SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks (2016)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1594

RESEARCH ARTICLE

A group key-policy attribute-based encryption with partial outsourcing decryption in wireless sensor networks

Qihua Wang[1,2], Chang Wu Yu[3], Fagen Li[2], Huaqun Wang[4]* and Lijie Cao[1]

[1] Department of Information Engineering, Dalian Ocean University, Dalian, 116023, China
[2] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China
[3] Department of Computer Science and Information Engineering, Chung Hua University, HsinChu, 300, Taiwan
[4] College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China

ABSTRACT

Outsourcing decryption that enables the authorized users to obtain the original data without decryption computation is crucially important for wireless sensor networks in public data center. The existing outsourcing decryption schemes have been designed based on key-policy attribute-based encryption. The security of outsourcing decryption cannot be guaranteed, because the data center is not loyal, and existing schemes have high computational complexity and energy consumption. In this work, a novel partially outsourcing decryption scheme is proposed to guarantee data security and computational efficiency for resource-constrained sensor nodes and terminal equipments. According to the attributes of cluster nodes in the proposed scheme, the encryption secret key is encrypted based on group key-policy attribute-based encryption and sent to data center, and authorized users who satisfy the attributes of the ciphertext can obtain the secret key to decrypt the ciphertext. Furthermore, in order to reduce the decryption overhead for users, the authorized users can simply decrypt the transformation ciphertext that is partially decrypted by the data center using token key. Compared with the previous decryption schemes, the proposed scheme efficiently decrypts ciphertext and enhances security of the data. The simulation results also indicate that the proposed scheme is efficient in terms of energy consumption and computation by comparing to previous work. Copyright © 2016 John Wiley & Sons, Ltd.

KEYWORDS
attribute-based encryption; wireless sensor networks; outsourcing decryption; access control

***Correspondence**
Huaqun Wang, College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China.
E-mail: wanghuaqun@aliyun.com

1. INTRODUCTION

With the rapid growth of networking and computing technology, outsourcing decryption [1] has become a reality. Security of outsourcing decrypted data in wireless sensor networks (WSNs) has received much attention in recent years [2]. Sensor nodes in WSN can be randomly deployed in various environments, such as military surveillance, target tracking, health care, real-time monitoring system, etc. One of the biggest challenges in WSN is how to store and access the collected data. When a large number of sensitive data is stored in WSN, one fundamental problem is how to guarantee the security of collected information and data.

Key-policy attribute-based encryption (KP-ABE) [3,4] can provide a flexible and scalable access control strategy for the original data in WSNs. If sensing data is encrypted with KP-ABE by each sensor node, it requires too many time and energy consumption. In order to reduce the computational cost of the users or the terminal equipments, the data center that has super strong computing power can complete the outsourcing decryption task. However, there are some drawbacks for outsourcing decryption in public data center. The main drawback is that untrusted public data center may modify the decrypted data to save storage space. Or, unauthorized intruders of public data center may obtain or tamper the data [5]. Thus, it is necessary to study partially outsourcing decryption in public data center.

In order to ensure security of sensitive data, this work proposes an efficient outsourcing decryption scheme based on group KP-ABE. In group KP-ABE method, authority sends public key to the cluster head, not for each sensor node distribution. It thus can reduce network traffic while ensuring data security and prolonging the network lifetime. Furthermore, partial outsourcing decryption scheme is adopted because data center and authority are semi-trusted.

To the best of our knowledge, there is little work about outsourcing partial decryption for KP-ABE in WSNs. This paper proposes some novel methods for outsource decryption of KP-ABE system. In the proposed methods, authorized users provide the data center with a transformation key, and data center can decrypt partial data without the knowledge of any original data. Consequently, the authorized users can simply decrypt ciphertext that is partially decrypted by data center. Simulation results show that the computation cost of a receiver is greatly reduced, and the security is also reinforced efficiency.

The rest of this work is organized as follows. Section 2 mentions related previous work. Section 3 introduces the cryptographic backgrounds, and Section 4 describes access control construction with partial outsourcing decryption, followed with the performance analysis of the proposed scheme in Section 5. Finally, Section 6 concludes this paper.

2. PREVIOUS WORK

In order to efficiently realize data confidentiality and access control, symmetric key cryptography (SKC) and public key cryptography (PKC) schemes are used in WSN. In SKC, data senders and data receivers share the same key. However, SKC cannot manage the shared symmetric key, so it is hard to deal with the fine-grained data access control. Applications of the elliptic curve cryptography in WSN [6] reveal that PKC can be used in WSN. For example, identity-based encryption is an important PKC scheme [7].

Fine-grained access control is one-to-many encryption in WSN [3]. In some practical applications, only authorized users can access the specific data. The problem of specifying access rights for a particular user is called fine-grained access control. Access control method using symmetric key techniques was proposed by Subramanian et al. [8]. The main idea is that the lifetime of wireless sensor nodes is divided into several phases. The encrypted data from sensor nodes at a certain phase can be decrypted by an authorized user. When the sensor nodes are deployed in the complex environment, it is hard to know the exact information about the intended receivers. This paper concerns that wireless sensor nodes can encrypt data without knowing the exact information of receivers. Moreover, fuzzy identity-based encryption was proposed by Sahai and Waters [9]. ABE was classified into two types according to access control rules: KP-ABE and ciphertext-policy ABE (CP-ABE).

In order to deal with the drawbacks of the symmetric encryption in WSN, Yu et al. [10] proposed a public key scheme to realize the fine-grained access control. One trusted authority is responsible for the distribution of the private key and public key. If the authority is malicious, the security of encrypted data is threatened. Sushmita [11] then proposed the encryption in multiple authorities of WSN. This work improves the security of key distribution, but the proposed key management faces with serious challenges. Junbeom [12] proposed an access control scheme using KP-ABE for WSNs. They realized fine-grained access control by using the proxy encryption strategy. The author in reference [12] proposed an access control scheme using KP-ABE with efficient attribute and user revocation capability. It was assumed that sensor network is composed of high-end sensor devices. A mobile entity is used to collect data from each sensor node. The proposed method improved the lifetime of network and solved availably the revocation of attribute and user. One drawback of the KP-ABE policy is that encryption computational costs scale with the number of attributes. If each sensor node encrypts data using KP-ABE policy, the resource-constrained sensor nodes will be under a high burden. Nevertheless, in the reality, sensor nodes are deployed in a special environment or a place where no one is watching. It is not suitable for ordinary WSN in practical application, especially large networks. In reference [13], authors addressed online/offline ABE scheme for the resource-constrained devices, but they did not give outsourced decryption scheme in the details.

Similar to the case in [11], this work considers the following situation in a monitoring region in WSNs. The sensor nodes are randomly deployed, but they can move in certain range. They are used to sense and gather data information about fog or wind, which is called *type attributes*. The wireless sensor nodes are deployed in different terrain features such as sand, water, road and forest, which are called *location attributes*. The whole monitoring area can be divided into smaller area such as reg_1, reg_2, …, reg_{10}, which are called *region attributes*. The set of attributes are only given to some cluster nodes by the authority. The public key of cluster nodes can be obtained from the authority. An access control policy is given to each user by authority. In Figure 1, an example as the access tree structure is shown [10]. An authorized user with access

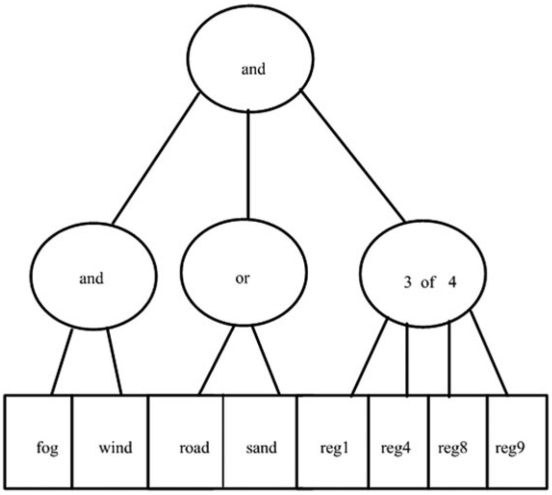

Figure 1. User access structures.

control structure matching sensor attributes can decrypt the encrypted data from data center. Notice that the user decrypts data sent by sensor nodes that are deployed in any three out of the four areas reg_1, reg_4, reg_8 and reg_9 according to the policy.

The scheme proposed in [11] is completely collusion resistant and supports revocation of attributes and users. If users cannot satisfy the access control, they cannot obtain any plaintext. However, when each key changes, authorization center sends public key to each sensor. The main problem is that the ciphertext size and time for decryption of these ABE systems grow with the size of the policy. In addition, the scheme in [11] mainly considered static sensor nodes. In this work, we focus on sensor nodes that can move in a certain range.

Literature in [1,2,14–16] proposed outsourcing decryption strategies that assumed that the data storage center had sufficient computing power and enough energy. The user sends its transformed secret key to data center. The data center decrypted partial ciphertext and then transmitted the partially decrypted data to users. Literature in [14,15] proposed CP-ABE data sharing scheme with outsourcing decryption. In [16], Wang et al. focused on fairly retrieving encrypted private medical records outsourced on remote untrusted cloud servers in the case of disputes in medical and accidents. Qiu et al. [17] proposed a group key management scheme based on the key-insulated encryption. In CP-ABE scheme, the sender specifies access to the encryption policy; but in WSNs, sensor nodes do not know the receiver of encrypted data in advance. Adding access control structure increases the energy consumption of the nodes in the ciphertext, and the decryption scheme based on the CP-ABE outsourcing part is not suitable for WSNs.

3. CRYPTOGRAPHIC BACKGROUNDS

The proposed partial outsourcing decryption scheme is based on group KP-ABE. Here, we introduce outsourcing decryption scheme based on group KP-ABE into WSNs as follows.

3.1. Lagrange interpolating basic theorem

Shamir's secret sharing scheme, which proposed a secret sharing scheme [18] for access control, will be applied and described as follows. In [18], central authority maintains a secret polynomial:

$$f(y) = a_0 + a_1 y + \cdots + a_{t-1} y^{t-1} \tag{1}$$

At the initial stage, each user U_i (U_i denotes the identity of the user) can get a secret share z_i, where $z_i = f(U_i)$.

Note that t secret shares can reconstruct the secret polynomial by Lagrange interpolation:

$$f(y) = \sum_{i=1}^{t} z_i \prod_{j=1, j\neq i}^{t} \frac{U_j - y}{U_j - U_i} \tag{2}$$

When $y = 0$, we can get the secret, i.e.,

$$\sum_{i=1}^{t} z_i l_i = a_0. \tag{3}$$

Here l_i denotes the Lagrange coefficient that is determined as

$$l_i = \prod_{j=1, j\neq i}^{t} \frac{U_j}{U_j - U_i}. \tag{4}$$

3.2. Bilinear maps

Let G_0 and G_1 be two multiplicative cyclic groups with the same prime order p. Let g be a generator of G_0. Let e: $G_0 \times G_0 \rightarrow G_1$ be a bilinear map that has the following property [11]:

(1) Bilinearity: $\forall q, r \in G_0$ and $a, b \in \mathbb{Z}_p$, $e(q^a, r^b) = e(q, r)^{ab}$.
(2) Non-degeneracy: $\exists q, r \in G_0$, such that $e(r, q) \neq 1$.
(3) Computability: $\forall q, r \in G_0$, there exists an efficient algorithm to calculate $e(q, r)$.

We say that G_0 is a bilinear group if the group operation in G_0 and the bilinear mapping e: $G_0 \times G_0 \rightarrow G_1$ are both computable efficiently. Notice that the bilinear mapping e is symmetric as $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

3.3. Hard problems

According to the bilinear map of cryptographic background, its security assumption is given as the following [12]:

(1) Computational Diffie–Hellman problem: Given (g, g^a, g^b) for some $a, b \in \mathbb{Z}_q^*$, compute g^{ab}.
(2) Decisional bilinear Diffie–Hellman (BDH): Given $a, b, c \in \mathbb{Z}_q^*$ at random, the decisional BDH assumption is that there is no probabilistic polynomial time algorithm A to distinguish the tuple $(A = g^a, B = g^b, C = g^c, e(g, g)^{abc})$ from the tuple $(A = g^a, B = g^b, C = g^c, e(g, g)^z)$ with a non-negligible probability. The probability advantage of A is

$$\left| \Pr\left[\mathrm{A}\left(A, B, C, e(g, g)^{abc}\right) = 1\right] - \Pr[\mathrm{A}(A, B, C, e(g, g)^z)] = 1 \right|$$

(3) Bilinear Diffie–Hellman problem: Given (g, g^a, g^b, g^c) for some $a, b, c \in \mathbb{Z}_q^*$, compute $e(g, g)^{abc}$.

3.4. Key-policy attribute-based encryption

Definition (Access structure) Let $\{p_1, p_2, \cdots, p_n\}$ be a set of parties. A collection $A \subseteq 2^{\{p_1, p_2, \cdots, p_n\}}$ is *monotone* if $B \in A$ and $B \subseteq C$ imply that $C \in A$. An access structure is a monotone collection $A \subseteq 2^{\{p_1, p_2, \cdots, p_n\}}$ of non-empty subsets of $\{p_1, p_2, \cdots, p_n\}$. Sets in A are called *authorized*, and sets not in A are called *unauthorized*.

In this subsection, we review KP-ABE proposed in literature [18,19]. KP-ABE scheme is composed of four parts:

(1) Setup (1^k): This algorithm inputs a security parameter k and returns a system master secret key MK and the public key PK.
(2) Encryption (m, γ, PK): This algorithm inputs a message m, the public key PK and a set of attributes γ and outputs the ciphertext E.
(3) Key generation (P, MK): This algorithm inputs an access control structure P, the public key PK, the master secret key MK and outputs a secret key SK that enables the user to decrypt a ciphertext message if and only if γ matches P.
(4) Decryption (E, D): This algorithm inputs the private key D and the ciphertext E that was encrypted by the attribute set γ, the public key PK and the access control structure P. It then outputs the message m if and only if the attribute set γ meets the user's access control structure P.

4. ACCESS CONTROL SCHEME CONSTRUCTION

In KP-ABE strategy, a set of attributes is associated with each ciphertext. The access control policy is associated with the private key. When the authorized user meets the condition, it can decrypt the ciphertext. In this section, inspired by KP-ABE strategy, we introduce new system architecture for access control. As shown in Figure 2, the system architecture for access control comprises five types of participants: system controller, sensor, cluster head, user and data center, which are categorized as follows.

(1) System controller: It is a central key authority that can generate public key and secret key for the senders and receivers. In addition, it can generate the access control strategy for every user. Here, we assume that the system controller is honest but curious. It cannot decrypt the ciphertext because ciphertext is established by data center and sensor cluster head.
(2) Sensor: Sensor nodes collect data from the monitoring region and send the symmetric encrypted data to its cluster head node.
(3) Cluster head: Each cluster head node is responsible for a number of nearby sensor nodes. Each sensor node encrypts data and sent it to the cluster head. Then, the cluster heads encrypt symmetric secret key according to its own attributes and then send the received data to the data storage center. In a circular way, we select cluster nodes at random so that the energy load of the whole network is distributed to every sensor node. Because the network energy consumption is balanced, this approach improves the lifetime of the entire network.
(4) User: It is an entity who wants to access the encrypted information stored in data center. If the

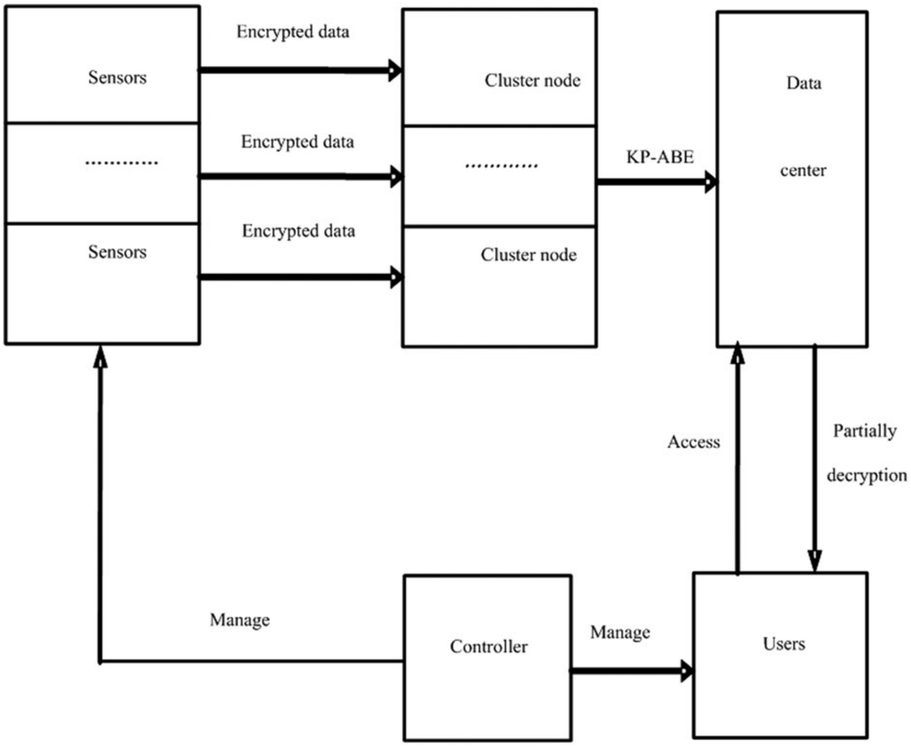

Figure 2. A new system architecture for access control.

Security Comm. Networks (2016) © 2016 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

access control policy satisfies a set of attributes in the encrypted ciphertext, it can obtain symmetric encryption key and decrypt the received sensor data. Moreover, in order to reduce the computing power for encryption, the user can generate the conversion key to data center.

(5) Data center: Data center stores the encrypted data that is sent by cluster nodes. It is in charge of controlling the access for users who provide corresponding data. Assume that the data center is honest but curious, and it also owns public key and private key to encrypt partial data. Legitimate users are allowed to access the data according to their own access control policy.

In this work, we consider a WSN composed of a network controller (that is a semi-trust mechanism), many users, a unique data center and many wireless sensor nodes. We denote the network controller, the universe of the users and the universe of wireless the sensor nodes with the symbol Υ, U and S, respectively. All users have their unique IDs as well as wireless sensor nodes. Symbols U_i and S_j represent user i and sensor node j, respectively. Symbol I denotes the universe of the attributes of all wireless sensor nodes in the given network. Specifically, symbol I_i denotes the set of attributes that is owned by every wireless sensor node i. Accordingly, we have $I = \cup_{\forall i \in S} I_i$. Let $\mathrm{O} = \max|I_i|$. The access control structure is generally denoted by P.

In this section, the detailed descriptions of the proposed access control scheme are introduced. The proposed scheme uses hash function and KP-ABE scheme to achieve data encryption. In hash function scheme, the information data is encrypted with hash functions. Encrypted data is sent it to cluster nodes by sensor nodes. In KP-ABE-based scheme, cluster nodes encrypt the master key with their attribute set and send the encrypted sensor data to data center. According to their access control policy, authorized users obtain the master key and get sensor data by calculating the secret key finally. The previous data process in WSN is shown in Figure 3.

This work assumes that sensor node can move, and data center and authority are semi-trusted. It means that the attributes of sensor nodes varies with the location of sensor nodes. Our scheme is based on the symmetric encryption and KP-ABE. Authority sends public key to the cluster head, not for each sensor node.

A brief idea about the proposed scheme is given. First of all, the given sensor network is divided into several groups, and a cluster head for each group is selected randomly; every sensor node joins the nearest cluster node and encrypts the data sending to the cluster head. According to their attributes, cluster nodes encrypt the encryption secret key and send to the data center. The user whose access control structure meets the conditions can get the secret key to decrypt the ciphertext, and then, data is encrypted. Inspired by outsourced decryption method literature in [14], the proposed scheme can be divided into six parts including system initialization, attribute key generation, sensor data and master key encryption, token generation, partially decryption and data decryption. The following subsections will describe them in detail sequentially.

4.1. System initialization

In this subsection, we discuss the initial stage of the whole system. First, we discuss how to select the cluster nodes.

In the form of circulation, the random choice of cluster node is based on two factors: (1) the total number of cluster nodes required in the network and (2) the number of sensor nodes that have become cluster nodes so far. Each sensor node randomly chooses a value (real number) between 0 and 1. If the selected value is less than a certain threshold, the node becomes cluster node [20]. The computation formula of threshold value $Q(i)$ is as follows:

$$Q(i) = \begin{cases} \dfrac{c}{1 - c\left(\varepsilon \mathrm{mod} \dfrac{1}{c}\right)} & i \in W \\ 0 & otherwise \end{cases} \tag{5}$$

Here, c represents the percentage of cluster nodes in WSNs, ε denotes the number of current rounds, and W denotes the set of not election of the cluster nodes in recent $\frac{1}{c}$ rounds. The following diagram based on clustering in WSN is shown in Figure 4.

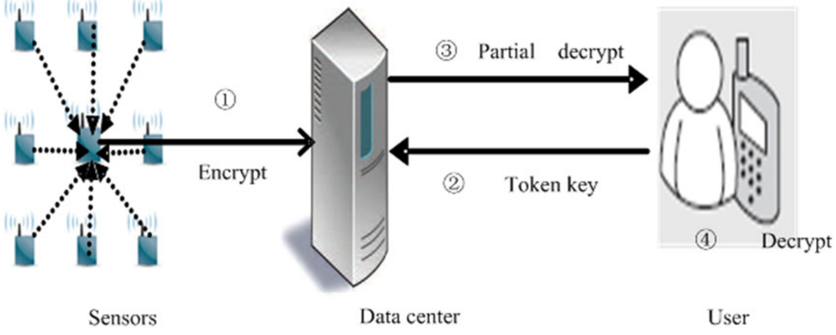

Figure 3. Data process in wireless sensor network.

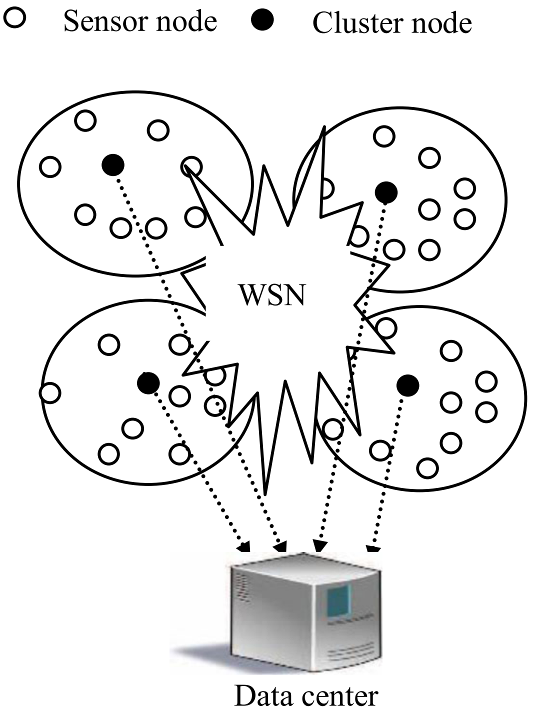

Figure 4. Clustering-based wireless sensor networks.

In the initial stage, the system controller and data center perform the following operations.

(1) The system controller chooses a number t_i uniformly at random from $\mathbb{Z}_p^*$ for each attribute $i \in I$ and y randomly from $\mathbb{Z}_p^*$ and outputs the public parameters PK and master secret key MK as follows:

$$PK = \left(G_0, g, Y = e(g,g)^y, T_1 = g^{t_1}, T_2 = g^{t_2} \cdots, T_{|I|} = g^{t_{|I|}}\right) \quad (6)$$

The master secret key is $MK = (y, t_1, t_2, \cdots, t_{|I|})$.

(2) The system controller chooses a secure one-way hash function, denoted as $h(.)$. Preload the following information to each cluster sensor node S_{Hi}: $\Upsilon \rightarrow S_{Hi} : I_i, h(.), PK$.

(3) The data center chooses a random number $r \in \mathbb{Z}_p^*$. Then, its private key and public key are given by $(PK_d = g^r, MK_d = h(ID_c)^r)$, where ID_c is the identity of the data storage center.

4.2. Attribute key generation

After setting up the PK and SK, system controller defines the access control tree for the whole attribute sets. It generates private keys for each user by a randomized algorithm.

For each user U_j, controller Υ generates an access tree structure P and computes its secret key SK in the following form: Starting from the root node ω of access tree P, in the top-down manner, we construct a random polynomial q_x of degree $d_x - 1$ using Lagrange interpolation for each node x in P, where d_x is the threshold value of node x. For each non-root node x in P, set $q_x(0) = q_{parent(x)}(index(x))$, where $parent(x)$ is the parent of x and $index(x)$ is the unique index number of x given by its parent. In particular, set $q_r(0) = y$. SK is output as follows:

$$SK = < \left\{ D_i = g^{\frac{q_{i(0)}}{t_i}} \right\}_{i \in P} > \quad (7)$$

Here, P_j denotes the access structure of user j. The following information is also preloaded.

$$\Upsilon \rightarrow U_j : P_j, SK, h(.), PK. \quad (8)$$

4.3. Sensor data and master key encryption

In many applications, sensor nodes can move, so attributes (such as location attributes and region attributes) of sensor nodes may change frequently. It brings much trouble to the management of system attributes. The attributes of nodes in the same cluster have great similarity. Assume that sensor nodes and cluster node have the same attributes. In each round, sensor S_i sends the encrypted sensor data to cluster node S_{Hi}, and the adopted encryption strategies are as follows:

(1) Each sensor node encrypts collected information in a symmetric encryption way. In every cluster, each sensor node calculates symmetric encryption key K_t where $K_t = h(K_{t-1})$, which can be obtained from the cluster head. In the initial state, we set $K_0 = K$.

(2) Each sensor node in its cluster encrypts the sensor data D with current symmetrical secret key K_t. Then, send the data item $< t, \{D\}_{K_t} >$ to cluster head, where $\{D\}_{K_t}$ represents the encrypted sensor data. In each round, the symmetrical secret key of each node are different, and it comes from the one-way hash function of cluster heads.

After receiving the message sent by the nodes in the cluster, according to its own properties, cluster head uses KP-ABE policy to encrypt the secret key and sends the data to data center. Every cluster head encrypts a new master key as follows:

(1) Each cluster head chooses a random number $\alpha \in \mathbb{Z}_p^*$ and then computes

$$K_S = e\left((g^r)^\alpha, h(ID_c)\right)$$

(2) In each round, each cluster head selects a number s uniquely at randomly and then calculates data item $E_i = T_i^s$ for each attribute $i \in I_i$. Finally, each cluster head constructs the set $\left\{E_i = T_i^s\right\}_{i \in I_i}$.

(3) Selects a number $K \in G_1$ as the master key of the key chain. Then, compute $C = KK_S Y^s$. Finally, obtain the ciphertext as follows:

$$E^{\mathrm{t}} = \left(\tilde{C} = KK_{\mathrm{S}}Y^{\mathrm{s}}, C = g^{\alpha}, \left\{E_{\mathrm{i}} = T_{\mathrm{i}}^{\mathrm{s}}\right\}_{\mathrm{i}\in\mathrm{I_i}}\right). \quad (9)$$

(4) Each cluster head sends data items $< E^t, \{D\}_{K_t} >$ to data storage center. In this phase, the data center and outside users can not know any information about the ciphertext. System controller cannot decrypt the ciphertext because E^{t} is blinded by K_{S}. It is established only by data center and sensor cluster head.

4.4. Token generation

When a user meeting the requirements to access structure wants to access the data from the data center, the user receives g^{α} from data storage center and generates the transformation key (token key). The user chooses a random $\tau \in \mathbb{Z}_{\mathrm{p}}^*$ and constructs the token key TK for the set of attributes as follows:

$$TK = \left(\forall j \in U_{\mathrm{j}}, \left(D_{\mathrm{j}}\right)^{\tau}, h(ID_{\mathrm{S}})\right) \quad (10)$$

where ID_{S} is the identity of the user. After building the token key, the user sends the transformation key TK to the data center and asks for the partial decryption.

4.5. Partial decryption

Assume that the access structure of one user satisfies the attributes of ciphertext that is encrypted by cluster head and sensor node. Data center can partially decrypt the sensor data $< E^t, \{D\}_{K_t} >$ generated by cluster sensor node S_{Hi} in tth round. Data center executes the following steps to obtain the sensor data CT' according to the transformation key TK:

(1) Without loss of generality, the control tree structure consists of leaf nodes and non-leaf nodes. In order to decrypt the master key K from E^{t}, the partial decryption process starts from the leaf nodes in the bottom-up manner. First, it computes the value F_{i} for each leaf node i in P as follows:

$$F_{\mathrm{i}} = \begin{cases} e\left((D_{\mathrm{i}})^{\tau}, E_{\mathrm{i}}\right) = e\left(g^{\frac{\mathrm{q_{i(0)}}}{\mathrm{t_i}}.\tau}, g^{\mathrm{s.t_i}}\right) = e(g,g)^{\mathrm{s}.\tau.\mathrm{q_i}(0)}, \mathit{if}\ i \in I_{\mathrm{i}} \\ \perp,\ \mathit{otherwise} \end{cases} \quad (11)$$

(2) Next, we consider the recursive case when i is a non-leaf node. Let S_x be an arbitrary d_x-sized set of child node x. If such set does not exist, F_i=⊥. Otherwise,

$$\begin{aligned} F_{\mathrm{x}} &= \prod\nolimits_{\mathrm{i}\in\mathrm{S_x}} F_{\mathrm{i}}^{\mathrm{l_i}(0)} = \prod\nolimits_{\mathrm{i}\in\mathrm{S_x}} \left(e(g,g)^{\mathrm{s}.\tau.\mathrm{q_{parent(i)}}(\mathrm{index(i)})}\right)^{\mathrm{l_i}(0)} \\ &= \prod\nolimits_{\mathrm{i}\in\mathrm{S_x}} e(g,g)^{\mathrm{s}.\tau.\mathrm{q_x}(\mathrm{i}).\mathrm{l_i}(0)} \\ &= e(g,g)^{\mathrm{s}.\tau.\mathrm{q_x}(0)}. \end{aligned} \quad (12)$$

where $l_i(0)$ denotes the Lagrange coefficient. The data center computes $K_{\mathrm{s}} = e(g^{\alpha}, MK_{\mathrm{d}}) = e(g^{\alpha}, h(ID_{\mathrm{c}})^{\mathrm{r}}))$ and obtains $C' = \tilde{C} / e(g,g)^{\mathrm{ys}\tau} = KK_{\mathrm{S}}Y^{\mathrm{s}} / e(g,g)^{\mathrm{ys}\tau}$. After the partial decryption, the data center sends the partial decryption ciphertext $CT' = (C', C = g^{\alpha}, A)$ to the user who wants to obtain the ciphertext and satisfies the attributes of ciphertext, where $A = e(g,g)^{\mathrm{ys}\tau}$.

4.6. Data decryption

The user obtains $e(g,g)^{\mathrm{ys}\tau} = Y^{\mathrm{s}\tau}$ from data center if and only if ciphertext attribute satisfies access control P. The user computes

$$\frac{C'}{(A)^{\frac{1}{\tau}}.K_{\mathrm{S}}} = \frac{KK_{\mathrm{S}}Y^{\mathrm{s}}}{\left(e(g,g)^{\mathrm{ys}\tau}.(A)^{\frac{1}{\tau}}.(e((g^{\mathrm{r}})^{\alpha}), h(ID_{\mathrm{c}})\right)} = K \quad (13)$$

When getting the encryption key K, the user gets the data encryption key $K_{\mathrm{t}} = h_{\mathrm{t}}(K)$ and decrypts the sensor data by $K_{\mathrm{t}} = h_{\mathrm{t}}(K)$ finally. In this part, the user only decrypts the CT', because the data center has partially decrypted the CT using TK. Consequently, the computation of the user can be greatly reduced.

5. PERFORMANCE ANALYSIS OF THE PROPOSED SCHEME

5.1. Collusion resistance and data confidentiality

The proposed scheme is collusion resistance when unauthorized users collude. Now, we can analyze the proposed scheme as follows: The master key K is encrypted by $e(g, g)^{\mathrm{ys}}$. In order to recover K, the user must cancel $e(g, g)^{\mathrm{ys}}$. To extract $e(g, g)^{\mathrm{ys}}$, the user has to obtain S that is a random number selected from $\mathbb{Z}_p^*$. Authorized users can obtain y. On the other hand, unauthorized user cannot give other users any information by computing $e(g, g)^{\mathrm{ys}}$. When the sensor data is encrypted based on KP-ABE, semi-trusted controllers and semi-trusted data centers cannot obtain any information about the ciphertext. The confidentiality of ciphertext data can be seen from Figure 5.

In the proposed scheme, the cluster node encrypts the message KK_{S} by using the form of $KK_{\mathrm{S}}Y^{\mathrm{S}}$ and sends all data to the data center. The controller is responsible for the distribution of the public key and the private key, but it cannot know any information about them because of

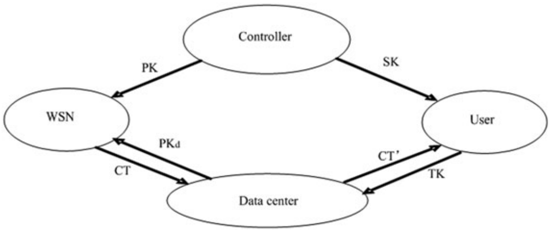

Figure 5. Flowchart of the proposed system.

the existence of K_S. In addition, data center encrypts the partial ciphertext when legitimate users send conversion key *TK*. To sum up, our scheme guarantees confidentiality of data and collusion resistance, and only a legitimate user can decrypt data completely.

We can use digital signature scheme for data authentication. In system initialization phase, the controller allocates the key pairs to the sensor nodes. The sensor node uses the private key to sign the data and then uses the symmetric key to encrypt the data. The sensor node sends the signature information and the encrypted data to the corresponding cluster node. In the same way, the cluster nodes can also sign the encryption key. In KP-ABE, the receiver that satisfies the attributes of the ciphertext can decrypt the data. Authorized users can verify the data. In [21], authors proposed a new cryptographic primitive called key-policy attribute-based ring signcryption scheme for wireless body area network that can ensure the authentication of data. In this paper, we do not offer information about the authentication of data. We can refer to the literature [21] in detail.

Theorem 1. Our key generation scheme can realize an access structure, and it guarantees the correctness and privacy of secret key.

Proof. The authorized sets can realize access structure $A_t = \{A \subseteq \{p_1, p_2, \cdots, p_n\} : |A| \geq t\}$, where $1 \leq t \leq n$ is an integer. In Shamir's scheme [18], the domain of secrets and shares is the elements of a finite field $\mathbb{F}_q$. Let $\alpha_1, \alpha_2, \cdots, \alpha_n \in \mathbb{F}_q$ be n distinct non-zero elements known to all parties. To share a secret $a_0 \in \mathbb{F}_q, t-1$ elements a_1, $a_2, \cdots, a_{t-1}$ from $\mathbb{F}_q$ are selected. These random elements together with the secrets define a polynomial $f(x) = k + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1}$. The share of p_j is $z_i = f(\alpha_i)$. The correctness and privacy of our scheme follow from the Lagrange's interpolation theorem [22]. For t distinct values $x_1, x_2, \cdots, x_t$ and any t values y_1, $y_2, \cdots, y_t$, there exists a unique polynomial f of degree at most $t-1$, such that $f(x_i) = y_i$ for each $i 1 \leq i \leq t$.

(1) Correctness: The set B has t points for the polynomial Q. We can reconstruct it using Lagrange's interpolation and compute $k = f(0)$. Formally, a set $B = \{p_{i1}, p_{i2}, \cdots, p_{it}\}$, $f(x)$ can be computed as follows.

$$f(x) = \sum_{l}^{t} z_{il} \prod_{1 \leq j \leq t, j \neq 1} \frac{\alpha_{ij} - x}{\alpha_{ij} - \alpha_{il}}$$

Thus, the parties in B reconstruct k by computing

$$f(0) = \sum_{l}^{t} z_{il} \prod_{1 \leq j \leq t, j \neq 1} \frac{\alpha_{ij}}{\alpha_{ij} - \alpha_{il}} = k$$

(2) Privacy: On the other hand, any unauthorized set B with $t-1$ parties holds $t-1$ points for the polynomial. Combined the possible secrets, a unique polynomial of degree at most $t-2$ can be determined. Formally, by the interpolation theorem, for set $B = \left\{p_{i1}, p_{i2}, \cdots, p_{i(t-1)}\right\}$, and any $a_0 \in \mathbb{F}_q$, there is a unique polynomial f with degree at most $t-2$ such that $f(0) = f(0) = a_0$ and $f(\alpha_{il}) = z_{il} \neq f(\alpha_{il}) = z_{il}, l \in [1, t-1]$. Hence, the following probability distribution can be obtained.

$$\Pr[f(\alpha_{il}) = z_{il}, (1 \leq l \leq t-1) | k = a_0] = \frac{1}{q^{t-1}}$$

Because this probability is the same for every $a_0 \in \mathbb{F}_q$ and it can be negligible, the privacy of secret follows.

Theorem 2. Assume that there exists an adversary A, which can break KP-ABE in a selective-set model under chosen plaintext attack, and then, a simulator E can play the DBDH problem with more than a negligible advantage.

The definition of selective-set model for KP-ABE is given in reference [19] in detail.

Proof. Assume a polynomial time adversary A can break the KP-ABE scheme in chosen plaintext attack with advantage ε. A simulator E can play the DBDH problem with advantage $\frac{1}{2}\varepsilon$.

First, the challenger builds the two groups G_1 and G_2 with a bilinear pair and generator g. The challenger selects a fair binary $\mu \in \{0, 1\}$, outside of simulation's view.

Init: The simulator E runs A. The adversary A chooses the set of attributes I that it wishes to be challenged.

Setup: If $i \in I$, the simulator chooses a random $r_i \in \mathbb{Z}_p$ and set $T_i = g^{r_i}$. It sets the parameter; the simulator B obtains the public parameters to A.

Phase 1: A makes requests for the keys corresponding to any access structures T such that the challenge attribute set I does not satitsfy T. If A makes a request for the secret key for an access structure T where $T(r) = 1$, in order to generate the secret key, **E** needs to design a polynomial function f_x of degree d_x for every node in the access tree T. Simulator defines the polynomial function $f_x(.) = b q_x(.)$ for every node. The key sharing for each leaf node is given by polynomial function f_x as follows:

$$D_x = \begin{cases} g^{\frac{f_x(0)}{t_i}} = g^{\frac{bq_x(0)}{r_i}} = B^{\frac{q_x(0)}{r_i}} & if\ att(x) \in \gamma, i = att(x) \\ g^{\frac{f_x(0)}{t_i}} = g^{\frac{bq_x(0)}{b\beta_i}} = g^{\frac{q_x(0)}{\beta_i}} & otherwise \end{cases}$$

From previous description, we know that the simulator can construct a private key for the access control structure.

In our KP-ABE scheme, let $C = MY^s = KK_S Y^s$ be the ciphertext. K_S can be computed as follows:

$$K_S = e((g^r)^\alpha, h(ID_c))$$

If we want to obtain symmetric encryption key, K_S must be calculated.

Challenge: The adversary A selects two challenge plaintext message m_0 and m_1 to the simulator. The simulator randomly selects $v \in \{0, 1\}$. The simulator returns an encryption of m_v to the adversary. The ciphertext E is calculated as

$$E = \left(I_i, \tilde{C}, = m_v Z, \{E_i = C\}_{i \in I_i}\right)$$

If $\mu = 0$ then $Z = e(g, g)^{abc}$. The ciphertext is an effective encryption of message m_v.

If $\mu = 1$ then $Z = e(g, g)^z$. From the adversary's view, the ciphertext E is a random element. It contains no information about m_v.

Phase 2: The simulator does exactly as it did in phase 1.

Guess: The adversary A submits a guess $m_{v'}$ of m_v to simulator that will output $\mu' = 0$; otherwise, it will output $\mu' = 1$. The total advantage of the simulator in the DBDH assumption is given by the following equation:

$$\frac{1}{2}\Pr\left[\mu' = \mu \middle| \mu = 0\right] + \frac{1}{2}\Pr\left[\mu' = \mu \middle| \mu = 1\right] - \frac{1}{2} = \frac{1}{2}\left(\frac{1}{2} + \varepsilon\right) + \frac{1}{2}\frac{1}{2} - \frac{1}{2} = \frac{1}{2}\varepsilon$$

Suppose that the construction of KP-ABE [19] is selective-set CPA secure, and then, the proposed outsourcing encryption scheme is also selective-set CPA secure.

PBC type A pairing is used for security analysis in [23]. Three kinds of parameters that represent 80-bit, 112-bit and 128-bit key size security level are used, respectively. Table I gives the specification for different security analysis. The type A pairing is given on the curve

$$y^2 \equiv (x^3 + x) \bmod q$$

In Table I, the order of G_0 is p.

Table I. Specification for different security analysis (bits).

Security level	Size of *q*	Size of *p*
80 Bits	512	160
112 Bits	1024	224
128 Bits	1536	256

5.2. Efficiency comparison

Partially data-outsourced decryption scheme not only ensures the safety of the data but also extends the life cycle of the network. The efficiency comparison is summarized in Table II. The notations used in the table are described as follows:

n_0	Bit size of an element in G_0.
n_1	Bit size of an element in G_1.
n_a	Bit size of an attribute string.
I	The number of attributes in system.
n_T	Bit size of an access tree T in the ciphertext.
T	The number of attributes appeared in access tree P.
N	The number of all numbers.
k	The number of attributes associated with a ciphertext.
n_k	Bit size of a KEK.
n_{cgk}	Bit size of public key in data center.
n_{cmk}	Bit size of private key in data center.

As is shown in Tables II and III, the proposed scheme is different from the previous schemes. However, our scheme requires addition private and public keys. Addition private and public keys are generated by data center. The previous works did not require addition parameter, but their scheme had no partially encrypted data for the users. Each scheme is compared in terms of ciphertext size, private and public key size. Ciphertext size represents the communication

Table III. Comparison of different schemes.

Scheme	Outsourcing decryption
Literature [10]	No
Literature [12]	No
Literature [19]	No
Proposed scheme	Yes

Table II. Efficiency comparison of different schemes.

Scheme	Ciphertext size	Private key size	Public key size
Literature [10]	$k(n_0 + n_a) + n_1 + n_0$	$Tn_0 + n_T + n_0$	$n_1 + In_0 + n_0$
Literature [12]	$k(n_0 + n_a) + n_1$	$Tn_0 + n_T + \log N_{n_k}$	$n_1 + In_0$
Literature [19]	$k(n_0 + n_a) + n_1$	$Tn_0 + n_T$	$n_1 + In_0$
Proposed scheme	$k(n_0 + n_a) + n_1$	$Tn_0 + n_T + n_0 + n_{cgk}$	$n_1 + In_0 + n_0 + n_{cmk}$

cost that a sensor needs to send its data to the data center. In the proposed scheme, the controller and data center separately send private and public keys to cluster heads, so the size of private and public key is larger than those of previous schemes.

In order to compare the lifetime of network, we randomly deploy 100 sensor nodes in an interested area. Each node has the same initial energy, and the probability of elected cluster head is 0.1. Compared with literature [10], we can see that the proposed scheme reduces the energy consumption of WSNs and prolongs the lifetime of whole network from Figure 6. Notice that the time that represents the number of rounds in network in *x*-axis is relative. In Figure 7, the proposed scheme consumes energy evenly. Sensor nodes in the proposed scheme take turns to send data, so the first died node appears later, which indicates that our scheme is more suitable for WSNs.

Notes: In literature [12], it was assumed that the deployed sensor network is composed of high-end sensor devices and a powerful data aggregator. The data aggregator moves in the field of interest where sensor nodes are deployed and collects data from sensor nodes. In this scheme, sensor nodes consumed little energy. Figure 8 is given according to the assumptions in [12]. In order to validate the performance of the proposed algorithm, our scheme is compared with other schemes under the same conditions. As shown from Figure 7, we can

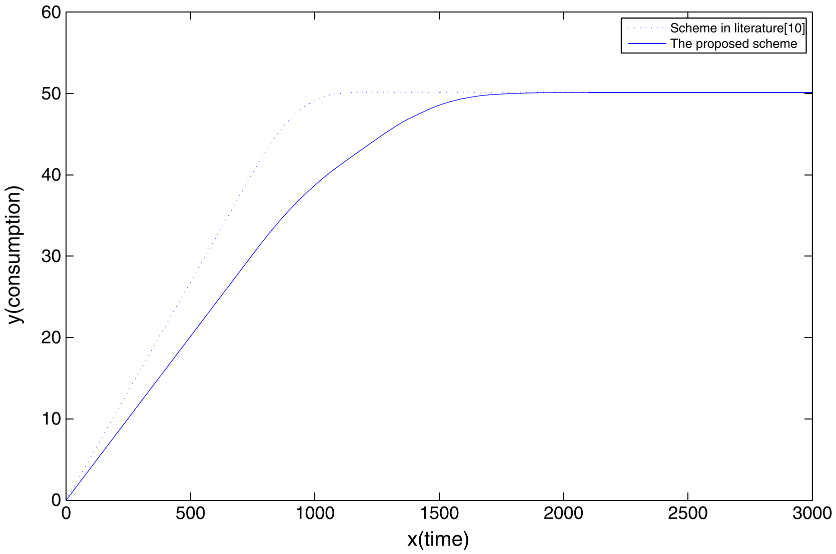

Figure 6. Energy consumption in different schemes.

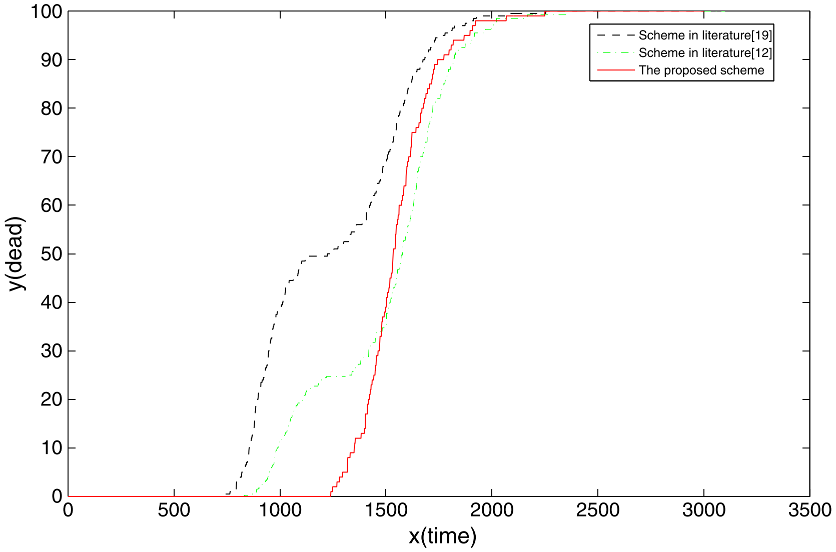

Figure 7. Number of death sensor node changes in different schemes.

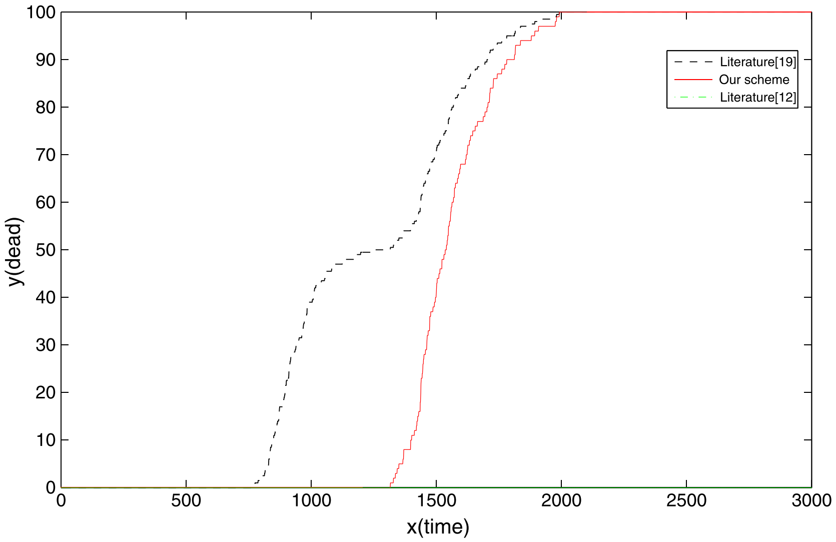

Figure 8. Number of death sensor node changes in different schemes at the same condition.

see that the energy balance of sensor nodes in the proposed scheme is better than that of the scheme in [12].

In WSN, communication energy consumption is the largest energy consumption. In this work, we assume that the size of an attribute string is 160 bits. When the 80-bit security level is adopted, the size of q is 512 bits. The size of an element in G_0 is 1024 bits. By standard compression technique [24], the size of an element in group G_0 can be reduced to 65 bytes. The size of an element in G_1 is 1024 bits. We assume that the number of attributes associated with a ciphertext is 4. That is to say, $k = 4$. One hundred sensor nodes are deployed in an interested area.

For the communication cost (we consider the ciphertext size and ignore the size of the private key and public key. Here, we only consider the WSN part because the resource of sensor node is constrained.), the size of the transmitted data should be calculated. So, in literature [10], sensor nodes need to transmit

$$[4(n_0 + n_a) + n_1 + n_0]\times 100 = 390,400\text{bits}$$

messages. In literature [12], sensor nodes need to transmit

$$[4(n_0 + n_a) + n_1]\times 100 = 374,400\text{bits}$$

messages. In our scheme, the ratio of cluster head node is 0.1. Sensor nodes need to transmit

$$[4(n_0 + n_a) + 3n_1]\times 10 + 2{*}90(n_a + n_1) = 271,040\text{bits}$$

messages. From [25], we know that sensor node consumes energy 0.052 and 0.019 mJ for transmitting

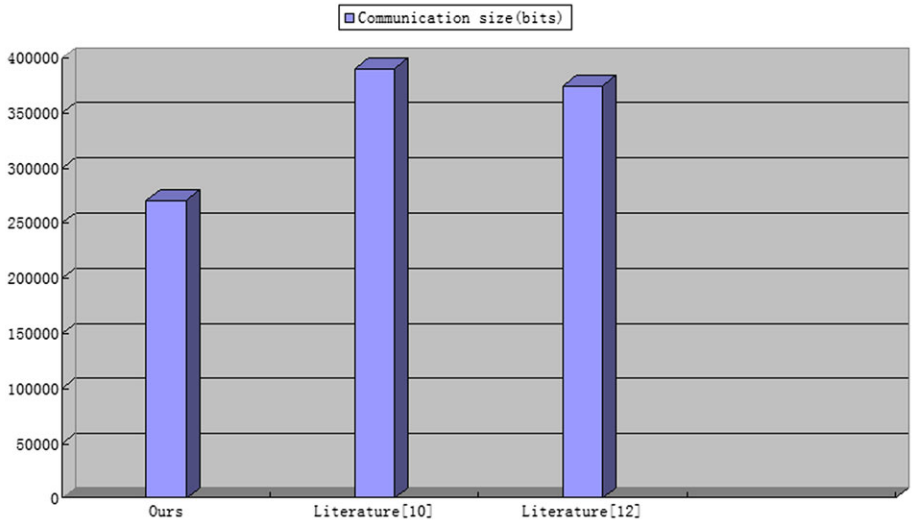

Figure 9. Communication size of the three schemes.

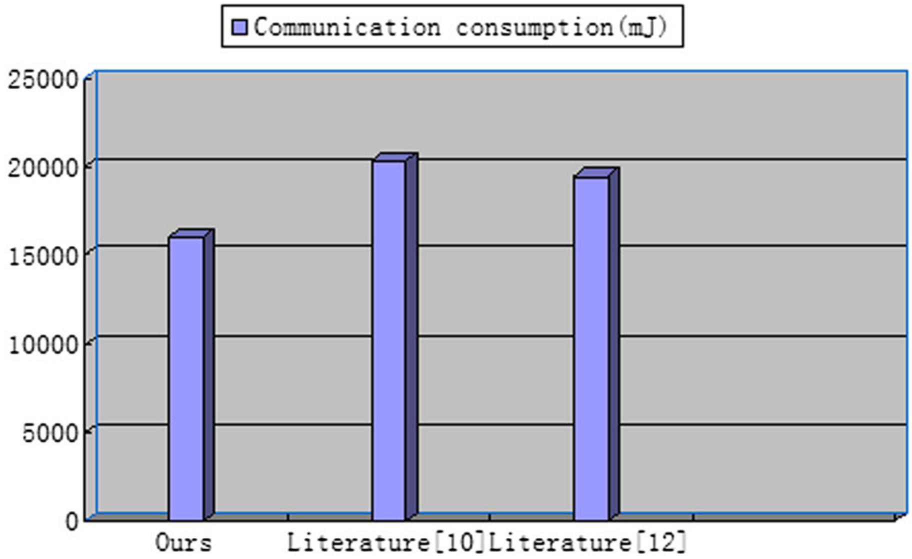

Figure 10. Communication consumption of the three schemes.

and receiving one byte message, respectively. In [10], the communication energy consumption of WSN is 390,400 × 0.052 = 20,300.8 mJ. In [12], the communication energy consumption of WSN is 374,400 × 0.052 = 19,468.8 mJ. In our scheme, the communication energy consumption is composed of two parts, which are the energy consumption of the transmission data and the energy consumption of receiving data. The communication energy consumption of WSN is 271,040 × 0.052 + 106,560 × 0.019 = 16,118.72 mJ.

The communication size and communication consumption on the WSN are summarized in Figures 9 and 10, respectively. From Figures 9 and 10, we can see that our scheme is more suitable for resource-constrained sensor nodes.

6. CONCLUSIONS

Based on clustering techniques and KP-ABE policy, partially data-outsourced decryption scheme is proposed for WSNs in this work. The proposed methods assumed that data center and authority are semi-trusted. The attributes of sensor nodes vary with the location of sensor nodes. Data center has enough energy and computing resources. In the proposed scheme, sensor nodes use symmetric decryption policy, and then, they send it to cluster heads that can encrypt the data based on KP-ABE and send the ciphertext to data center. The receiver that satisfies the attributes of the ciphertext can decrypt the data. The efficiency of computation for users is improved because the data center can partially decrypt the data using token key. Authorized users can efficiently decrypt the transformed ciphertext. Partially data-outsourced decryption not only ensures the security of the data but also prolongs the lifetime of the network. The security of the data is preserved against the data center and unauthorized users. Only authorized users can obtain the partially decrypted data from data center and decrypt the partially decrypted data. System controller and data center cannot obtain the original data in the proposed scheme. The decrypted cost for users is reduced because most of the decryption operations are delegated to the data center. The decryption efficiency of the users is improved in the proposed scheme. The simulation results demonstrated that the proposed scheme is efficient and secure.

An interesting future work is that we will pay more attention to attribute key revocation and multi-authorities for WSN.

REFERENCES

1. Li KY, Ma H. Outsourcing decryption of multi-authority ABE ciphertexts. *International Journal of Network Security* 2015; **16**(4):286–294.
2. Lounis A, Hadjidj A, Bouabdallah A, Challal Y. Secure medical architecture on the cloud using wireless sensor networks for emergency management. 8th International Conference on Broadband, Wireless Computing, Communication and Applications, 2013, 248–252.
3. Santan C, Sandip R. Cryptanalysis and enhancement of a distributed fine-grained access control in wireless sensor networks. Proceedings of the International Conference on Advances in Computing, Communications and Informatics, 2014, 2074–2083.
4. Su JS, Cao D, Wang XF, Sun YP, Hu QL. Attribute-based encryption schemes. *Journal of Software* 2011; **22**(6):1299–1315.
5. Zhang YQ, Wang XF, Liu XF, Liu L. Survey on cloud computing security. *Journal of Software* 2016; **27**(6):1328–1348.

6. Yan JZ, Ma JF, Li FH, Moon SJ. Key pre-distribution scheme with node revocation for wireless sensor networks. *Ad Hoc & Sensor Wireless Networks* 2010; **10**(2/3):235–251.
7. Tian MM, Yang W, Huang LS. Security of a biometric identity-based encryption scheme. *International Journal of Network Security* 2012; **14**(6):362–365.
8. Subramanian N, Yang CJ, Zhang WS. Securing distributed data storage and retrieval in sensor networks. *Pervasive and Mobile Computing* 2007; **3**:659–676.
9. Sahai A, Waters B. *Fuzzy Identity-Based Encryption. Advances in Cryptology Eurocrypt*. Springer: Berlin Heidelberg, 2005; 457–473.
10. Yu SC, Ren K, Lou WJ. Fdac: Toward fine-grained distributed data access control in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems* 2011; **22**(4):673–684.
11. Sushmita R, Amiya N, Ivan S. *Distributed fine-grained access control in wireless sensor networks. IEEE International Parallel & Distributed Processing Symposium*. IPDPS: Alaska, 2011; 352–356.
12. Junbeom H. Fine-grained data access control for distributed sensor networks. *Wireless Networks* 2011; **17**(5):1235–1249.
13. Hohenberger S, Waters B. Online/offline attribute-based encryption. *Public-Key Cryptography-PKC* 2014; **2014**:293–310.
14. Junbeon H. Attribute-based secure data sharing with hidden policies in smart grid. *IEEE Transactions:on parallel distributed systems* 2013; **24**(11):2171–2179.
15. Lai JZ, Deng RH. Guan Chaowen, Weng Jian. Attribute-based encryption with verifiable outsourced decryption. *IEEE Transactions on Information Forensics and Security* 2013; **8**(8):1343–1354.
16. Wang HQ, Wu QH, Qin B, Josep DF. FRR: fair remote retrieval of outsourced private medical records in electronic health networks. *Journal of Biomedical Informatics* 2014; **50**:226–233.
17. Qiu WD, Zhou YW, Zhu B, Zheng YF, Gong Z. Key-insulated encryption based group key management for wireless sensor network. *Journal of Central South University* 2013; **20**(5):1277–1284.
18. Shamir A. How to share a secret. *Communications of the ACM* 1979; **22**(11):612–613.
19. Goyal V, Pandey O, Sahai A, Waters B. *Attribute-based encryption for fine-grained access control of encrypted data. Proceeding of the 13th ACM Conference on Computer and Communications Security*. ACM Press: New York, 2006; 89–98.
20. Heinzelman WB, Chandrakasen AP, Balakrishnan H. An application specific protocol architecture for wireless micro-sensor networks. *IEEE Transactions on Wireless Communications* 2002; **l**(4):660–670.
21. Wang CJ, Liu J. Attribute-based ring signcryption scheme and its application in wireless body area networks. *Algorithms and Architectures for Parallel Processing* 2015:521–530.
22. Amos B. Secret-sharing schemes: a survey. *Coding and Cryptology* 2011:11–46.
23. Li FG, Han YN, Jin CH. Certificateless online/offline signcryption for the Internet of Things. *Wireless Networks* 2015:1–14.
24. Shim KA. CPAS: an efficient conditional privacy preserving authentication scheme for vehicular sensor networks. *IEEE Transactions on Vehicular Technology* 2012; **61**(4):1874–1883.
25. Li FG, Han YN, Jin CH. Practical signcryption for secure communication of wireless sensor networks. *Wireless Personal Communications* 2016:1–22.