Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 814892, 11 pages
http://dx.doi.org/10.1155/2013/814892

Research Article

EMQP: An Energy-Efficient Privacy-Preserving MAX/MIN Query Processing in Tiered Wireless Sensor Networks

Hua Dai,[1,2,3] Geng Yang,[1,2,3] and Xiaolin Qin[4]

[1] *College of Computer Science & Technology, Nanjing University of Post & Telecommunications, Nanjing 210013, China*
[2] *Key Lab of Broadband Wireless Communication and Sensor Network Technology, Ministry of Education, Nanjing 210013, China*
[3] *Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, Jiangsu 210003, China*
[4] *College of Computer Science & Technology, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China*

Correspondence should be addressed to Hua Dai; daihua@njupt.edu.cn

Received 22 March 2013; Revised 2 July 2013; Accepted 3 July 2013

Academic Editor: Tai-hoon Kim

We consider a hybrid two-tiered sensor network consisting of regular resource-limited sensor nodes and powerful master nodes with abundant resources. In the architecture, master nodes take charge of storing data collected by sensor nodes and processing queries from the base station. Due to the important role of master nodes, they might easily become the target for the adversary to compromise in an untrusted or hostile circumstance. A compromised master node may leak sensitive data in its storage to the adversary, which breaches the data privacy. This paper proposes EMQP, a novel and energy-efficient privacy-preserving MAX/MIN query protocol which is capable of preventing adversaries from obtaining sensitive data collected by sensor nodes. To preserve privacy, the 0-1 encoding verification, keyed-hash message authentication coding, and symmetric encryption are applied to achieve the secret comparison of data items without knowing their real values. On the basis of secret comparison mechanism, the data submission and query processing protocols are proposed to describe the details of EMQP. And the analyses on privacy protection and energy consumption are also given. Moreover, a hash-based optimization method is presented to save more energy of the resource-limited sensor nodes. The simulation result shows that EMQP is more efficient than the current work in energy consumption.

1. Introduction

Wireless sensor networks (WSNs) have been widely used in a variety of important areas, such as environment sensing, battlefield monitoring, and volcanic eruption predication. In this paper, we consider a two-tiered wireless sensor network (two-tiered WSNs) [1, 2] as shown in Figure 1, which consists of a large number of sensor nodes at the lower tier and relatively fewer master nodes at the upper tier. Sensor nodes are resource limited (computation, storage, energy, etc.) and take charge of collecting data and periodically submitting it to a nearby master node for storage, while master nodes have rich resources, and answer for the ad hoc data queries from the base station which are issued via an on-demand wireless (e.g., satellite) link. It is necessary to maintain such in-network data storage and query processing in remote and tough environments, where it is infeasible or difficult to keep connection between the sensor networks and the base station with the high-speed and always-on manner. The two-tiered architecture is also known to be indispensable for increasing network capacity and scalability, reducing system complexity, and prolonging network lifetime.

As master nodes are responsible for data storage and query answering in networks, they are much more attractive and vulnerable to adversaries in a hostile environment. Once a master node is compromised, serious threats could be brought out. For example, adversaries could use compromised master nodes to steal information about patients in a human health monitoring sensor network, leading to the privacy breach of patients. It is a challenge for master nodes to process queries in such an environment with privacy, since they have to gain information about the collected data items

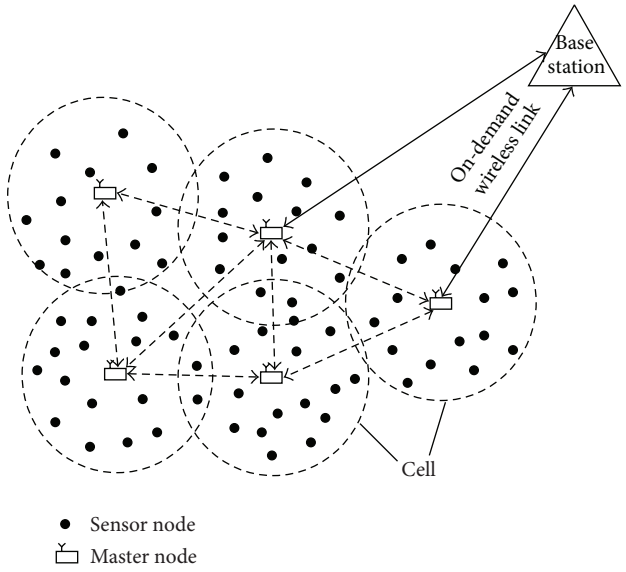


Figure 1: A two-tiered sensor network architecture.

for query result computing, which is conflictive with the privacy preserving objective.

Data query is an important operation for events monitoring or data analysis in sensors networks. Recently, privacy-preserving range query [3–8] and data aggregation [9–12] have been well addressed, however, research efforts on MAX/MIN query are limited, which is to query the maximum or minimum value in an interested area. In this paper, we focus on the MAX/MIN queries, which are important in many applications. For example, the MAX/MIN query can be applied to monitor the forest fires according to the maximum temperature acquiring.

To the best of our knowledge, only [13] proposed a preliminary solution to privacy-preserving MAX/MIN query in two-tiered WSNs, but it is still with the problem of inefficient energy consumption. This paper proposes an energy-efficient privacy-preserving MAX/MIN query processing (EMQP) for two-tiered WSNs. The basic idea is that sensor nodes first encode their collected data and send them to their nearby master nodes for storage, for the convenience that the master nodes can correctly process MAX/MIN queries over encoded data without knowing their real values. An adversary cannot steal any data items or query results in the master nodes, even when they were compromised. The main contribution of our work is that we introduce 0-1 encoding verification scheme to achieve the secret comparison between the collected data items, without knowing their real values. Based on that method, we propose a novel privacy-preserving MAX/MIN query protocol. To reduce the energy consumption of sensor nodes, we also give a hash-based optimization method, which demonstrates a significant energy-saving benefit. We evaluate EMQP by comprehensive simulation, and the results indicate that EMQP has a good performance compared with other methods.

The rest of this paper is organized as follows. Section 2 gives a brief review of the related work. Section 3 describes the models and the problem statement. In Section 4, we present the details of our energy-efficient privacy-preserving MAX/MIN query protocol. Section 5 gives an optimization for saving energy of sensor nodes. We evaluate the performance of our approach in Section 6 and conclude this paper in Section 7.

2. Related Work

Data storage models for sensor networks have drawn much attention in existing research work. In [14, 15], a novel data storage system is proposed by introducing an intermediate tier between the base station and sensor nodes, which can provide abundant storage for data caching and an efficient access to the data collected by sensor networks for query processing. We consider the same system model in this paper, in which some master nodes are deployed as the intermediate tier for data storage and query answering. In practice, several products of master nodes have been manufactured and are commercially available, such as StarGate [16] and RISE [17].

Recently, verifiable privacy-preserving range query in two-tiered WSNs has been widely studied [3–8], aiming to protect the privacy and integrity of range queries. Hacigümüş et al.firstly proposed a bucket partitioning [18] based scheme [3, 4], whose basic idea is to divide the domain of collected data values into multiple continuous but no overlapping buckets. In each epoch of time, sensor nodes collect data items, put them into corresponding buckets, encrypt them together in each bucket, and then send the ciphertext along with the corresponding bucket ID to a nearby master node. For each bucket without data items, an encoding number will be generated and transmitted to a nearby master node, which can be used by the base station to verify that the bucket is empty. When the base station executes a range query, it first generates the smallest set of bucket IDs covering the range in the query and then sends the ID set as the query to master nodes. Upon receiving the bucket IDs, the master nodes return the corresponding ciphertext in all those buckets. The base station can then decrypt the ciphertext to get the query result and verify its integrity by encoding numbers. Shi et al. proposed an optimized version [5, 6] of integrity verification scheme of [3, 4], with the objective to reduce the communication cost of sensor nodes. Since all the works in [3–6] are based on the bucket partitioning scheme, there is an inherited drawback that the bucket partitioning allows compromised master nodes to obtain a reasonable estimation on the real values of both data items and queried ranges [19]. To solve data estimation problem, Chen and Liu proposed a secure and efficient range query processing protocol, SafeQ [7, 8], which is based on Prefix Membership Verification (PMV) [20, 21] and neighborhood chains. The PMV scheme can be used to check a data item x whether it is in a range $[a, b]$ without knowing the real values of x, a and b, while the neighborhood chains mechanism can be used to detect the falsifies or forges of query results. Using such PMV and neighborhood chains, SafeQ can correctly process range queries in privacy and integrity preserving circumstance.

For privacy-preserving MAX/MIN query in two-tiered WSNs, only [13] has presented a preliminary solution. In [13], the same PMV scheme as in [7, 8] is used to privately compute the maximum or minimum data item. However, it still has a problem of inefficient energy consumption of sensor nodes. This paper will propose an energy-efficient privacy-preserving MAX/MIN query, the evaluation results of which indicate that it has a better performance than [13] in energy consumption of sensor nodes.

3. Models and Problem Statement

3.1. Network Model. We consider a similar two-tiered sensor networks model as in [3–8]. As shown in Figure 1, the network is partitioned into multiple cells, each containing several sensor nodes and a master node. The two types of nodes are different in resource owning. In particular, the master nodes are powerful devices and have abundant resource in energy, storage, and computation, while the sensor nodes are cheap sensing devices with limited resource. Each sensor node periodically transmits its collected data to its nearby master node in the same cell. The base station is in charge of converting users' questions into queries and then disseminating the queries to the corresponding master nodes, which process the queries based on their stored data items and return the query results to the base station via an upper-tier multihop network formed by the resource-rich master nodes and an on-demand wireless (e.g., satellite) link between some master nodes and the base station.

As in [3–8], we assume that master nodes and sensor nodes know their respective locations and affiliated cells. The time is assumed to be divided into epochs. At the end of each epoch, each sensor node submits all its collected data items to the affiliated master node in its cell.

3.2. MAX/MIN Query Model. A MAX/MIN query is an operation to obtain the maximum or minimum value from an interested area. For simplicity, the following atomic MAX/MIN query will be considered, which is denoted as a four-element tuple:

$$Q^t = \left(\varphi, t, C, \Gamma^t\right), \tag{1}$$

where $\varphi \in \{\text{MAX}, \text{MIN}\}$ indicates the query type, t and C are the queried epoch number and cell ID, and Γ^t denotes the set of queried sensor nodes IDs which indicate a query region in C. Other complicated MAX/MIN queries that contain multiple epochs, cells, and/or query regions can be easily decomposed into multiple atomic ones. For example, there is a network consisting of 3 cells and 21 sensors as shown in Figure 2. The complicated MAX query "obtaining the maximum value of the rectangle region in epoch t" can be decomposed into 3 atomic MAX queries, such as $Q_1^t = (\text{MAX}, t, C_1, \{1, 4, 6, 11\})$, $Q_2^t = (\text{MAX}, t, C_2, \{8, 9, 12\})$, and $Q_3^t = (\text{MAX}, t, C_3, \{16, 20\})$. And the query result of the above complicated MAX query is the maximum of the results of Q_1^t, Q_2^t, and Q_3^t. In this paper, we take atomic MAX/MIN query as an abbreviation, "MAX/MIN query", for simplicity.

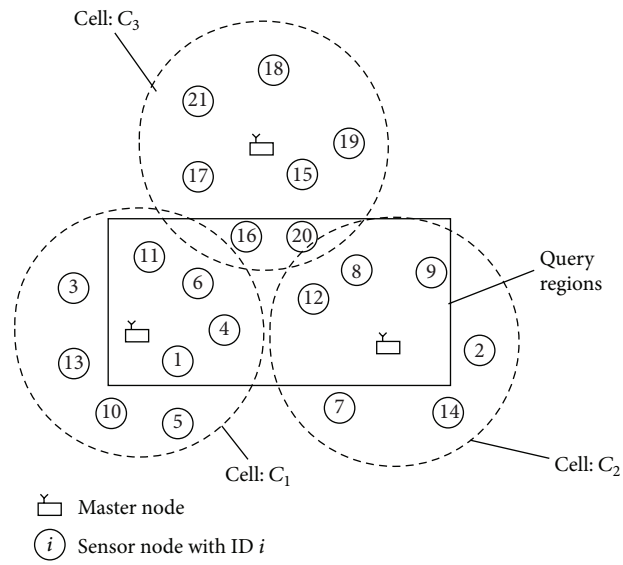


Figure 2: A complicated MAX query example.

Since each master takes charge of a unique cell, the adversary will not gain more from the collaboration of multiple honest-but-curious masters. The subsequent discussion in this paper focuses on a cell C consisting of a master M and n sensor nodes $\{s_1, s_2, \ldots, s_n\}$ whose IDs constitute the set $\Gamma = \{1, 2, \ldots, n\}$. Each sensor node probes several data items during each epoch t. We just concentrate on the MAX query processing schemes, while the MIN query is similar and easy to implement.

3.3. Threat Model and Problem Statement. In two-tiered WSNs, the master nodes are too attractive to be easily under attacks from adversaries, since they not only store all the data items collected by sensor nodes, but also take charge of processing queries received from the base station. We assume that the sensor nodes and the base station are trusted but the master nodes. And we adopt the same honest-but-curious threat model as [13], where master nodes may try to breach privacy to obtain sensitive data items but faithfully obey protocols during query processing.

In this paper, we focus on how to provide data privacy preservation and efficient query processing schemes for MAX/MIN queries, while confronting the honest-but-curious master nodes. In addition, we will use the metric of energy consumption of sensor nodes, which directly affects the lift time of the whole networks, to evaluate the performance of our proposed schemes.

4. 0-1 Encoding-Verification-Based MAX/MIN Query Processing

To preserve privacy, it seems natural to have sensor nodes encrypting their collected data items; however, the key challenge is how the master nodes process MAX/MIN queries over encrypted data without knowing their real values.

The basic idea for preserving privacy MAX/MIN query is as follows. We assume that each senor s_i in a network and the base station share a secret key k_i. For the N data items that s_i collects in epoch t, s_i first encrypts the maximum or minimum data item d_i using key k_i, the result of which is denoted as $(d_i)_{ki}$. For computation efficiency, we use symmetric encryption like DES, IDEA, and so forth. Then, s_i applies an encoding function $\mathfrak{R}$ to d_i and obtains $\mathfrak{R}(d_i)$. And s_i submits the encrypted and encoded data to its closest master node M. When M performs a MAX/MIN query, a *secret comparing function* $\mathfrak{T}$ will be used for query processing over encrypted and encoded data. The functions $\mathfrak{R}$ and $\mathfrak{T}$ satisfy the following conditions: (1) given $\mathfrak{R}(d_i)$ and $(d_i)_{ki}$, it is computationally infeasible for the master node to compute d_i. (2) Given two data items x and y, $x \leq y$ if and only if $\mathfrak{T}(x, y)$ is not null. The former condition guarantees data privacy, while the later allows the master node to determine the very encrypted data containing the maximum or minimum without knowing the real values of the collected.

4.1. 0-1 Encoding Verification. 0-1 encoding verification was first introduced by Lin and Tzeng in [22] for solving the millionaires' problem [23], which is to find the richest from several millionaires without leaking the sensitive personal information of their properties.

Definition 1. Let $x = b_1 b_2 \cdots b_{w-1} b_w \in \{0,1\}^w$ be a binary string of length w. The 0-encoding and 1-encoding are the sets $E^0(x)$ and $E^1(x)$ of binary strings, such that

$$E^0(x) = \{b_1 b_2 \cdots b_{i-1} 1 \mid b_i = 0 \wedge 1 \leq i \leq w\},$$
$$E^1(x) = \{b_1 b_2 \cdots b_i \mid b_i = 1 \wedge 1 \leq i \leq w\}. \quad (2)$$

According to Definition 1, we can get the properties as follows:

(1) $1 \leq |E^0(x)| \leq w$, $1 \leq |E^1(x)| \leq w$, and $|E^0(x)| + |E^1(x)| = w$.

(2) $E^0(x) \cap E^1(x) = \varnothing$.

Theorem 2. *For two numbers x and y, if they are encoded into $E^1(x)$ and $E^0(y)$, one can see that*

(1) $x > y \Leftrightarrow E^1(x) \cap E^0(y) \neq \varnothing$.

(2) $x \leq y \Leftrightarrow E^1(x) \cap E^0(y) = \varnothing$.

The proof of Theorem 2 refers to [22]. In order to verify whether a number x is not greater than the other number y using Theorem 2, we can convert x and y to $E^1(x)$ and $E^0(y)$; thus, $x \leq y$ if and only if $E^1(x) \cap E^0(y) = \varnothing$, otherwise $x > y$.

To verify whether $E^1(x) \cap E^0(y)$ is null or not, the operation of verifying the equalization of two binary strings is needed. For simplicity, we convert each 0-encoding or 1-encoding binary string to a corresponding unique number using a numeralization function $\mathcal{N}$, which should satisfy the following properties: (1) for any 0-encoding or 1-encoding binary string p, $\mathcal{N}(p)$ is also a binary string; (2) for any two 0-encoding or 1-encoding binary strings p and q, $p = q$ if and only if $\mathcal{N}(p) = \mathcal{N}(q)$. There are many ways to construct $\mathcal{N}$. We use a similar numeralization function as [24]. Given a binary string $b_1 b_2 \cdots b_{w-1} b_w$ of w bits, we insert 1 before b_1. For example, 0101 is converted to 10101. Given a set of 0-encoding or 1-encoding binary strings P, we denote by $\mathcal{N}(P)$ the resulting set of numericalized binary strings. Therefore, $x \leq y$ if and only if $\mathcal{N}(E^1(x)) \cap \mathcal{N}(E^0(y)) = \varnothing$, and $x > y$ if and only if $\mathcal{N}(E^1(x)) \cap \mathcal{N}(E^0(y)) \neq \varnothing$. Figure 3 shows the process of verifying $9 > 4$.

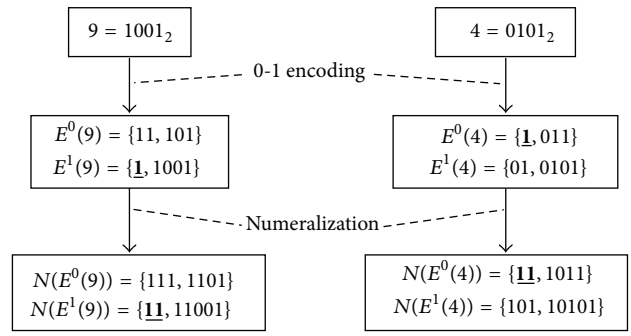


Figure 3: 0-1 encoding verification.

4.2. Data Submission Protocol. The data submission protocol (DSP) is concerned with how a sensor node transmits its collected data items to the master node M. For each sensor node s_i in C, after collecting the N data items $\{d_i^1, d_i^2, ..., d_i^N\}$ in epoch t, s_i performs the following steps.

(1) Compute the maximum of $\{d_i^1, d_i^2, ..., d_i^N\}$, which is denoted as $d_i = \max\{d_i^1, d_i^2, ..., d_i^N\}$.

(2) Convert d_i to $E^0(d_i)$ and $E^1(d_i)$ and compute $\mathcal{N}(E^0(d_i))$ and $\mathcal{N}(E^1(d_i))$ by the numeralization function $\mathcal{N}$.

(3) Compute the keyed-hash message authentication code (HMAC) [25] of each data item in $\mathcal{N}(E^0(d_i))$ and $\mathcal{N}(E^1(d_i))$ using key g, which is shared by all sensor nodes in C, but M knows nothing about it. An HMAC function using key g, denoted as HMAC_g, satisfies the one-wayness and the collision resistance properties. (The one-wayness property of HMAC means that, given $\mathrm{HMAC}_g(x)$, it is computationally infeasible to compute x and g, while the collision resistance property means that it is also computationally infeasible to find two different data items x and y such that $\mathrm{HMAC}_g(x) = \mathrm{HMAC}_g(y)$.) Given a set of numbers S, we use $\mathrm{HMAC}_g(S)$ to represent the resulting set after applying HMAC_g to every numbers in S. In summary, this step computes $\mathrm{HMAC}_g(\mathcal{N}(E^0(d_i)))$ and $\mathrm{HMAC}_g(N(E^1(d_i)))$.

(4) Encrypt d_i to $(d_i)_{ki}$ using key k_i which is shared with the base station.

(5) Submit the following message to M:

$$s_i \longrightarrow M : \left\langle i, t, (d_i)_{ki}, \mathrm{HMAC}_g\left(\mathcal{N}\left(E^0(d_i)\right)\right), \mathrm{HMAC}_g\left(\mathcal{N}\left(E^1(d_i)\right)\right)\right\rangle. \tag{3}$$

The above steps indicate that the aforementioned encoding function $\mathfrak{R}$ is defined as follows:

$$\mathfrak{R}(d_i) = \left\{\mathrm{HMAC}_g\left(\mathcal{N}\left(E^0(d_i)\right)\right), \mathrm{HMAC}_g\left(\mathcal{N}\left(E^1(d_i)\right)\right)\right\}. \tag{4}$$

We name $\mathfrak{R}(d_i)$ as *comparison factors* (*c-factors*) of d_i, which will be used for the secret comparing in the next section.

Since the HMAC function is with one-wayness and collision resistance properties, and sensor nodes only share the secret key with the base station, given $\mathfrak{R}(d_i)$ and $(d_i)_{ki}$, it is computationally infeasible for the master node to obtain the value of d_i. Therefore, we can see that the DSP can preserve data privacy from the master node.

4.3. Query Processing Protocol. The query processing protocol (QPP) is concerned with how the master node M executes a query and returns response to the base station. When M receives a query $Q^t = (\mathrm{MAX}, t, C, \Gamma^t)$ from the base station, M processes Q^t on its stored data $\{(d_i)_{ki}, \mathrm{HMAC}_g(\mathcal{N}(E^0(d_i))), \mathrm{HMAC}_g(\mathcal{N}(E^1(d_i))) \mid i \in \Gamma^t\}$, which is received from sensor nodes in epoch t.

Lemma 3. *Given two data items x and y with corresponding 0-1 encoding c-factors, $HMAC_g(\mathcal{N}(E^1(x)))$ and $HMAC_g(\mathcal{N}(E^0(y)))$, one has*

(1) $x > y \Leftrightarrow HMAC_g(\mathcal{N}(E^1(x))) \cap HMAC_g(\mathcal{N}(E^0(y))) \neq \varnothing$,

(2) $x \leq y \Leftrightarrow HMAC_g(\mathcal{N}(E^1(x))) \cap HMAC_g(\mathcal{N}(E^0(y))) = \varnothing$.

We omit its proof here since it can be easily derived from the collision resistance property of HMAC and Theorem 2.

Lemma 3 shows that the aforementioned *secret comparing function* $\mathfrak{F}$ is defined as follows, where $\mathfrak{F}(x, y) = \varnothing$ means $x \leq y$ otherwise $x > y$,

$$\mathfrak{F}(x, y) = \mathrm{HMAC}_g\left(\mathcal{N}\left(E^1(x)\right)\right) \cap \mathrm{HMAC}_g\left(\mathcal{N}\left(E^0(y)\right)\right). \tag{5}$$

Theorem 4. *Given n data items $D = \{d_1, d_2, \ldots, d_n\}$, $d_i \in D$ is the maximum of D if and only if the following condition satisfied:*

$$\forall d_j \in D \wedge d_j \neq d_i \left(HMAC_g\left(\mathcal{N}\left(E^1\left(d_j\right)\right)\right) \cap HMAC_g\left(\mathcal{N}\left(E^0\left(d_i\right)\right)\right) = \varnothing\right). \tag{6}$$

Proof. Suppose that the above condition is satisfied, for each $d_j \in D$ and $d_j \neq d_i$, we have $\mathrm{HMAC}_g(\mathcal{N}(E^1(d_j))) \cap \mathrm{HMAC}_g(\mathcal{N}(E^0(d_i))) = \varnothing$, and then $d_j \leq d_i$ can be derived due to Lemma 3. Therefore, we can see that d_i is not smaller than any other data items in D, which means that d_i is the maximum of D. □

On the basis of Theorem 4, the master node M performs the following steps to implement query processing.

(1) Load $\{\mathrm{HMAC}_g(\mathcal{N}(E^0(d_i))), \mathrm{HMAC}_g(\mathcal{N}(E^1(d_i))) \mid i \in \Gamma^t\}$ received from the queried sensor nodes whose IDs belong to Γ^t in epoch t.

(2) Find the encrypted data $(d_i)_{ki}$ whose corresponding *c-factors* satisfying the condition of Theorem 4 and transmit the following response message to the base station:

$$M \longrightarrow \text{base station} : \left\langle i, (d_i)_{ki}\right\rangle. \tag{7}$$

Upon receiving the above message, the base station loads the secret key k_i shared with s_i and then decrypts $(d_i)_{ki}$ to obtain the query result d_i, which is the maximum of the data item collected by the queried sensor nodes in epoch t.

4.4. Privacy Protection Analysis. As the privacy protection is the focus in this paper, we propose the privacy analysis about EMQP on the following two aspects.

(1) *Privacy of Collected Data.* According to the data submission protocol, the submitted information from each sensor node to the affiliated master node is not plaintext but encrypted and HMAC data. Since the HMAC function is with one-wayness and collision resistance properties, and sensor nodes only share the secret key with the base station, given $\mathfrak{R}(d_i)$ and $(d_i)_{ki}$, it is computationally infeasible for master nodes to obtain the value of d_i. Thus, the difficulty for master node to breach privacy is equal with cracking encryption and HMAC. Therefore, we have that EMQP can protect collected data items from master nodes.

(2) *Privacy of Query Result.* The query processing protocol shows that the master node can use the secret comparing function to obtain the query result, which is the maximum or minimum of data items collected by the queried sensor nodes. Because the secret comparing is built upon the HMAC data items and the collected data items are all encrypted for master node storage, it is also computationally infeasible for master nodes to obtain the value of the query result without keys. As a consequence, we have that EMQP is capable of preserving query result from master nodes.

Since [13] also uses similar HMAC and encryption to protect privacy, the capability of privacy preservation is the same between our work and [13].

4.5. Energy Consumption Analysis. In two-tiered WSNs, sensor nodes have limited energy resource while master nodes are abundant in energy. Therefore, the life time of network is mainly determined by the energy consumption of sensor nodes. In this section, we discuss the energy consumption of sensor nodes in our proposed schemes.

We assume that (1) a cell C have of n sensor nodes; (2) each epoch number and node ID are of l_t and l_{id} bits; (3) the average hops between a sensor node and M is L; (4) each collected data item is of w bits; (5) each encrypted and HMAC data item is of l_c and τ bits; (6) the energy consumed by encrypting and HMAC computing a data item are e_c and e_h; (7) the energy consumed by transmitting and receiving a data item are e_t and e_r.

The total energy consumption of sensor nodes is composed of two aspects, one is communication cost including sending and receiving messages and the other is computation cost such as encryption and HMAC computing. We use E_{total}, E_{sr}, and E_c to represent the total, communication, and computation energy consumption of the sensor nodes, then we have

$$E_{\text{total}} = E_{sr} + E_c. \tag{8}$$

As shown in DSP, each sensor node will encrypt the maximum or minimum of its collected data in an epoch and generate its 0-encoding and 1-encoding *c-factors* having w HMAC data items in total. The encrypted data and *c-factors* will both be transmitted to M. Then, we have

$$\begin{aligned} E_{sr} &= n \cdot \left(l_{id} + l_t + w \cdot \tau + l_c\right) \cdot \left(L \cdot e_t + (L-1) \cdot e_r\right), \\ E_c &= n \cdot e_c + n \cdot w \cdot e_h. \end{aligned} \tag{9}$$

According to (8), (9), we have

$$\begin{aligned} E_{\text{total}} = {} & n \cdot \left(l_{id} + l_t + w \cdot \tau + l_c\right) \cdot \left(L \cdot e_t + (L-1) \cdot e_r\right) \\ & + \; n \cdot e_c + n \cdot w \cdot e_h. \end{aligned} \tag{10}$$

In [13], for each senor node s_i and the local maximum or minimum data item d_i collected by s_i in epoch t, s_i will first generate the HMAC computed and numericalized prefix families of d_i and $[d_i, d_{\text{top}}]$, which are denoted as F and S. Here, the d_{top} is a very large number that is greater than any collected data items. Then s_i encrypts d_i, and the encrypted data will be transmitted to its closest master node along with the HMAC data sets F and S. According to [13], if d_i is of w bits, then F has $w+1$ HMAC data items and S has j HMAC data items, where $1 \le j \le 2w-2$. So the lower bound of transmitted HMAC data items is w+2, while the upper bound is $3w-1$. Since each sensor node computes and transmits the same encrypted data but more HMAC data items, [13] will consume more energy in sensor nodes comparing with our scheme. We will evaluate their energy consumptions in Section 6.

5. Energy Optimization

The above query scheme will consume much energy in sensor nodes because each sensor node needs to submit *c-factors* which consist of multiple HMAC data, and each HMAC data may have several bits such as 128 bits with HMAC-MD5 [26] or 160 bits with HMAC-SHA1 [27]. In this section, we focus on the *c-factor* compressing method with the basic idea to compress the HMAC data of *c-factors*, which can significantly reduce the communication cost in sensor nodes. As a result, the energy consumption of sensor nodes will be decreased and the lifetime will be promoted.

Assume that each HMAC data have τ bits and are randomly distributed in $I_H = \{0, 1, \ldots, 2^\tau - 1\}$. We use a simple hash function $\mathcal{H}$ to compress the HMAC data, which is defined as follows, where $x \in I_H$ and $\mu < \tau$,

$$\mathcal{H}(x) = x \bmod \left(2^\mu - 1\right). \tag{11}$$

After applying $\mathcal{H}$, each HMAC data of a *c-factor* can be converted to a fewer-bits number, which is called *c*HMAC data and is randomly distributed in $I_C = \{0, 1, \ldots, 2^\mu - 1\}$ because of the random distribution of HMAC data in I_H, such that the *c-factor* is compressed. Given a set of HMAC data X, we use $\mathcal{H}(X) = \{\mathcal{H}(x_i) \mid x_i \in X\}$ to represent the resulting set of *c*HMAC data after applying $\mathcal{H}$ to every items in X.

If the HMAC data is replaced with the corresponding *c*HMAC data in (5), we can get a similar secret comparing function $\mathfrak{F}'$ as follows, where x and y are two collected data items:

$$\begin{aligned} \mathfrak{F}'(x, y) = {} & \mathcal{H}\left(\text{HMAC}_g\left(\mathcal{N}\left(E^1(x)\right)\right)\right) \\ & \cap \mathcal{H}\left(\text{HMAC}_g\left(\mathcal{N}\left(E^0(y)\right)\right)\right). \end{aligned} \tag{12}$$

Lemma 5. *For two data items x and y, one has*

(1) *If $\mathfrak{F}'(x, y) = \varnothing$, then $x \le y$ must be true.*

(2) *If $\mathfrak{F}'(x, y) \neq \varnothing$, then $x > y$ may be true but with a certain false positive.*

Proof. For any $x_1, x_2 \in I_H$, if $x_1 = x_2$, there must be $\mathcal{H}(x_1) = \mathcal{H}(x_2)$, but if $x_1 \neq x_2$, there still may be $\mathcal{H}(x_1) = \mathcal{H}(x_2)$ when x_1 and x_2 modules $2^\mu - 1$ are equal, which is named collision. The above facts imply that, for any two HMAC data sets X and Y, if $X \cap Y \neq \varnothing$ then $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$, otherwise we may still have $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$. Therefore, if $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$, we may have $X \cap Y \neq \varnothing$ hold, otherwise $X \cap Y = \varnothing$ must be true. Assuming that $X$and $Y$are the *c-factors* of x and y where $X = \text{HMAC}_g(\mathcal{N}(E^1(x)))$ and $Y = \text{HMAC}_g(\mathcal{N}(E^0(y)))$, if $\mathfrak{F}'(x, y) = \mathcal{H}(X) \cap \mathcal{H}(Y) = \varnothing$, then we have $X \cap Y = \varnothing$ which implies $x \le y$ according to Lemma 3. But if $\mathfrak{F}'(x, y) = \mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$, then we may have $X \cap Y \neq \varnothing$ which implies $x > y$ may be true. □

As shown in Lemma 5 and its proof, there could have false positive in secret comparing while using the compressed *c-factors*, which is represented as the probability of a false decision in data comparison. And only if $X \cap Y = \varnothing$ but $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$ holds, the false decision is to be emerged. We denote the maximum false positive rate as Pr. In practical, if Pr is low enough, the *c-factor* compressing method is still acceptable. In the subsequent of this section, we will give the analysis of Pr in comparing two data items by Lemma 5.

We assume that each collected data item is of w bits, X and Y are the *c-factors* of data items x and y where $X = \text{HMAC}_g(\mathcal{N}(E^1(x)))$ and $Y = \text{HMAC}_g(\mathcal{N}(E^0(y)))$, and $X \cap Y = \varnothing$. Apparently, the more HMAC data that X and Y have, the higher probability that $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$ emerges.

Therefore, we assume that X and Y both have w HMAC data items, which is the upper bound of the quantity of HMAC data that each *c-factor* has, $X = \{x_1, x_2, \ldots, x_w\}$ and $Y = \{y_1, y_2, \ldots, y_w\}$. For each cHMAC data $x \in I_c$, there are at least $\lfloor 2^\tau/(2^\mu - 1)\rfloor$ HMAC data items in I_H whose results equal x when module $2^\mu - 1$, such as x, $x + (2^\mu + 1)$, $x + 2^*(2^\mu + 1)$. Supposing $\mathcal{H}(X)$ has δ cHMAC data items where $0 < \delta \le w$, there will be a set C having $\lfloor 2^\tau/(2^\mu - 1)\rfloor \cdot \delta$ HMAC data items, which satisfies $\mathcal{H}(c_i) \in \mathcal{H}(X)$ for each $c_i \in C$. Therefore, for each $y_i \in Y$, the probability of $\mathcal{H}(y_i) \notin \mathcal{H}(X)$ is equal with $y_i \notin C$, which is $1 - \lfloor 2^\tau/(2^\mu - 1)\rfloor \cdot \delta/2^\tau$, and only if $\mathcal{H}(y_i) \notin \mathcal{H}(X)$ then $\mathcal{H}(X) \cap \mathcal{H}(Y) = \varnothing$, otherwise we have $\mathcal{H}(X) \cap \mathcal{H}(Y) \neq \varnothing$ with the probability $1-(1 - \lfloor 2^\tau/(2^\mu - 1)\rfloor \cdot \delta/2^\tau)^w$. It is apparent that the probability will reach the maximum when $\delta = w$. As a result, we have

$$\Pr = 1 - \left(1 - \frac{\lfloor 2^\tau/(2^\mu - 1)\rfloor \cdot w}{2^\tau}\right)^w. \tag{13}$$

If the 128-bit HMAC-MD5 is used ($\tau = 128$), we have the results of the impact of w and μ on Pr as shown in Figure 4, which indicate that Pr could be very low if an appropriate μ is chosen. For instance, assuming that $w = 16$ and $\mu = 24$, then we have $\Pr = 1.53 * 10^{-5}$. Obviously, such low false positive rate rarely affects the result of secret comparing.

6. Performance Evaluation

To evaluate the performance of our proposed EMQP and the current work [13], which is denoted as PMQP, we implement both schemes and perform energy consumption comparison on the simulator of [28] with the same data set as [13] which is from Intel Lab [29]. We use PMQP(bot) and PMQP(top) to represent the lower and upper bounds of energy consumption of PMQP, and EMQP(bas) and EMQP(opt) to represent energy consumption of EMQP before and after hash-based optimization. We carry out evaluations on a MAX query in a cell with n sensor nodes and a master node, and we consider the following two aspects: firstly, the total energy consumption E_{total} of sensor nodes in EMQP and PMQP will be given, while the communication and computation energy cost E_{sr} and E_c in EMQP(opt) will be secondly measured as EMQP(opt) is the most energy-saving scheme.

The evaluations are conducted on a PC with a P4 3.0 GHz CPU and 512 MB memory running Ubuntu operating system. The placement of sensors nodes of a cell follows a uniform distribution over a two-dimensional region covering a $100 \times 100\,\text{m}^2$ area, and the radius of sensor communication is assumed as 10 m. According to [30], the energy consumed by transmitting and receiving 1-bit data in wireless communication are computed as follows: $e_t = \alpha + \gamma \times d^n$ and $e_r = \beta$, where d is the distance to which a bit is being transmitted, m is the path loss index, α and β capture the energy dissipated by the communication electronics, and γ represents the energy radiated by the power-amp. In our simulation, the values for these parameters as in [30] are adopted as follows: $\gamma = 10\,\text{pJ/bit/m}^2$, $\alpha = 45\,\text{nJ/bit}$, $\beta = 135\,\text{nJ/bit}$, and $m = 2$. In addition, we assume that the energy of encrypting a data item is adopted as $e_c = 8.92\,\mu\text{J}$ which is from [31], where using RC4 for encryption in TelosB, and the energy of HMAC computing a data item is assumed to be equal with encryption for simplicity. Other default parameters are summarized in Table 1.

Table 1: Default evaluation parameters.

Para.	n	w	l_{id}	l_t	l_c	τ	μ
Val.	480	10 bits	32 bits	32 bits	128 bits	128 bits	24 bits

In each measurement, we randomly distribute the sensor nodes and generate 20 networks with different topologies which are represented by different network IDs. The total energy including communication and computation costs of each measurement is the average of 20 networks.

(1) *Energy Consumption versus Network ID.* Figure 5(a) shows that E_{total} of EMQP and PMQP are both uniformly distributed in different networks, but E_{total} of EMQP is obviously lower than PMQP. In detail, compared with the lower bound of PMQP, the EMQP before optimization saves about 15% energy in average, and about 75% is saved after optimization. The main reason is that PMQP needs to submit and compute more HMAC data than EMQP, and the optimization of EMQP converts every HMAC data to a lower-bits cHMAC data, which significantly reduces the communication cost.

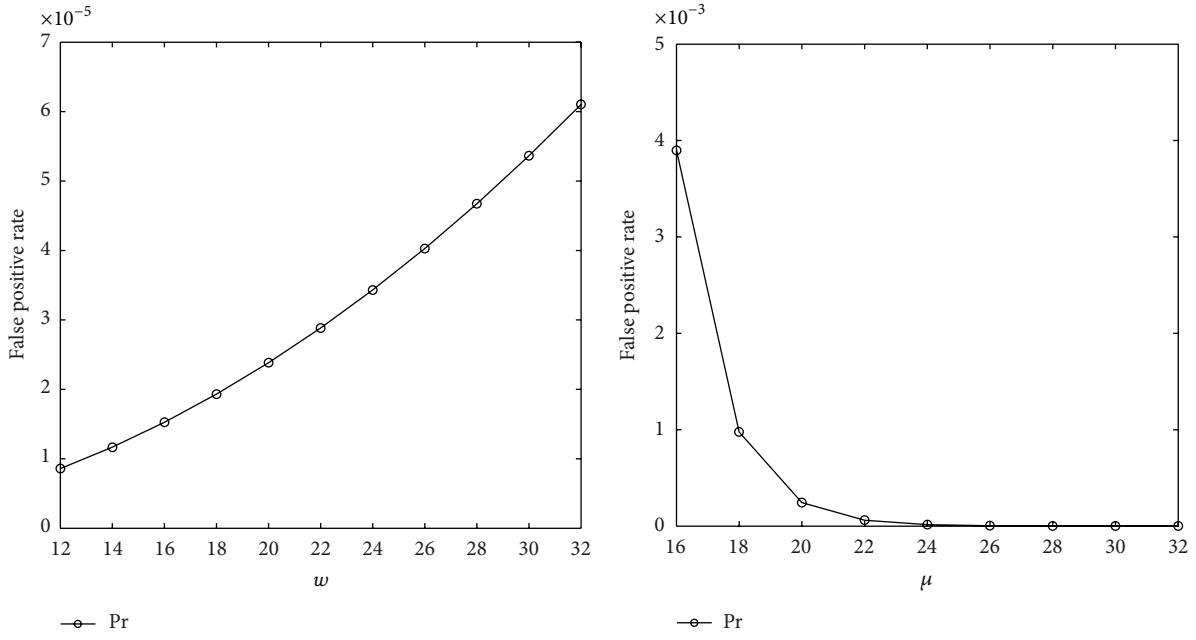
Figure 5(b) indicates that E_{sr} and E_c in the optimized EMQP are also both uniformly distributed, but E_{sr} is much higher than E_c. The energy consumption is almost entirely covered by communication that consumes more than 99% energy in total consumption.

(2) *Energy Consumption versus n and w.* Figures 6(a) and 7(a) both show that E_{total} of EMQP and PMQP are both increased as n and w increasing, but E_{total} of EMQP is always lower than PMQP. In detail, the EMQP before and after optimization save about 12% and 76% energy in average. The reason is similar with (1) in this section. Although E_{sr} and E_c in the optimized EMQP are both increased as n and w increasing, such increments are both very inconspicuous as shown in Figures 6(b) and 7(b), since the computation only covers little part (no more than 1%) in total energy consumption.

According to the above evaluations, we can conclude that our proposed EMQP are more energy efficient than the current PMQP. Particularly, the optimized EMQP has a significant saving (about 75%) in energy consumption even compared with the lower bound of PMQP. And communication costs much more energy than computation (more than 99%).

7. Conclusion

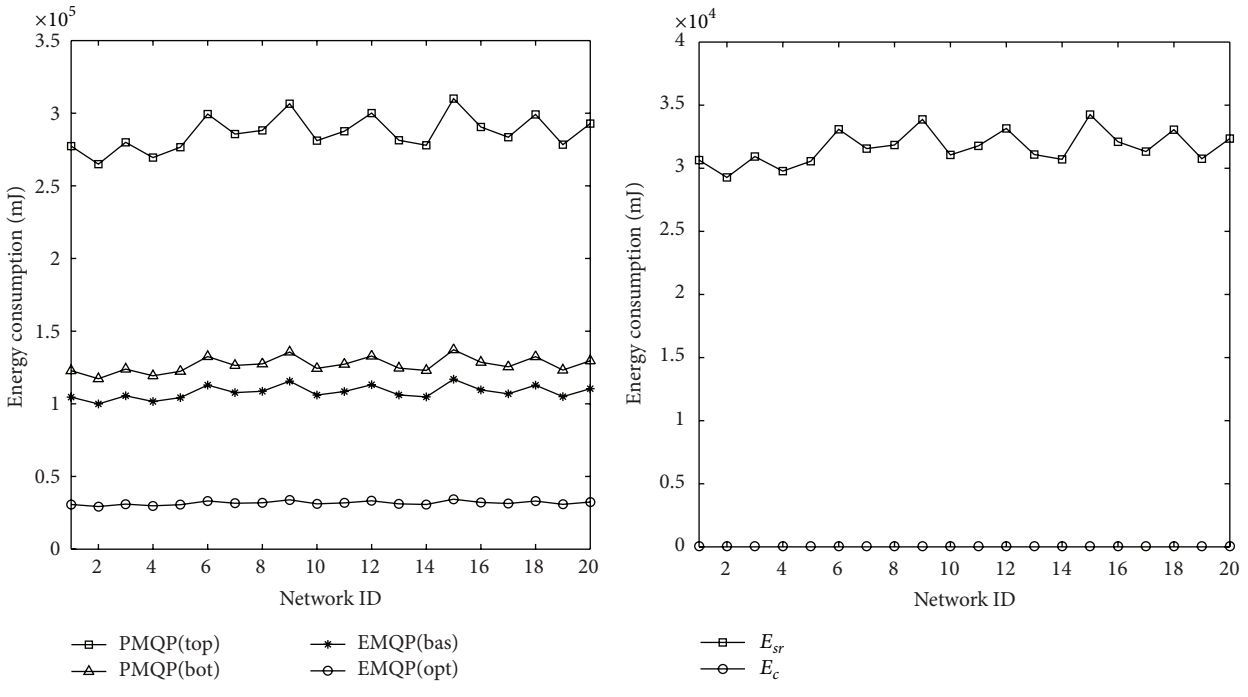
As the wireless sensor networks are deployed and used in many important areas, preserving the privacy of sensitive collected data items during query processing is a critical problem in sensor network applications. In this paper, we propose EMQP, a novel and energy-efficient protocol for handling privacy-preserving MAX/MIN queries in two-tiered sensor networks. To implement privacy-preserving



(a) Pr versus w ($\mu = 24$)

(b) Pr versus μ ($w = 16$)

Figure 4: Impacts of w and μ on Pr.

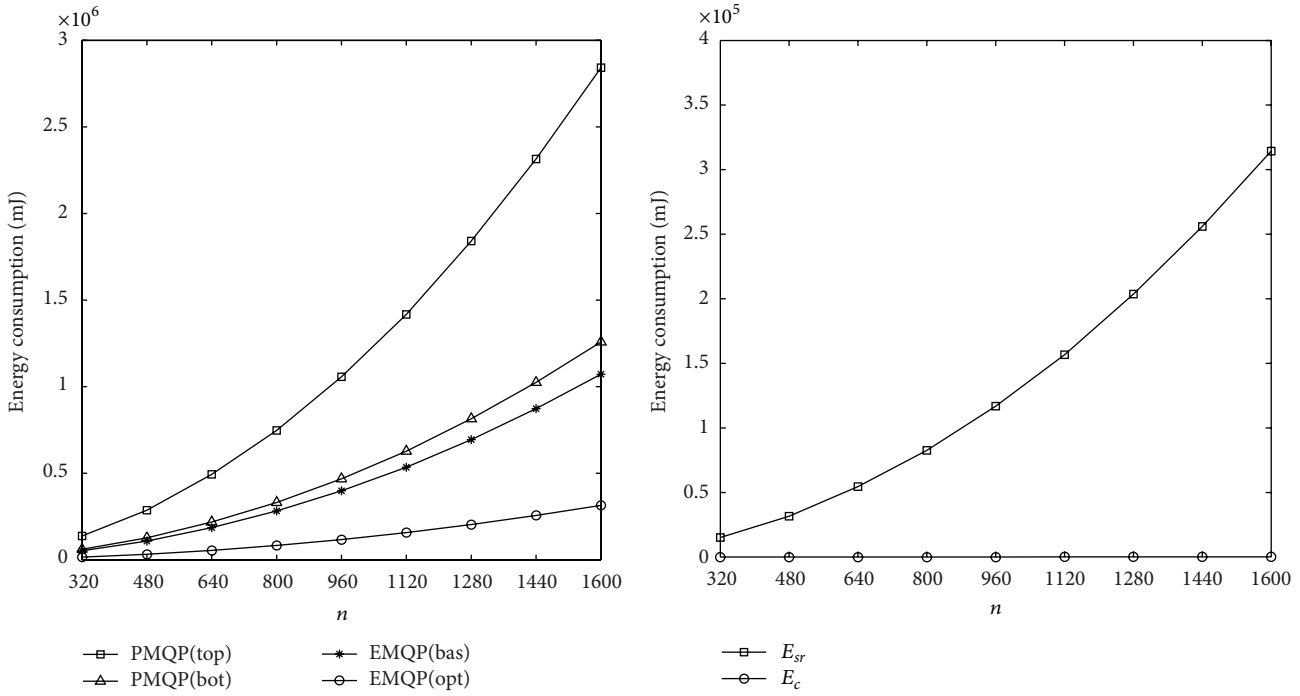


(a) E_{total} versus network ID

(b) E_{sr}, E_c versus network ID in EMQP(opt)

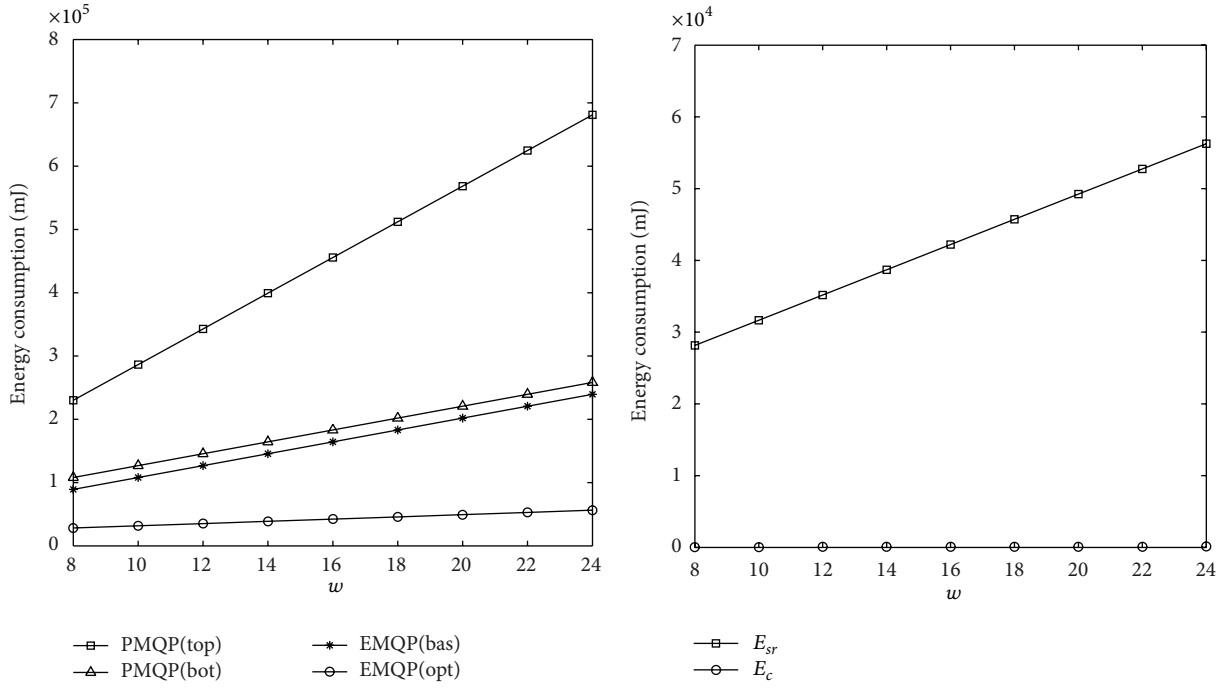
Figure 5: Impact of network ID on energy consumption.

MAX/MIN query processing without exposing the real value of collected data to master nodes, the technique of 0-1 encoding verification and encryption are applied. Furthermore, we also give a hash-based optimization for saving more energy of the resource-limited sensor nodes. The result of our evaluations shows that the proposed EMQP has a better performance than the current work in energy consumption. Under our optimized circumstance, in comparison with the lower bound of the current work, the EMQP has a significant improvement in energy saving. Last but not least,



(a) E_{total} versus n (b) E_{sr}, E_c versus n in EMQP(opt)

Figure 6: Impact of n on energy consumption.



(a) E_{total} versus w (b) E_{sr}, E_c versus w in EMQP(opt)

Figure 7: Impact of w on energy consumption.

communication costs much more than computation in all consumptions.

Acknowledgments

This research is supported by the National Key Basic Research Program (973 Program) of China under the Grant no. 2011CB302903, the National Natural Science Foundation of China under the Grants nos. 61272084, 61202004, 61201163, and 61202353, the Specialized Research Fund for the Doctoral Program of Higher Education under the Grant nos. 20113223110003 and 20093223120001, the Key Project of Natural Science Research of Jiangsu University under the Grant no. 11KJA520002, and the Natural Science Foundation of Jiangsu Province under the Grants nos. BK2011754 and BK2011072, the Introduction of Talent Research Foundation of Nanjing University of Posts and Telecommunications under the Grant no. NY211043, and the Priority Academic Program Development Foundation of Jiangsu Higher Education Institutions (Information and Communication yx002001).

References

[1] O. Gnawali, K.-Y. Jang, J. Paek et al., "The tenet architecture for tiered sensor networks," in *Proceedings of the 4th ACM International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 153–166, ACM, November 2006.

[2] P. Desnoyers, D. Ganesan, and P. Shenoy, "TSAR: a two tier sensor storage architecture using inteval skip graphs," in *Proceedings of the 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 39–50, San Diego, Calif, USA, 2005.

[3] B. Sheng and Q. Li, "Verifiable privacy-preserving range query in two-tiered sensor networks," in *Proceedings of the 27th IEEE International Conference on Computer Communications*, pp. 46–50, IEEE, Piscataway, NJ, USA, 2008.

[4] B. Sheng and Q. Li, "Verifiable privacy-preserving sensor network storage for range query," *IEEE Transactions on Mobile Computing*, vol. 10, no. 9, pp. 1312–1326, 2011.

[5] J. Shi, R. Zhang, and Y. Zhang, "Secure range queries in tiered sensor networks," in *Proceedings of the IEEE 28th Conference on Computer Communications (INFOCOM '09)*, pp. 945–953, IEEE, Piscataway, NJ, USA, April 2009.

[6] J. Shi, R. Zhang, and Y. Zhang, "A spatiotemporal approach for secure range queries in tiered sensor networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 1, pp. 264–273, 2011.

[7] F. Chen and A. X. Liu, "SafeQ: secure and efficient query processing in sensor networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '10)*, IEEE, Piscataway, NJ, USA, March 2010.

[8] F. Chen and A. X. Liu, "Privacy and integrity preserving range queries in sensor networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 6, pp. 1774–1787, 2012.

[9] M. Yoon, Y. K. Kim, and J. W. Chang, "A new data aggregation scheme to support energy efficiency and privacy preservation for wireless sensor networks," *International Journal of Security and its Applications*, vol. 7, no. 1, pp. 129–142, 2013.

[10] Z. Chen, G. Yang, L. Chen et al., "An algorithm for data aggregation scheduling with long-lifetime and low-latency in wireless sensor networks," *International Journal of Future Generation Communication and Networking*, vol. 5, no. 4, pp. 141–152, 2012.

[11] G. Yang, S. Li, X. Xu et al., "Precision-enhanced and encryption-mixed privacy-preserving data aggregation in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 427275, 12 pages, 2013.

[12] C.-M. Chen, Y.-H. Lin, Y.-C. Lin, and H.-M. Sun, "RCDA: recoverable concealed data aggregation for data integrity in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, pp. 727–734, 2012.

[13] Y. Yao, N. Xiong, J. H. Park, L. Ma, and J. Liu, "Privacy-preserving max/min query in two-tiered wireless sensor networks," *Computers and Mathematics with Applications*, vol. 65, no. 9, pp. 1318–1325, 2012.

[14] S. Ratnasamy, B. Karp, S. Shenker et al., "Data-centric storage in sensornets with GHT, a geographic Hash Table," *Mobile Networks and Applications*, vol. 8, no. 4, pp. 427–442, 2003.

[15] P. Desnoyers, D. Ganesan, H. Li et al., "PRESTO: a predictive storage architecture for sensor networks," in *Proceedings of the 10th Workshop on Hot Topics in Operating Systems*, pp. 1–6, Santa Fe, NM, USA, 2005.

[16] Stargate Gateway(sp400), http://www.xbow.com/.

[17] Rise Project, http://www.cs.ucr.edu/~rise/.

[18] H. Hacigümüş, B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over encrypted data in the database-service-provider model," in *Proceedings of the ACM SIGMOD International Conference on Managment of Data*, pp. 216–227, New York, NY, USA, June 2002.

[19] B. Hore, S. Mehrotra, and G. Tsudik, "A privacy-preserving index for range queries," in *Proceeding of 30th Very Large Data Bases Conference (VLDB '04)*, pp. 720–731, 2004.

[20] J. Cheng, H. Yang, S. H. Y. Wong, P. Zerfos, and S. Lu, "Design and implementation of cross-domain cooperative firewall," in *Proceedings of the 15th IEEE International Conference on Network Protocols (ICNP '07)*, pp. 284–293, IEEE, Piscataway, NJ, USA, October 2007.

[21] A. X. Liu and F. Chen, "Collaborative enforcement of firewall policies in virtual private networks," in *Proceedings of the 27th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, pp. 95–104, ACM, New York, NY, USA, August 2008.

[22] H. Y. Lin and W. G. Tzeng, "An efficient solution to the millionaires'problem based on homomorphic encryption," in *Proceedings of the 3rd International Conference on Applied Cryptography and Network Security*, pp. 97–134, New York, NY, USA, 2005.

[23] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, pp. 160–164, Chicago, Ill, USA, 1982.

[24] Y.-K. Chang, "Fast binary and multiway prefix searches for packet forwarding," *Computer Networks*, vol. 51, no. 3, pp. 588–605, 2007.

[25] H. Krawczyk, R. Canetti, and M. Bellare, "HMAC: keyed-hashing for message authentication," Tech. Rep. RFC 2104, Internet Society, Reston, Va, USA, 1997.

[26] R. Rivest, "The MD5 message-digest algorithm," Tech. Rep. RFC 1321, Internet Society, Reston, Va, USA, 1992.

[27] D. Eastlake and P. Jones, "US secure hash algorithm 1 (SHA1)," Tech. Rep. RFC 3174, Internet Society, Reston, Va, USA, 2001.

[28] A. Coman, J. Sander, and M. A. Nascimento, "Adaptive processing of historical spatial range queries in peer-to-peer sensor networks," *Distributed and Parallel Databases*, vol. 22, no. 2-3, pp. 133–163, 2007.

[29] Samuel M. Intel lab data, http://db.csail.mit.edu/labdata/labdata.html.

[30] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.

[31] M. M. Groat, W. Hey, and S. Forrest, "KIPDA: K-indistinguishable privacy-preserving data aggregation in wireless sensor networks," in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2024–2032, IEEE, Piscataway, NJ, USA, April 2011.