this experiment, we measured the energy consumption of the aggregators due to receiving, processing, and transmitting data, as described in Adawy et al. (2018) in the free-space propagation model.

$$E_{Enc}(k) = (E_{elec} * k) \quad (26)$$

$$E_{Tx}(k, d) = E_{Enc}(k) + (E_{amp} * k * d)^2 \quad (27)$$

where, E$_{Enc}$ is the energy consumption for encrypting k-bit data. E$_{Tx}$ is the energy consumption for transmitting encrypted data to the server. d is the distance between the aggregator device and the server. E$_{elec}$ is the energy consumption for transmitting or receiving, or processing one-bit data. E$_{amp}$ is the amplification energy.

$$E_{Rx}(m) = (E_{elec} * m) \quad (28)$$

$$E_{Process} = E_{Dec}(m)+E_{Agg}(m)+E_{Enc}(k) \quad (29)$$

where, E_{Rx} is energy consumption for receiving m-bits of encrypted data from a sensor node; E_{Dec} is energy consumption for decrypting encrypted data; E_{Agg} is energy consumption for aggregating encrypted data.

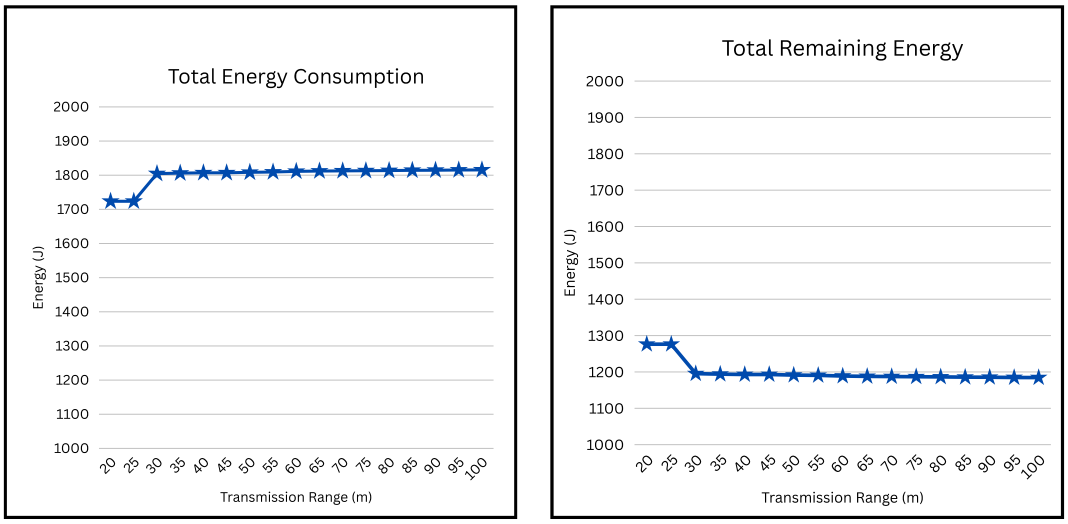
In the proposed scheme, the energy depletion for decrypting and aggregating the encrypted data is zero for both. Since there is no need to decrypt received encrypted data and encrypt aggregated data. The energy consumption and remaining energy consumption of the aggregation device are given by:

$$Energy\ Consumption = N * E_{Rx}(m) + N * E_{Agg}(m) + E_{Tx}(k,d) \quad (30)$$

$$Remaining\ Energy = Initial\ Energy - Energy\ Consumption \quad (31)$$

where, N is the number of sensor nodes that send encrypted data to the aggregator.

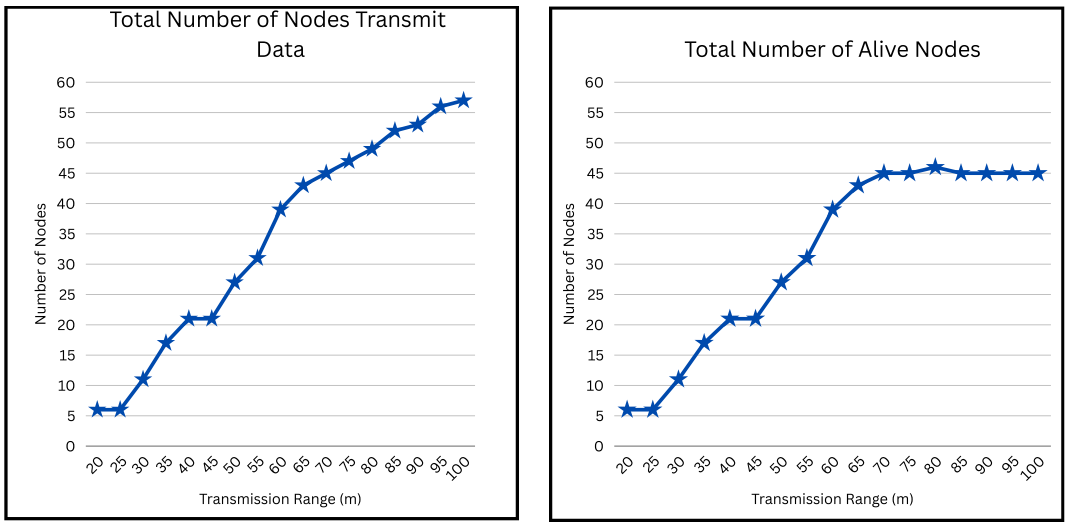
Fig. 6a displays the total energy consumption of three aggregators, while Fig. 6b displays the total remaining energy for them. Remaining energy is defined as the residual energy after these aggregators have completed their tasks. The results demonstrate that as the transmission range increases, the total number of encrypted data sent to the aggregators increases slowly, due to the rise in the total number of sensor nodes communicating with these aggregators. This leads to an increase the energy consumption due to receiving and aggregating the encrypted data. On the other hand, the total remaining energy decreases as the transmission range increases, because of the increase in the total energy consumption of the aggregators.



(a) (b)

Fig. 6: (a) Total energy consumption; (b) Total remaining energy

Fig. 7a shows the total number of nodes transmitting data to three aggregators, while Fig. 7b shows the total number of alive nodes in the network. The results show that as the transmission range increases, the total number of transmitting nodes increases, and the number of alive nodes increases dramatically until the transmission range reaches 70 meters. Beyond this range, the number of alive nodes remains roughly constant because many of them exhaust their energy and die.



(a) (b)

Fig. 7: (a) Total number of node transmit data; (b) Total number of live nodes

Fig. 8 demonstrates the energy consumption and remaining energy of the sensor nodes that transmit their encrypted data to the three aggregators. Experience illustrates that the number of sensor nodes transmitting encrypted data upsurges as the transmission range increases. Consequently, the total energy consumption and remaining energy of these nodes will increase until the transmission range reaches 70 meters due to some nodes being unable to operate. Beyond this range, the remaining energy decreases as many nodes stop operating due to energy exhaustion.

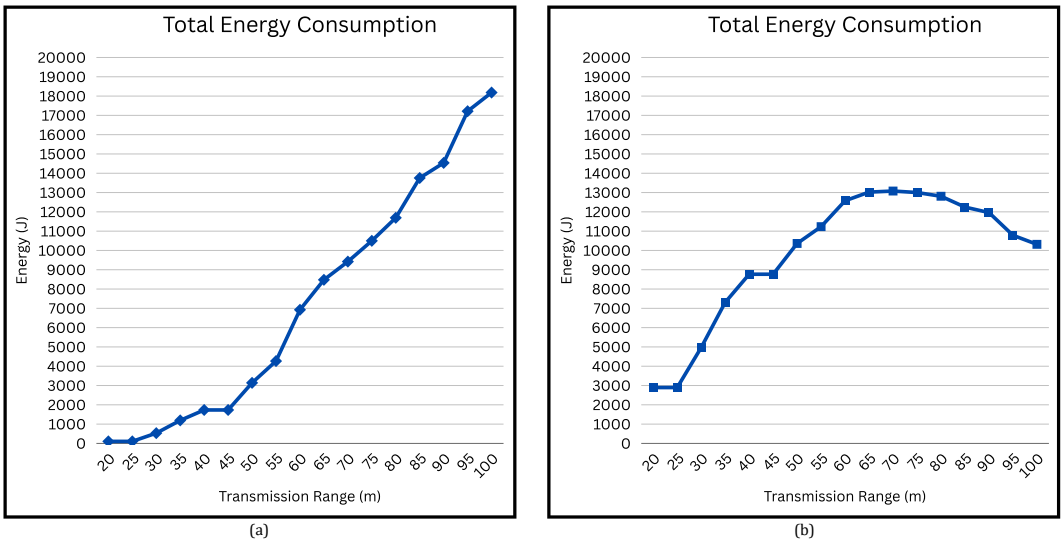


Fig. 8: (a) Total energy consumption for sensor nodes transmitting data; (b) Total remaining energy for sensor nodes transmitting data

Fig. 9 demonstrates the end-to-end delay of encrypted data being transmitted from nodes until the aggregated data reaches the server. The results illustrate that the end-to-end delay increases with increasing transmission range due to the increase in the total number of sensor nodes transmitting encrypted data.

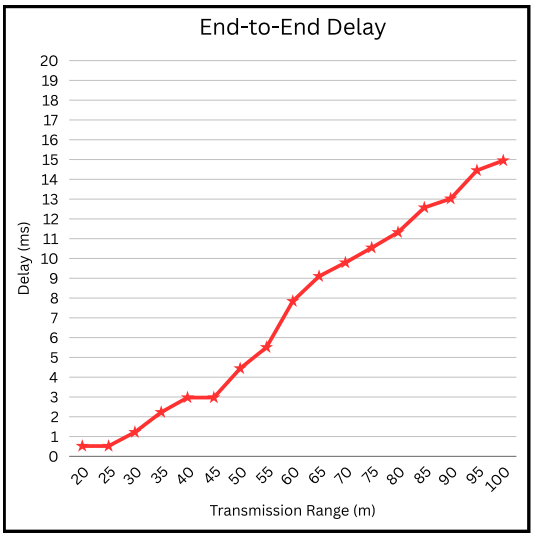


Fig. 9: End-to-end delay for three aggregator nodes

6. Proposed scheme evaluation

Simulation parameters and simulation environment are used in this study. The performance of the proposed scheme is assessed using the OMNeT++ simulator. The network density is defined by the number of sensor nodes in the same area. Several parameters are defined in the simulator, as shown in Table 2. The performance metrics of the proposed scheme, CSHEAD, and the CSDA schemes are measured.

Table 2: Simulation parameters

Parameters	Values
Number of nodes	50,60,70,80
Number of aggregator devices	3
Area	100x100 m
Transmission range	50 m
Initial aggregator node energy	1000 J
Initial sensor node energy	500 J
The node deployment	Randomly
The sensing range	50 m
Energy consumption to send one bit	$1*10^{-6}$ joule
Energy consumption for processing a bit	$1*10^{-7}$ joule
Energy consumption for the amplifier	$1*10^{-8}$ joule

The performance metrics are end-to-end delay, total energy consumption, and remaining energy in the aggregators. For each metric, CSDA and CSHEAD are compared with the proposed scheme. In the WSN-IoT, the sensor nodes have low-power batteries, and energy consumption depends on data transmission and reception. Data transmission energy consumption takes most of a node's energy consumption due to its dependence on the distance between the node and the aggregator. Therefore, when this distance is large, energy consumption is high. The results demonstrate that our scheme significantly improves the performance of WSN-IoT. Our scheme achieves better than CSHEAD and CSDA in terms of end-to-end delays, energy consumption, and remaining energy.

6.1. End-to-end delay

Fig. 10 demonstrates the end-to-end delay on diverse numbers of sensor nodes in the CSDA, CSHEAD, and the proposed scheme. The results

demonstrate that the end-to-end delay in our scheme is lower than that in both CSDA and CSHEAD. Furthermore, the end-to-end delay in CSHEAD is lower than that in CSAD.

6.2. Energy consumption and remaining energy

Fig. 10 shows the total energy consumption and total residual energy of three collectors in the network at different node numbers in the CSDA, CSHEAD, and proposed schemes. The results show that the proposed scheme significantly improves the network lifetime by reducing the collectors' energy consumption compared to the CSDA and CSHEAD schemes, as shown in Fig. 11a. The results in Fig. 11b also show that the proposed scheme significantly contributes to conserving the aggregators' remaining energy by consuming less of it compared to the CSDA and CSHEAD schemes. Furthermore, the aggregators exhausted their full energy when the number of nodes in the network was 60 in the CSDA scheme, while they exhausted their energy when the number of nodes in the network was approximately equal to 75 in the CSHEAD scheme.

7. Conclusion

Wireless sensor networkin IoT (WSN-IoT) security has received widespread attention, with most investigations related to the perception layer components. The IoT perception layer provides the storage of transmitted data. Encrypting transmitted data packets is crucial to securing aggregated data in WSN-IoT and preventing security attacks. On the other hand, encrypting and decrypting all received data increases the power consumption of the aggregator device and increases the end-to-end delay.

This paper presents a data aggregation security approach based on end-to-end elliptic curve and Homomorphic encryption in a WSN-IoT system, which ensures the privacy of data transmitted from sensor nodes to the server with minimal power consumption and end-to-end delay. Furthermore, the proposed scheme uses a message confirmation code (MAC) to verify the accuracy of the collected data, allowing for the fastest possible identification of forged information. The proposed scheme secures the aggregated data while reducing the end-to-end delay and energy consumption compared to CSHEAD and CSDA schemes.

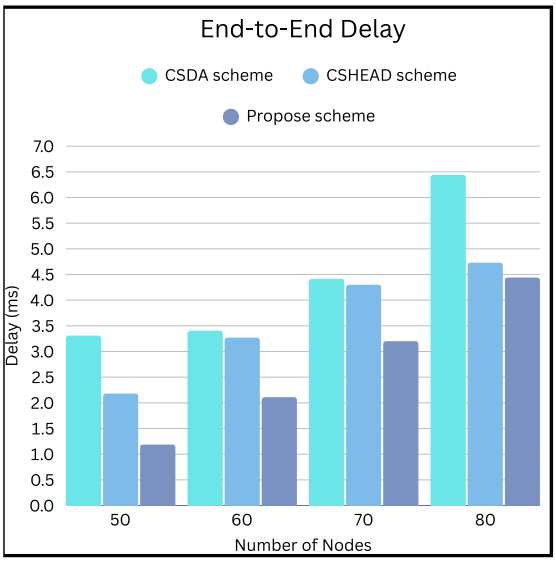


Fig. 10: Evaluation proposed scheme with other schemes based on end-to-end delay

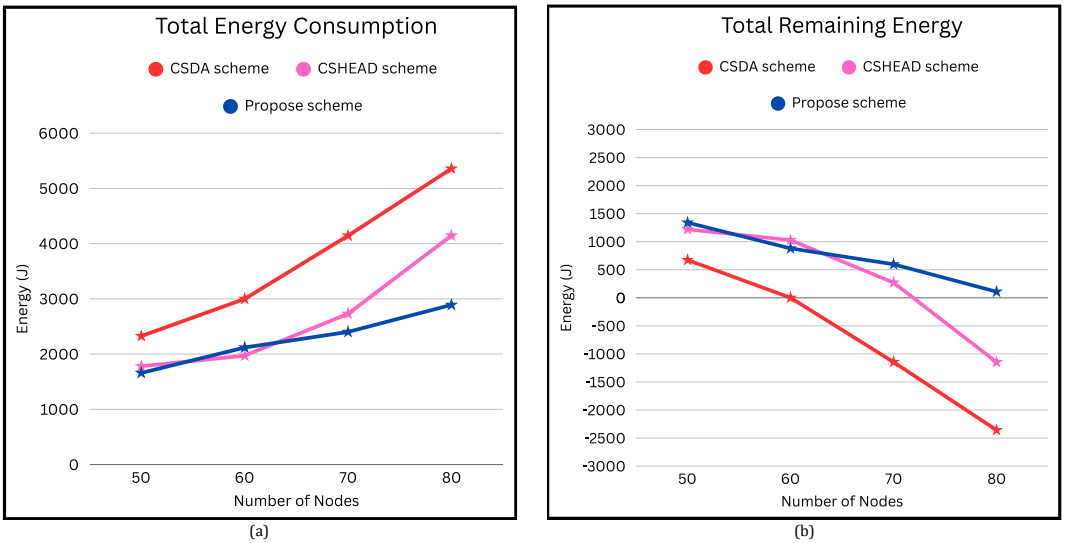


Fig. 11: (a) Total energy consumption for aggregators; (b) Total remaining energy for aggregators

List of abbreviations

Aggr_Data	Aggregated data
Aggr_MAC	Aggregated message authentication code
BS	Base station
CH	Cluster head
CSDA	Cluster-based secure data aggregation
CSHEAD	Cluster-based semi-homomorphic encryption aggregated data
DoS	Denial of service
EC-ElGamal	Elliptic curve ElGamal cryptosystem
ECC	Elliptic curve cryptography
EEECHE	End-to-end elliptic curve and homomorphic encryption

EEHE	End-to-end homomorphic encryption
EP	Elliptic curve parameters
FESA	Fully encrypted secure aggregation
FHE	Fully homomorphic encryption
GF(p)	Galois field of prime order p
HE	Homomorphic encryption
HMAC	Hash-based message authentication code
IoT	Internet of Things
IP	Internet protocol
MAC	Message authentication code
MACserver	Message authentication code computed at the server
OMNeT++	Objective modular network testbed in C++
Pk	Public key
SEDA-ECC	Secure-enhanced data aggregation based on elliptic curve cryptography
Sk	Secret (private) key
WSN	Wireless sensor network
WSN-IoT	Wireless sensor network–Internet of Things

Funding

The research fees are covered by WISE University.

Compliance with ethical standards

Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Adawy MI, Awang Nor S, and Mahmuddin M (2018). Data redundancy reduction in wireless sensor network. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 10(1-11): 1–6.

Adawy MI, Tahboush M, Aloqaily O, and Abdulraheem W (2023). Man-in-the-middle attack detection scheme on data aggregation in wireless sensor networks. International Journal of Advances in Soft Computing and its Applications, 15(2): 179–193.

Al-Baz A and El-Sayed A (2018). A new algorithm for cluster head selection in LEACH protocol for wireless sensor networks. International Journal of Communication Systems, 31(1): e3407. https://doi.org/10.1002/dac.3407

Bajpai A and Yadav A (2024). A novel approach for secure data aggregation scheme in battlefield surveillance using elliptic curve cryptography. In: Roy BK, Chaturvedi A, Tsaban B, and Hasan SU (Eds.), Cryptology and network security with machine learning. ICCNSML 2022. Algorithms for intelligent systems: 265–277. Springer, Singapore, Singapore. https://doi.org/10.1007/978-981-99-2229-1_23

Elhoseny M, Elminir H, Riad A, and Yuan X (2016). A secure data routing schema for WSN using elliptic curve cryptography and homomorphic encryption. Journal of King Saud University - Computer and Information Sciences, 28(3): 262–275. https://doi.org/10.1016/j.jksuci.2015.11.001

Fang W, Wen XZ, Xu J, and Zhu JZ (2019). CSDA: A novel cluster-based secure data aggregation scheme for WSNs. Cluster Computing, 22(Suppl 3): 5233–5244. https://doi.org/10.1007/s10586-017-1195-7

Gentry C and Halevi S (2011). Implementing Gentry's fully-homomorphic encryption scheme. In: Paterson KG (Ed.), Advances in cryptology – EUROCRYPT 2011. Lecture notes in computer science, 6632: 129-148. Springer, Berlin, Germany. https://doi.org/10.1007/978-3-642-20465-4_9

Gulati K, Boddu RS, Kapila D, Bangare SL, Chandnani N, and Saravanan G (2022). A review paper on wireless sensor network techniques in Internet of Things (IoT). Materials Today: Proceedings, 51(Part 1): 161-165. https://doi.org/10.1016/j.matpr.2021.05.067

Hong MQ, Wang PY, and Zhao WB (2016). Homomorphic encryption scheme based on elliptic curve cryptography for privacy protection of cloud computing. In the IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), IEEE, New York, USA: 152-157. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS.2016.51

Huang Y, Wang L, Hou Y, Zhang W, and Zhang Y (2018). A prototype IOT based wireless sensor network for traffic information monitoring. International Journal of Pavement Research and Technology, 11(2): 146-152. https://doi.org/10.1016/j.ijprt.2017.07.005

Ifzarne S, Hafidi I, and Idrissi N (2021). A novel secure data aggregation scheme based on semihomomorphic encryption in WSNs. Journal of Communications, 16(8): 323–330. https://doi.org/10.12720/jcm.16.8.323-330

Kumar M, Sethi M, Rani S, Sah DK, AlQahtani SA, and Al-Rakhami MS (2023). Secure data aggregation based on end-to-end homomorphic encryption in IoT-based wireless sensor networks. Sensors, 23(13): 6181. https://doi.org/10.3390/s23136181 **PMid:37448038 PMCid:PMC10346161**

Kumar M, Verma S, and Lata K (2015). Secure data aggregation in wireless sensor networks using homomorphic encryption. International Journal of Electronics, 102(4): 690–702. https://doi.org/10.1080/00207217.2014.936524

Lavanya G, Velammal BL, and Kulothungan K (2022). SCDAP–Secured cluster based data aggregation protocol for energy efficient communication in wireless sensor networks. Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 44(3): 4747-4757. https://doi.org/10.3233/JIFS-223256

Li X, Chen D, Li C, and Wang L (2015). Secure data aggregation with fully homomorphic encryption in large-scale wireless sensor networks. Sensors, 15(7): 15952–15973. https://doi.org/10.3390/s150715952 **PMid:26151208 PMCid:PMC4541862**

Mocanu DC, Vega MT, and Liotta A (2015). Redundancy reduction in wireless sensor networks via centrality metrics. In the IEEE International Conference on Data Mining Workshop (ICDMW), IEEE, Atlantic City, USA: 501-507. https://doi.org/10.1109/ICDMW.2015.53

Pu Y, Luo J, Hu C, Yu J, Zhao R, Huang H, and Xiang T (2019). Two secure privacy-preserving data aggregation schemes for IoT. Wireless Communications and Mobile Computing, 2019: 3985232. https://doi.org/10.1155/2019/3985232

Qu Z and Li B (2022). An energy-efficient clustering method for target tracking based on tracking anchors in wireless sensor networks. Sensors, 22(15): 5675. https://doi.org/10.3390/s22155675 **PMid:35957232 PMCid:PMC9371144**

Randhawa S and Jain S (2017). Data aggregation in wireless sensor networks: Previous research, current status and future directions. Wireless Personal Communications, 97: 3355–3425. https://doi.org/10.1007/s11277-017-4674-5

Siddiqui S, Khan AA, and Ghani S (2015). A survey on data aggregation mechanisms in wireless sensor networks. In the International Conference on Information and Communication Technologies (ICICT), IEEE, Karachi, Pakistan: 1-7. https://doi.org/10.1109/ICICT.2015.7469596

Szczechowiak P and Collier M (2009). TinyIBE: Identity-based encryption for heterogeneous sensor networks. In the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), IEEE, Melbourne, Australia: 319-354. https://doi.org/10.1109/ISSNIP.2009.5416743

Ugus O, Westhoff D, Laue R, Shoufan A, and Huss SA (2009). Optimized implementation of elliptic curve based additive homomorphic encryption for wireless sensor networks. Arxiv Preprint Arxiv:0903.3900. https://doi.org/10.48550/arXiv.0903.3900

Zhou Q, Yang G, and He L (2014). A secure-enhanced data aggregation based on ECC in wireless sensor networks. Sensors, 14(4): 6701–6721. https://doi.org/10.3390/s140406701 **PMid:24732099 PMCid:PMC4029653**

Zijie F, Al-Shareeda MA, Saare MA, Manickam S, and Karuppayah S (2023). Wireless sensor networks in the Internet of Things: Review, techniques, challenges, and future directions. Indonesian Journal of Electrical Engineering and Computer Science, 31(2): 1190–1200. https://doi.org/10.11591/ijeecs.v31.i2.pp1190-1200