

Privacy Protection Range Query Protocol for Multidimensional Data in Two-Tiered Wireless Sensor Networks

Jinyu Chen, Shuxi Chen, Liting Lei

School of Yongyou Digital Intelligence, Nantong Institute of Technology, Nantong, Jiangsu, China

Abstract: Range query is the main query method in two-tiered wireless sensor networks. In the process of query operation, privacy protection technology is needed to ensure the security and integrity of private data. Therefore, it is of great significance to study the range query problem with privacy protection capability in two-tiered wireless sensor networks. This paper analyzes the privacy security of multidimensional data and the communication energy consumption of range query, proposes an improved cross 0-1 encoding technology to optimize the comparison factor and realize data comparison in ciphertext state. An improved multidimensional encrypted constraint chain is proposed to verify the authenticity and integrity of the query results of each dimension. In the experimental part, the protocol in this paper was implemented using the development board of Cortex-M4 core, the development board of Cortex-A9 core and PC, and the security and effectiveness of the protocol were verified. Comparing the communication energy consumption of the proposed protocol with the Optimized HMAC Protocol in two aspects, the experimental results show that in the same experimental environment, the communication energy consumption of the proposed protocol is lower than that of the optimized HMAC protocol, indicating better performance.

Keywords: Two-tiered Wireless Sensor Networks; Privacy Protection Range Query; Cross 0-1 Encoding; Multidimensional Encrypted Constraint Chain.

1. Introduction

Two-tiered wireless sensor networks are a type of sensor network with a simpler network topology. The lower tier consists of a large number of sensor nodes with limited storage and computing capabilities, while the upper tier consists of a small number of storage nodes with larger storage capacity and stronger computing capability [1]. This architecture provides more stable communication links, and query processing is more efficient because the Sink node communicates only with storage nodes. However, security issues are also more prominent. As the intermediate layer, storage nodes not only store a large amount of sensing data but also respond to query instructions. Therefore, in two-tiered wireless sensor networks, storage nodes are the most vulnerable to attacks. It is thus of great practical significance to study and solve privacy-preserving data query problems in two-tiered wireless sensor networks.

Range query is a common data query operation and has broad application prospects in many fields, such as national defense, wildlife observation, and smart healthcare. For example, in the safety monitoring of large industrial parks, sensors can be used to monitor parameters such as temperature, humidity, and illumination within a specified range. The security objectives of range queries in wireless sensor networks mainly include two aspects. The first is to protect the privacy of sensed data and query ranges. The second is to ensure the authenticity and completeness of query results. To achieve these objectives, two major challenges must be addressed. One is how to identify the relevant data when both sensed data and query range values are encrypted. The other is how to verify the correctness and completeness of the query results. Existing studies mostly focus on one-dimensional data, and they still suffer from limitations in both security and communication energy consumption.

To address these issues, this paper proposes an energy-efficient privacy-preserving range query protocol for multidimensional data in two-tiered wireless sensor networks, named **PERQ-M (Privacy and Efficient Range Query for Multidimensional Data)**. By introducing cross 0-1 encoding technology and combining it with a prime-number fusion mechanism and the Diffie-Hellman key exchange protocol, the proposed scheme realizes encrypted storage of sensed data and secure query processing. Meanwhile, a multidimensional encrypted constraint chain is constructed to support integrity verification of query results. Finally, the privacy and security of the protocol are analyzed in detail, and the communication energy consumption of sensor nodes is compared with that of the Optimized HMAC scheme.

2. Related Work

Existing studies on secure range queries can be mainly divided into the following two categories according to the query technique used.

(1) Secure range queries based on bucket partitioning.

These methods rely on the same assumption: sensor nodes and the Sink node share the bucket partitioning strategy, that is, the mapping between bucket intervals and random labels is known only to sensor nodes and the Sink node, but unknown to storage nodes. The randomness of the labels guarantees the security of the bucket partitioning strategy. In the scheme proposed by Sheng and Li, symmetric encryption and hash operations are introduced on the basis of bucket partitioning, and bucket encoding is used to verify the completeness of query results [2]. However, since the Sheng and Li scheme needs to generate a bucket encoding for each empty bucket and transmit it to the storage node, the communication cost of sensor nodes increases rapidly in large-scale wireless sensor networks as the number of empty buckets grows. To reduce the communication cost of sensor

nodes, Shi J. et al. proposed an optimized method based on spatiotemporal cross-checking. Their scheme replaces the bucket encoding in the Sheng and Li scheme with a bitmap index, thereby reducing the communication cost of sensor nodes, although the communication cost between storage nodes and the Sink node increases [3].

(2) Secure range queries based on secure comparison.

The basic idea of these methods is that sensor nodes encrypt the sensed data and generate secure comparison codes that can be used to compare encrypted data. Chen and Liu proposed the **SafeQ** scheme based on a prefix membership verification coding mechanism, which enables ciphertext comparison between collected data and query ranges. In this way, SafeQ can determine whether encrypted data satisfy the query range without decrypting the data. To verify the completeness of query results, a neighborhood-chain mechanism is also introduced during data encryption, and Bloom filters are used to reduce the communication cost of secure comparison codes [4]. Yi Y. et al. proposed the **QuerySec** scheme, which uses order-preserving functionality to achieve encrypted comparison between collected data and query ranges. By embedding linked watermark information into encrypted data blocks, the scheme realizes integrity verification of query results [5].

Dai H. et al. proposed **CSRQ**, a privacy-preserving range query protocol based on 0–1 encoding. This protocol ensures the privacy of data, results, and range intervals through 0–1 encoding and hash-based message authentication techniques, but it lacks the ability to resist collusion attacks among sensor nodes [6]. Hu Q. proposed a multidimensional data range query method based on the compressed HMAC algorithm, referred to as the **Optimized HMAC Protocol**. This scheme can effectively protect the privacy of sensed data and can also verify the integrity of query results through a constraint-chain mechanism, but there is still room for improvement in terms of energy consumption [7].

A comparison of the above two types of secure range query

methods reveals the following. First, in terms of security, the first category mainly depends on the bucket partitioning strategy, whereas the second category depends on the complexity of the secure comparison function. Second, assuming the same encryption method is used, the first category produces less encrypted data than the second category. Third, in terms of sensor-node communication cost, which directly affects network lifetime, the first category depends on the granularity of bucket partitioning and the distribution of collected data among buckets, whereas the second category depends on the amount of sensed data and the corresponding secure comparison codes.

3. Model and Problem Description

3.1. Network Model

This paper adopts a two-tier wireless sensor network model, whose network topology is shown in Fig. 1 below. The entire two-tier wireless sensor network is divided into multiple query cells (grids), and each cell consists of several sensor nodes and one storage node, denoted as $G = \{M, (s_1, s_2, \dots, s_n)\}$. Each sensor node is equipped with sensors of different attributes for acquiring multidimensional data. The storage node is responsible for storing the data uploaded by sensor nodes within the cell and for responding to query instructions from the Sink node. The Sink node, in turn, is responsible for responding to user query requests. In terms of network connectivity, Wi-Fi is used for communication between sensor nodes and storage nodes, while wired Ethernet is used for communication between storage nodes and the Sink node. The communication link between sensor nodes and storage nodes is mainly used for uploading encrypted sensed data, whereas the communication link between storage nodes and the Sink node is mainly used for the entire range query process.

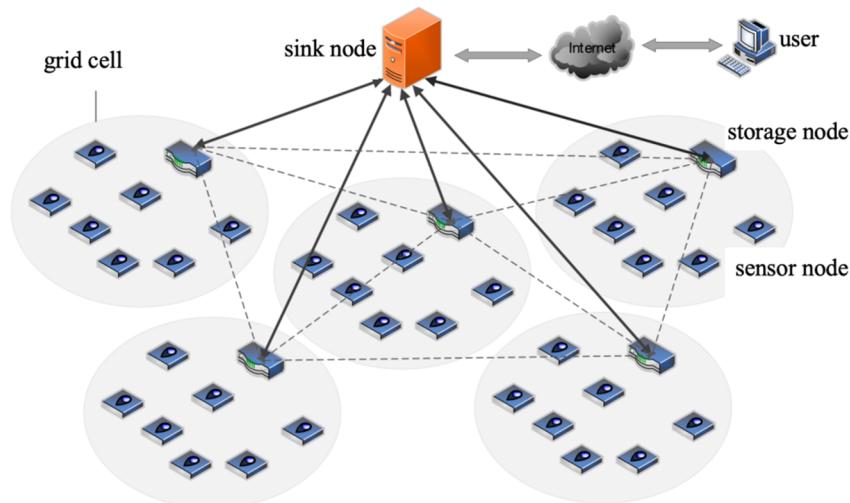


Fig 1. Network model of the two-tier WSN

3.2. Range Query Model

Based on the two-tier WSN model proposed in Section 2.1, the query model in this paper can be defined as follows.

(1) The two-tier WSN is divided into several query cells. Assume that any query cell G contains one storage node G

and several lower-layer sensor nodes. Each sensor node is assigned a unique device ID. The sensor nodes in query cell G are denoted by S_i . Then, a query cell can be represented as:

$$G = \{M, S_i\} \quad (1)$$

(2) All sensor nodes in the network maintain loose time synchronization. The time interval between two consecutive

data submissions is denoted by t . Let a_i represent the number of data dimensions queried by the Sink node, and let $ID(G)$ denote the ID of query cell G . When the Sink node sends a range query request with range $[Low_i, High_i]$ to query cell G , the range query request can be expressed as:

$$R = \{ID(G), t, (a_1, [Low_1, High_1]), \dots, (a_n, [Low_n, High_n])\} \quad (2)$$

(3) Suppose that sensor node S_i collects data N times during time period t , obtaining multidimensional sensed data D_1, D_2, \dots, D_N , which can be represented as:

$$S_i = \{ID(S_i), t, (D_1, D_2, \dots, D_N)\} \quad (3)$$

where $ID(S_i)$ denotes the ID of sensor node S_i .

Based on the above query model, and for the convenience of observation and computation, the protocol proposed in this paper assumes that the data collected by lower-layer sensor nodes are integers. However, in practical applications, some data are not naturally integers, such as temperature, humidity, illuminance, and atmospheric pressure. Nevertheless, these values can be conveniently transformed into integers, and after such conversion, they are still applicable to the proposed protocol.

3.3. Attack Model

In a two-tier WSN, storage nodes are located in the intermediate layer. They not only store data uploaded by a large number of neighboring sensor nodes, but also respond to query instructions. Therefore, storage nodes are more likely

to become the primary targets of malicious attacks. In contrast, even if a sensor node is compromised, the amount of data stored on a single sensor node accounts for only a small proportion of the overall network data, and thus its impact on the entire network is limited.

This paper adopts the **honest-but-curious** threat model and focuses on privacy-preserving measures when storage nodes are attacked by adversaries. A compromised storage node may lead to the following three situations. First, all private data stored in the storage node may be disclosed. Second, the storage node may return fabricated data that do not actually exist to the Sink node. Third, the storage node may return incomplete data to the Sink node.

4. Modeling of the Privacy-Preserving Range Query Protocol

The protocol proposed in this paper is based on the Diffie–Hellman algorithm, the AES encryption algorithm, cross 0–1 encoding, and the multidimensional encrypted constraint chain. It consists of a key management mechanism, a data submission protocol, a query processing protocol, and a result verification algorithm. The overall protocol can be divided into four stages: the network access stage, the data submission stage, the query processing stage, and the result verification stage. The workflow of the protocol is shown in Fig. 2.

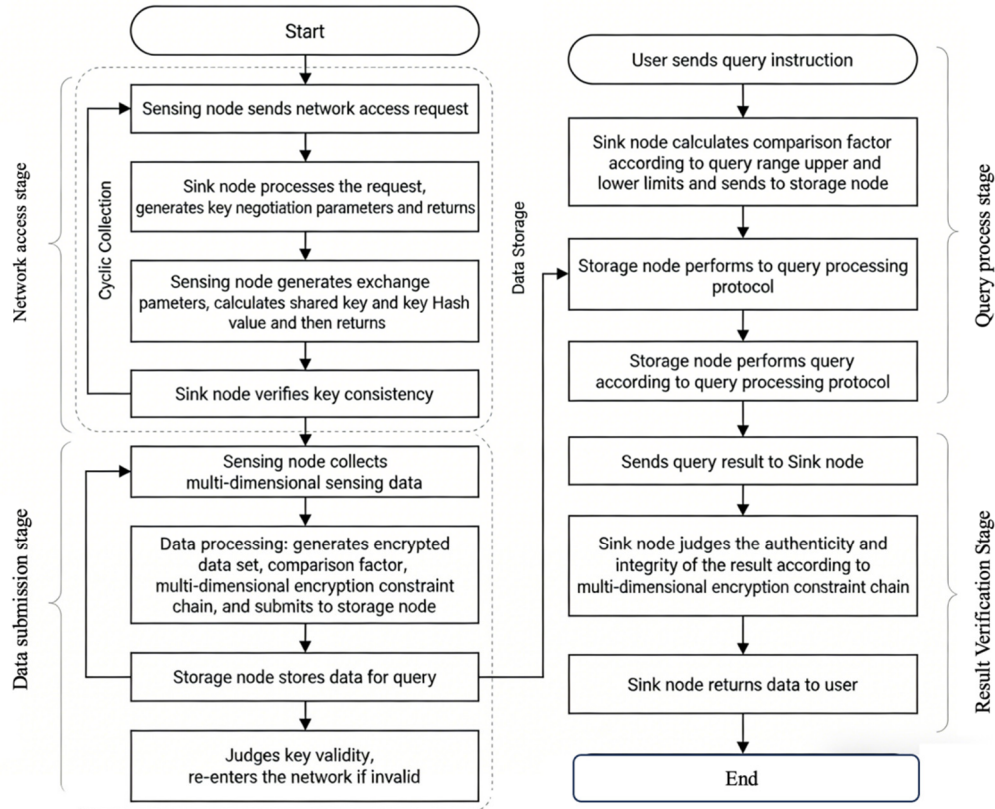


Fig 2. Workflow of the proposed protocol

(1) Network access stage:

A sensor node initiates a network access request, and the Sink node determines the legitimacy of the request according to the node ID. If the request is approved, the two parties exchange Diffie–Hellman parameters and perform modular exponentiation based on the exchanged information to derive a shared key, after which the network access process is completed. If an adversary attempts to obtain the shared key

through computation, it must confront the difficulties of large-prime arithmetic and discrete logarithm computation. Therefore, the security of the shared key in the network access stage can be guaranteed.

(2) Data submission stage:

In each cycle, sensor nodes collect multidimensional data and group them according to attribute dimensions. Cross 0–1 encoding is then employed to generate comparison factors for

each group. For each group, the AES algorithm is used to generate an encrypted data set. Meanwhile, the sensor node ID, the data collection period, and the grouped data are used to construct a multidimensional encrypted constraint chain according to the characteristics of neighboring nodes. Finally, the sensor node uploads the comparison factors, ciphertext data, and the multidimensional encrypted constraint chain to the storage node for storage.

(3) Query processing stage:

The user sends the query range to the Sink node. After receiving it, the Sink node applies cross 0–1 encoding to the query range values to generate comparison factors, and then sends the query cell together with the comparison factors to the storage node as a query instruction. According to the corresponding rules, the storage node compares the comparison factors of the collected data with those of the query range values, identifies the private data that satisfy the query range, completes the privacy-preserving range query for multidimensional data, and sends the results to the Sink node.

(4) Result verification stage:

The Sink node decrypts the returned data and verifies the authenticity and integrity of the query results according to the data collection period contained in the multidimensional encrypted constraint chain and the structural properties of the chain. Finally, the correct results are returned to the user.

In the above process, sensor nodes can verify key validity based on time. If a key expires, a new key is generated using the Diffie–Hellman algorithm. All keys are stored at the Sink node. When a new key is generated, the old expired key is overwritten. Therefore, the protocol has a certain capability to resist collusion attacks involving sensor nodes and the Sink node.

4.1. Relevant Techniques

4.1.1. Cross 0–1 Encoding Technique

To compare data values without knowing their actual plaintext values, this paper proposes a novel **cross 0–1 encoding mechanism** based on the fundamental principle of 0–1 encoding. Different cross 0–1 encoding rules are applied to sensed data and query range values, respectively. The cross 0–1 encoding proposed in this paper can be directly converted into decimal values for comparison, without requiring any additional numerical transformation before comparison.

Definition 3.1: Cross 0–1 Encoding. For sensed data, let the positive integer $d=d_n d_{n-1} d_{n-2} \dots d_1 \in \{0,1\}^w$ be a binary value of length w . Denote the cross 0-encoding of d by CE_d^0 , and the cross 1-encoding of d by CE_d^1 . The formulas are as follows:

$$CE_d^0 = \{d_n d_{n-1} \dots d_i 1 \dots 1 / d_i = 0, 1 \leq i \leq n\} \quad (4)$$

$$CE_d^1 = \{d_n d_{n-1} \dots d_i 0 \dots 0 / d_i = 1, 1 \leq i \leq n\} \quad (5)$$

For the values in the query range interval, let the positive integer $s=s_n s_{n-1} s_{n-2} \dots s_1 \in \{0,1\}^w$ be a binary value of length w . Denote the cross 0-encoding of s by CE_s^0 , and the cross 1-encoding of s by CE_s^1 . The formulas are as follows:

$$CE_s^0 = \{s_n s_{n-1} \dots s_{i+1} 10 \dots 0 / s_i = 0, 1 \leq i \leq n\} \quad (6)$$

$$CE_s^1 = \{s_n s_{n-1} \dots s_{i+1} 01 \dots 1 / s_i = 1, 1 \leq i \leq n\} \quad (7)$$

Property 3.1: Suppose that the positive integer d is represented by a binary string of length w , and its cross 0-encoding and cross 1-encoding are CE_d^0 and CE_d^1 , respectively. Then the cardinalities of CE_d^0 and CE_d^1 are both within the interval $[0, w]$, and the sum of their cardinalities is equal to w .

Theorem 3.1 (Comparison Rule): For sensed data d and a query range value s , $s > d$ can be inferred if and only if $CE_s^1 \cap CE_d^0 \neq \emptyset$; $s \leq d$ can be inferred if and only if $CE_s^1 \cap CE_d^0 = \emptyset$ and $s = d$ can be inferred if and only if $CE_s^1 = CE_d^0$.

The basic principle of cross 0–1 encoding is to first convert a positive integer into its binary representation, then rearrange and re-encode it on the basis of traditional 0–1 encoding, and finally pad the code so that the binary representation remains w bits in length. For sensed data, each bit d_i in the binary representation is examined from right to left. If $d_i = 0$, then all bits lower than position i are set to 0, and if the resulting code is shorter than w bits, the vacant positions are padded with 1; the resulting value is then included in its 0-encoding set. If $d_i = 1$, then all bits lower than position i are set to 0, and if the resulting code is shorter than w bits, the vacant positions are padded with 0; the resulting value is then included in its 1-encoding set. For values in the query range interval, each bit d_i in the binary representation is also examined from right to left. If $d_i = 0$, this bit is changed to 1, and all bits lower than position i are set to 0; if the resulting code is shorter than w bits, the vacant positions are padded with 0, and the resulting value is included in its 0-encoding set. If $d_i = 1$, this bit is changed to 0, and all bits lower than position i are set to 0; if the resulting code is shorter than w bits, the vacant positions are padded with 1, and the resulting value is included in its 1-encoding set.

An illustrative example is given below. Suppose that the sensed data are $d=5=(0101)_2$, and the query range value is $s=9=(1001)_2$. Their binary representations are both 4 bits in length. Then, the cross 1-encoding of d is $CE_d^1 = \{0101, 0100\}$, and its cross 0-encoding is $CE_d^0 = \{0101, 0111\}$. For the query range value s , its cross 1-encoding is $CE_s^1 = \{1000, 0111\}$, and its cross 0-encoding is $CE_s^0 = \{1010, 1100\}$. These encodings can be directly represented in decimal form as $CE_d^1 = \{5, 4\}$, $CE_d^0 = \{5, 7\}$, $CE_s^1 = \{8, 7\}$, and $CE_s^0 = \{10, 12\}$. In this way, the comparison can be performed directly in numerical form. Since there exists an intersection element, namely 7, between the cross 1-encoding of the query range value s , $CE_s^1 = \{8, 7\}$, and the cross 0-encoding of the sensed data d , $CE_d^0 = \{5, 7\}$, it can be concluded that $s > d$.

4.1.2. Multidimensional Encrypted Constraint Chain Technique

In practical applications, sensor nodes are usually equipped with multiple sensing modules and can simultaneously acquire multidimensional data such as temperature, humidity, and illuminance. Both the collected data and the range query instructions issued by the Sink node are generally multidimensional. Suppose that the collected multidimensional data and the multidimensional range query instructions are represented as follows:

$$D = \{S_i, t, (a_1, [d_1^1, d_2^1, \dots]), \dots, (a_n, [d_1^n, d_2^n, \dots])\} \quad (8)$$

$$R = \{S_i, t, (a_1, [Low_1, High_1]), \dots, (a_n, [Low_n, High_n])\} \quad (9)$$

where S_i denotes the ID of the sensor node, t denotes the query period, $a_i, (1 \leq i \leq n)$ denotes the collected data in the i -th dimension, and $[Low_i, High_i], (1 \leq i \leq n)$ denote the lower and upper bounds of the range query interval in the i -th dimension, respectively.

The multidimensional encrypted constraint chain technique mainly consists of two steps. The first step is the construction of the multidimensional encrypted constraint chain, and the second step is the verification of the authenticity and integrity

of query results using the multidimensional encrypted constraint chain. The detailed procedure is described as follows.

$$\{(a_1, [d_1^1, d_2^1, \dots, d_m^1]), \dots, (a_i, [d_1^i, d_2^i, \dots, d_m^i]), \dots, (a_n, [d_1^n, d_2^n, \dots, d_m^n])\}, (d_1^1 \leq d_2^1 \leq \dots \leq d_{m-1}^1 \leq d_m^1), (d_1^i \leq d_2^i \leq \dots \leq d_{m-1}^i \leq d_m^i) \quad (10)$$

Then, the multidimensional encrypted constraint chain is constructed as follows:

$$\left\{ \begin{array}{l} (a_1, [d_1^1 \| d_2^1], [d_2^1 \| d_3^1], \dots, [d_{m-2}^1 \| d_{m-1}^1], [d_{m-1}^1 \| d_m^1]), \dots, \\ (a_i, [d_1^i \| d_2^i], [d_2^i \| d_3^i], \dots, [d_{m-2}^i \| d_{m-1}^i], [d_{m-1}^i \| d_m^i]) \end{array} \right\} \quad (11)$$

where “||” denotes the concatenation operator. Finally, the entire chain is encrypted using the *Key*:

$$\left\{ \begin{array}{l} (a_1, [d_1^1 \| d_2^1]_{Key}, [d_2^1 \| d_3^1]_{Key}, \dots, [d_{m-2}^1 \| d_{m-1}^1]_{Key}, [d_{m-1}^1 \| d_m^1]_{Key}), \dots, \\ (a_i, [d_1^i \| d_2^i]_{Key}, [d_2^i \| d_3^i]_{Key}, \dots, [d_{m-2}^i \| d_{m-1}^i]_{Key}, [d_{m-1}^i \| d_m^i]_{Key}) \end{array} \right\} \quad (12)$$

The multidimensional encrypted constraint chain, together with the encrypted sensed data and the comparison factors, is then transmitted to the storage node.

After receiving the range query request *R*, the storage node denotes the multidimensional encrypted constraint chain corresponding to the valid data set that satisfies the query range as *QR*. In addition to returning the correct result data set to the Sink node, the storage node must also send a verification code *VO*. Here, *VO* is the right-neighbor data node of the maximum value in the result data set *QR*, which can assist in determining the authenticity and integrity of the result data set. Assume that $QR \neq \emptyset$, $d_m^i < Low_i \leq High_i < d_{m+1}^i$, then we have:

$$VO = \{a_i, [d_m^i \| d_{m+1}^i]_{Key}\} \quad (13)$$

Step 2: After receiving *QR* and *VO*, the Sink node first decrypts them to determine whether the returned query result data are indeed within the required query range. It then judges the completeness of the result data according to whether the encrypted constraint chain is complete. The detailed verification procedure is presented in Section 3.6, Query Result Verification Stage.

4.2. Security Analysis

(1) Sensed data

The key to protecting the privacy of sensed data in two-tier wireless sensor networks lies in ensuring that storage nodes cannot obtain the actual values of encrypted data items without knowing the secret key. Even if a storage node is compromised, the scheme proposed in this paper can still effectively protect the privacy of sensed data. Before data are uploaded, the private key used for encryption is shared only with the Sink node. Moreover, the probability that an attacker can obtain the cross encoding and convert it into the corresponding prime number is only $1/(2^{n+1})$, where *n* is the bit length of the sensed data in binary form. In addition, the prime number library changes dynamically, which further reduces the possibility of attackers obtaining the actual data values and thus effectively protects the privacy of sensed data.

(2) Query results

The core of protecting the privacy of query results is to prevent storage nodes from obtaining their actual values. In the proposed scheme, the storage node processes range queries only in the ciphertext domain according to the intersections of comparison factors, without performing any decryption. The query results are returned to the Sink node in

Step 1: The sensor node first sorts the collected *a_i*-dimensional data in ascending order:

encrypted form and are decrypted only after reception by the Sink node. Therefore, even if the ciphertext is intercepted during transmission, data privacy cannot be disclosed without the key.

(3) Query range

The Sink node applies cross 0–1 encoding and prime fusion to the lower and upper bounds of the query range, converts them into ciphertext, and then sends them to the storage node for query processing. Owing to the low probability of reversing the cross encoding and the dynamic nature of the prime number library, it is difficult for an attacker to derive the actual values of the query range through computation, thereby ensuring that the query range is not disclosed.

In summary, the multidimensional data range query scheme proposed in this paper, based on cross 0–1 encoding and prime fusion, can effectively guarantee the privacy of sensed data, query results, and query ranges. In the following sections, its feasibility and security will be further validated through implementation and experimental analysis.

5. Protocol Implementation and Energy Consumption Analysis

5.1. Protocol Implementation and Energy Consumption Measurement

In the experimental section, the proposed protocol was implemented by designing sensor nodes on Developer Kit boards with a Cortex-M4 core running the AliOS Things operating system, storage nodes on an iTOP-4412 core board with a Cortex-A9 core running the Linux operating system, and the Sink node on a PC. A schematic diagram of the physical connection between the sensor nodes and the storage node is shown in Fig 3.

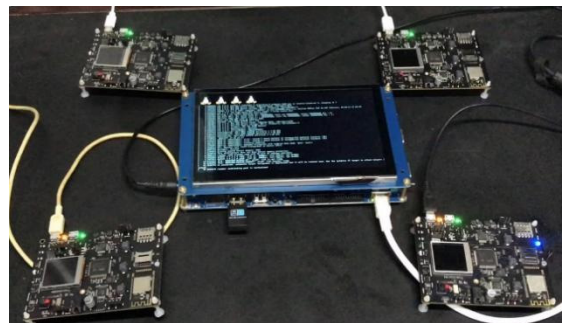


Fig 3. Schematic diagram of the physical connection of the nodes

In the communication energy consumption experiment of this paper, a NAPUI PM9816 power meter was used. A photograph of the device is shown in Fig 4. Due to the influence of the external environment and the operating characteristics of electronic circuits, the communication power and communication time of sensor nodes may fluctuate within a certain range. To ensure the accuracy of the experimental data, this paper collected data 20 times and then used the average value obtained through a median filtering algorithm as the experimental result. Let the measured power data be denoted by P , and let the communication transmission time obtained from the power meter be denoted by t . Then, according to Eq. 14, the communication energy consumption W of a sensor node can be calculated as follows:

$$W = P \times t \quad (14)$$



Fig 4. Photograph of the NAPUI PM9816 power meter

5.2. Experimental Data Collection and Energy Consumption Analysis

The communication energy consumption of sensor nodes in the proposed scheme is compared with that of the Optimized HMAC Protocol, which is currently one of the best secure range query protocols for two-tier WSNs based on 0–1 encoding. According to the literature [13], its sensor-node communication energy consumption is lower than that of the SafeQ, QuerySec, and CSRQ schemes.

The default experimental parameters are listed in Table 1 below.

Table 1. Default Experimental Parameter Settings

Parameter	Value
Network node ID length / bit	96
Shared key length / bit	128
Sensed data length (www) / bit	12
Time interval of sensed data collection / ms	200
Number of dimensions of sensed data (d)	3
Number of data samples collected per cycle (N)	15
Number of sensor nodes (s)	4

5.2.1. Effect of the Number of Data Samples Collected per Cycle on Energy Consumption

In the experiments conducted in this paper, the number of data samples collected within a single cycle (N) was treated as an independent variable ranging from 10 to 40, while the

other default experimental parameters were kept unchanged. The communication energy consumption of sensor nodes was measured in milliwatt-hours (mWh). The experimental data were collected and used to calculate the communication energy consumption, as shown in Table 2.

Table 2. Communication Energy Consumption under Different Numbers of Data Samples Collected per Cycle (N)

Number of Data Samples Collected per Cycle /N	Energy Consumption of the Proposed Protocol /mWh	Energy Consumption of the Optimized HMAC Protocol /mWh
15	0.223156	0.743365
20	0.318304	0.834674
25	0.381501	0.927653
30	0.484211	1.043272
35	0.664753	1.155303

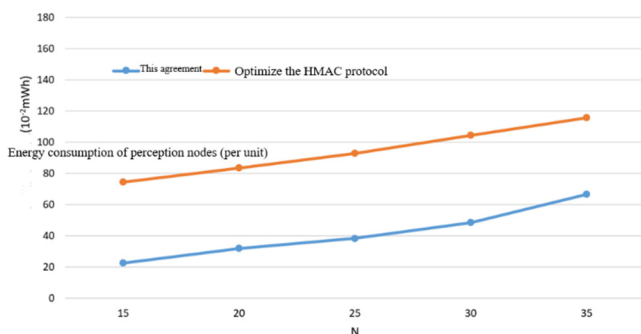


Fig 5. Effect of N on the Energy Consumption of Sensor Nodes

Based on the data in the table, a line chart illustrating the effect of the number of data samples collected per cycle on the communication energy consumption of sensor nodes is plotted, as shown in Fig 5.

As can be seen from the figure, with the increase in the amount of data collected in a single cycle, the energy consumption of sensor nodes in both the proposed protocol and the Optimized HMAC Protocol increases linearly. As the data volume becomes larger, more encrypted data sets and multidimensional encrypted constraint chains need to be generated, which leads to higher communication energy consumption. However, the energy consumption of sensor nodes in the proposed protocol remains consistently lower

than that in the Optimized HMAC Protocol, and its growth rate is also more moderate, indicating better energy efficiency.

5.2.2. Effect of the Number of Sensor Nodes in a Query Cell on Energy Consumption

In the experiments conducted in this paper, the dimensionality of the collected data (d) could be up to four, namely illuminance, temperature, atmospheric pressure, and

humidity. The data dimensionality was treated as an independent variable ranging from 1 to 4, while the other default experimental parameters were kept unchanged. The communication energy consumption of sensor nodes was measured in milliwatt-hours (mWh). The experimental data were collected and used to calculate the communication energy consumption, as shown in Table 3.

Table 3. Communication Energy Consumption under Different Data Dimensionalities(d)

Data Dimensionality	Energy Consumption of the Proposed Protocol / mWh	Energy Consumption of CSRQ/mWh
1dimension	0.199103	0.215969
2dimension	0.265273	0.323726
3dimension	0.339812	0.447388
4dimension	0.410122	0.579643

Based on the data in the table, a line chart illustrating the effect of changes in data dimensionality on the communication energy consumption of sensor nodes is plotted, as shown in Fig 6.

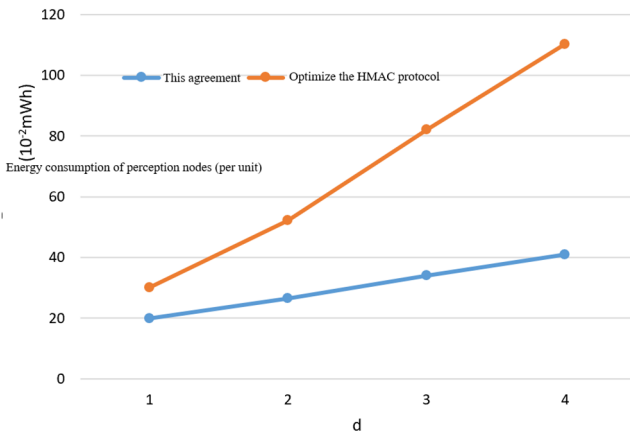


Fig 6. Effect of d on the Energy Consumption of Sensor Nodes

As can be seen from the figure, with the increase in data dimensionality, the energy consumption of sensor nodes in both the proposed protocol and the Optimized HMAC Protocol increases linearly. As the dimensionality becomes higher, the encrypted data set becomes larger, and the communication energy consumption correspondingly increases. However, the energy consumption of the proposed protocol remains consistently lower and grows more slowly, indicating better energy performance.

6. Conclusion

This paper proposes an energy-efficient privacy-preserving range query protocol for multidimensional data in two-tier wireless sensor networks, referred to as **PERQ-M**. By combining the Diffie–Hellman key exchange protocol with AES encryption, a key management mechanism is established to ensure data privacy and security. An improved cross 0–1 encoding technique is adopted to enable efficient comparison

of ciphertext data. In addition, an improved multidimensional encrypted constraint chain is designed to verify the authenticity and integrity of multidimensional query results. The experimental implementation validates the security and effectiveness of the proposed protocol. Comparison with the Optimized HMAC Protocol shows that the proposed protocol achieves lower communication energy consumption and better overall performance.

This protocol mainly targets scenarios in which sensor nodes are captured. However, it still has limitations in resisting security threats arising from the simultaneous capture of the Sink node or a large number of sensor nodes. This issue will be further investigated in future work.

References

- [1] Zhang X, Dong L, Peng H, et al. Collusion-Aware Privacy-Preserving Range Query in Tiered Wireless Sensor Networks [J]. *Sensors*, 2014,14(12):23905-23932.
- [2] Sheng B, Li Q. Verifiable privacy-preserving range query in two-tiered sensor networks[C]//IEEE INFOCOM 2008-The 27th Conference on Computer Communications. IEEE, 2008: 46-50.
- [3] Shi J, Zhang R, Zhang Y. A spatiotemporal approach for secure range queries in tiered sensor networks[J]. *IEEE transactions on wireless communications*, 2010, 10(1):264-273.
- [4] Chen F, Liu A X. Privacy-and integrity-preserving range queries in sensor networks[J]. *IEEE/ACM Transactions on Networking*, 2012, 20(6):1774-1787.
- [5] Yi Y, Li R, Chen F, et al. A digital watermarking approach to secure and precise range query processing in sensor networks [C] //2013 Proceedings IEEE INFOCOM. IEEE, 2013:1950-1958.
- [6] Dai H, Ye Q, Yang G, et al. CSRQ: communication-efficient secure range queries in two-tiered sensor networks[J]. *Sensors*, 2016, 16(2):1-17.
- [7] Hu Q, Deng Y. Range Query Method Based on Compressed HMAC Algorithm for Sensor Networks[J]. *Computer Engineering*, 2021, 47(12):200-208.