An Authentication-Based Secure Data Aggregation Method in Internet of Things

Maryam Ataei Nezhad · Hamid Barati · Ali Barati

Received: 8 November 2021 / Accepted: 6 July 2022 © The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract Internet of Things (IoT) means connecting different devices through the Internet. The Internet of things enables humans to remotely manage and control the objects they use with the Internet infrastructure. After the advent of the Internet of Things in homes, organizations, and private companies, privacy and information security are the biggest concern. This issue has challenged the spread of the Internet of things as news of the user’s theft of information by hackers intensified. The proposed method in this paper consists of three phases. In the first phase, a star structure is constructed within each cluster, and a unique key is shared between each child and parent to encrypt and secure subsequent communications. The second phase is for intra-cluster communications, in which members of the cluster send their data to the cluster head in a multi-hop manner. Also, in this phase, the data is encrypted with different keys in each hop, and at the end of each connection, the keys are updated to ensure data security. The third phase is to improve the security of inter-cluster communications using an authentication protocol. In this way, the cluster heads are authenticated before sending information to prevent malicious nodes in the network. The proposed method is also simulated using NS2 software. The results showed that the proposed method has improved in terms of energy consumption, end-to-end delay, flexibility, packet delivery rate, and the number of alive nodes compared to other methods.

Keywords Internet of things · Security · Authentication · Secure routing · Elliptic curve cryptography · Rail fence cipher

1 Introduction

The Internet of Things is a new concept in the world of information and communication technology [14, 26]. In short, it is a modern technology in which it provides the ability for any creature (human, animal, or object) to send and receive data through communication networks, including the Internet or intranet [18, 27]. IoT refers to everyday objects that can be identified, located, addressed, and controlled over the Internet [23].

Given the variable and potentially vulnerable topology of any network, routing and addressing are two critical issues in IoT networks that require special attention.
Routing means choosing the best route to transfer data to the desired destination [21, 34]. IoT devices are arranged as a mesh network and are connected to the Internet through a gateway router, which makes them different from traditional wireless sensor networks. The characteristics of the IoT network vary greatly in network size, traffic pattern flow, and mobility [4]. The need for routing protocols in different scenarios should be analyzed based on traffic patterns, energy efficiency, scalability, mobility, reciprocity, and transmitter amplitude. The nature of IoT devices that make up a network has led to the development of new routing protocols.

One of the significant challenges that need to be addressed to bring the Internet of Things into the real world is security [12, 29]. With the increase in the number and variety of devices connected to the Internet of Things, its security is severely threatened [10, 16]. Security means protecting the system against malfunction or theft of hardware, software, and computer data [28, 38]. The security and reliability of nodes in the IoT are very essential because nodes can be added to or removed from the network over time [20, 36].

Identification and authentication are the two central cores in most access control systems [2]. Identification is the practice that a user uses to identify themselves to an information system, usually in the form of a username and password. Identity is a set of attributes and characteristics that distinguish a person from other people or objects from other objects. Authentication is a method of examining whether the other party in the relationship is what he/she should be or whether it is an intruder who has replaced the real party [8]. The only way a user can use a network or system’s services and facilities is to identify the user with what he/she claims to be authorized [30].

The proposed method’s primary motivation is to reduce energy consumption, improve the network lifetime, and enhance network security. To achieve this motivation, the proposed method provides a way to navigate the clustered IoT network securely. For this purpose, a hierarchical structure is formed in each cluster, which provides a ready infrastructure to aggregate data and reduces energy consumption. Due to the use of the rail fence cipher algorithm in the data transmission process and the use of unique keys, data transmission security is increased. So, when the key is exposed, only the connection between the two nodes is compromised. In the cluster heads’ data transfer phase, an efficient authentication protocol is used to prevent malicious nodes from infiltrating.

The innovations of the proposed method are as follows:

- We propose a hierarchical framework in each cluster for data aggregation to make the energy consumption more uniform among nodes, decrease energy consumption, and achieve an improved network lifetime.
- Due to the use of the rail fence cipher algorithm during data aggregation and transmission, security between IoT nodes is improved.
- We update the keys in each connection in a light and efficient way to prevent synchronization attacks.
- In the proposed method, due to the use of the shared key locally between the parent and the children, if the attacker succeeds in discovering the node key, it will only access the information of the two nodes and will not be able to access the other nodes’ information.
- In the proposed method, an efficient authentication method is used for communication between cluster heads to prevent malicious nodes from infiltrating.
- The experiment results showed that the proposed method had improved energy consumption, end-to-end delay, flexibility, packet delivery rate, and the number of alive nodes compared to other methods.

The rest of the paper is organized as follows. Section 2 reviews the previous secure data aggregation and authentication schemes in the IoT. Section 3 describes the proposed method, in detail. Section 4 presents the simulation of the proposed protocol and evaluates its performance. Finally, the conclusion is provided in Section 5.

2 Related Works

Given the importance of data aggregation and secure authentication to improve IoT security, much research has been done in this area. In this section, some examples of these articles are reviewed.
2.1 Secure Data Aggregation Schemes

In this section, a number of the latest secure data aggregation schemes are reviewed, and finally, in Table 1, their advantages and disadvantages are compared.

Haseeb et al. [11] presents a lightweight, secure Data Aggregation Routing called LSDAR for IoT combined with sensor networks. This method improves energy routing performance against malicious threats. This method’s primary goal is to enhance the reliability and network lifetime by using data aggregation and optimizing data forwarding process. At first, the network nodes and the energy holes around the base station are organized into several independent clusters. Then, for an efficient routing process, a tree is constructed based on residual energy and received signal strength. The constructed routing tree organizes data transmission routes with a fair number of scalable and dynamic levels. Finally, the forwarder hops in a constructed tree are guarded using an efficient and secure algorithm with low complexity for network reliability supporting.

Niu [22] proposed a secure opportunistic transmission system based on dynamic optimization that improves wireless networks’ energy efficiency and data transmission security. The proposed method specifies the dynamic signal energy switching transmission flow based on the information energy joint transmission system’s network model. This method protects the system’s dynamic signal data based on the group intelligence understanding model using fog nodes. A task allocation with privacy awareness and data aggregation method is proposed for this aim. This method comprises four steps: system initialization, generation of task and allocation, data collection and aggregation, and decryption of data.

In [3], due to the energy constraints of the device-to-device nodes in IoT usage for secure routing challenges, an optimal routing method is proposed to increase the secrecy and energy efficiency. The optimization problem is defined to maximize the SEE based on the secrecy rate constraints, SINR, PU outage loss, and maximum attempt to packet retransmission by the D2D links. In low and moderate node density, simulation results show better results than the other methods.

Saleem et al. [31] proposed a fog-enabled privacy-preserving data aggregation method. In this paper, the fog node aggregates users’ consumption data and forwards it to the control center. This method has used the Paillier cryptosystem; fog nodes aggregate the encrypted data from the user’s node. This method comprises four steps: Key generation, Encryption and Message Authentication Codes, generation of tag, Secure Aggregation, and MAC verification at the fog node, Decryption and verification of MAC at the control center, and Fault-Tolerant Aggregation and decryption.

Ullah et al. [35] proposed a FoG assisted healthcare-based data aggregation method to apply p2p communication between sensors and relevant wearables. In this method, when an aggregator is away from the FoG server, it can assign the encrypted data to the node aggregator to transfer data to the FoG server. A fog server can achieve the required data, save it in its storage, and be further updated later in cloud storage. This paper presented a Message Receiving Algorithm for the aggregator and a Message Extraction Algorithm for the FoG server. Moreover, to lower communication expenses, a compression mechanism is also proposed.

Tang et al. [33] introduced a privacy-preserving data aggregation scheme that collects data from multiple sources in a secure manner. Specifically, in this paper, a framework is proposed that has a three-layer data aggregation. In the first layer, to lower the computation cost, the relevant health care centers use secret keys signed by patients to aggregate data. In the second layer, the healthcare centers add noises to the data for resisting differential attacks. Also, they encrypt the data during transmission. The data is decrypted and aggregated in the third layer by the cloud center. Data users have access to the cloud to obtain the results and reward the patients.

Alagirisamy and Chow [1] proposed an energy-efficient cluster head selection algorithm using a dual sink. The clustering approach uses two sinks in this method to balance the energy consumption in cluster heads and improve network lifetime. The static and mobile sinks are responsible for gathering the data of the cluster heads. The static sink is in the network center, and the mobile sink transfer in a direct path on the network. In each round, the network is reclustered, and the static and mobile sinks collect data.

Kumaramangalam et al. [15] proposed an energy-aware edge-based network partitioning method to form sensor nodes into equal-sized clusters. Also, a hierarchical routing algorithm was proposed named zone-based routing protocol (ZBRP). The main purpose of ZBRP is to improve network lifetime by distributing energy consumption between network nodes.
| Reference          | Approach                                                                 | Advantages                                                                 | Disadvantages                                                                 |
|--------------------|--------------------------------------------------------------------------|----------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| Haseeb et al. [11] | Lightweight structure-based Data AggregationRouting (LSDAR) protocol for IoT integrated Next-generation Sensor Networks | Improvements in terms of energy consumption, network lifetime, an end to end delay, and packet drop ratio | Not supporting mobile nodes, Selecting the cluster heads only based on the remaining energy |
| Niu [22]           | A secure and reliable transmission scheme for the wireless communication system of the Internet of Things based on complexity analysis | Greater throughput, high reliability, and security                      | High computational overhead                                                   |
| Basak and Acharya [3] | A physical layer security method in a multi-hop cognitive radio enabled device-to-device (CRD2D) communication | Target SINR                                                                 | Primary users’ outage probability, maximum packet retransmission            |
| Saleem et al. [31] | A fog-enabled privacy-preserving data aggregation scheme (FESDA)         | Low communication cost                                                     | Dependence on trusted authority for the production of public and private keys |
| Ullah et al. [35]  | A FoG assisted scheme for healthcare data aggregation securely and efficiently (EHDA) | Low energy consumption and storage requirements                           | Not supporting mobile nodes                                                  |
| Tang et al. [33]   | A privacy-preserving health data aggregation scheme that securely collects health data from multiple sources and guarantees fair incentives for contributing patients | Improved key generation overhead, Aggregation privacy                      | High computational overhead                                                   |
| Alagirisamy and Chow [1] | Energy-based cluster head selection, Unequal clustering algorithm with dual sink (ECH-DUAL) for data transmission in continuous monitoring applications | Energy-efficient approach                                                 | High hardware redundancy                                                     |
| Kumaramangalam et al. [15] | A novel network organization scheme, energy-efficient edge-based network partitioning scheme, to organize sensor nodes into clusters of equal size | Distributes the load uniformly across the network, enhances the sensor network lifetime | Equal size clusters                                                          |
| Proposed method    | An Authentication-Based Secure Data Aggregation Method in Internet of Things | Improved in terms of energy consumption, flexibility, packet delivery rate, and the number of alive nodes | Increase in end-to-end delay                                                  |
An Authentication-Based Secure Data Aggregation Method in Internet of Things

2.2 Authentication Schemes

In this section, a number of the latest authentication schemes are reviewed. Finally, a table is presented to assess their resistance to some attacks related to the field of authentication.

Kalra and Sood [13] presented a mutual authentication and key generation method based on the ECC algorithm. [5] pointed out two significant defects: achieving mutual authentication and session key generation, as it has been claimed, is impossible. Then, Chang et al. presented an improved method that attempted to overcome the disadvantages of Kalra and Sood’s method. [40] show that the method proposed by Chang et al. is still insecure. An attacker can simply impersonate the server and create a connection with a known device. Wang et al. proposed a method that makes known devices communicate securely with a server on an IoT network.

Dang et al. [6] proposed an authentication approach that provides secure control for devices by cloud servers and establishes direct communications. This method provides efficient resource and energy consumption using ECC and low-cost operations. This method is used for communication, authentication, and MAC between IoT nodes. Using centralized servers leads to employing complex management policies. Their authentication protocol includes three phases: Registration, authentication between servers and devices, and authentication between two devices. Panda and Chattopadhyay [24] proposed a secure mutual authentication approach for IoT and cloud servers based on the ECC algorithm. This protocol consists of two phases: Registration and authentication. The embedded device computes protected identity in the registration phase and selects a unique password for each node. The cloud server saves the values received through the registration phase and a status bit. In the authentication phase, the device sends a request login to the server, and then mutual authentication is performed.

Sowjanya et al. [32] mentioned an authentication scheme in [17] that is unsafe on some attacks, such as lack of perfect forward secrecy, lack of key control, and cannot resist the clock synchronization. Sowjanya et al. enhanced a lightweight protocol for end-to-end authentication based on ECC that overcame Li et al.’s scheme’s security disadvantages. This scheme is proposed in three phases. In the first phase, called Initialization, the system’s parameters are defined. In the second phase, called registration, each user must be registered to the network manager to get the authentication parameters. In the third phase, called authentication, the user must be authenticated with the application server to receive the medical services.

Das et al. [7] presented a protocol for authentication in the RFID system that authenticates tag and reader mutually and preserves the privacy of tags. In this method, the tag and reader’s public key behave as the ID, and its private key should not be revealed. This scheme is proposed in three phases. In the setup phase, a one-time computation is performed and recalled when a tag is added/removed from the system. In the Authentication phase, the tag and reader communicate as on-demand.

Let A1, A2, A3, A4, A5, A6, A7, and A8 denote resistance to man-in-the-middle attack, resistance to synchronization attack, resistance to replay attack, resistance to impersonation attack, session key establishment, traceability, and mutual authentication, respectively. Table 2 provides an overview of the resistance of the expressed authentication schemes to different attacks.

3 Proposed Method

In this paper, a scheme is proposed to improve the security of IoT networks. A clustered network is considered to have one cluster head and several cluster members in the proposed method. The clustered network improves scalability, reduces traffic, and speeds up. In the proposed method, a star structure is constructed within each cluster for data transmission. Then, unique keys are shared between each parent and child for the subsequent
communication. These keys are used to encrypt data using rail fence cipher, which is a lightweight, encryption method. An authentication protocol is also used to improve the security of inter-cluster communication to ensure the cluster heads’ authenticity and to prevent the presence of malicious nodes in the network. Details of the proposed method are described below.

3.1 Network Model

In the proposed method, a homogeneous Internet of Things network is considered, including a powerful base station (BS) with unlimited energy sources, several nodes, and cluster heads. In this IoT network, every node has an ID and is equipped with GPS, which can be applied to find out its location. Furthermore, the nodes are developed in the network area. The network is supposed to be clustered. Network nodes are fixed, have minimal resources, and face memory, power, computing power, bandwidth, and communication range limitations. These nodes’ main task is to sense data from the environment and send this data to the cluster head. Each cluster’s sensed data reaches the star structure’s root through the child sending

---

**Table 2** Comparison and summary of previous authentication methods

| Reference                        | A1 | A2 | A3 | A4 | A5 | A6 | A7 |
|----------------------------------|----|----|----|----|----|----|----|
| Wang et al. [40]                 | Yes| No | Yes| Yes| Yes| No | Yes|
| Dang et al. [6]                  | Yes| No | Yes| Yes| Yes| No | Yes|
| Panda and Chattopadhyay [24]     | Yes| No | Yes| Yes| Yes| Yes| Yes|
| Sowjanya et al. [32]             | No | Yes| Yes| Yes| Yes| Yes| Yes|
| Das et al. [7]                   | Yes| No | Yes| Yes| Yes| No | Yes|
| Proposed method                  | Yes| Yes| Yes| Yes| Yes| Yes| Yes|

---

**Fig. 1** Network model in the proposed method
successive messages to the parent. Cluster heads receive data from cluster members, aggregating them and transmitting them to the base station in a few hops. The base station is permanently outside the network field, the nodes are aware of BS location, and BS should process, analyze, and decide on the data received from the cluster heads. The network model in the proposed method is shown in Fig. 1.

In the proposed method, we have two types of channels, the first is the secure channel, and the second one is the insecure channel. According to [37], a secure channel is one that an adversary can not access it.

3.2 Energy Consumption Model

In WSN [19, 25], the radio energy dissipation model is used to transmit $m$ bit message over a distance $d$ in the transmission is according to (1). This energy model is also used in the proposed method.

$$E_{Tx}(m,d) = m \times E_{elec} + m \times \phi_f \times d^2, \quad d < T_i$$
$$m \times E_{elec} + m \times \phi_m \times d^4, \quad d \geq T_i$$

(1)

$E_{elec}$ is electronic energy and the ratio of $\phi_f$ and $\phi_m$ are constants that denote the amplifier energy to maintain the acceptable signal to noise ratio. The same procedures are implemented in the receiver, and the energy spent on the radio is

$$E_{Rx}(m) = m \times E_{elec}$$

(2)

Compared with the communication energy [39], the energy consumed on computing and storage process is much lower. So for simplicity, we only consider the energy consumption on communication. To combine the number’s’ such messages, the energy consumes

$$E_{Ds} = s \times k \times E_{merge}$$

(3)

In (1) and (2), $E_{elec}$ represents the energy consumption of transmitting or receiving a 1-bit message. In (3), $E_{merge}$ represents the energy consumption of merge 1-bit message, and $T_i$ shows the threshold value when the distance is less than $T_i$, the free space channel model ($d^2$ power loss) is used; when the distance is more than $T_i$, the multi-path fading channel model ($d^4$ power loss) is used.

### 3.2.1 Phase I: Star Construction and Key Distribution

This section describes the relationship between the cluster heads and cluster members. Initially, a star structure is constructed inside each cluster so that the cluster head is considered as the root of the star. The star’s next levels are then constructed based on a fitness function consisting of the nodes’ remaining energy and their locations. When constructing a star structure, a key is shared between each node and its parent, which is used to encrypt the data before sending it to make the data transfer more secure. Also, keys shared between nodes and their parents are updated at specified times to improve security in the encryption process. The steps of this phase are described below.

- **Step 1: Sending Hello packet**
  Each node sends a Hello packet containing its ID and location to its neighbors, as shown in Fig. 2.

  Each node, also based on the Hello packets received from their neighbors, stores their information in their neighbors’ table, as shown in Table 3. The number of Hello packets received by each node indicates the number of node neighbors, which will then be stored in the variable $K$.

  An overview of the network in the first step of the first phase of the proposed method is shown in Fig. 3.

- **Step 2: Calculating the fitness to become a parent**
  Each node calculates its fitness for parenting according to the information stored in the node neighbors’ table and based on (4).
Where $E_{\text{remaining}}$ is the remaining energy of the node, $E_{\text{initial}}$ is the energy of the node when the battery is fully charged, $D_{\text{max}}$ is the distance from the node to the farthest neighbor node, parameter $i$ is the node neighbor counter, parameter $K$ is the number of node neighbors, ($x_n, y_n$) and ($x_i, y_i$) are node coordinates and neighbor node coordinates, respectively.

- **Step 3:** Calculating the maximum number of children

The maximum value of the function $F_{\text{it}}_n$ is when the battery charge of the node is full, and as a result, the fraction $E_{\text{remaining}} / E_{\text{initial}}$ equals one; also, the average distance of the node to its neighbors equals the distance of the node to the farthest neighbor node, the fraction $\frac{\sum_{i=1}^{K} \sqrt{(x_n-x_i)^2+(y_n-y_i)^2}}{K D_{\text{max}}}$ equals one, and the final result of the function $\left(1 - \frac{\sum_{i=1}^{K} \sqrt{(x_n-x_i)^2+(y_n-y_i)^2}}{K D_{\text{max}}} \right)$ equals two.

The minimum value of the function $F_{\text{it}}_n$ is when the battery charge of the node is empty, and as a result, the fraction $E_{\text{remaining}} / E_{\text{initial}}$ equals zero; also, the average distance of the node to its neighbors equals the distance of the node to the farthest neighbor node, the fraction $\frac{\sum_{i=1}^{K} \sqrt{(x_n-x_i)^2+(y_n-y_i)^2}}{K D_{\text{max}}}$ equals one, and the final result of the function $\left(1 - \frac{\sum_{i=1}^{K} \sqrt{(x_n-x_i)^2+(y_n-y_i)^2}}{K D_{\text{max}}} \right)$ equals zero.

As a result, the function $F_{\text{it}}_n$ has a value between zero and two. A higher fitness function value indicates more competence, higher centrality, and more energy for the node. As a result, the node can have more children. In order to calculate the maximum number of children allowed for each

| $ID_n$ | $F_{\text{it}}_n$ | $L_n$ |
|-------|------------------|-------|

**Fig. 3** Overview of the network in the first step of the first phase of the proposed method

**Fig. 4** Fitness packet structure
node, the $Fit_n$ value is divided by a fixed number of 0.4. Each parent node calculates the maximum number of children, according to (5).

$$CN_n = \frac{Fit_n}{0.4}$$ \hspace{1cm} (5)

- **Step 4: Sending a BuildStar packet**

The parent node (which is the cluster head in the first round) sends BuildStar packet containing its ID, its fitness function value, and its level number to its neighbor nodes; the initial level value for the cluster head is zero, as shown in Fig. 4. The overview of the network at this stage is shown in Fig. 5.

- **Step 5: Joining the parent**

Each node which has not joined the star structure (for which no level is specified) and receives the BuildStar packet, calculates a unique shared key for the hop-by-hop encryption of its messages for subsequent communication with its parent according to (6).

$$Key_n = \left( \frac{E_{\text{remaining}}}{20} \right) + (LB_n \% 5 + r)$$ \hspace{1cm} (6)

$E_{\text{remaining}}$ is the node’s remaining energy in terms of percentage, and it is a number between 0 and 100. The result of the expression $\left( \frac{E_{\text{remaining}}}{20} \right)$ is an integer between 0 and 5. $LB_n$ is the least valuable byte of the node ID, so the result of the expression $(LB_n \% 5 + r)$ is also an integer between 0 and 4. $r$ is a random integer between 3 and 5 selected by the node. Thus, the calculated key will be numerically between 3 and 14.

If the key is smaller than 3, security will be reduced in the encryption method, which will be described in the next sections; and if the key is larger than 14,
short messages may not be encrypted. The node then sends a Join packet to the sender, as shown in Fig. 6.

- **Step 6: Selecting the children**
  After receiving the Join packet from its neighbors, the BuildStar sender node counts their number and compares them to the maximum number of children it calculates in the third step. If this number is less than or equals the maximum number of children, it selects all nodes as children. Otherwise, from the answers received, it chooses those nodes as children which possess the highest amount of fitness function. It then stores its children’s information based on the received packets in the children’s information table, similar to Table 4.

- **Step 7: Assigning the level of the children**
  The parent node (which is the cluster head in the first round) sends an AssignedLevel packet to its children, as shown in Fig. 7, containing its ID and child level equaling one level more than its level. The children selected at this stage are considered as parents in the next round.

  If a child receives multiple AssignedLevel packets, it will join the parent with the highest amount of fitness function and send a LeaveChild packet containing its ID to the other parents.

- **Step 8: Checking the repetition condition**
  Steps four through seven are repeated by the selected children and continue until all nodes have a stored \( L_n \) value.

  An overview of the network after determining the network’s final level is shown in Fig. 8.

  The pseudo-code and flowchart of the first phase of the proposed method are shown in Fig. 9 and algorithm 1, respectively. The complexity of constructing a star in the first phase is \( O(n) \), where \( n \) is the number of nodes.

### 3.2.2 Phase II: Intra-Cluster Communication

In this step, we use multi-hop encryption for each cluster’s nodes to establish secure communication within the cluster. Each node has unique private keys with each of its children so that in case of an attack or disclosure, only that connection is compromised. The details of calculating and sharing these keys were explained in detail in the first phase of the proposed method. This phase describes how to use the keys to encrypt the data. The keys are also updated at the end of each connection to keep the keys secure. The advantage of

---

**Table 4  Children’s information table**

| Field  | Content                      |
|--------|------------------------------|
| \( ID_n \) | Identifier of node \( n \) |
| \( Key_n \) | Key of node \( n \)          |

**Fig. 7 Assigned Level packet structure**

| \( ID_n \) | \( L_n + 1 \) |

---

**Algorithm 1 First phase of the proposed method**

1. Initialize: (\( n \): number of nodes, \( L \): level of node, \( CN \): maximum number of children)
2. for \( i = 1; i \leq n; i++ \) do
3. \( Node_i \) sends the Hello packet for its neighbors
4. end for
5. for \( i = 1; i \leq n; i++ \) do
6. \( Node_i \) stores the information of neighbors in neighbor’s table
7. \( K \) is defined as the number of neighbors based on received Hello packets
8. \( Node_i \) calculates its fitness
9. \( Node_i \) calculates the maximum number of its children (\( CN_i \))
10. end for
11. if \( Node_i \) is a cluster head then
12. \( L = 0 \)
13. \( Node_i \) sends BuildStar packet to its neighbors
14. else
15. \( Node_i \) sends BuildStar packet to its neighbors
16. end if
17. for \( i = 1; i \leq n; i++ \) do
18. if \( Node_i \) receives BuildStar packet and an \( L \) value is assigned then
19. \( Node_i \) calculates \( Key_i \)
20. \( Node_i \) sends Join packet
21. if the total number of packets received by the parent is \( \geq CN_i \) then
22. Selecting the nodes which have the highest fitness functions as children
23. else
24. Selecting all the nodes as children
25. end if
26. The parent sends AssignedLevel packet to its children
27. if Children \( Node_i \) receives more than one AssignedLevel packet then
28. Children \( Node_i \) selects parent \( Node_j \) which has the best fitness as its parent
29. Children \( Node_i \) sends LeaveChild packet to other parents
30. end if
31. The parent stores its children’s information
32. Children that are selected in this round are parents in the next round
33. if Every \( Node_i \) has an \( L \) value then
34. Break
35. else
36. Go to step (15)
37. end if
38. end if
39. end for
this method is that if at some point, the communication key is revealed for any reason, the continuation of communication remains secure.

- **Step 1: Sensing, encrypting, and sending data from cluster member nodes to the cluster head**

  First, the nodes (the leaf nodes of the star structure in the first round) collect the environment’s required data. Using their shared key with their parent, they encrypt the data using the fence rail cipher and send it to their parent. Each node also recalculates its new key for use in subsequent communications according to (6) and sends it along with the encrypted data. Figure 10 shows the packet sent by the nodes to the cluster head, which contains the node ID, the encrypted data of the node \( (Data_n) \), and the new node key \( (Key_{n_{new}}) \).

- **Step 2: Collecting and sending data by the parent**

  After receiving encrypted data from its children, the parent decrypts it with each of their children’s current shared key. It also updates the node keys according to the newly received keys in the children’s information table. If the parent level is zero, it means that the data has reached the cluster head. Otherwise, the parent aggregates the sensed data with the data received from their children, encrypts it with the child’s shared key, and sends the encrypted data and the newly calculated key to its parent.

- **Step 3: Collecting and sending data by the parent**

  This process is repeated until the parent node level is zero, indicating that the cluster head received the packet.

For example, it is assumed that in a clustered network such as Fig. 11, the first node intends to send a message to its cluster head.

The encryption method is fully described below and according to the algorithm 2. The complexity of encrypting a message with the rail fence cipher method is \( O(key \times Text.Length) \), where \( key \) is the
encryption key and $\text{Text.Length}$ is the length of the plain text.

The plain text of a message that a node intends to send equals:

\[
\text{ABCDEFGHIJKLMNOPQRST}
\]

The plain text is written as a fence in six lines, as shown in Fig. 12.

Text encrypted with the key $Key_1$ equals:
AKBJLTCIMSDHNREGOQFP

The complexity of decrypting a with the rail fence cipher method is $O(key \times \text{Cipher.Length})$, where $key$ is the encryption key and $\text{Cipher.Length}$ is the length of the cipher text.

**Algorithm 2** Pseudo-code of the rail fence encryption.

1: Initialization: (DirDown: moving down, Row: number of rows, col: number of columns)
2: for (i=0; i < key; i++) do
3: for (j=0; j < Text.Length; j++) do
4:   Rail [i][j] = Null
5: end for
6: end for
7: DirDown = False
8: Row = 0
9: Col = 0
10: for (i=0; i < Text.Length; i++) do
11:   if (Row==0 || Row == key-1) then
12:     DirDown = DirDown
13: else
14:   Rail [Row][Col++] = Text[i]
15: end if
16: if (DirDown == True) then
17:   Row ++
18: else
19:   Row --
20: end if
21: end for
22: for (j=0; j < Text.Length; j++) do
23:   if (Rail[i][j] != Null) then
24:     Result.PushBack(Rail [i][j])
25: end for
26: end for

\[
\begin{align*}
\text{Key}_1 &= \left\lfloor \frac{E_{\text{remaining}}}{20} \right\rfloor + \left\lfloor \frac{LB_1}{5} \right\rfloor + r \\
&= 17/20 + 12/5 + 4 \\
&= 0 + 2 + 4 \\
&= 6
\end{align*}
\]

**Algorithm 3** Pseudo-code of the rail fence decryption.

1: Initialization: (DirDown: moving down, Row: number of rows, col: number of columns, Index: counter )
2: for (i=0; i < key; i++) do
3: for (j=0; j < cipher.Length(); j++) do
4:   Rail [i][j] = Null
5: end for
6: end for
7: DirDown = False
8: Row = 0
9: Col = 0
10: for (i=0; i < Cipher.Length(); i++) do
11: if (Row == 0) then
12:   DirDown = True
13: end if
14: if (Row == key -1) then
15:   DirDown = False
16: end if
17: if (DirDown == True) then
18:   Row ++
19: else
20:   Row --
21: end if
22: end for
23: Index = 0
24: for (i=0; i < key; i++) do
25: for (j=0; j < Cipher.Length(); j++) do
26:   if (Rail[i][j] == 'x' && Index < Cipher.Length()) then
27:     Result.PushBack(Rail [i][j])
28: end if
29: end for
30: end for
31: Row = 0
32: Col = 0
33: for (i=0; i < Cipher.Length(); i++) do
34: if (Row == 0) then
35:   DirDown = True
36: end if
37: if (Row == key-1) then
38:   DirDown = False
39: end if
40: if (Rail[Row][Col] != 'x') then
41: Result.PushBack(Rail [Row][Col++])
42: end if
43: if (DirDown == True) then
44: Row ++
45: else
46: Row --
47: end if
48: end for
An Authentication-Based Secure Data Aggregation Method in Internet of Things

Fig. 9 Flowchart of the first phase of the proposed method
The second node also has the remaining energy of 32% and an ID of 162.168.10.10. The randomly selected number of node also equals 3. The key, according to (6), equals:

\[
\text{Key}_1 = \left( \frac{E_{\text{remaining}}}{20} \right) + (LB_1 \%5) + r = \left( \frac{32}{20} \right) + (10\%5) + 3 = 1 + 0 + 3 = 4
\]

The second node aggregates the data received from the first node and its sensed data, then encrypts it with key Key₂ and sends it to the fourth node. The fourth node, after receiving its children’s data (the first and second nodes), decrypts them with the common keys Key₂ and Key₃, respectively. It then aggregates its children’s data with its data, encrypts, and sends it to the cluster head with its shared key. The flowchart of the proposed intra-cluster communication algorithm is shown in Fig. 13.

The Pseudo-code of the second phase of the proposed method is shown in algorithm 4. The complexity of sending a message within the cluster is \(O(n \times \text{key} \times \text{Text.Length})\), where \(n\) is the number of nodes, \(\text{key}\) is the encryption key, \(\text{Cipher.Length}\) is the length of the cipher text and \(\text{Text.Length}\) is the length of the plain text.

![Fig. 10](image1.png) Encrypted data message structure

![Fig. 11](image2.png) An example of a cluster transmitting a message from the first node to a cluster head
3.2.3 Phase III: Inter-Cluster Communication

At the end of the second phase, all nodes have delivered their data to the cluster heads. Now, this data must be sent to the base station. A standard routing algorithm specifies the data transfer path from the cluster heads to the base station. In this phase, using an authentication protocol, the cluster heads’ authenticity is ensured to prevent malicious nodes in the network and improve network security. The procedure is that mutual authentication is performed between the first and second cluster heads. A one-way version of the protocol is performed between the other cluster heads to reduce the network’s computational overhead. Because one of the cluster heads has always been authenticated in the previous hop, there is no need to recheck. The general flowchart of the stages of the third phase of the proposed method is shown in Fig. 14.

Wang et al. [40] presents an authentication method that devices must be registered with a centralized server to create secure cookie data. In this method, the cookie data is used to agree on a session to communicate between the current device and the server. In this method, the author shows the lack of security in the previous versions of this protocol [13] and [5], and presented potential improvements. Also, the authors demonstrate that their protocol was secure against several attacks. Notice that this protocol is based on ECC and only uses simple operations. It also requires storing low data on the end nodes. Due to those factors, this method is suitable for the IoT network.

However, it should be noted that in the proposed protocol, the device ID is explicitly sent in the authentication step, which exposes the device to a tracking attack. In this section, we present a modified version of this scheme. We prevent the tracking attack by adding a pseudonym to each cluster head and sending it instead of sending the ID directly and explicitly. Also, by adding an update step at the end of the protocol, we protect the parameters sent directly from unauthorized nodes.

**Algorithm 4 Second phase of the proposed method.**

1. Initialization: \((n: \text{number of nodes})\)
2. for \((i = 1; i \leq n; i++)\) do
3. \(Node_i\) senses the data of its environment
4. \(Node_i\) encrypts sensed data and sends it to its parent
5. The parent decrypts received data with the shared key
6. The parent updates its children’s table
7. if The parent’s level == 0 then
8. Intra-cluster communication is finished
9. else
10. The parent aggregates its children’s data and its sensed data
11. The parent encrypts the aggregated data
12. The parent sends encrypted data for the next hop
13. end if
14. end for

![Fig. 12 Encrypting data with the key Key1](image)
The following is the proposed authentication protocol.

Table 5 shows the notations used in the proposed protocol.

- **Step 1: Registration**
  At this step, each cluster head is assigned a unique identifier. Also, information is stored in each cluster head and database on the base station, and it is used for the authentication step. Note that this information is updated again at the beginning of each clustering. The information at this stage is sent through a secure channel [9, 41]. The initialization steps are shown in Fig. 15 Cluster head $CH_i$ selects a unique ID $ID_i$ and pseudonym and sends it to the cluster head $CH_j$.

Cluster head $CH_i$ generates a random number $R_i$ of length $l_H$-bit and computes $CK, CK', T_i, A_i, A'_i$ as follow, and stores $\{A'_i, T_i, ID_i, x_i, EXP_Time\}$ in the database. It then sends these values through a private channel to the cluster head $CH_j$.

After receiving the message, the cluster head $CH_j$ stores $CK'$, $ID_i$, $x_{i_{old}}$ and $x_{i_{new}}$ in its memory. Note that $x_{i_{old}}$ at first is zero and $x_{i_{new}}$ is the cluster head’s pseudonym. These values are required in the next step of authentication taking place between the two cluster heads.
An Authentication-Based Secure Data Aggregation Method in Internet of Things

**Table 5** Notation used in the proposed authentication protocol

| Notation | Description |
|----------|-------------|
| $CH_i, CH_j$ | Two cluster head registered in the system |
| $x_{new}, x_{old}$ | The cluster head’s current and immediate prior pseudonym |
| $ID_i$ | The ID number of the cluster head $CH_i$ |
| $H$ | A cryptographic hash function with an output of 128-bit $H = \{0,1\}^{128}$ |
| $h$ | A cryptographic hash function with an output of 160-bit $h = \{0,1\}^{160}$ |
| $G$ | A generator of the group $G$ - a public parameter |
| EXP_Time | The expiry time of a particular device. |
| $X$ | Cluster head’s secret key. |
| SK | A session key output at the end of a session. |
| $\|$ | Concatenation operation |
| $\times$ | Linear multiplication with a point on the elliptic curve |
| $\oplus$ | XOR operation |

\[
CK = h(R_i \parallel X \parallel EXP\_Time \parallel ID_i) \quad (7) \\
CK' = CK \times G \quad (8) \\
T_i = R_i \oplus H(X) \quad (9) \\
A_i = h(R_i \oplus H(X) \parallel CK') \quad (10)
\]

\[
A_i' = A_i \times G \quad (11)
\]

- **Step 2: Authentication between cluster heads**

At this step, the first and second cluster heads authenticate each other. The authentication protocol is implemented mutually at this step so that both cluster heads can ensure each other’s authenticity. One-way authentication is then performed between the other cluster heads in the communi-

![Flowchart of the third phase of the proposed method](image-url)
cation path specified by the standard routing algorithm. The description of these steps is as follows:

When the first cluster head desires to be authenticated by the second cluster head, it sends handshake information "Hello" to the second cluster head. After receiving the handshake information, the second cluster head sends its pseudonym $x_i$ to the first cluster head. After receiving the pseudonym of the second cluster head, if the corresponding new pseudonym cannot be found in the backend database, the first cluster head will regenerate handshake information "Hello" to the second cluster head. By this time, the second cluster head will set $x_i=x_{i,old}$ and retransmits it to the first cluster head. If the corresponding old pseudonym cannot be found in the backend database again, the first cluster head is declared illegal, and the session ends. Otherwise, the first cluster head is authenticated. The second cluster head then selects a random number $N_2$ with a $l_h$-bit length and computes $P_3$ and $P_4$ according to (16) and (17), respectively.

$$P_2 = N_1 \times G \text{ (12)}$$

$$P_2 = H(P_1 \parallel N_1 \times CK') \text{ (13)}$$

Finally, the second cluster head sends $\{P_1, P_2\}$ to the first cluster head for authentication. After receiving the values, the first cluster head should reconstruct $CK$ using the obtained corresponding $\{T_i, X, EXP\_Time\}$ from the database. Then it computes $P_2'$ according to (14) and compares it with the received $P_2$ according to (15). If these two values not equal, the first cluster head is declared illegal, and the session ends. Otherwise, the first cluster head is authenticated. The second cluster head then selects a random number $N_2$ with a $l_h$-bit length and computes $P_3$ and $P_4$ according to (16) and (17), respectively.

$$P_2' = H(P_1 \parallel CK \times P_1) \text{ (14)}$$

$$P_2' = P_2 \text{ (15)}$$

$$P_3 = N_2 \times G \text{ (16)}$$

$$P_4 = H(P_2' \parallel N_2 \times A_i') \text{ (17)}$$
second cluster head, the first cluster head computes $A_i$ according to (18), using $T_i$ and $CK'$, and then computes $P'_4$ according to (19) and compares $P'_4$ with the received $P_4$ according to (20). It ends the session if the two values not equal. Otherwise, it computes $V_i$ and the session key $SK$ according to (21) and (22), respectively, and sends $V_i$ for the second cluster head.

\[ A_i = h(T_i \parallel CK') \quad (18) \]

\[ P'_4 = H(P_2 \parallel A_i \times P_3) \quad (19) \]

\[ P'_4 \overset{?}{=} P_4 \quad (20) \]

\[ V_i = H(P'_4 \parallel N_1 \times P_3) \quad (21) \]

\[ SK = H(P_3 \parallel N_1 \times P_3) \quad (22) \]

After receiving the $V_i$, the second cluster head computes $V'_i$ according to (23) and checks its validity according to (24) and, if confirmed, it computes the session key for the next communications according to (25).

\[ V'_i = H(P_4 \parallel N_2 \times P_1) \quad (23) \]

\[ V'_i \overset{?}{=} V_i \quad (24) \]

\[ SK = H(P_3 \parallel N_2 \times P_1) \quad (25) \]

The proposed two-way authentication protocol steps between the first and second cluster heads are shown in Fig. 16. The proposed mutual authentication protocol steps between the other cluster heads are also shown in Fig. 17.

- **Step 3: Updating**

  Provided that all steps of the authentication phase are accurately taken, two cluster heads update their secret key $X$ and the pseudonym of the cluster head $x_i$. Since $x_i$ is explicitly transmitted during the authentication steps; it is required to take updating steps to prevent unauthorized use. Updating the secret key and pseudonym by each cluster head is as follows:

\[ x_{new} = H(P_1) \oplus x_{old} \oplus X_{old} \quad (26) \]

\[ X_{new} = H(P_2) \oplus 2X_{old} \quad (27) \]

\[ x_{old} = x_{new} \quad (28) \]

\[ X_{old} = X_{new} \quad (29) \]

4 Simulation

The proposed method is simulated using NS2 software. The simulation results are compared with LSDAR [11], ECH-DUAL [1], and ZBRP [15] methods. To achieve acceptable results, the simulation conditions are considered the same for all three algorithms. The simulation parameters are shown in Table 6. The simulation results are investigated to evaluate the proposed method’s performance in terms of consumed energy, flexibility, package delivery rate, end-to-end latency, and the number of alive nodes, which are described below.

4.1 Energy Consumption

Figure 18 shows the average energy consumption of the proposed method compared to other LSDAR, ECH-DUAL, and ZBRP methods concerning the network’s number of nodes. As shown in Fig. 18, the proposed method has a lower average energy consumption than other methods. The network structure is clustered. Inside the clusters, a star structure is used. Cluster member nodes send their data to the cluster head based on a star structure. The influential parameters in constructing the structure within the cluster are the remaining energy and the distance. Therefore, the mechanism of star formation is such that the nodes of higher levels, i.e., nodes with a higher workload and more energy, have less distance to the cluster head; this causes a proper distribution of energy consumption among the nodes of the cluster member. Also, due to data transmission at shorter distances, nodes naturally have lower energy consumption and longer life.
On the other hand, determining the number of children based on the fitness function causes the parent nodes to have an appropriate number of children. Thus, energy consumption is distributed among them. An intra-cluster star structure also provides a good infrastructure for transferring information from nodes to cluster heads without discovering a path. Also, the aggregation of data in each parent node in the proposed method reduces the network’s data. Reducing the amount of sent data will also reduce energy consumption. In inter-cluster communication, multi-hop communication between cluster heads results in less energy consumption in clusters to send data. Due to these conditions, the need to resend packets is reduced and as a result, no additional energy is
consumed. The energy consumption of network nodes and the average energy consumption of the entire network increases with an increase in nodes. Still, the trend of increasing energy in the proposed method compared to other methods has been significantly improved.

4.2 End-to-end Delay

The end-to-end delay represents the length of time a packet is transmitted over the network from the source node to the destination. It includes all possible delays during route discovery, packet queue, retransmission delays, packet release, and packet transmission time. Figure 19 shows the end-to-end delay in the proposed method.
compared to other methods. In the proposed method, due to the star structure, the relationship between the nodes within the cluster is predetermined, and there is no need for a path discovery phase. Therefore, no time is spent to discover the path and the data is sent from a predefined route. Due to the parameters of energy and distance and the construction of the star structure within the cluster, a regular and relatively stable structure is constructed to transmit information, which reduces delay. Due to the fact that in the star structure we have considered a shorter distance to the cluster head, the data is transferred to the cluster head with less number of hops, thus we have less delay. However, since lightweight encryption is used in sending data within the cluster and the data encryption is done hop by hop, and on the other hand, there is a step-wise authentication between the cluster heads; as a result, the delay of the proposed method increases compared to LSDAR method. However, compared to other methods, the end-to-end delay in the network is lower.

4.3 Flexibility

The flexibility of the proposed methods compared to other methods, is shown in Fig. 20. In the proposed method, due to the using the shared key locally between the parent and the children, if the attacker succeeds in discovering the node key, it only has access to the two nodes’ information. It is not able to access information from other nodes. Given that the encryption and decryption of data are done hop by hop between the child and the parent, the password’s discovery will not have much effect on changing the data, because this key is only valid for one hop and will be changed in the next hop. In communication,
the data sent in each cluster is encrypted in each hop and then sent to the next hop. In the communication between the cluster heads, the first level of mutual authentication is done, and before sending the data, the identities of both cluster heads are verified for communication. Therefore, if the cluster nodes have delivered the data to a captured or fake cluster head, since each cluster head performs authentication operations with a lower level cluster head, the lower level cluster head will not receive the information. On the other hand, due to the hop-by-hop authentication, the intermediate cluster heads’ identities are also checked and verified before sending the data. Therefore, it can be said that captured cluster heads in the authentication process are not confirmed and their data is not accepted. As a result, as shown in Fig. 20, the proposed method has more flexibility in the presence of destructive nodes than other methods.

4.4 Packet Delivery Rate

Packet delivery rate is defined as the ratio of the number of packets successfully received at the destination to the number of packets sent by network nodes. The ratio of the packet delivery to the number of network nodes in the proposed method is shown in Fig. 21. The ratio of packet delivery rate to the packet sending rate compared to other methods is shown in Fig. 22. In the proposed method, the cluster’s star structure is based on the remaining energy and the average distance. The use of the mentioned parameters in determining the parent node results in forming a more stable structure within the cluster. Sending packets via stable routes reduces the rate of packet loss. Also, the use of a star structure results in the sending of data with low number of hops. As a result, the packet lost probability in data transmission is reduced. In the inter-cluster phase, hop-by-hop authentication prevents sending packets to fake nodes; this prevents packets from being lost. Therefore, as shown in Figs. 21 and 22, in both cases, in the proposed method, the packet delivery rate is at a higher level.

4.5 Number of Alive Nodes

Figure 23 shows the number of alive nodes in the proposed method compared to other methods. As shown in Fig. 23, the proposed method consumes less energy than other methods, which prolongs the network life, and the network nodes will serve in a larger number of rounds. As shown in Fig. 18, the proposed method consumes less energy than other methods, which prolongs the network’s life and the network nodes serve in a larger number of rounds. Also, the distribution of energy consumption in the network and the selection of nodes with higher energy at higher levels of the star structure with higher energy consumption in the network causes the last node’s death to occur after more time. Also, increasing the packet delivery rate, as shown in Fig. 21, reduces the need for retransmission in the network, which also reduces the energy consumption of network nodes. Therefore, in general, it can be said that the performance of the proposed method is better than other methods in terms of the number of alive nodes and creates a better load balance in the network.
5 Security Analysis of the Proposed Authentication Protocol

In the third phase of the proposed method, an improvement of the authentication protocol [40] is presented. In the authentication protocol, the node ID is sent directly, which puts the protocol at risk of being tracked. There is also no phase to update the parameters sent instantly, which also puts the protocol at risk of desynchronizing attack. In the proposed method,
these defects have been corrected by adding a pseudonym and an update phase. Also, we used this protocol to authenticate hop-by-hop cluster heads, which is mutual authentication between the cluster heads of the first and second hops of the protocol, and to reduce the computational overhead; the protocol is executed one-way in subsequent hops. Since one of the cluster heads’ identity has already been proven right in the previous phase, there is no need to reconsider.

5.1 Mutual Authentication

In the proposed method, inter-cluster communication has been used through two authentication methods in order to authenticate the first-level cluster heads and intermediate cluster heads to the base station. In the proposed mutual authentication method, the first cluster head sends its pseudonym to the second cluster head, and according to the received pseudonym, if it exists in the second cluster head database, the first cluster head authentication operation is followed. For this purpose, the value of the expression $P_2$ received from the first cluster head is compared with the calculated local value $P'_2$, which is described below.

$$P_2 = H(P_1 \parallel N_1 \times CK') = H(P_1 \parallel N_1 \times CK \times G)$$

On the other hand, the first cluster head confirms or denies the identity of the second cluster head by comparing the value of the expression $P_4$ received from the second cluster head with the calculated local value $P'_4$, which is described below.

$$P_4 = H(P_2 \parallel N_2 \times A') = H(P_2 \parallel N_2 \times A \times G)$$

$$= H(P_2 \parallel A \times P_3) = H(P_2 \parallel A \times P_3) = P'_4$$

5.2 Resistance to the Tracing Attack

In the proposed method, in none of the authentication operations, the node ID is sent directly. In the authentication steps, the pseudonym is used to identify the cluster heads, as shown in (18) and (19). The pseudonym is updated to prevent tracking attacks after each authentication operation under (24) and (25). As a result, the proposed method can resist the tracking attack.

5.3 Resistance to the Synchronization Attack

In the proposed method, the secret key and pseudonym are updated in each round of authentication operations. If the attacker blocks the transmission of information to one of the cluster heads, with the intention of de-synchronization attack, considering the update operation is performed after the authentication operation is done successfully and the session key is established between parties, the attacker cannot block the synchronization of the secret key or pseudonym by breaking the connection between cluster heads. Also, if the messages transmission interrupt by the malicious attacker or various natural factors and result in desynchronization between the cluster heads, since the protocol keeps the old and new secret key and pseudonym ($X_{old}$, $X_{old}$, $x_{old}$, $x_{new}$), it can return to the synchronization state.

5.4 Resistance to the Replay Attack

In the replay attack, the attacker intends to introduce itself as an authorized cluster head by cutting off the communication between the meeting parties, using previous messages. In mutual one-way authentication operations in Figs. 16 and 17, each party to the session generates a random number ($N_1, N_2$), which are used in the authentication process. Random numbers change in each round of the authentication algorithm. Since both random numbers are used to authenticate the meeting parties, the attacker will not be able to use the previous messages in the new session, and the sent messages will be recognized as invalid.

5.5 Resistance to the Impersonation Attack

In the impersonation attack, the attacker uses previously exchanged information and the access to cluster head information to impersonate the cluster head node. In the proposed method, each cluster head is assigned an ID, pseudonym, and $CK'$ value to perform the authentication process. During the authentication process, the cluster heads communicate via pseudonym, and the cluster head ID and the $CK'$ value are not sent directly. The value of the node pseudonym is also updated at the end of the authentication operation. On the other hand, in the proposed method, the ID, pseudonym, and CK value are unique for each cluster head and are randomly selected. So, the attacker will not be able to calculate and access the information of other
cluster heads by accessing the information of a number of them. As a result, the proposed method can resist the impersonation attack.

5.6 Resistance to the Man-in-the-Middle Attack

In the man-in-the-middle attack, an attacker settles between cluster head nodes and eavesdrops on any information sent between them. This attack can be successful if the attacker is located between the cluster heads of the network and before sending the packet to the other cluster head node, it is able to make changes in the packet and then send it. The attack may be carried out bilaterally and continue without the parties noticing the attacker. In the proposed method, in the first step of the authentication process, the messages $x_i, P_1,$ and $P_2$ are sent. If the attacker changes the value of $x_i$, the session will end due to the invalid pseudonym. Due to the use of $P_1$ in the message $P_2$, changing the value of $P_1$ indicates that the message is invalid and the session ends. The change in the value of $P_2$ is also evident according to the correct examination of the value of $P_2$ and $P'_{2}$. In the second step of the proposed method, $T_i, P_3,$ and $P_4$ are sent. The value of $T_i$ is effective in calculating $A_i$, and the values of $A_i$ and $P_3$ are used to check the correctness of the received $P_4$ with $P'_{4}$. The other node detects the change in each of these values, and the received values are recognized as incorrect. As a result, the session ends, and the attacker is revealed. The change in the value of $V_i$ is also evident since this value is calculated using the local cluster head information. As a result, the proposed method can resist the man-in-the-middle attack.

5.7 Computational Cost

In the proposed method, the computational cost includes the cost that is performed in exchange for calculating the elliptic curve and the hash operation. Other operations such as concatenation and xor are not considered compared to other operations due to their short computation time, and their computation time is ignored. The number of elliptic curve multiplication operations in the proposed mutual authentication method is nine times, and the number of hash multiplication operations is nine times in both cluster heads (5 elliptic curve multiplication operations and five hash operations in the first cluster head and four elliptic curve multiplication operations and four hash operations in the second cluster head). In the one-way authentication method, the number of elliptic curve multiplication operations is three times, and the number of times the hash function is used is two times per cluster head. In the Tmote Sky node (8 Hz) as one of the most common sensor nodes, each elliptic curve multiplication operation takes 1.76 seconds, and a hash operation takes 0.011 seconds. Therefore, the mutual authentication method generally takes 15.9 seconds, and the one-way authentication method takes 10.6 seconds.

6 Conclusion

One of the major challenges in the Internet of Things is security. With the increase in the number and variety of devices connected to the Internet of Things, its security is severely threatened. As a result, we need to maintain security and privacy in these networks. In this paper, a secure routing method is proposed to prevent the intrusion of malicious nodes. The proposed method consists of three phases: star structure construction and key distribution, intra-cluster communication, and inter-cluster communication. In the first phase, a star structure is constructed between each cluster’s nodes, which provides a regular structure for the transfer of information within the cluster and keys to secure the information exchanged in subsequent communications. In the first phase, a star structure is constructed between each cluster’s nodes, which provides a regular structure for the transfer of information within the cluster and keys to secure the information exchanged in subsequent communications. In the second phase, to improve security, the data is encrypted hop by hop with different keys and sent to the cluster heads in a few hops. Finally, each cluster head is authenticated in the third phase before sending its information to prevent malicious nodes from infiltrating. The proposed method is also simulated using NS2 software. The results showed that the proposed method had improved energy consumption parameters, end-to-end delay, flexibility, packet delivery rate, and the number of alive nodes compared to other methods.

7 Future Work

The following are suggestions for future work in this article.

- Use another encryption that provides adequate security with low overhead
• Use fuzzy logic to validate cluster head nodes instead of authentication
• Extend the proposed security method to other networks, including mobile networks

Data Availability Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Alagirisamy, M., Chow, C.O.: An energy based cluster head selection unequal clustering algorithm with dual sink (ech-dual) for continuous monitoring applications in wireless sensor networks. Clust. Comput. 21(1), 91–103 (2018)
2. Alzahrani, B.A., Chaudhry, S.A., Barnawi, A., Xiao, W., Chen, M., Al-Barakati, A.: Ilas-iot: an improved and lightweight authentication scheme for iot deployment. J. Ambient. Intell. Humaniz. Comput., pp. 1–13 (2020)
3. Basak, S., Acharya, T.: On energy efficient secure routing in multi-hop underlay d2d communications for iot applications. Ad Hoc Netw. 108, 102275 (2020)
4. Boulaalam, A.: Internet of things: new classification model of intelligence. J. Ambient. Intell. Humaniz. Comput. 10(7), 2731–2744 (2019)
5. Chang, C.C., Wu, H.L., Sun, C.Y.: Notes on "secure authentication scheme for iot and cloud servers". Pervasive and Mobile Computing 38, 275–278 (2017)
6. Dang, T.K., Pham, C.D., Nguyen, T.L.: A pragmatic elliptic curve cryptography-based extension for energy-efficient device-to-device communications in smart cities. Sustainable Cities and Society 56, 102097 (2020)
7. Das, M.L., Kumar, P., Martin, A.: Secure and privacy-preserving rfid authentication scheme for internet of things applications. Wirel. Pers. Commun. 110(1), 339–353 (2020)
8. Dinarvand, N., Barati, H.: An efficient and secure rfid authentication protocol using elliptic curve cryptography. Wirel. Netw. 25(1), 415–428 (2019)
9. Ghani, A., Mansoor, K., Mehmoond, S., Chaudhry, S.A., Rahman, A.U., Najmus Saqib, M.: Security and key management in iot-based wireless sensor networks: an authentication protocol using symmetric key. Int. J. Commun. Syst. 32(16), e4139 (2019)
10. Hameed, A., Alomary, A.: Security issues in iot: A survey. In: 2019 international conference on innovation and intelligence for informatics, computing, and technologies (3ICT), pp 1–5, IEEE (2019)
11. Haseeb, K., Islam, N., Saba, T., Rehman, A., Mehmood, Z.: Lsdar: A light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks. Sustainable Cities and Society 54, 101995 (2020)
12. Islam, M.J., Rahman, A., Kabir, S., Karim, M.R., Achajee, U.K., Nasir, M.K., Band, S.S., Sookhak, M., Wu, S.: Blockchain-sdn based energy-aware and distributed secure architecture for iots in smart cities. IEEE Internet of Things Journal (2021)
13. Kalra, S., Sood, S.K.: Secure authentication scheme for iot and cloud servers. Pervasive and Mobile Computing 24, 210–223 (2015)
14. Kertész, A., Pflanzner, T., Gyimóthy, T.: A mobile iot device simulator for iot-fog-cloud systems. J. Grid Comput. 17(3), 529–551 (2019)
15. Kumaramangalam, M.V., Adiyapatham, K., Kandasamy, C.: Zone-based routing protocol for wireless sensor networks. International Scholarly Research Notices 2014 (2014)
16. Latif, S.A., Wen, F.B.X., Iwendi, C., Li-li, F.W., Mohsin, S.M., Han, Z., Band, S.S.: Ai-empowered, blockchain and sdn integrated security architecture for iot network of cyber physical systems. Comput. Commun. 181, 274–283 (2022)
17. Li, X., Peng, J., Kumari, S., Wu, F., Karuppiah, M., Choo, K.K.R.: An enhanced 1-round authentication protocol for wireless body area networks with user anonymity. Comput. Elect. Eng. 61, 238–249 (2017)
18. Lin, J.W., Arul, J.M., Kao, J.T.: A bottom-up tree based storage approach for efficient iot data analytics in cloud systems. J. Comput. 19(1), 1–19 (2021)
19. Mathew, G., Gupta, A.K., Pant, M.: Timer and distance based routing protocol for continuous monitoring application in wsn. In: 2012 International Conference on Computing Sciences, pp. 332–337, IEEE (2012)
20. Mousavi, S.K., Ghaffari, A., Besharat, S., Afshari, H.: Improving the security of internet of things using cryptographic algorithms: a case of smart irrigation systems. Journal of Ambient Intelligence and Humanized Computing, pp. 1-19 (2020)
21. Naghibi, M., Barati, H.: Shsda: secure hybrid structure data aggregation method in wireless sensor networks. Journal of Ambient Intelligence and Humanized Computing, pp. 1–20 (2021)
22. Niu, X.: A secure and reliable transmission scheme for low loss high performance wireless communication system based on iot. Journal of Ambient Intelligence and Humanized Computing (2020)
23. Pahl, C., Ramachandran, M., Wills, G.: Intelligent management of cloud, iot and big data applications. Journal of Grid Computing (2019)
24. Panda, P.K., Chattopadhyay, S.: A secure mutual authentication protocol for iot environment. Journal of Reliable Intelligent Environments, pp. 1–16 (2020)
25. Qu, W.Q.: Cluster_head selection approach based on energy and distance. In: Proceedings of 2011 international conference on computer science and network technology, vol. 4, pp 2516–2519, IEEE (2011)
26. Rahman, A., Chakraborty, C., Anwar, A., Karim, M., Islam, M., Kundu, D., Rahman, Z., Band, S.S., et al.: Sdn–iot empowered intelligent framework for industry 4.0
applications during covid-19 pandemic. Cluster Computing, pp 1–18 (2021)

27. Rahman, A., Islam, M.J., Montieri, A., Nasir, M.K., Reza, M.M., Band, S.S., Pescape, A., Hasan, M., Sookhak, M., Mosavi, A.: Smartblock-sdn: an optimized blockchain-sdn framework for resource management in iot. IEEE Access 9, 28361–28376 (2021)

28. Rao, V., Prema, K.: A review on lightweight cryptography for internet-of-things based applications. Journal of Ambient Intelligence and Humanized Computing, pp. 1–23 (2020)

29. Sabry, S.S., Qarabash, N.A., Obaid, H.S.: The Road to the Internet of Things: a Survey. In: 2019 9Th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECOn), pp 290–296. IEEE (2019)

30. Sahoo, S.S., Mohanty, S., Majhi, B.: A secure three factor based authentication scheme for health care systems using iot enabled devices. Journal of Ambient Intelligence and Humanized Computing, pp. 1–16 (2020)

31. Saleem, A., Khan, A., Malik, S.U.R., Pervaiz, H., Malik, H., Alam, M., Jindal, A.: Fesda: Fog-enabled secure data aggregation in smart grid iot network. IEEE Internet of Things Journal (2019)

32. Sowjanya, K., Dasgupta, M., Ray, S.: An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems. Int. J. Inf. Secur. 19(1), 129–146 (2020)

33. Tang, W., Ren, J., Deng, K., Zhang, Y.: Secure data aggregation of lightweight e-healthcare iot devices with fair incentives. IEEE Internet of Things Journal 6(5), 8714–8726 (2019)

34. Thangaramya, K., Kulothungan, K., Logambigai, R., Selvi, M., Ganapathy, S., Kannan, A.: Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in iot. Comput. Netw. 151, 211–223 (2019)

35. Ullah, A., Said, G., Sher, M., Ning, H.: Fog-assisted secure healthcare data aggregation scheme in iot-enabled wsn. Peer-to-Peer Networking and Applications 13(1), 163–174 (2020)

36. Vaishnavi, S., Sethukarasi, T.: Sybilwatch: a novel approach to detect sybil attack in iot based smart health care. J. Ambient. Intell. Humaniz. Comput., pp. 1–15 (2020)

37. Van Oorschot, P.C., Menezes, A.J., Vanstone, S.A.: Handbook of applied cryptography. CRC press (1996)

38. Viswanathan, S., Kannan, A.: Elliptic key cryptography with beta gamma functions for secure routing in wireless sensor networks. Wirel. Netw 25(8), 4903–4914 (2019)

39. Wang, J., Yang, X., Ma, T., Wu, M., Kim, J.U.: An energy-efficient competitive clustering algorithm for wireless sensor networks using mobile sink. International Journal of Grid and Distributed Computing 5(4), 79–92 (2012)

40. Wang, K.H., Chen, C.M., Fang, W., Wu, T.Y.: A secure authentication scheme for internet of things. Pervasive and Mobile Computing 42, 15–26 (2017)

41. Wu, F., Xu, L., Kumari, S., Li, X.: A new and secure authentication scheme for wireless sensor networks with formal proof. Peer-to-Peer Networking and Applications 10(1), 16–30 (2017)

**Publisher’s Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.