IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 8, 2019, accepted August 13, 2019, date of publication August 16, 2019, date of current version August 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935797

A Secure Trust-Based Key Distribution With Self-Healing for Internet of Things

SONG HAN[1], MIANXUE GU[1], BAILIN YANG[1], JIANHONG LIN[2], HAIBO HONG[1], AND MENGJIAO KONG[1]

[1]School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China
[2]Zhejiang Ponshine Information Technology Company Ltd., Hangzhou 310012, China

Corresponding author: Bailin Yang (ybl@zjgsu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1401304, in part by the Zhejiang Province Qianjiang Distinguished Scholar Support Program, in part by the Zhejiang Xinmiao Project under Grant 1300KZN0219084G, in part by the Key R&D Program Project of Zhejiang Province under Grant 2019c01004, and in part by the Key Research and Development Program Project of Hangzhou under Grant 20182011A46.

ABSTRACT Internet of things (IoT) can enable cyber-physical objects to communicate with one another and realize real-time living-needs control such as for vehicles, smart phones, refrigerators, healthcare gadgets and air-conditioners. Of the applications of IoT, collecting and receiving health-related data securely is the most crucial and significant use in the fields of community and hospital healthcare, where any single communication failure or data loss might cause a life damage risk. To address this problem, self-healing mechanism can be used for facilitating secure communications and recovering from the lost data. In this paper, a new key distribution with self-healing for IoT objects of community healthcare is proposed, and the underlying system is composed of two layers. In the top layer, the new scheme implements both deterministic security link and access control which are based on polynomial-based methods. In the bottom layer, we propose a trust-based key distribution scheme with self-healing and a singular value decomposition (SVD) based authentication method. Security and performance analysis show that our protocol can be performed more efficiently in data communication. In addition, results obtained from both security analysis and simulations indicate that our scheme is more suitable for IoT networks.

INDEX TERMS Internet of Things, healthcare, key distribution, self-healing, trust.

I. INTRODUCTION

The Internet of Things (IoT) is known as the next generation Internet with the development of modern wireless telecommunications [1]. IoT takes advantage of pervasive computation which uses a large number of ''smart'' things or objects around us, such as Radio-Frequency IDentification (RFID) tags, wireless sensors, smart terminals, and so on. With wireless communication technologies, these objects become capable of interacting and cooperating to reach common goals [2].

Wireless sensor network (WSN) is an indispensable part in the IoT. Like RFID technology, WSN collects data from external physical world, and then transmits the data to the upper applications. There exist two main models for this kind of transmission [3]. In the first model, a proxy system (a base station [4] or a sink [5]), is needed. In the second model, IP protocol is used for WSNs connecting to the Internet [6].

The associate editor coordinating the review of this article and approving it for publication was Adnan M. Abu-Mahfouz.

The development of IoT promotes the progress of community healthcare in smart cities integrated with IoT communications [46]. Healthcare in the community is one of the most important services to prevent and detect diseases for users in the community [7]. Another very important function of community healthcare is to do basic treatment before a patient being transferred to a hospital. This function is the basic step to implement universal access to primary healthcare. In rural areas, community healthcare plays a more important role. It is people-centric for emphasizing the family unit and providing continuous and humane medical services. The tasks of the community healthcare are prevention and care-oriented for community individuals and families to implement healthcare and management. Compared with large hospital healthcare, community healthcare not only saves time, but also increases the chances for real-time communication between doctors and a patient. In addition, because of the relatively low cost, the patients can save a large number of fees on outpatient, therapy and drugs which partly solves the high

expenses problem. Community healthcare systems can not only improve medical service levels but also reduce medical service cost.

IoT devices (IoTDs) are responsible for automatic data collection of the people in the community. It is more convenient for the doctor to obtain health-related data of people. The doctor or nurse can use his smartphone or other smart devices to send data-request tasks to the sub-network of the community healthcare system and get the corresponding data. However, most of IoTDs are resource-constrained. It indicates that traditional schemes in other kinds of networks using asymmetric keys are infeasible and impractical, which leads to data confidentiality a nontrivial task for IoTDs. In addition, the nature of wireless communication of most devices and their applications in sub-networks of the community healthcare system, make security and privacy be two major challenges. Specifically, there are diverse types of users in the community, and the character of the human involvement further augments the sensitivity [9]. Therefore, how to safely and efficiently control the access to the sub-networks of the community healthcare system is also a nontrivial task. Secure key management would prevent an unauthorized entity from both compromising valid entities' communications and accessing their exchanged contents [8]. Of the applications of IoT, collecting and receiving health-related data securely is the most crucial and necessary use in the fields of community healthcare. However, any single communication failure or data loss might cause a life or health threatening risk.

Consider a scenario where a sensor network supported IoT can provide healthcare services to people in a community. People (i.e. users) will be equipped with IoTDs to collect health-related data automatically. To provide health services for a person in this community, the doctor or the nurse in the certified hospital can access the network of community healthcare system and obtain the corresponding data remotely. It is evident that securely access control and reliable data communication are equally needed. Data communication loss might be vital in this healthcare scenario. In addition, preventing strong adversary from compromising users sensitive data is important for the sensitivity of people with potential illness in this community. Obviously, designing a new key distribution mechanism with self-healing and trust for IoT objects is significant and also an open problem.

To address this problem, in this paper, we propose a framework of key management with self-healing and trust for IoT objects of community healthcare. In this framework, self-healing mechanism is used for facilitating secure communications and recovering from the lost data while trust is used for both access control and efficient broadcast. Specifically, our main contributions include but not limited to the followings:

1) We propose a hierarchical key management scheme for three types of devices for a community healthcare system.

2) We define a system model which contains two layers. We propose a polynomial-based key agreement method for the top layer, and with a proper adaption, we make it capable of controlling the access of MSs to the sub-network of the community healthcare system. In addition, our scheme is secure against mobile sink replication attack which can protect the distribution of data-request tasks and collections of required data.

3) In the bottom layer of our system model, we propose a polynomial-based group key distribution scheme with self-healing and revocation capabilities. In addition, we propose a singular value decomposition (SVD) based authentication method in this layer.

4) We introduce trust management into key management for facilitating both access control and efficient broadcast.

5) We conduct security and mathematical performance analysis on the proposed key management scheme.

The rest of the paper is organized as follows: In Section II, related work is presented. In Section III, we describe our system model. In Section IV, we present the proposed scheme. We analyze the scheme in terms of security and performance in Section V, and in Section VI, we make the conclusion. Table 1 shows the notations used throughout of this paper.

TABLE 1. Notations.

MS	The mobile sink
MS_p	The mobile sink of people
MS_n	The mobile sink of nurse
MS_d	The mobile sink of doctor
AD	The access device
$IoTD$	The *IoT* device
$IoTD_p$	The *IoT* device working for people
$IoTD_n$	The *IoT* device working for nurse
$IoTD_d$	The *IoT* device working for doctor
KDC	The key distribution center
x_{ij}	The secret value of $IoTD_i$ in the j-th session
$P_{dr}(x)$	The double-revocation polynomial
$P_v(x)$	The validation polynomial
$P_k(x)$	The key material polynomial
K_j	The group key in the j-th session
R_1	The set of revoked IoTDs in the first case
R_2	The set of revoked IoTDs in the second case
B_j	The current broadcast in the j-th session
B	The broadcast in each session
T	The degree of polynomials used in the top layer
$f()$	The one-way hash function

II. RELATED WORK

In this section, we review two types of related work: trust model, and self-healing key distribution scheme. We highlight the stand points of our proposed scheme with respect to existing schemes in TABLE 2.

A. TRUST MODEL

Trust model is mainly used to protect resource constrained networks from network insider attacks while other

TABLE 2. Comparison of closely related schemes.

Schemes	Resisting mobile replication attack	Trust model	Self-healing	Key management scheme
Waluyo et al. scheme[13]	No	Somewhat Conventional	No	Conventional
Rasheed et al. scheme [34]	Yes	No	No	No
Shen et al. scheme [42]	No	Conventional	Yes	Conventional
Bilal et al. scheme [44]	No	Conventional	No	Conventional
Ours	Yes	New	Yes	New

security approaches, such as authentication, confidentiality, can defend the networks against security threats from outsiders, such as eavesdropping attacks, impersonation attacks, etc. [15], [48]. Trust model could facilitate many applications including malicious attack detection, secure routing, authentication, and so on [10], [14], [44], [49]. When utilized in pervasive computing, trust can be used to alleviate privacy risks [11]. Trust computation and trust dynamics are two cores in trust management. Trust dynamics, which is mainly composed of trust propagation, aggregation and prediction, can help to compute trust values [12], [45].

Trust management is often used in routing. Trust dependent greedy anti-void routing was proposed by Sivasankari *et al.* to save energy and maximize the reliability of data delivery [14]. They take use of Bayesian estimation models to calculate the trust values from the history, i.e. to calculate the trust values from the source to the sink and conducted the data transmission based on the trust value path. However, in their scheme, each node needs to store the position information for its neighbors, which would produce more storage costs. In [47], Yang *et al.* built a model of the trust relaying quantum key distribution network and proposed an optimal secret-key-aware routing method for trust relaying. Their simulation results show the improvements of holistic performance of the trust relaying quantum key distribution network and achieve load balance. However, they did not consider self-healing to improve the completeness and security in their work.

Zahariadis *et al.* proposed a novel trust-aware geographical routing scheme against routing attacks [15]. The total trust value of a node *i* towards another node *j* consists of direct trust and indirect trust. The simulations show their scheme has flexible configuration, trade-offs and fine tuning of the algorithm. They also prove that, when some malicious nodes take up 50% in the underlying WSN, the scheme could still reveal these malicious nodes successfully. However, geographical routing principle relies on geographic position information, which might be not available in some environments.

Cluster-based WSNs is highly scalable and energy-efficient and have been used in many scenarios. A hierarchical trust management protocol was proposed by Bao *et al.* for this type of WSN [16]. Different than most existing trust schemes, their scheme takes both the quality of service and the social trust into consideration. It can also adapt to new conditions by the knowledge dynamically learned from past experiences. Its simulation results show the performance of the routing protocol is much closer to that of the routing protocols which are based on flooding in terms of not only delivery ratio but also message delay. The scheme also has low message overheads compared with those without using trust methods. However, it is impractical to implement such a complex trust evaluation scheme at each cluster manager of the cluster.

Trust models are also designed for vehicular ad hoc networks (VANETs). Mármol *et al.* proposed a trust and reputation infrastructure-based proposal (TRIP) for VANETs [17]. They analyzed and identified several indispensable requirements needed to be considered for designing trust models for VANETs.

Wireless medical sensor networks are another kind of WSNs which have their unique operational and security requirements. He *et al.* pointed out the security and performance challenges in medical wireless sensor networks with monitors and proposed a trust management scheme called ReTrust, which is both lightweight for sensor nodes and computation-simple for master nodes [18]. They conducted experiments which showed the scheme could effectively identify malicious behaviors, and boost the performance of the network. Their simulation results also show that on–off and bad-mouthing attacks can be efficiently defended. However, they did not consider the difference between different types of task roles of medical sensors and did not adopt a more formal security analysis to prove the security of their scheme.

Al-Turjman *et al.* in [50] proposed a content-sensitive seamless identity provisioning (CSIP) framework for Industrial Internet of Things (IIoT). In their scheme, they presented a secure mutual authentication approach to achieve the major security goals, such as supporting session-key agreement and being resilient to privileged-insiders. Although their scheme meets the requirements of both security and being real-time, it is only effective for certain environments. Further, it did neither provide a formal logical analysis to prove the security nor adopt a key management with a trust model.

Waluyo *et al.* proposed a trustworthy-based scheme for a resource-constrained P2P environment [13]. The proposed scheme consists of three components: trust model, trust management scheme and data broadcast model. In their scheme, a tit-for-tat network policy is applied to ensure that payment will return. At the same time, each peer is capable of broadcasting in an efficient and reliable way. However, their

model is not directly suitable for multi-entity and hierarchical environments such as a community healthcare system.

B. SELF-HEALING METHODS

During a group key distribution, a member in the group might lose a message due to unreliable transmissions or any network attack, such as replication attack [34], collusion attack [21] and node-capture attack [32]. However, requesting for resending this message would introduce additional communication overheads. In [41], Kurnio et al. proposed a secure re-keying scheme with the key recovery property for multiple users revocation and multiple users joining, which is similar to the self-healing property. Following that, a number of key distribution schemes with self-healing were proposed. From the fundamentals they relied on, there exist four types of self-healing schemes, which are based on polynomial, bilinear pairings, vector space secret sharing, and exponential arithmetic [19]–[21], [23], [40], respectively.

Polynomial based schemes make use of polynomials operations over a finite field. Dutta *et al.* proposed a self-healing key distribution scheme based on secret polynomials and masking polynomials [21]. Their scheme achieves four security features: group confidentiality, t-revocation, t-wise forward secrecy and t-wise backward secrecy. However, Wang *et al.* pointed out the aforementioned scheme was unsecure against collusion attacks. To tackle this issue, Wang *et al.* proposed a hash chain-based group key distribution [30]. In their scheme, the user joins the group with the capability of recovering previous group session keys. Their scheme is also able to resist any collusion attack between revoked users and new joined users. In addition, their scheme has the capability of dealing with more revoked users. Janani and Manikandan in [33] presented a rekeying scheme for addressing the issue in managing a group key among dynamic group of nodes in mobile ad hoc networks. In their scheme, a revocation protocol was presented to gather an accurate rate of node misbehaviors. However, there was no formal logical analysis to prove the security of their work.

Compared with other types of self-healing schemes, exponential arithmetic based ones are considered to be the most efficient long lived methods. However, this type of self-healing schemes lacks backward secrecy. Rams *et al.* proposed two exponential arithmetic based schemes with backward secrecy [28]. To update the personal key of each user in each session, the newly added users will not be able to get the past session keys. Their work achieves both forward and backward secrecy, and resists collusions between new joined users and revoked users. However, they did not consider a more secure formal analysis for proving the security of the scheme.

To replace the threshold used in Sharmir's secret sharing and to achieve better performance and higher security, vector space secret sharing (VSS) based self-healing key distribution scheme was proposed [25]. Dutta et al proposed a generalized vector space access structure to achieve revocation capability [26]. The proposed scheme can be used for a large number of dynamic users to negotiate a session key. Also, their work shows computationally secure and achieves forward and backward secrecy. An effective collusion resilient key distribution scheme was proposed for the secure group communication [27]. In addition, VSS based self-healing scheme can be used to prevent collusion attacks more effectively.

There also exist another two self-healing key management schemes: logical key tree based scheme and hash chain-based self-healing scheme. To decrease the number of needed rekeying messages, logical key tree was applied into key distribution schemes. Jiang et al. proposed a group key management scheme which was based on logic route key [29]. The proposed scheme divides a WSN into clusters and generates a route key tree based on the route topology of the WSN. However this type of schemes suffers from heavy overheads for rekeying whenever group nodes change.

Although our new scheme deals with the similar issues as the works compared in TABLE 2, such as resisting mobile replication attack, trust model, self-healing and key management, our research focuses are different: i) we propose a novel hierarchical key management scheme for three types of Internet of Things devices compared with conventional ones [13], [42], [44]. ii) our proposed scheme is secure against MS replication attacks and gives a more formal logic security analysis compared with the work in [34]. iii) We propose a polynomial-based group key distribution scheme with both self-healing and revocation capabilities in the bottom layer, which were not considered in the works of [13], [34] and [44]. iv) We introduce a more novel trust method into key management for facilitating both access control and efficient broadcast compared with [13], [42], [44].

III. SYSTEM MODEL

In this section, we present the system model, security model and trust management model, respectively.

MS can be a smart device of a person (all the people in the community can receive healthcare services, therefore, the subscript p not only refers to the patient in the community, but also refers to healthy people living in the community), a nurse or a doctor, which can be a smart device and is responsible for distributing data-request tasks to the sub-network of the community healthcare system and receiving the corresponding data. Therefore, we define three MSs in our system model, MS_p as an MS of people, MS_n as an MS of nurses, and MS_d as an MS of doctors.

IoTD is a kind of device with sensing functions in the sub-network of the community healthcare system. There are a number of types of IoTDs, which have different tasks for collecting different health-related data of people and also sensing ongoing situations in the community. According to the objects serviced by IoTDs, such as doctors, nurses and patients, their authorities and functions are different. In our system model, IoTDs can be divided into three levels and their logic relationships are presented by: $IoTD_p$ $<IoTD_n$ $<IoTD_d$. The reason for this hierarchical relationship is due to: The IoTDs working for doctors can

send data-request task to access the health-related data and give specific treatment recommendations of patients while for nurses only recording and uploading symptom information to the previous level.

A. SYSTEM MODEL

The entities of Internet of Things in our system model include: mobile sink (MS), offline key distribution center (KDC), access device (AD), IoT device (IoTD). The Fig 1 shows the structure of the system model which consists of two layers.

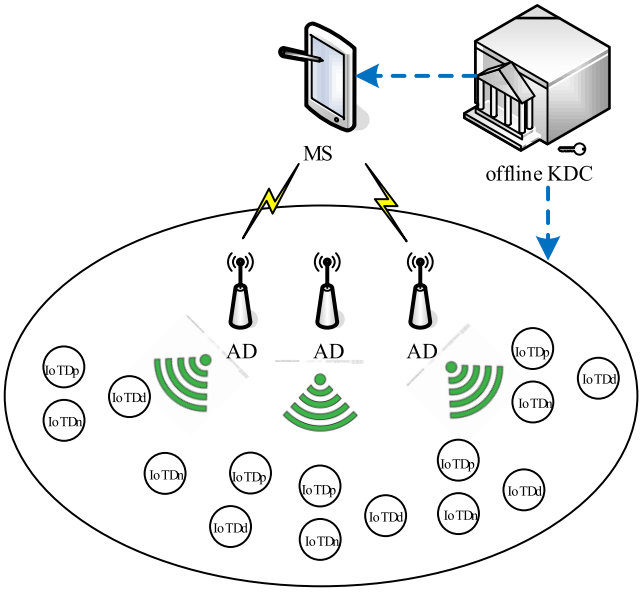

FIGURE 1. System model.

In the top layer, we assume there is a large polynomials pool, for which there is an offline and trusted key distribution center (KDC) in the community that maintains the key pool. KDC only works during the offline deployment phase. Each AD will obtain one polynomial before deployment. MS can get the polynomials of the corresponding ADs from the KDC for the task. AD is the access device for the MS to use the sub-network of the community healthcare system. We assume there exist three types of ADs for different MSs. Specifically, type I AD is the access device for MS_p (the mobile sink of people) to collect health-related data of people automatically in a community. Type II and type III AD are access devices for MS_n and MS_d (the mobile sink of nurse and the mobile sink of doctor) to send data requests, respectively. And these ADs control the IoTDs in the bottom layer of the system.

Here, we assume there are m ADs and n IoTDs, and $m < n$. From the architecture point of view, our system model has two layers: the top layer, i.e., between MS and AD, and the bottom layer, namely, between AD and IoTDs. In the top layer, MS needs to create secure links with AD to send data requests and receive the collected data.

Policies in the bottom layer include but not limited to the followings: the AD receives a task from the MS and then distributes it to the selected working IoTDs in its domain. In addition, the AD can receive a number of tasks at the same time. The AD selects the working IoTDs by their trust values. In addition, working as a group manager, the AD is also responsible for key distributing and rekeying in its corresponding group. In our scheme, we assume the MS would not communicate with the IoTDs which will simplify the system model and also decrease the cost of IoTDs. And the worker in the community will place some IoTDs or remove an old IoTD for the use-up of battery or other factors. If there is a new IoTD joining the sub-network of the community healthcare system, the AD can add it into the working group. In addition, if the number of revoked IoTDs surpasses the necessary number of completing a task but there is no new placed IoTD, the AD can add low trust-value IoTDs to take part in the task again.

B. SECURITY MODEL

In the top layer, we mainly consider mobile sink replication attacks. In a key pre-distribution scheme based on a polynomial pool, an adversary can obtain many polynomials after compromising a number of MSs and then launches a mobile sink replication attack [34]. If an attacker launches a mobile sink replication attack successfully, it can collect useful data from the sub-network of the community healthcare system or receive data requested by other users.

In the bottom layer, the AD should have the cability of revoking and adding IoTDs in its domain. As the group broadcast might be missed, the IoTDs which do not receive the broadcast could recover the key by using other broadcast messages. Considering in the community the ADs are all set in a fixed place and the MS should move to the right place to send messages or receive data, so we do not discuss the AD replication attacks here which will be our future work. The IoTDs in the domain of the AD form a group, and AD is responsible for distributing the group key for them.

Our security model in this layer should meet the following requirements: 1) group key confidentiality; 2) forward secrecy; 3) backward secrecy; and 4) revocation capability.

C. TRUST MANAGEMENT

We allocate trust management in the bottom layer for access control and efficient broadcast. In the healthcare community, there are various IoTDs with different tasks. In addition, there also exist two types of communications between IoTDs, namely, pull-interaction and push-interaction, like that in [13]. The difference between our method and the scheme in [13] is that in our case, the IoTDs have an additional component for access control, where they can be used by certain users in the system. For example, some IoTDs can only be used by doctors, so patients and nurses cannot access these IoTDs. Obviously, this type of IoTDs will work less which leads to less energy overhead. With the same condition, the trust value of IoTDs only working for MS_d will be higher than that of MS_n and MS_p. IoTDs working for MSd has less possibility of revealing data privacy than IoTDs

working for MSn while IoTDs working for MSn has less possibility of revealing data privacy than IoTDp. Therefore, we should enhance the trust method to be suitable for our case. We define our trust model by the following construction: where S, C, Q, R represent Speed, Correctness, Quality (a measure quality of resource), and Risk-free (a measure for malicious resources), respectively. We also define O as Owners hierarchical relationship. For different roles in a community, such as patients, nurses and doctors, their owners hierarchical relationships are declining, i.e. $O_p > O_n > O_d$ because $IoTD_p$ collects more data than $IoTD_n$ while $IoTD_n$ collets more data than $IoTD_d$. This trust model gives the service reputation Ts of one peer i *in time* t. The new personalized reputation can be computed as follows:

$$[S, C, Q, R, O]_{pi}^{t} \rightarrow \left[\left(\sum (SCQR)/4\right) \cdot O\right]_{pi}^{t} \rightarrow T_{S_i} \quad (1)$$

Note that, after once interaction between two devices, they will update the corresponding value S, C, Q, R mutually.

As bottom layer IoTDs will need to frequently transmit data for top layer IoTDs and thus consume energy faster. Therefore, we raise the trust value of bottom layer IoTDs by multiplying a larger reputation coefficient, which will be an incentive for bottom layer IoTDs to follow the system operation rules.

IV. THE PROPOSED SCHEME

In this section, we come up with our trust-based self-healing group key distribution scheme, an authentication for ADs and IoTDs, and the access control strategy.

A. TRUST-BASED SELF-HEALING KEY DISTRIBUTION SCHEME

The trust-based self-healing group key distribution scheme consists of two parts: key agreement in the top layer and group key distribution in the bottom layer. When an MS wants to send data-request messages, it should do key agreement with a certain number of ADs. And then each AD will distribute the group key for the IoTDs to finish the task.

1) KEY AGREEMENT IN THE TOP LAYER

In the top layer, we assume there is a large polynomials pool, and each AD will obtain one polynomial before deployment. Besides, there is a key distribution center (KDC) in the community which maintains the key pool. MS can get the polynomials of the corresponding ADs from the KDC for the task.

Before deployment in a community, each AD will be assigned with n key polynomials to meet the requirements of n corresponding users in this community, which could create $n!$ different combinations for $n!$ different tasks. Here we define that generating each session key needs n different polynomials, because if we do not limit it, for example, MS_a with P1, P2, P3 can pretend that it is MS_b with only P1. We also define that any two ADs have no same polynomial which means with m ADs, there needs a polynomials pool with nm polynomials.

When the MS wants to collect some data from the subnetwork of the community healthcare system, it will do key agreement with the ADs with the following steps:

Step 1: An MS registers a task in the KDC:

An MS registers a task at the KDC and gets n polynomials (the same polynomials for the ADs whom the MS wants to distribute a task to) and a random number RNc standing for one combination of the polynomials.

Step 2: MS -> AD: ID, IDs of the polynomials, RNc, $RNms$ and $K(RNms)$:

The MS computes the session key: $K = H(P_{i1}|P_{i2}|\ldots|P_{in}|RNc)$, generates random number $RNms$ and encrypts $RNms$ with K. It sends its ID, the IDs of the polynomials,RNc, $RNms$, and $K(RNms)$ to the AD whom it wants to communicate with.

Step 3: AD authenticates MS; AD -> MS: $RNad$ and $K(RNad)$.

The AD checks the IDs of the polynomials and RNc, AD computes the session key and sends some information back to the SM; otherwise, the AD ignores the request. The AD computes the session key: $K = H(P_{i1}|P_{i2}|\ldots|P_{in}|RNc)$. And then, it sends $RNad$, also a random number, and $K(RNad)$ to MS. Note that in $K = H(P_{i1}|P_{i2}|\ldots|P_{in}|RNc)$, the order of i_n is determined by RNc.

Step 4: MS authenticates AD: MS gets the message and uses K to decrypt $K(RNad)$. If the decrypted content is equal to $RNad$, then AD gets authentication of MS.

After the key agreement in the top layer gets done, MS can send data-request message to AD secretly.

2) GROUP KEY DISTRIBUTION IN THE BOTTOM LAYER

Now AD distributes the task to the IoTDs in its domain. AD is responsible for selection, revocation and addition of IoTDs. The AD decides which IoTD can be added to take part in the task by its trust value.

Step 1: At the beginning of the task, AD sets a trust threshold value.

Step 2: In each session, AD re-computes the trust value of the IoTDs in its domain according to the new values S, C, Q, R of each IoTD. Note that for a certain IoTD, O is a constant which is set by its role of the IoTD.

$$[S, C, Q, R, O]_{pi}^{t'} \rightarrow \left[\left(\sum (SCQR)/4\right) \cdot O\right]_{pi}^{t'} \rightarrow T_{S_i} \quad (2)$$

Step 3: AD judges whether the number of trusted IoTDs can finish the task, if not, then it needs to drop the trust threshold value.

Step 4: According to the number of needed working IoTDs and the trust value of all the IoTDs in its domain, AD sets a new trust threshold value. The number of IoTDs whose trust values are higher or equal to the new trust threshold value should be equal to the number of needed working IoTDs.

That is to say, at the beginning of each task, AD will set a trust threshold value. The IoTD which has a higher trust

value will be added into the task. After some interactions, the trust values of AD to each IoTD in its domain will change. And when the trust value of an IoTD in the task drops to the trust threshold value, AD will revoke it from the task in this session. If after revoking, the remaining IoTDs cannot finfish the task, then AD needs to drop the trust threshold value. Some IoTDs might run out batteries. In addition, as the sub-network is unreliable, each IoTDs might lose broadcasts. To tackle these issues, self-healing is needed.

Regarding the group key distribution in the bottom layer, we assume there are three stages: 1) key material generation and key computation, 2) self-healing, and 3) adding and revoking IoTDs.

3) GROUP KEY GENERATION AND COMPUTATION

During the initiation, each AD generates n secret value x_i, $i = 1, 2, \ldots, n$. Each secret value will be assigned to each corresponding IoTD. Note that, here, we use secret value to compute the group key, rather than the identifier of each IoTD as other schemes, such as [21]. The secret value will be updated in each session in our scheme while the identifier is a constant. In addition, in order to decrease the degree of key material polynomial, IoTDs will be separated into many groups. For each group, AD will generate a lower degree key material polynomial. And the identifiers are sent before the corresponding broadcast. Only when an IoTD receives its identifier will it begin to receive the corresponding broadcast. Therefore, in our scheme, the secret value of each IoTD is used to compute the key material polynomial and the identifier of each IoTD determines to receive which broadcast.

In each session, each AD will generate the broadcast in the following four steps:

Step 1: AD randomly generates the group key K_j and updates the secret values of each IoTDs:

$$x_{ij} = f\left(x_{i(j-1)}\right) \tag{3}$$

where $f(.)$ is a one-way hash function. Note that for simple statement, we will add the subscript j only when updating it, and in other case, we will ignore the subscript j. In addition, the hash function $f(.)$ that each IoTD uses is distinct.

Step 2: AD generates two polynomials: double-revocation polynomial $P_{dr}(x)$ and validation polynomial $P_v(x)$:

$$P_{dr}(x) = \prod_{x_i \in R_1} (x - x_i) \cdot \prod_{x_j \in R_2} \left(x - x_j\right) \tag{4}$$

$$P_v(x) = \prod_{x_i \in V} (x - x_i) \tag{5}$$

Here, R_1 is the set of secret values of the IoTDs which cannot be added into the system again (for the running up of energy or being captured), R_2 is the set of secret values of the IoTDs which can be added into the system again (for the low trust value), and V is the set of secret values of the valid IoTDs

Step 3: AD uses the group key, double-revocation polynomial and validation polynomial to construct the key material polynomial:

$$P_k(x) = P_{dr}(x) K_j + P_v(x) \tag{6}$$

Step 4: AD generates the current broadcast $B_j = \{P_k(x), R_1, R_2\}$, and generates the broadcast by redundancy:

$$B = \begin{cases} \{B_1, B_2, \ldots, B_T\} & j = 1, 2, \ldots, T \\ \{B_{j-T+1}, B_{j-T+1}, \ldots, B_j\} & j > T \end{cases} \tag{7}$$

and then broadcasts B. Note that, here, we use the notation B_j as the current broadcast which is used to compute the current group key, and the notation B as the broadcast with self-healing capability.

An IoTD computes the current session group key in the j-th session as follows:

Step 1: it extracts B_j from the broadcast B in this session.

Step 2: it calculates the group key as equation (8):

$$K_j^{'} = \frac{P_k(x_i)}{P_{dr}(x_i)} \tag{8}$$

Note that, $P_v(x_i)$ is equal to 0 if x_i is the secret value of a valid IoTD and $P_k(x_i)$ is eaual to $P_{dr}(x_i)K_j + P_v(x_i)$. Then we can get the following equation.

$$\begin{aligned} K_j^{'} &= \frac{P_k(x_i)}{P_{dr}(x_i)} = \frac{P_{dr}(x_i) K_j + P_v(x_i)}{P_{dr}(x_i)} \\ &= \frac{P_{dr}(x_i) K_j + 0}{P_{dr}(x_i)} = \frac{P_{dr}(x_i) K_j}{P_{dr}(x_i)} = K_j \end{aligned} \tag{9}$$

The degree of the key material is determined by the number of valid IoTDs, which is a large number in a large system. To decrease the communication of each IoTD, AD will broadcast many messages. And before broadcasting one message, AD will broadcast the identifiers of the target IoTDs for this message. This is one of the reasons why we use secret value rather than identifier to generate and evaluate polynomials.

We assume the number of valid IoTDs in each session is n_v. We re-group this set V into d sets and the cardinality of each set is n_v/d. AD will generate $dp_v(x)$ and each can be described as the following:

$$P_{v_j}(x) = \prod_{x_i \in V_j} (x - x_i), \quad j = 1, 2, \ldots, d \tag{10}$$

And then, AD will generate corresponding d current broadcasts.

4) SELF-HEALING

An IoTD which loses key material $P(x)$ in a certain session can obtain this key material from the later broadcast to recover the group key in the certain session because the later broadcast contains a certain number of past key materials. We assume that an IoTD loses the broadcast in the j_1-th session and it successfully receives the broadcast in the j_2-th session, where $j_1 < j_2$. If j_2-j_1 is larger than T session, then it cannot recover the group key. If j_2-j_1 is smaller than T session, then it recovers the lost group key as the following steps:

Step 1: it extracts B_{j1} from B in the j_2-th session.

Step 2: it calculates the group key used in the j_1-th session as follows:

$$K_{j1}^{'} = \frac{P_k(x_i)}{P_{dr}(x_i)} = \frac{P_{dr}(x_i) K_{j_1} + P_v(x_i)}{P_{dr}(x_i)} = \frac{P_{dr}(x_i) K_{j_1} + 0}{P_{dr}(x_i)} = K_{j_1} \quad (11)$$

Under this circumstance, an IoTD can still recover the group key of K_{j1} and get the missing information from the lost data.

5) ADDING OR REVOKING GROUP USERS

When an IoTD is placed into the group of AD in the session j, AD will do the following two steps to add this IoTD into the group.

Step 1: AD generates a new identifier i_{new} and corresponding new secret value x_{new} and sends them to the new IoTD.

Step 2: AD sets initial trust values for the new added IoTD.

Remark: There exist two cases of revoking IoTDs. The first one is when some IoTDs run up their energy or are compromised. The identifiers of these kinds of revoked IoTDs will be added into the R_1. The second one is when the trust values of some IoTDs become lower than the threshold value. In this case, the identifiers of these kinds of revoked IoTDs will be added into R_2. Note that these kinds of IoTDs can be added into the group again when their trust values become higher than the threshold value or when there are not enough IoTDs to finish the task.

B. AUTHENTICATION BETWEEN AD AND IOTD

In this subsection, we propose a singular value decomposition (SVD) based authentication method for sub-network of the community healthcare system in which there are m ADs and n IoTDs. Using SVD, each IoTD only needs store one vector and with this vector it can get the authentication of m ADs. In addition, vector operations are very efficient because of linear computation.

1) SINGULAR VALUE DECOMPOSITION

Matrix decomposition is a very useful tool in many areas. For example, LU decomposition can be used in key management [35], [36]. Regarding SVD decomposition, its definition is as follows:

An $m \times n$ matrix M can be decomposed into three matrices as follows:

$$M = XYZ \quad (12)$$

where X is an m-order unitary matrix, Y is an $m \times n$ diagonal matrix whose elements are all nonnegative real numbers, and Z is an n- order matrix unitary matrix.

2) THE AUTHENTICATION METHOD

We set $A = X$ and $S = YZ$, therefore the equation (12) can be rewritten as the equation (13):

$$M = XYZ = AS \quad (13)$$

Before deployment, an AD whose identifier is i will get a row from the matrix M, $M_r(i)$, and a row from the matrix A, $A_r(i)$. Each IoTD with the identifier j will get a column from the matrix S, $S_c(j)$.

When an IoTD wants to send its collected data to the AD, AD will check whether the IoTD is an valid IoTD or not.

IoTD sends its $S_c(j)$ to AD which is encrypted by the session key between them. AD decrypts the message, multiplies $A_r(i)$ and $S_c(j)$, and then checks whether the result $A_r(i) \bullet S_c(j)$ is in the corresponding position of $M_r(i)$. If yes, then the IoTD gets the authentication of AD.

When an IoTD is revoked, each AD only needs to delete the corresponding element in $M_r(i)$. If a new IoTD with a new $S_c(i)$ is added, each AD needs to generate the new corresponding element for $M_r(i)$:

$$M_r(i)[j_{new}] = A_r(i) \cdot S_c(j_{new}) \quad (14)$$

where $S_c(j_{new})$ is the S_c of the new added IoTD whose serial number is j_{new}.

3) DETAILS OF THE AUTHENTICATION

The group key is used for communication among group members. However, if an IoTD wants to communicate with an AD, it also needs to generate a session key which is different from those used by other IoTDs. This session key K_{sj} can be computed as follows:

$$K_{sj} = H\left(K_j || N_{IoTD}\right) \quad (15)$$

Here, N_{IoTD} is another random number which is known only by AD and an IoTD. That means each IoTD has a different random number. However, AD needs to store more n random numbers which will be certainly a large storage cost. To tackle this issue, the following method is introduced.

Note that the elements in the matrix M are generated randomly. Using the elements in $M_r(i)$ for each N_{IoTD} will not introduce new n random numbers. We combine the authentication with the establishment of the session key between AD and IoTD, which contains the following steps:

Step 1: AD with id i sends $A_r(i)$, *RNad* and K_{sj} (*RNad*) to IoTD with id j. *RNad* is a random number and the N_{IoTD} for computing K_{sj} is determined by the equation (16):

$$N_{IoTD} = M_r(i)[j] \quad (16)$$

Step 2: IoTD computes N_{IoTD} as follows:

$$N_{IoTD} = A_r(i) \cdot S_c(j) \quad (17)$$

IoTD uses N_{IoTD} to generate K_{sj} as the equation (15), and then decrypts K_{sj} (*RNad*) by K_{sj}. If the result is equal to *RNad*, IoTD sends N_{IoTD} and K_{sj} (N_{IoTD}) to AD.

Step 3: AD decrypts K_{sj} (N_{IoTD}) by K_{sj}. If the result is equal to N_{IoTD}, IoTD is a valid IoT device in the system.

After these three steps, AD and IoTD finish mutual authentication and generate a session key for thereafter communication.

We can see from Fig. 2 that the 1st AD, that is AD_1, will get $M_r(1)$ from matrix M, and $A_r(1)$ from matrix A. At the

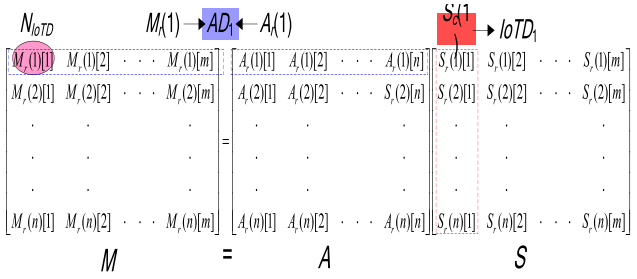

FIGURE 2. The detail of the used singular value decomposition.

same time, the 1*st* IoTD, that is IoTD1, will get $S_c(1)$ from matrix S. Besides, AD_1 has N_{IoTD}, that is $M_r(1)$ [1]. $IoTD_1$ can calculate the N_{IoTD} by the equation (17). If the $IoTD_1$ is revoked, each AD needs to delete the first column in M.

C. ACCESS CONTROL IN OUR PROPOSED SCHEME

We implement access control in both the top layer and the bottom layer. In the top layer, to simplify the access problem, we set the access level as MS_d >MS_n >MS_p, and allow MS only having the same IDs of polynomials and random numbers can access the same task. As we use the polynomial based key agreement approach, it is easy to implement access control. As we define that only MS having the same IDs of polynomials can access the same task, MS_d, MS_n and MS_p of the same task should have the same polynomials. In order to implement the access control, we define that MS_d can get t-degree polynomials, MS_n can only get (t-1)-degree of the same polynomials, and (t-2)-degree for MS_p, respectively. However, the polynomials of AD are still t-degree.

The following steps show the detailed process:

Step 1: KDC will assign certain polynomials to MS according to the role of the MS. If MS is MS_d, then KDC assigns t-degree polynomials to the MS; if MS is MS_n, KDC assign (t-1)-degree polynomials to the MS; if MS is MS_p, KDC assign (t-2)-degree polynomials to the MS.

Step 2: MS sends its ID, and all the IDs of the assigned polynomials to AD.

Step 3: AD judges the role of the MS by its ID. If MS is MS_d, then AD uses t-degree polynomials to compute the session key K; if the MS is MS_n, AD extracts (t-1)-degree polynomials from t-degree polynomials to compute the session key K; if the MS is MS_p, AD extracts (t-1)-degree polynomials from t-degree polynomials to compute the session key K.

So if an MS wants to communicate with an AD in order to send request or receive data, it should send its ID and the IDs of the polynomials it has to the AD. AD can determine its role by the ID and use the corresponding part of the polynomials to generate the corresponding session key.

In the bottom layer, access control can be implemented by trust management. When an AD gets a task from an MS, it firstly defines the trust threshold value that each IoTD should have in order for taking part in the task. AD then selects the working IoTDs by the trust value and sends the task to selected IoTDs. After some interactions, the trust value of each IoTD will change, and once their values drop to the trust threshold value, they cannot access the group. Therefore, trust based access control in the bottom layer should follow the following policies: 1) AD uses trust value to decide which IoTD can take part in a task at the beginning of the new task. 2) AD uses trust value to decide which low trust value IoTD can be added into the task again. 3) AD uses trust value to decide which IoTD can sponsor IoTDs out of the group.

V. PERFORMANCE

This section provides security analysis and performance evaluation for the proposed scheme.

A. SECURITY ANALYSIS

As we can see from the access control in the top layer, it is simple and effective using the polynomial pool-based key pre-distribution approach. However, in this kind of scheme, an attacker could obtain many polynomials after compromising a number of IoTDs and then might launch an IoTD replication attack, which in our scheme is the mobile sink replication attack. In the community, it is normal for the existence of a KDC which stores the polynomial pool. In our scheme, each AD stores n distinct key polynomials. If an attacker compromises an AD, the polynomials cannot be used to attack other ADs. It is impossible for attackers to launch an AD replication attack. In addition, each valid MS can communicate with a certain AD which defines the connectivity between an MS and an AD as 1. In other words, an attacker could not make an AD replication attack in the top layer our scheme.

In the bottom layer, we show that the scheme meets security requirements mentioned in Section III:

1) GROUP KEY CONFIDENTIALITY

Group key confidentiality means that any IoTD out of the domain of a certain AD cannot derive the group key of this group. Only an IoTD which has the corresponding secret value x_i can compute the right group key. Other IoTDs that do not have the corresponding secret value x_i will obtain nonzero value when evaluating the validation polynomial which leads to the $K_j^{'}$ is not equal to K_j as shown in Equation (18). Because $P_v(x_i)$ is not equal to 0 if x_i is the secret of an invalid IoTDs which means $K_j^{''}$ is not equal to K_j. That is to say, our scheme has group key confidentiality.

$$K_j'' = \frac{P_k(x_i)}{P_{dr}(x_i)} = \frac{P_{dr}(x_i)K_j + P_v(x_i)}{P_{dr}(x_i)} \neq \frac{P_{dr}(x_i)K_j + 0}{P_{dr}(x_i)} \neq \frac{P_{dr}(x_i)K_j}{P_{dr}(x_i)} \neq K_j \quad (18)$$

2) FORWARD SECRECY

Forward secrecy means that an IoTD could no longer get the subsequent group key after being revoked. To compute the

group key, each IoTD needs to compute:

$$K_j^{"} = \frac{P_k(x_i)}{P_{dr}(x_i)} \tag{19}$$

If an IoTD is revoked, its secret value will be added into generating the double-revocation polynomial which leads to the evaluation of $P_v(x_i)$ equal to zero. Therefore, the revoked IoTD cannot compute $K_j^{"}$ as Equation (19) and cannot get the subsequent group key after being revoked.

In addition, the valid polynomial is used only once. A valid IoTD can obtain the valid polynomial and further obtain all the valid secret values by

$$P_v(x) = P_k(x) - P_{dr}(x) K_j \tag{20}$$

Although a valid IoTD can obtain all the valid secret values in this session, these valid secret values are only used once which will not impact the security of future session group key. In other words, our scheme enjoys forward secrecy.

3) BACKWARD SECRECY

Backward secrecy means that any new added IoTD cannot get the group key before it is actually added. If an IoTD is actually added in the k-th session by the AD, it will get the secret value x_{ik} and this secret value will be used to generate the valid polynomial.

$$x_{ik} = f\left(x_{i(k-1)}\right) \tag{21}$$

However, according to x_{ik}, the IoTD cannot obtain its secret value used in previous sessions for the one-way feature of hash function $f()$.

$$x_{i(k-1)} = f^{-1}(x_{ik}) \tag{22}$$

In addition, even the IoTD gets the secret value $x_{i(k-1)}$, this secret value do not belong to V_{k-1} which means it will not be used to generate the valid polynomial in (k-1)-th session. Therefore, the IoTD added in k-th session cannot obtain the group key used in (k-1)-th session. Neither can it obtain the one used in (k-2)-th session or the previous. That is to say, our scheme supports backward secrecy.

4) REVOCATION CAPABILITY

in our scheme, the main revocation has two cases: revoking some IoTDs forever, or for a certain number of sessions. Not like other schemes which only have t-revocation capability, our scheme can revoke any number of IoTDs. Note that the number of IoTDs is much less than the total number.

In our scheme, no masking polynomial is used (which is always a t-degree polynomial). This means, the revocation capability will not be limited to the degree of the used polynomial. The key material mainly consists of two parts: double revocation polynomial and valid polynomial. The double-revocation polynomial determines which IoTDs are revoked. The valid polynomial determines which IoTDs are valid. In addition, the valid polynomial is used only once.

Each AD controls two revocation sets R_1 and R_2. The secret value in the first set belongs to the IoTDs which are revoked forever, while secret values in the second set belong to the IoTDs which are revoked for a specific number of sessions. That is to say, our scheme supports the revocation capability which is mentioned in the security model.

B. FORMAL ANALYSIS USING BAN LOGIC

Due to the simplicity and straightforwardness of BAN logic [51], it is widely used for analyzing and verifying security protocols [44]. Compared with other schemes [28], [30] and [32], our protocol has more security. To prove the security, we present the following logical notations of BAN-logic for our analysis in Table 3 and the main rules, while the details can be referred to [44], [51]–[53].

TABLE 3. Notations of the ban logic.

Notation	Meaning
$P \mid\equiv X$	P believes X, *or* principal P can take X as true
$P \lhd X$	P sees the statement X
$P \mid\sim X$	P once said X,means that $P \mid\equiv X$ when P sent it
$P \mid\Rightarrow X$	P controls X and should be trusted for X
$\#(X)$	The message X is fresh
$Q \overset{K}{\leftrightarrow} P$	Principal Q ,P share a key K to communicate with each other in a secure way
$(X)_K$	The statement X is encrypted by K
$\langle X \rangle_Y$	X combined with formula Y

In addition, some primary BAN-logic rules are presented in the following:

Rule (1). Message meaning rules:

$$\frac{P \mid\equiv Q \overset{K}{\leftrightarrow} P, P \lhd (X)_K}{P \mid\equiv Q \mid\sim X}, \frac{P \mid\equiv Q \overset{Y}{\leftrightarrow} P, P \lhd \langle X \rangle_Y}{P \mid\equiv Q \mid\sim X} \tag{23}$$

Rule (2). Nonce verification rule:

$$\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\equiv X}{P \mid\equiv Q \mid\equiv X} \tag{24}$$

Rule (3). Jurisdiction rule:

$$\frac{P \mid\equiv Q \mid\Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X} \tag{25}$$

Rule (4). Freshness rule:

$$\frac{P \mid\equiv \#(X)}{P \mid\equiv (X, Y)} \tag{26}$$

Rule (5). Believe rule:

$$\frac{P \mid\equiv Q \mid\equiv (X, Y)}{P \mid\equiv X, P \mid\equiv Y} \tag{27}$$

Rule (6). Session key rule:

$$\frac{P \mid\equiv Q\#(X), P \mid\equiv Q \mid\equiv X}{P \mid\equiv Q \overset{K}{\leftrightarrow} P} \tag{28}$$

1) REASONABLE ASSUMPTIONS

Before verification of the proposed protocol, some reasonable assumptions should be made for authentication service (AS) between AD and IoTD as follows:

A1. $AS| \equiv \#\overset{K_{IoTD}}{\rightarrow} IoTD$: The AD believe that his challenge is fresh and is never sent by any other principal before. $AS \triangleleft n$, where n is a random number.

A2. $AS \triangleleft n$: The AD has once seen the challenge which has been generated by the AD.

A3. $AS| \equiv \overset{K_{IoTD}}{\rightarrow} IoTD$: The AD believes that K_{IoTD} is the public key of one device.

A4. $AS| \equiv IoTD \overset{K}{\rightarrow} AS$: The AD believes the session key K.

A5. $AS| \equiv \#(IoTD \overset{K}{\rightarrow} AS)$: The AD believes that the session key is fresh.

A6. $AS| \equiv (IoTD \overset{K}{\rightarrow} AS)$: The AD has once seen the session key.

2) AUTHENTICATION GOAL

The following goal of our proposed protocol should be achieved:

$$AS| \equiv IoTD| \equiv IoTD \overset{K}{\rightarrow} AS.$$

3) IDEALIZATION FORM OF OUR PROTOCOL

The idealized form of our proposed protocol should also be presented:

I. $IoTD \rightarrow AS : \left\{H_B\left(H_A\left(IoTD \overset{K}{\leftrightarrow} AS, S\right)\right)\right\}_{K_{IoTD}^{-1}}$: where H_A and H_B is the first hash and second hash. The S is the send number.

4) AUTHENTICATION PROTOCOL PROOF USING BAN LOGIC

Now, we analyze the protocol which achieves the goal G.

Since we have the assumptions A3, I, A2 and A6, by applying message meaning rule (23), we can infer that

$$AS| \equiv IoTD| \equiv \left\{H_B\left(H_A\left(IoTD \overset{K}{\leftrightarrow} AS, S\right)\right)\right\}_{K_{IoTD}^{-1}} \quad (29)$$

$$AS| \equiv IoTD| \equiv IoTD \overset{K}{\leftrightarrow} AS. \quad (30)$$

By applying nonce verification rule (24), if the conditions of A5 and $AS| \equiv IoTD| \overset{K}{\leftrightarrow} AS$ are both met, we can deduce $AS| \equiv IoTD| \equiv IoTD \overset{K}{\leftrightarrow} AS$.

According to the inferential rigorous logic process, we can obtain the final result the same as expected goal. We can conclude that our proposed protocol is secure between AD and IoTD under the analysis of BAN logic.

C. PERFORMANCE EVALUATION

In this subsection, we conduct performance evaluation of the proposed scheme in terms of storage overhead, computation overhead, communication overhead, and healing rate. Some simulation experiments about these elements of our work have been implemented to prove the efficiency with the other two other two schemes [28], [30] in storage and communication overheads. The simulation results will contribute in analyzing how to select parameters. We wrote the programming codes in C language and performed simulation experiments on Linux operating system in a virtual machine with 4 GB of memory and 2.5 GHz CPU. Without loss of generality, we also set the simulation scenarios regarding parameters and network which is similar to that of [32].

Parameters: To simplify the security analysis, we simulate the size of maximum broadcast packet is 64 KB. We assume the number of sessions T is 100 which is from 10 to 100 and for each session, no more than 5 IoTDs will randomly be revoked. In general, we suppose p is a 128-bit integer and q is a 512-bit integer which are required to get energy consumption of all devices (IoTDs and AD). We set v as new user(s) joining the group during m sessions, $1 \leq v < m$. Assuming the length of coefficients in each polynomial is 64bit. We conduct the experiment between healing rate and energy consumption to prove the completeness and security of our key distribution scheme when d is equal to 5 and the number of IoTD is 500. We set the drop probability of broadcast as 5% with respect to each session to discover the energy consumption of IoTDs.

Network: We assume that the number of neighbors of each peer is constant and equal to four. All nodes establish a secure channel with four neighbors. In our scheme, we assume the network topology is invariable. At each interaction, revoked users are replaced with new ones and distributed with fresh keys. They establish secure communications with four neighbors.

We discuss the case that an MS wants to distribute a task which needs the participation of N_{AD} ADs. And each AD controls N_{IoTD} IoTDs. We define Nb as the number of broadcasts, and Nh as the self-healing number of an IoTD, T as the number of $P_j(x)$ each broadcast has. we define that the random number has the same length with the polynomial value which can be denoted by Value.

Each AD needs to store n polynomials and $n!$ random numbers for a key agreement with MS since each AD can receive n! tasks. For key distribution and self-healing, an AD will store T-1 polynomials. Note that although the broadcast consists of T polynomials, the current key material polynomial is generated in that session. In addition, each AD stores all the secret values of its IoTDs and the number of them is N_{IoTD}. Each AD stores $n + N_{IoTD}+1$ IDs, n IDs of n polynomials, N_{IoTD} IDs of IoTDs in its group, and its device ID.

Each IoTD needs to store a secret value x_i and its device ID. We can see here, the storage overhead of IoTD in our scheme is low because it need not store any polynomial.

TABLE 4. Storage overhead.

Device	Polynomial	Value	ID
MS	nN_{AD}	N_{AD}	$1+(n+1)\,N_{AD}$
AD	$n+T$-1	$n!+N_{IoTD}$	$n+N_{IoTD}+1$
IoTD	0	1	1

Table 4 shows the storage overheads of three types of devices. MS needs to store n polynomials and a random

TABLE 5. Computation overhead.

Device	Hash	Polynomial generation	Polynomial addition	Polynomial multiplication	Polynomial evaluation	Encryption and decryption	Division
MS	N_{AD}	0	0	0	nN_{AD}	$2Nad$	0
AD	$1+N_{IoTD}Nb$	$2Nb$	Nb	Nb	n	2	0
IoTD	Nb	Nb	0	0	$2Nb$	0	Nb

TABLE 6. Communication overhead.

Device	Polynomial	Value
MS	0	$5\ Nad$
AD	$\frac{(1+T)T}{2}+\sum_{T+1}^{Nb}T$	5
IoTD	$\frac{(1+T)T}{2}+\sum_{T+1}^{Nb}T$	0

number RNc which can be used to generate a session key with AD. This implies that for a task, an MS needs to store nN_{AD} polynomials and N_{AD} random numbers in total. Each MS also stores $1+(n+1)N_{AD}$ IDs for each polynomial and each AD, together with its own device ID.

1) COMPUTATION OVERHEAD

Table 5 presents computation overheads of three types of devices.

In the key agreement between MS and AD, MS as well as AD needs to do one hash operation and n polynomial evaluations. Therefore, to distribute one task, an MS needs to do N_{AD} hash operations and nN_{AD} polynomial evaluations. In addition, MS and AD will encrypt and decrypt two random numbers $RNad$ and $RNms$, respectively.

In the group key distribution, there exist two cases: one is broadcast with Nb times and the other is self-healing with Nh times. In broadcast, each AD does two polynomial generation operations to generate $P_{dr}(x)$ and $P_v(x)$, respectively. AD will compute $P_{dr}(x)K_j$ by one polynomial multiplication evaluation, and then, does one polynomial addition evaluation to generate $P_j(x)$.

AD needs to do hash evaluations on the secret values of its members IoTDs, the number of which is N_{IoTD}. Each IoTD will do one hash evaluation to update its secret value.

Each IoTD computes the group key by Equation (8). In fact, it will do one polynomial generation operation to generate $P_{dr}(x)$, two polynomial evaluations, and one division operation, respectively. If an IoTD loses a broadcast, it does the same computation evaluations because it can use previous Pj(x) while the computation process is the same.

2) COMMUNICATION OVERHEAD

Table 6 shows communication overheads of three types of devices. We will ignore sent IDs in each session, because the number of IDs are different. We also ignore the communication overhead caused by IDs because the identities can be chosen over a small finite field [40].

TABLE 7. Energy consumption.

Device	Energy consumption (bit)
AD	$n\log^p$
IoTD	$n/d\log^p$

Regarding the key agreement between MS and AD, MS sends $n+1$ IDs, and random number $RNms$ and RNc. AD will send back $RNad$. In addition, both MS and AD will also send and receive two encryption data, $K(RNad)$ and $K(RNsm)$, respectively. We define the length of the two data is as the same as Value. Therefore, both MS and AD will send or receive 5 values: RNc, $RNad$, $RNsm$, $K(RNad)$ and $K(RNsm)$. As a consequence, MS which needs to communicate with N_{AD} ADs will send N_{AD} times of the above mentioned overhead.

In the group key distribution, each AD broadcasts T polynomials. The total number of polynomials each AD sends in Nb sessions is in fact the following:

$$1+2+3+\cdots+T-1+\underbrace{T+T+\cdots+T}_{Nb-T-1}=\frac{(1+T)T}{2}+\sum_{T+1}^{Nb}T \tag{31}$$

Each IoTD will receive at most T polynomial in each session. Note that, an IoTD which loses a broadcast will have less communication overhead. Obliviously, the self-healing process will not increase communication overheads since it has no interaction.

3) ENERGY CONSUMPTION

We discuss the energy consumption in group key distribution between AD and IoTD. In addition, we use the number and size of transmitted messages to evaluate the energy consumption as that in [37]. Other literatures including [39] also show the feasibility of this method.

We assume one AD controls n IoTDs and divides its IoTDs into d parts for increasing the successful communication rate of IoTDs. And the coefficients of the key material polynomial and secret values belong to F_p, which means the length of them is $\log^p$.

It is easy to see that from Table 7 that the energy consumption of each device is mainly determined by the degree of the sending or receiving polynomial.

Assuming that in the initial phase, there are 500 IoTDs, and in each session; 0-5 IoTDs will be revoked randomly. Fig 3

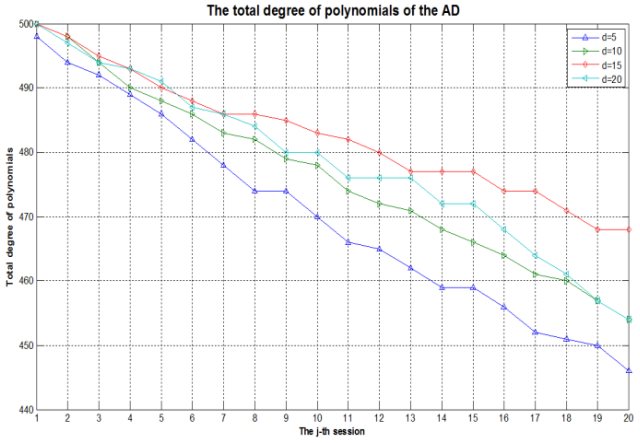

FIGURE 3. The total degree of polynomials of the AD.

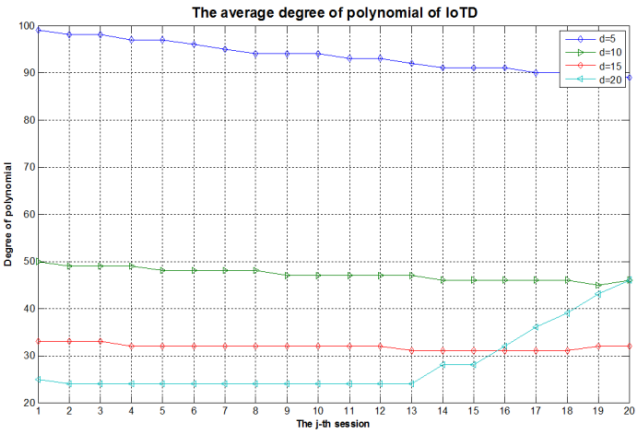

FIGURE 4. The average degree of polynomial of IoTD.

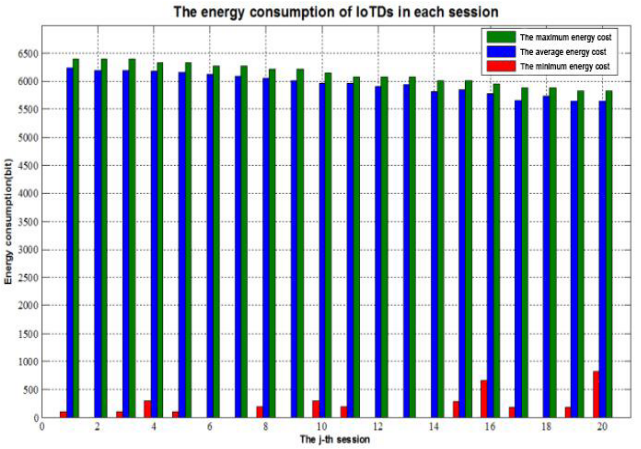

FIGURE 5. The energy consumption of IoTDs in each session.

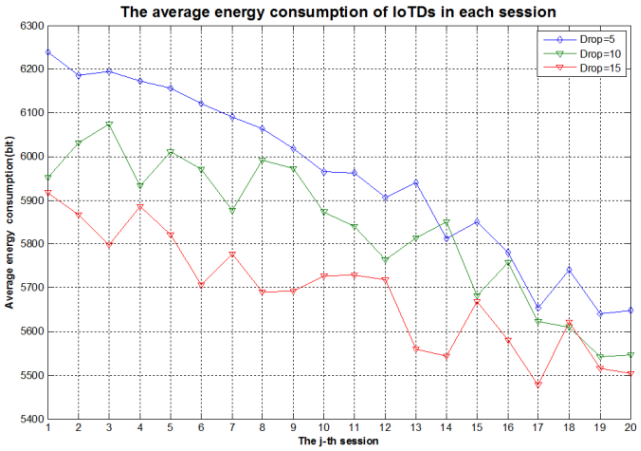

FIGURE 6. The average energy consumption of IoTDs in each session.

and Fig 4 show the total degrees of polynomials of AD and IoTD (average degree) when the number of parts d increases.

Obviously from Fig 3, the total polynomial degree of AD will decrease when revocation IoTD increases. While in Fig 4, when d is equal to 20, the degree of IoTD first decreases, and then increases because the degree of each polynomial is determined by the number of revocation IoTDs and the number of IoTDs in each part, together. Therefore, it is suggested that the parts of all the valid IoTDs can be divided by the number of revocation IoTDs.

Assuming the length of coefficients in each polynomial is 64bit, we conduct the following experiment in the case that d is equal to 5 and the number of IoTD is 500. The following two experiments aim to discover the energy consumption of IoTDs when the drop probability of broadcast is 5% in respect to each session.

In Fig 5, the maximum energy cost and the average energy cost gradually decline. This is because the degree of the received key material polynomial gradually decreases, which can be seen from Fig 4. The minimum energy cost of IoTDs is caused by the loss of broadcast. An IoTD loses a broadcast when it only receives a part of the broadcast.

We conduct the experiment when the drop probability of broadcast changes which can be seen from Fig 6. It is easy to see from Fig 6 that the average energy cost of all IoTDs decreases when the drop probability of broadcast increases. As can be seen from this figure, no matter how the drop probability of broadcast increases, energy consumption cost of all IoTs in each session is always decreasing. This reflects that the energy consumption is negatively correlated with the drop probability of broadcast. The driving factor of that is the same reason as that of Fig 5.

We also show the detailed energy consumption of all IoTDs in one session. The session is the 18-th session when d is 5 in Fig 3. The number of valid IoTDs in that session is 452. Fig 7 shows the detail. We can see the energy consumption of all IoTs in one session in respect to drop probability is always decreasing.

4) HEALING RATE

In this subsection, we discuss the healing rate of our proposed scheme. In our scheme, if an IoTD loses the broadcast, it could heal the group key in this session with the help of the later broadcast with T sessions. The healing rate of our

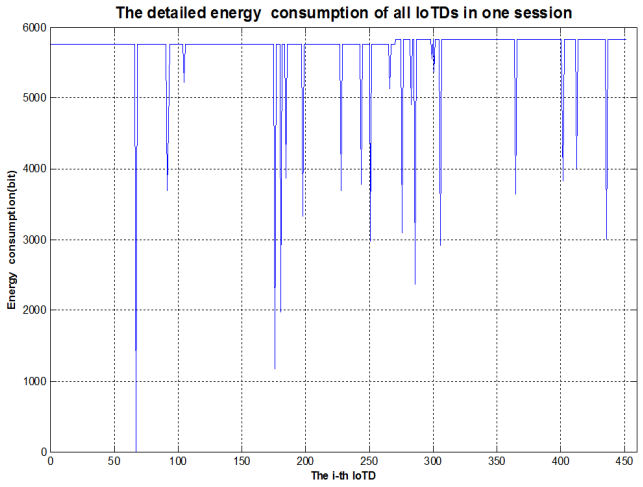

FIGURE 7. The detailed energy consumption of all IoTDs in one session.

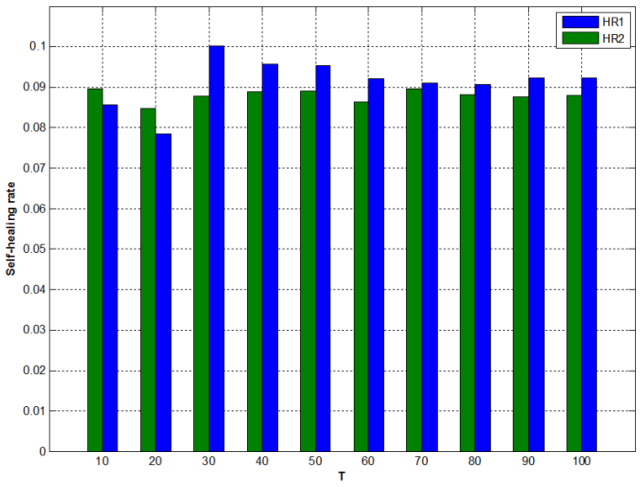

FIGURE 8. The healing rate when T is from 10 to 100.

scheme is similar to Formula (32). Here, HR refers to the healing rate while P_l is the broadcast loss probability.

$$HR = 1 - \left(1/P_l\right)^T \tag{32}$$

We conduct our experiment in which the loss probability is 5%, and T from 10 to 50. We set the number of IoTDs 500 and the number of sessions 100. As is shown in Fig 8, the experiment result shows the healing rate is very high even T is small. According to this experiment, we can see that no matter what the session T it is from 10 to 100, an IoTD works well to heal the group key.

Table 8 shows the detailed data of the conducted experiment, in which C_t is the total number of broadcast loss. C_h is the healing times. C_l is the loss times of the last broadcast which cannot be healed because there is no more broadcast left. HR1 refers to the healing rate with the consideration of the last broadcast. With regarding to computing HR2, we do not take the last broadcast loss into account. It is easy to see that values of HR1 and HR2 in different session are all

TABLE 8. Healing rate.

T	C_t	C_h	C_l	HR1	HR2
10	24802	24747	27	0.997782437	0.99778002
20	24948	24907	24	0.998356582	0.998354999
30	25083	25029	26	0.997847147	0.997844914
40	24830	24777	25	0.997865485	0.997863334
50	25135	25082	27	0.997891387	0.997889119
60	25214	25154	30	0.99762037	0.997617535
70	24915	24852	27	0.997471403	0.99746866
80	24931	24874	24	0.99771369	0.997711487
90	24655	24605	26	0.997972014	0.997969873
100	25048	25008	24	0.998403066	0.998401535

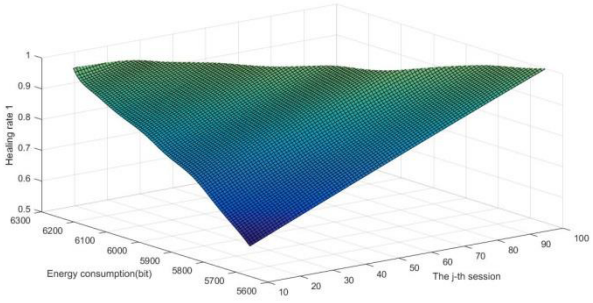

FIGURE 9. Energy consumption and healing rate HR1 when T is from 10 to 100.

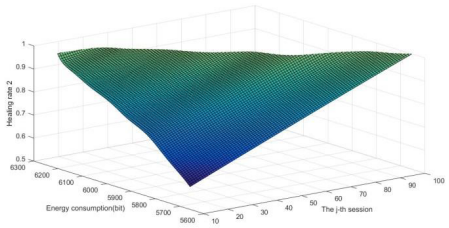

FIGURE 10. Energy consumption and healing rate HR2 when T is from 10 to 100.

above 0.997. HR1 and HR2 both show the optimistic healing rates of our proposed scheme.

5) SIMULATION FOR ENERGY CONSUMPTION AND SELF-HEALING

We provide simulation results for energy consumption, session number and self-healing altogether. Here healing rate reflects security of the scheme since higher healing rate indicates the scheme is more secure. This is because that higher healing rate assures key distribution more complete and secure. We assume sessions ranging from 10 to 100, energy consumption ranging from 5600 to 6300, and healing rate ranging from 0.5 to 1. Simulation results are displayed in Fig. 9 and Fig.10. The simulations results show the varying trends of energy consumption and security i.e. self-healing, when session changes from 10 to 100.

When energy consumption is fixed, healing rate becomes higher when session increases. The behind reason is when session increases, the IoTDs have received more data and therefore self-healing becomes more successful, which results in the increase of the healing rate. While healing

rate is fixed, energy consumption decreases when session increases. This is because when session increases, the IoTDs have received more data and therefore data re-transmission for key distribution decreases, which results in the decrease of energy consumption. On the other hand, when session is fixed, healing rate grows larger when energy consumption becomes higher. Obviously, the system is also more secure during the status of having larger healing rates. The behind reason is that when healing rate is larger, self-healing of key distribution of the system is more successful and therefore the system is more secure.

6) COMPARISON WITH OTHER SIMILAR SCHEMES

Now, we focus on the efficiency performance, including the storage overhead and the communication overhead. To show the performance of the proposed self-healing scheme, comparison with two other schemes is presented. Table 8 summaries comparisons between three self-healing key distribution schemes. Here, we consider the communication overhead of each IoTD. The reason that we compare our scheme with [28] and [30] is: [28] and [30] both utilize polynomials. Note that [28] is an extension to [38] which is based on polynomials. Obviously from this table, only scheme [28] and ours have the same storage overhead, i.e., log_p, which is optimal compared with the other scheme in [30]. Moreover, we can find that the communication overhead of our scheme $Tn/d\ log_p$ is obviously less than that of the scheme in [28] and [30]. Therefore, our proposed scheme achieves the most favorable efficiency in terms of storage and communication overheads.

In [30], v is the number of sessions in which there are new user(s) joining during m sessions, $1 \leq v < m$. The scheme in [28] is an extension to [38]. q is used in [38] while p is used in [28]. To simplify the comparison, we set $p = q$. In addition, d in [28] has the same function with the parameter T in our scheme. Therefore, we use T to replace d in [28]. And d in our scheme is the number of parts each broadcast contains. Then Table 9 can be rewritten as the following table:

TABLE 9. Storage and communication comparison.

Scheme	Storage overhead(bits)	Communication overhead(bits)
Scheme in [28]	*log p*	*(d+1)((t+2) log q +(t+1)log p)*
Scheme in [30]	*(t+3) log p*	*((t+1)v+j) log q*
Our scheme	*log p*	*Tn/d log p*

TABLE 10. Storage and communication comparison.

Scheme	Storage overhead (bits/ log p)	Communication overhead (bits/ log p)
Scheme in [28]	1	*2(T+1)(t+1)*
Scheme in [30]	*t*+3	*(t+1)v+j*
Our scheme	1	*Tn/d*

We can see from Table 10 that the scheme in [30] and ours are much better than that of [32] in storage overhead. In our scheme, T can be a small constant which is much smaller than the value v in [32]. In [30] and [32], the degree of the polynomials is t while in our scheme the degree of the polynomials each IoTD receives is n/d. We can set d a large number which makes $n/d < t$. Note that the value of d will not affect the secret of our scheme but will increase the computation overhead of the AD. Therefore, our scheme is more efficient in communication overhead than the other two schemes. In addition, the degree of the polynomials each IoTD receives will decrease when more and more IoTDs are revoked.

VI. CONCLUSION

In this paper, we proposed a key management scheme for community healthcare which is one of the most important services in the community. Our system model which contains three kinds of devices can be divided into two layers, i.e., the top layer: between MS and AD, and the bottom layer: between AD and IoTDs. In the top layer, we use a polynomial based key agreement approach; while in the bottom layer, we propose a polynomial based self-healing group key distribution method, which meets the requirements of our secret model. In the top layer, our scheme is secure against MS replication attacks and can receive $n!$ tasks at the same time. In addition, we define access level for the users of the sub-networks of the community healthcare system, which is simple but practical. In the bottom layer, our scheme has the capabilities of group key confidentiality, forward security, backward security, revocation, and self-healing, which are imperative properties for group key distributions. Besides, we apply trust method into our scheme, which helps for facilitating efficient access control and message broadcast in the sub-network of the community healthcare system. At last, we conduct security analysis which shows our proposed scheme meets all the requirements we presented in the security model. We also conduct performance analysis which shows the overheads in term of storage, computation and communication, and some experiments which relate to the healing rate of the proposed scheme. We also compare our key distribution scheme with other schemes and show its advantages. We intend to focus on the solutions for AD replication attacks and other attacks in our future work.

ACKNOWLEDGEMENT

The authors thank the anonymous reviewers for their valuable comments for improving the quality of the paper. The first author thanks L. Jiang for the work of formatting and figuring in this paper.

REFERENCES

[1] D. Giusto, A. Iera, G. Morabito, and L. Atzori, Eds., "The Internet of Things," in *Proc. 20th Tyrrhenian Workshop Digit. Commun.* New York, NY, USA: Springer, 2010.
[2] R. Roman, P. Najera, and J. Lopez, "Securing the Internet of Things," *Computer*, vol. 44, no. 9, pp. 51–58, Sep. 2011.
[3] H. Yu, J. He, T. Zhang, P. Xiao, and Y. Zhang, "Enabling end-to-end secure communication between wireless sensor networks and the Internet," *World Wide Web*, vol. 16, no. 4, pp. 515–540, Jul. 2013.
[4] T. J. Swamy, T. Sandhya, and G. Ramamurthy, "Link evaluation of uniform grid based wireless sensor networks to base station with leveling and clustering," in *Proc. Int. Conf. Comput. Commun. Inform. (ICCCI)*, Jan. 2013, pp. 1–5.

[5] E. C.-H. Ngai and I. Rodhe, "On providing location privacy for mobile sinks in wireless sensor networks," *Wireless Netw.*, vol. 19, no. 1, pp. 115–130, 2013.
[6] X. Wang and S. Zhong, "An IPv6 address configuration scheme for wireless sensor networks based on location information," *Telecommun. Syst.*, vol. 52, no. 1, pp. 151–160, 2013.
[7] W.-T. Sung and K.-Y. Chang, "Evidence-based multi-sensor information fusion for remote health care systems," *Sens. Actuators A, Phys.*, vol. 204, pp. 1–19, Dec. 2013.
[8] Y. L. Sun and K. J. R. Liu, "Analysis and protection of dynamic membership information for group key distribution schemes," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 2, pp. 213–226, Jun. 2007.
[9] M. Al Ameen, J. Liu, and K. Kwak, "Security and privacy issues in wireless sensor networks for healthcare applications," *J. Med. Syst.*, vol. 36, pp. 93–101, Feb. 2012.
[10] G. Han, J. Jiang, L. Shu, J. Niu, and H.-C. Chao, "Management and applications of trust in wireless sensor networks: A survey," *J. Comput. Syst. Sci.*, vol. 80, no. 3, pp. 602–617, May 2014.
[11] S.-C. S. Cheung, Y. Sun, K. Aberer, J. Haritsa, B. Horne, and K. Hwang, "Guest editorial: Special issue on privacy and trust management in cloud and distributed systems," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 6, pp. 835–837, Jun. 2013.
[12] K. Govindan and P. Mohapatra, "Trust computations and trust dynamics in mobile adhoc networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 2, pp. 279–298, 2nd Quart., 2012.
[13] A. B. Waluyo, D. Taniar, W. Rahayu, A. Aikebaier, M. Takizawa, and B. Srinivasan, "Trustworthy-based efficient data broadcast model for P2P interaction in resource-constrained wireless environments," *J. Comput. Syst. Sci.*, vol. 78, no. 6, pp. 1716–1736, 2012.
[14] H. Sivasankari, R. Aparna, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, "TGAR: Trust dependent greedy anti-void routing in wireless sensor networks (WSNs)," in *Proc. 3rd Int. Conf. Trends Inf., Telecommun. Comput.*, vol. 150. New York, NY, USA: Springer, 2013, pp. 39–45.
[15] T. Zahariadis, P. Trakadas, H. C. Leligou, S. Maniatis, and P. Karkazis, "A novel trust-aware geographical routing scheme for wireless sensor networks," *Wireless Pers. Commun.*, vol. 69, no. 2, pp. 805–826, 2013.
[16] F. Bao, I.-R. Chen, M. Chang, and J.-H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection," *IEEE Trans. Netw. Service Manage.*, vol. 9, no. 2, pp. 169–183, Jun. 2012.
[17] F. G. Mármol and G. M. Pérez, "TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks," *J. Netw. Comput. Appl.*, vol. 35, no. 3, pp. 934–941, 2012.
[18] D. He, C. Chen, S. Chan, J. Bu, and A. V. Vasilakos, "ReTrust: Attack-resistant and lightweight trust management for medical sensor networks," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 4, pp. 623–632, Jul. 2012.
[19] B. Tian, S. Han, S. Parvin, J. Hu, and S. Das, "Self-healing key distribution schemes for wireless networks: A survey," *Comput. J.*, vol. 54, no. 4, pp. 549–569, 2011.
[20] T. Rams and P. Pacyna, "A survey of group key distribution schemes with self-healing property," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 820–842, 2nd Quart., 2013.
[21] R. Dutta and S. Mukhopadhyay, "Designing scalable self-healing key distribution schemes with revocation capability," in *Parallel and Distributed Processing and Applications* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2007, pp. 419–430.
[22] X. Du, Y. Wang, J. Ge, and Y. Wang, "An ID-based broadcast encryption scheme for key distribution," *IEEE Trans. Broadcast.*, vol. 51, no. 2, pp. 264–266, Jun. 2005.
[23] S. Han, B. Tian, Y. Zhang, and J. Hu, "An efficient self-healing key distribution scheme with constant-size personal keys for wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2010, pp. 1–5.
[24] B. Tian, S. Han, J. Hu, and T. Dillion, "A mutual-healing key distribution scheme in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 80–88, Jan. 2011.
[25] G. Sáez, "On threshold self-healing key distribution schemes," in *Cryptography and Coding* (Lecture Notes in Computer Science), vol. 3796, N. P. Smart, Ed. Berlin, Germany: Springer, 2005, pp. 340–354.
[26] R. Dutta, S. Mukhopadhyay, A. Das, and S. Emmanuel, "Generalized self-healing key distribution using vector space access structure," in *NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet* (Lecture Notes in Computer Science), vol. 4982, 2008, pp. 612–623.
[27] Z. Wang and M. Ma, "A collusion-resilient self-healing key distribution scheme for wireless sensor networks," in *Proc. IEEE ICC*, Jun. 2012, pp. 566–570.
[28] T. Rams and P. Pacyna, "Long-lived self-healing group key distribution scheme with backward secrecy," in *Proc. NETSYS*, vol. 13, Mar. 2013, pp. 59–65.
[29] R. Jiang, J. Luo, and X. Wang, "A logic-route key tree based group key management scheme for wireless sensor networks," in *Proc. IEEE ICCC*, Aug. 2013, pp. 686–691.
[30] Q. Wang, H. Chen, L. Xie, and K. Wang, "One-way hash chain-based self-healing group key distribution scheme with collusion resistance capability in wireless sensor networks," *Ad Hoc Netw.*, vol. 11, pp. 2500–2511, Nov. 2013.
[31] J. Chen, H. Zhang, B. Fang, X. Du, H. Yu, and X. Yu, "An efficient and sustainable self-healing protocol for unattended wireless sensor networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2012, pp. 5356–5361.
[32] A. Miyaji and K. Omote, "How to build random key pre-distribution schemes with self-healing for multiphase wsns," in *Proc. IEEE 27th Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2013, pp. 205–212.
[33] V. S. Janani and M. S. K. Manikandan, "An efficient genetic based broadcast stateless group key management scheme with dynamic rekeying in mobile ad-hoc networks," *Wireless Pers. Commun.*, vol. 105, no. 3, pp. 857–876, Apr. 2019.
[34] A. Rasheed and R. N. Mahapatra, "The three-tier security scheme in wireless sensor networks with mobile sinks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 5, pp. 958–965, May 2012.
[35] A.-S. K. Pathan, T. T. Dai, and C. S. Hong, "A key management scheme with encoding and improved security for wireless sensor networks," in *Proc. ICDCIT*, vol. 4317, Bhubaneswar, India, pp. 102–115, 2006.
[36] C. W. Park, S. J. Choi, and H. Y. Youn, "A noble key pre-distribution scheme with LU matrix for secure wireless sensor networks," in *Lecture Notes in Artificial Intelligence*, vol. 3801. Berlin, Germany: Springer, 2005, pp. 494–499.
[37] O. Delgado-Mohatar, A. Fúster-Sabater, and J. M. Sierra, "A light-weight authentication scheme for wireless sensor networks," *Ad Hoc Netw.*, vol. 9, pp. 727–735, Jul. 2011.
[38] R. Dutta, E.-C. Chang, and S. Mukhopadhyay, "Efficient self-healing key distribution with revocation for wireless sensor networks using one way key chains," in *Applied Cryptography and Network Security* (Lecture Notes in Computer Science), vol. 4521. Berlin, Germany: Springer, 2007, pp. 385–400.
[39] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 7, no. 3, pp. 537–568, 2009.
[40] D. Hong and J.-S. Kang, "An efficient key distribution scheme with self-healing property," *IEEE Commun. Lett.*, vol. 9, no. 8, pp. 759–761, Aug. 2005.
[41] H. Kurnio, R. Safani-Naini, and H. Wang, "A secure re-keying scheme with key recovery property," in *Information Security and Privacy* (Lecture Notes in Computer Science), vol. 2384. Berlin, Germany: Springer, 2002, pp. 40–55.
[42] J. Shen, S. Chang, Q. Liu, J. Shen, and Y. Ren, "Implicit authentication protocol and self-healing key management for WBANs," *Multimedia Tools Appl.*, vol. 77, no. 9, pp. 11381–11401, May 2018.
[43] T. J. Tsitaitse, Y. Cai, and A. Ditta, "Secure self-healing group key distribution scheme with constant storage for SCADA systems in smart grid," *Wireless Pers. Commun.*, vol. 101, no. 3, pp. 1749–1763, Aug. 2018.
[44] M. Bilal and S. G. Kang, "An authentication protocol for future sensor networks," *Sensors*, vol. 17, no. 5, p. 979, 2017.
[45] J.-H. Cho, I.-R. Chen, and K. S. Chan, "Trust threshold based public key management in mobile ad hoc networks," *Ad Hoc Netw.*, vol. 44, pp. 58–75, Jul. 2016
[46] F. Al-Turjman, H. Zahmatkesh, and R. Shahroze, "An overview of security and privacy in smart cities' IoT communications," *Trans. Emerg. Telecommun. Technol.*, Jul. 2019.
[47] C. Yang, H. Zhang, and J. Su, "Quantum key distribution network: Optimal secret-key-aware routing method for trust relaying," *China Commun.*, vol. 15, no. 2, pp. 33–45, 2018.
[48] S. A. Alabady, F. Al Turjman, and S. Din, "A novel security model for cooperative virtual networks in the IoT era," in *International Journal of Parallel Programming*. New York, NY, USA: Springer, 2018, pp. 1–16.
[49] X. Fangfang, Y. Zhao, and Y. Bo, "Mutual authentication and key distribution protocol based on the Trust-Third-Party in mobile RFID environment," in *Proc. 13th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, May/Jun. 2018, pp. 2501–2509.

[50] F. A. Turjman and S. Alturjman, "Context-sensitive access in industrial Internet of Things (IIoT) healthcare applications," *IEEE Trans. Ind. Inform.*, vol. 14, no. 6, pp. 2736–2744, Jun. 2018.
[51] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 426, no. 1871, pp. 233–271, 1989.
[52] M. S. Alkatheiri, M. H. Eldefrawy, and M. K. Khan, "BAN logic-based security proof for mobile OTP authentication scheme," in *Proc. 7th FTRA Int. Conf. Future Inf. Technol. (FutureTech)*, Vancouver, BC, Canada, vol. 1, Jun. 2012, pp. 53–59.
[53] L. L. Cao and W. C. Ge, "Formal analysis of an efficient handover authentication scheme for EAP-based wireless networks with extending BAN logic," *Appl. Mech. Mater.*, vols. 401–403, pp. 1864–1867, Sep. 2013.

SONG HAN was a Visiting Professor with the School of Data Sciences, East China Normal University, 2018 and 2019. He is currently a Professor with the School of Computer and Information Engineering, Zhejiang Gongshang University. He is also with Zhejiang Ponshine Information Technology Company Ltd., Hangzhou, China. His current research interests include information and network security, blockchain, the Internet of Things, and data privacy protection.

MIANXUE GU is currently pursuing the master's degree with the School of Computer and Information Engineering, Zhejiang Gongshang University. His current research interests include information and network security, blockchain, and data privacy protection.

BAILIN YANG is currently a Professor with the School of Computer and Information Engineering, Zhejiang Gongshang University. His current research interests include mobile game, blockchain, mobile graphics, network security, and graphic computing.

JIANHONG LIN is currently pursuing the Ph.D. degree with Zhejiang University. He is also the Chief Technology Officer of Zhejiang Ponshine Information Technology Company Ltd., Hangzhou, China. His current research interests include network security and security management.

HAIBO HONG is currently an Associate Professor with the School of Computer and Information Engineering, Zhejiang Gongshang University. His current research interests include information security and cryptography.

MENGJIAO KONG is currently pursuing the master's degree with the School of Computer and Information Engineering, Zhejiang Gongshang University. Her current research interests include information and network security, and data privacy protection.