Novel Design of Secure End-to-End Routing Protocol in Wireless Sensor Networks

Lein Harn, Ching-Fang Hsu, Ou Ruan, and Mao-Yuan Zhang

***Abstract*— In wireless sensor networks, the secure end-to-end data communication is needed to collect data from source to destination. Collected data are transmitted in a path consisting of connected links. All existing end-to-end routing protocols propose solutions in which each link uses a pairwise shared key to protect data. In this paper, we propose a novel design of secure end-to-end data communication. We adopt a newly published group key pre-distribution scheme in our design, such that there is a unique group key, called path key, to protect data transmitted in the entire routing path. Specifically, instead of using multiple pairwise shared keys to repeatedly perform encryption and decryption over every link, our proposed scheme uses a unique end-to-end path key to protect data transmitted over the path. Our protocol can authenticate sensors to establish the path and to establish the path key. The main advantage using our protocol is to reduce the time needed to process data by intermediate sensors. Moreover, our proposed authentication scheme has complexity $O(n)$, where n is the number of sensors in a communication path, which is different from all existing authentication schemes which are one-to-one authentications with complexity $O(n^2)$. The security of the protocol is computationally secure.**

***Index Terms*— Secure communications, group keys, authentication, secret sharing.**

I. Introduction

Wireless sensor networks (WSN) have been deployed in various applications to collect information from human body, battle fields, smart power grids, Interstate highways, etc. Sensors are subjected by their physical limitations on hardware, storage space, computational power, etc. Developing efficient solutions to protect information in sensor networks is a challenging task.

User authentication and key establishment are two fundamental security functions in most secure communications. The user authentication enables communication entities to authenticate identities of their communication partners. After users being successfully authenticated, a key establishment enables a secret session key to be shared among communication entities such that all exchange information can be protected using this shared key.

Traditional communications are one-to-one type of communications which involves only two communication entities. Most existing user authentication schemes [1]–[7] involve only two entities, one is the prover and the other one is the verifier. The verifier interacts with the prover to validate the identity of the prover. However, communication has been moved to many-to-many communications recently, also called *group communications*. Traditional user authentication which authenticates one user at one time is no longer suitable for a group communication which involves multiple users. Recently, a new type of authentication, called *group authentication* [8], is proposed which can be used to determine whether all users belong to the same group or not. The group authentication is very efficient since it can authenticate all members at one time. However, the group authentication can only be used as a pre-processing of user authentication since if there are non-members, group authentication cannot determine who are non-members. Additional one-to-one user authentications are needed to identify non-members.

Most popular key establishment schemes in WSNs are to establish pairwise shared keys for sensors. Eschenauer and Gligor [8] proposed the first *random key pre-distribution scheme* which is a probabilistic schemes and does not guarantee connectivity in WSNs. Each sensor is preloaded with k keys randomly selected from a large pool of keys. After the deployment, if two neighboring sensors share at least one key, they can use one of the shared key to establish a secure link. Otherwise, they should determine a path which is connected by successive secure links. The values of the key size k on each sensor and the key pool size are chosen to guarantee a high probability of connectivity. Chan *et al.* [9] proposed a *Q-composite scheme* to enhance the resilience of the random key scheme. Du *et al.* [10] proposed an enhanced random scheme assuming the node deployment knowledge. In 2013, Ruj *et al.* [11] proposed the first triple key establishment in WSNs. Any three sensors can establish unique triple keys among them.

The deterministic schemes do guarantee the connectivity in WSNs. Most deterministic schemes are based on threshold cryptography. Most deterministic schemes are to establish

Manuscript received October 19, 2015; revised November 20, 2015; accepted November 24, 2015. Date of publication November 30, 2015; date of current version February 8, 2016. This work was supported in part by the Self-Determined Research Funds of CCNU from the colleges' basic research and operation of the Ministry of Education under Grant CCNU15ZD003 and Grant CCNU15A02018, in part by the Major Project of National Social Science Fund under Grant 12&2D223, in part by the Educational Commission of Hubei Province of China under Grant D20151401, and in part by the Research Fund for the Doctoral Program of Hubei University of Technology under Grant BSQD13040. The associate editor coordinating the review of this paper and approving it for publication was Dr. Akshya Swain. *(Lein Harn and Ching-Fang Hsu contributed equally to this work.) (Corresponding author: Ching-Fang Hsu.)*

L. Harn is with the School of Computer Science and Technology, Hubei University of Technology, Wuhan 430068, China, and also with the Department of Computer Science Electrical Engineering, University of Missouri–Kansas City, Kansas City, MO 64110 USA (e-mail: harnl@umkc.edu).

C.-F. Hsu and M.-Y. Zhang are with the Computer School, Central China Normal University, Wuhan 430079, China (e-mail: cherryjingfang@gmail.com; zhangmyccnu@126.com).

O. Ruan is with the School of Computer Science and Technology, Hubei University of Technology, Wuhan 430068, China (e-mail: 12695133@qq.com).

Digital Object Identifier 10.1109/JSEN.2015.2504375

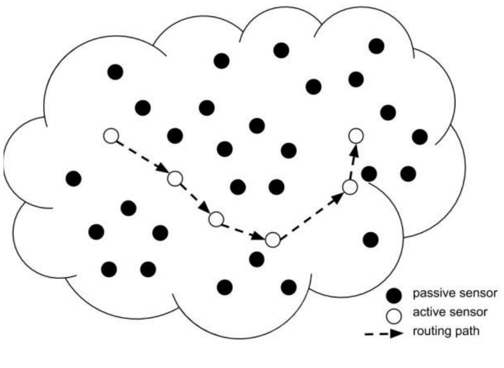


Fig. 1. Routing path in WSN.

pairwise shared keys between two sensors. Blom [12] proposed the first pairwise key establishment scheme based on threshold cryptography and Blundo *et al.* [13] further investigated the key establishment using polynomials. Khan *et al.* [14] proposed a pre-distribution scheme using a symmetric matrix and a generator matrix of maximum rank distance to establish pairwise keys for sensor nodes. In a recent paper by Harn and Xu [15] which is a pre-distribution scheme of group keys in sensor networks using a multivariate polynomial. The advantage of their scheme is to limit the storage space of each sensor to be linearly independent on the size of network. Recently, public-key based algorithms have been proposed and used in WSNs. For example, D'Souza and Varaprasad, *et al.* [16] proposed a secure node disjoint multipath routing protocol for WSNs in which data packets are transmitted in a secure manner by using the digital signature crypto system. Azarderakhsh *et al.* [17] proposed an algorithm and architecture for elliptic curve cryptography for secure applications. However, public-key computations take too much computational resources.

In WSNs, secure end-to-end data communication is needed to collect data from source to destination. Collected information needs to be forwarded in a path consisting of routing sensors from source sensor to sink sensor. Figure 1 illustrates a routing path in WSN. Collected data are transmitted in a path consisting of connected links. All existing end-to-end routing protocols propose solutions in which each link uses a pairwise shared key to protect data. In fact, this approach uses hop-to-hop solution to provide end-to-end secure data communication. One main concern in using a random key pre-distribution scheme to establish pairwise shared keys is that this approach does not guarantee the connectivity between two sensors. In the end-to-end secure communication scheme proposed by Gu *et al.* [18], it uses a methodology called *differentiated key pre-distribution*. The core idea is to distribute different number of keys to different sensors to enhance the resilience of certain links.

In this paper, we propose a novel design of secure end-to-end data communication. We adopt a newly published group key pre-distribution scheme [15] in our design such that there is a unique group key, called *path key*, to protect data transmitted in entire path. Specifically, instead of using multiple pairwise shared keys to repeatedly perform encryption and decryption over every link, our proposed scheme uses a unique end-to-end path key to protect data transmitted over the path. Our protocol can authenticate sensors to establish a routing path and to establish a path key. The main advantage of our protocol is to reduce the time needed to process data by intermediate sensors. Moreover, our proposed authentication scheme has complexity $O(n)$, where n is the number of sensors in a communication path, which is different from all existing authentication schemes which are one-to-one authentications with complexity $O(n^2)$.

Here, we summarize the contributions of our paper.

- We propose a novel secure end-to-end routing protocol in which a unique path key is established for all sensors in the path.
- Our protocol provides both authentication of sensors to establish the path and to establish the path key.
- Our authentication scheme has complexity $O(n)$.
- Our protocol is polynomial-based protocol so the computation is very efficient.

The rest of this paper is organized as follows: In the next section, we review a pre-distribution scheme of group keys which was published recently. In Section 3, we describe the model of our protocol and the security feature of our protocol. In Section 4, we propose our protocol. We analyze the security and performance in Section 5. We conclude in Section 6.

II. Review of Pre-Distribution Scheme for Establishing Group Keys [15]

Our proposed protocol is built upon a recent paper by Harn and Hsu [15] which is a pre-distribution scheme of group keys in sensor networks using a multivariate polynomial. In this section, we review their scheme.

There has a key generation center (KDC) and there are n sensors, $\{P_1, P_2, \ldots, P_1\}$. Each sensor is loaded with keys by the KDC initially. The KDC selects a RSA [19] modulus N, where N is the product of two large safe primes, p and q, i.e., $p{=}2p'{+}1$ and $q{=}2q'{+}1$, where p' and q' are also primes. p and q are KDC's secrets, N is made publicly known. The KDC selects a random polynomial having degree k as $f(x) = a_k x^k + \ldots + a_1 x + a_0 \bmod N$, and uses it to generate an m-variate polynomial, $F(x_1, x_2, \ldots, x_m) = \prod_{i=1}^{m} f(x_i) \bmod N$. KDC computes shares, $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, where $s_{i,j}(x) = f_{i,l}(i) f(x) \bmod N$, for $l = 1, 2, \ldots, m-1$, and $f_{i,1}(i) \cdot f_{i,2}(i) \cdot \ldots \cdot f_{i,m-1}(i) = f(i) \bmod N$, for each sensor. Shares are stored in sensor P_i secretly.

If m sensors, $\{P_{i_1}, P_{i_2}, \ldots, P_{i_m}\}$, want to establish a secret group key among them, each sensor P_i, where $i \in \{1, 2, \ldots, m\}$, uses its shares, $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, to compute $K = s_{i,1}(i_1) \cdot s_{i,2}(i_2) \cdot \ldots \cdot s_{i,m-1}(i_{m-1}) \bmod N$, where $i_1, i_2, \ldots i_{r-1} \in \{1, 2, \ldots, m\} - \{i\}$ *and* $i_{\mathrm{r}} \neq i_s, \forall r, s$. In fact, $K = f(i_1) \cdot f(i_2) \cdot \ldots \cdot f(i_m) \mathrm{mod} N$. We describe the scheme in Figure 2.

In [15], it has shown that the proposed scheme satisfies following properties.

(a) **Correctness**: The group key can be computed by each sensor in a group communication involving m (i.e., $2 \leq m \leq n$) sensors.

(b) ***k*-secure**: If k sensors are captured, there will have no information to be compromised.

Step 1. The KDC selects a random polynomial having degree k as $f(x) = a_k x^k + \ldots + a_1 x + a_0 \bmod N$, *where* a_i □ $(0,N)$. *The m-variate polynomial is* $F(x_1, x_2, \ldots, x_m) = \prod_{i=1}^{m} f(x_i) \bmod N$.

Step 2. For each sensor, P_i, *KDC first computes* $f(i) \bmod N$, *where i is a public information associated with the sensor,* P_i. *Then KDC randomly selects integers,* $\{f_{i,1}(i), f_{i,2}(i), \ldots, f_{i,m-2}(i)\}$, *where each integer is in* Z_N, *and solves* $f_{i,m-1}(i)$ *satisfying* $f_{i,1}(i) \cdot f_{i,2}(i) \cdot \ldots \cdot f_{i,m-1}(i) = f(i) \bmod N$. *KDC computes shares,* $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, *where* $s_{i,j}(x) = f_{i,l}(i) f(x) \bmod N$, *for* $l = 1,2,\ldots,m-1$, *of all sensors. Shares,* $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, *are stored by sensor,* P_i, *secretly.*

Group key establishment

Let us assume that m sensors, $\{P_1, P_2, \ldots, P_m\}$, *want to establish a secret group key among them, each sensor* P_i, *where* $i \in \{1,2,\ldots,m\}$, *uses its shares,* $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, *to compute* $K = s_{i,1}(i_1) \cdot s_{i,2}(i_2) \cdot \ldots \cdot s_{i,m-1}(i_{m-1}) \bmod N$, *where* $i_1, i_2, \ldots i_{r-1} \in \{1,2,\ldots,m\} - \{i\}$ *and* $i_r \neq i_s, \forall r, s$.

Fig. 2. Predistribution scheme for establishing group keys [15].

(c) ***Key confidentiality***: It is computationally infeasible for any attacker to discover any group key.

(d) ***Key independence***: Knowing a subset of group keys, $K' \subset K$, where k is the complete set of group keys, the attacker cannot discover any other group keys, $K'' = K - K'$.

III. Model of Secure End-to-End Routing Protocol

In this section, we describe the model of our proposed secure end-to-end routing protocol in WSNs including the protocol description, the attacks and security properties of our proposed protocol.

A. Protocol Description

In our proposed protocol, there is a key generation center (KGC) and there are n sensors, $\{U_1, U_2, \ldots, U_n\}$. There are four stages, key pre-distribution stage, routing path establishment stage, path key establishment stage, and data protection stage.

In the key pre-distribution stage, the KGC selects a special type of m-variate polynomial and generates keys. Secret keys of each sensor are $m - 1$ univariate polynomials. Secret keys are different in all sensors.

At the beginning in routing path establishment stage, a path from source sensor to destination sensor has to be identified. Let U_s and U_D be the source sensor and destination sensor, respectively. Let the intermediate sensors identified be $U_1, U_2, \ldots, U_{m-2}$. Collected data are transmitted from U_s to U_D through the intermediate sensors $U_1, U_2, \ldots, U_{m-2}$. In order to establish a secure group communication involving m (i.e., $2 \leq m \leq n$) sensors, it requires to authenticate all sensors in the path first. In the stage, sensors interact to prove that they are legitimate sensors. In the authentication, each sensor needs to broadcast its identity and a random integer. After receiving all identities and random integers, each sensor needs to use its secret keys obtained from the KGC initially to compute a key-hash output as its *authentication response*. Other sensors can use this authentication response to authenticate its legitimacy. Since each sensor is required to generate an authentication response and to be verified by other sensors, the complexity of this authentication is $O(m)$. This authentication can also identify illegitimate sensors. At the end of authentication, each sensor knows exactly the legitimacy of other sensors in the path of secure communication. In case any sensor being authenticated unsuccessfully, a new path need to be identified and repeat this stage at the beginning. This stage will be repeated until all sensors in a path have been authenticated successfully.

In the path key establishment stage, a secret path key is computed first by each sensor individually. There are two keys used to protect data. One is a pair of encryption and decryption keys used by the source and the destination sensors respectively. Another is a data authentication key used by all sensors in the path to provide authentication of the routed ciphertext. There needs no interaction with other sensors to compute these keys. Thus, our proposed protocol is very efficient in both authentication and key establishment since there is only broadcast transmission. Furthermore, the computations of each sensor needs are polynomial evaluation and key-hash function. We will give detail discussion of its performance evaluation in Section 5.

In the data protection stage, the collected data is encrypted and an authentication of the ciphertext is computed by the source sensor. Each intermediate sensor needs to authenticate the ciphertext in order to forward the ciphertext to its next sensor. Unauthenticated ciphertext will be removed from this routing process. At the destination sensor, the collected data can be recovered by deciphering the ciphertext.

B. Type of Attacks

We consider following types of attacks.

Attack by capturing sensors-After capturing sensors, attacker can recover secret keys stored in sensors and uses these recovered keys to derive the secret polynomial selected by the KGC.

Attack by wiretapping transmitted information-Attacker may try to recover the collected data by wiretapping transmitted message in WSNs.

Attack by injecting false data in the routing process-Attacker can inject false data to interfere routing transmission.

In the security analysis, we will show that none of these attacks can work properly against our protocol.

Step 1. The KGC selects a random polynomial having degree k as $f(x) = a_k x^k + \ldots + a_1 x + a_0 \bmod N$, where a_i □ $(0, N)$. The m-variate polynomial is $F(x_1, x_2, \ldots, x_m) = \prod_{i=1}^{m} f(x_i) \bmod N$.

Step 2. For each sensor, U_i, KGC first computes $f(i) \bmod N$, where i is a public information associated with the sensor, U_i. Then KGC randomly selects integers, $\{f_{i,1}(i), f_{i,2}(i), \ldots, f_{i,m-2}(i)\}$, where each integer is in Z_N, and solves $f_{i,m-1}(i)$ satisfying $f_{i,1}(i) \cdot f_{i,2}(i) \cdot \ldots \cdot f_{i,m-1}(i) = f(i) \bmod N$. KGC computes keys, $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, where $s_{i,j}(x) = f_{i,l}(i) f(x) \bmod N$, for $l = 1, 2, \ldots, m-1$, of all sensors. Keys, $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, are stored in sensor, U_i.

Fig. 3. Key predistribution stage.

C. Security Features of Proposed Scheme

Since our protocol is based on the group key establishment scheme proposed in [15] with properties, *k-secure, key confidentiality and key independence*, it is obvious that our protocol shares the same properties. In addition, our protocol has the following features.

(a) ***Correctness*****:** The protocol can successfully authenticate legitimacy of sensors in the path and then to establish a path key.

(b) ***Freshness of authentication response***: The authentication responses generated by sensors in the authentication stage can only be used for one time. This feature can prevent replay attack in which attackers replay recorded authentication response to fail the authentication.

(c) ***Freshness of communication keys***: The secret keys used by sensors to protect collected data can only be used for one time. This feature can prevent attackers either to reuse compromised communication keys to gain access to other communications or to replay recorded ciphertext and its authentication to interfere data transmission.

(d) ***Forward secrecy of group keys***: The forward secrecy is ensured if a captured sensor cannot access the content of communications of any future session.

(e) ***Backward secrecy of group keys***: The backward secrecy is ensured if a new sensor cannot access the content of communications of any past session.

IV. Proposed Protocol

In this paper, we propose our end-to-end routing protocol which provides authentication of sensors to establish a path and to establish a path key. The KGC selects a multivariate polynomial in Z_N, where N is a RSA modulus [19]. Our protocol is built upon the pre-distribution scheme of group key establishment [15].

We assume that a routing path has been identified by an algorithm initially which involves m sensors, $\{U_1, U_2, \ldots, U_m\}$, where U_1 is the source and U_m is the destination. The authentication allows each sensor to authenticate other sensors in the path.

Step 1. Each sensor, U_i, where $i \in \{1, 2, \ldots, m\}$, needs to broadcast its identity, i, and a random integer, r_i, to all other sensors.

Step 2. After receiving all identities and random integers, $\{(i, r_i), i = 1, 2, \ldots, m\}$, of sensors, each sensor, U_i, uses its secret keys, $\{s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,m-1}(x)\}$, to compute a group key, $K = s_{i,1}(i_1) \cdot s_{i,2}(i_2) \cdot \ldots \cdot s_{i,m-1}(i_{m-1}) \bmod N$, wher e $i_1, i_2, \ldots i_{m-1} \in \{1, 2, \ldots, m\} - \{i\}$ *and* $i_r \neq i_s, \forall r, s$.

Step 3. Each sensor, U_i, computes an authentication response, $AS_i = MAC(K, (i, r_i, (1, r_1), (2, r_2), \ldots, (m, r_m))$, where MAC is standard message authentication code which is a keyed hash function with K is the key and $i, r_i, (1, r_1), (2, r_2), \ldots, (m, r_m)$ as its input. The authentication response, AS_i, is broadcast to all other sensors.

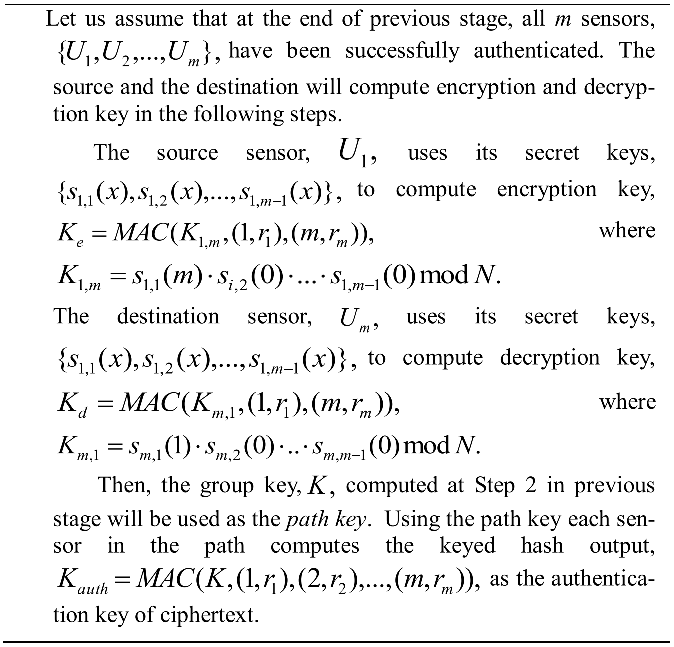
Step 4. After receiving each authentication response, AS_j, from sensor, U_j, each sensor, U_i, uses the group key, K, computed in Step 2 to verify the authentication response by checking $AS_j \overset{?}{=} MAC(K, (j, r_j, (1, r_1), (2, r_2), \ldots, (m, r_m))$. If the checking is passed successfully, the sensor, U_j, has been successfully authenticated; otherwise, the sensor, U_j, has been failed in authentication and a new path needs to be identified. This stage needs to be repeated when a new path is identified.

Repeat Step 4 for $m-1$ times to authenticate all other sensors in the path. This stage needs to be repeated until all sensors in a path have been authenticated successfully.

Fig. 4. Routing path establishment stage.

The KGC selects a RSA modulus N, where N is the product of two large safe primes, p and q, i.e., $p=2p'+1$ and $q=2q'+1$, where p' and q' are also primes. p and q are KGC's secrets, N is made publicly known. The protocol consists four different stages. In the key pre-distribution stage as shown in Figure 3, the KGC computes and loads keys to sensors. In the routing path establishment stage as shown in Figure 4, all sensors in the path have been authenticated successfully. In the path key establishment stage as shown in Figure 5, the encryption/ decryption keys and authentication key for the path are computed by sensors in the path. In the data protection stage as shown in Figure 6, the collected date is transmitted in the path in a secure and authenticated manner.

Remark 1: As discussed in [15], if there are less than m sensors in the routing path, for example r (i.e., $2 \leq r \leq m \leq n$*) sensors,* $\{U_1, U_2, \ldots, U_r\}$*. Then, the same steps*



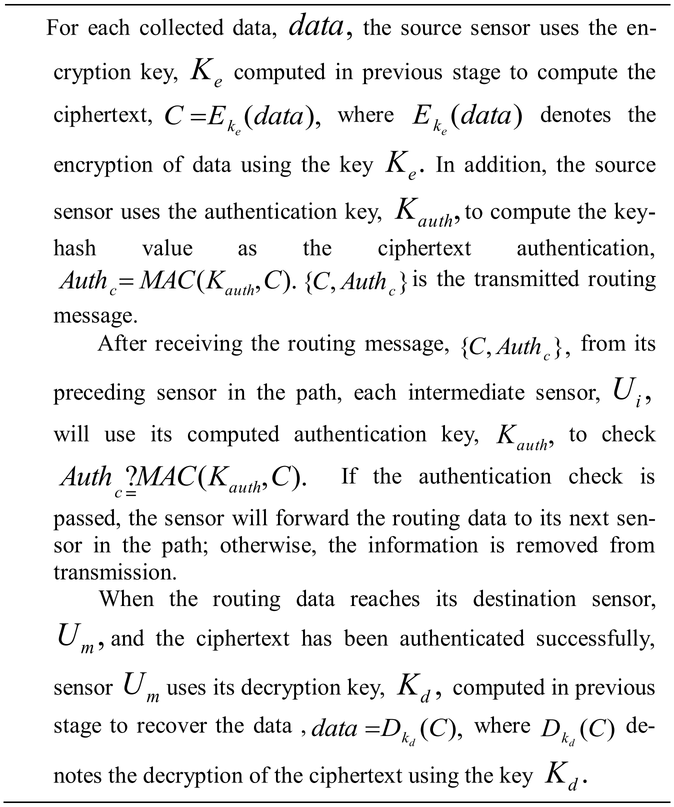
Let us assume that at the end of previous stage, all m sensors, $\{U_1,U_2,...,U_m\}$, have been successfully authenticated. The source and the destination will compute encryption and decryption key in the following steps.

The source sensor, U_1, uses its secret keys, $\{s_{1,1}(x),s_{1,2}(x),...,s_{1,m-1}(x)\}$, to compute encryption key, $K_e = MAC(K_{1,m},(1,r_1),(m,r_m))$, where $K_{1,m} = s_{1,1}(m)\cdot s_{i,2}(0)\cdot ...\cdot s_{1,m-1}(0) \bmod N$.

The destination sensor, U_m, uses its secret keys, $\{s_{1,1}(x),s_{1,2}(x),...,s_{1,m-1}(x)\}$, to compute decryption key, $K_d = MAC(K_{m,1},(1,r_1),(m,r_m))$, where $K_{m,1} = s_{m,1}(1)\cdot s_{m,2}(0)\cdot ..\cdot s_{m,m-1}(0) \bmod N$.

Then, the group key, K, computed at Step 2 in previous stage will be used as the *path key*. Using the path key each sensor in the path computes the keyed hash output, $K_{auth} = MAC(K,(1,r_1),(2,r_2),...,(m,r_m))$, as the authentication key of ciphertext.

Fig. 5. Path key establishment stage.



For each collected data, $data$, the source sensor uses the encryption key, K_e computed in previous stage to compute the ciphertext, $C = E_{k_e}(data)$, where $E_{k_e}(data)$ denotes the encryption of data using the key K_e. In addition, the source sensor uses the authentication key, K_{auth}, to compute the key-hash value as the ciphertext authentication, $Auth_c = MAC(K_{auth},C)$. $\{C, Auth_c\}$ is the transmitted routing message.

After receiving the routing message, $\{C, Auth_c\}$, from its preceding sensor in the path, each intermediate sensor, U_i, will use its computed authentication key, K_{auth}, to check $Auth_c \overset{?}{=} MAC(K_{auth},C)$. If the authentication check is passed, the sensor will forward the routing data to its next sensor in the path; otherwise, the information is removed from transmission.

When the routing data reaches its destination sensor, U_m, and the ciphertext has been authenticated successfully, sensor U_m uses its decryption key, K_d, computed in previous stage to recover the data , $data = D_{k_d}(C)$, where $D_{k_d}(C)$ denotes the decryption of the ciphertext using the key K_d.

Fig. 6. Data protection stage.

in Figure 3 are executed except each sensor, U_j, where $j \in \{1, 2, \ldots, r\}$, uses its keys, $\{s_{j,1}(x), s_{j,2}(x), \ldots, s_{i,m-1}(x)\}$, to compute a group key $K = s_{j,1}(i_l) \cdot s_{j,2}(i_2) \cdot \ldots \cdot s_{j,r-1}(i_r) \cdot s_{j,r}(0) \cdot s_{j,r+1}(0) \cdot \ldots \cdot s_{j,m-1}(0) \bmod N$, where $i_1, i_2, \ldots i_{r-1} \in \{1, 2, \ldots, r\} - \{j\}$ and $i_r \neq i_s, \forall r, s$, by all sensors.

Remark 2: Since the secret keys of each sensor are $m - 1$ univariate polynomials, the number of sensors in the routing path is limited to be no more than m sensors in Figure 3. There is a trade-off between the storage space of each sensor and the maximal number of sensors in a path. However, if there are more than m sensors in a path, the entire path can be divided into multiple sub-paths such that each sub-path has no more than msensors. In this way, secure communication can be established between sub-paths by a pairwise shared key in [15].

V. Analysis

A. Security Analysis

In the following discussion, we first analyze the attacks as described in Section 3.2.

Attack by capturing sensors-After capturing sensors, attacker can recover secret keys stored in sensors and then from these recovered keys tries to recover the secret polynomial selected by the KGC. Since the multivariate polynomial used to generate keys of sensors having degree k, it has been shown in [15] that attacker needs to capture at least $k + 1$ sensors in order to lunch this attack. The degree of the polynomial determines the minimal number of captured sensors needed to lunch this attack successfully.

Attack by wiretapping transmitted information-In our protocol, the routing data is encrypted by the secret key, K_e, which can only be computed by the source and the destination sensors. Attacker cannot recover the collected data by wiretapping transmitted ciphertext without knowing the key.

Attack by injecting false data in the routing process-In our protocol, the transmitted ciphertext is protected by the authentication key, K_{auth}, which can only be computed by sensors in the path. Attacker cannot inject false data to interfere routing transmission without knowing the authentication key. Unauthenticated ciphertext will be removed from the routing process.

In the following discussion, we discuss security features of our protocol as described in Section 3.3.

(a) ***Correctness:*** *Authentication of a Path*- If all sensors in the path are legitimate sensors as they claimed to be in Step 1, each sensor, U_i, in Step 2 should be able to compute the group key, $K = s_{i,1}(i_1) \cdot s_{i,2}(i_2) \cdot \ldots \cdot s_{i,m-1}(i_{m-1}) \bmod N == f(1) \cdot f(2) \cdot \ldots \cdot f(m) \bmod N$. Thus, in Step 3 the authentication response, $AS_i = h(K, (i, r_i, (1, r_1), (2, r_2), \ldots, (m, r_m))$, can be used to verify its legitimacy by other sensors in Step 4. Illegitimate sensor cannot forge this authentication response since illegitimate sensor does not have the keys of sensor, U_i.

Path key establishment and Data protection- The correctness of this property comes from the scheme [15].

(b) ***Freshness of authentication response:*** In Step 3 the authentication response, $AS_i = h(K, (i, r_i, (1, r_1), (2, r_2), \ldots, (m, r_m))$, is a key-hash output of all random integers selected by sensors initially. By recording a previously transmitted authentication response cannot impersonate a sensor since the verifier's random integer is different in every session.

(c) ***Freshness of communication keys:*** In the path key establishment stage, the authentication key, $K_c = h(K, (1, r_1), (2, r_2), \ldots, (m, r_m))$, is a key-hash output of random integers selected by sensors in the path initially. This authentication key is different in every session. Similarly, the encryption/decryption key, $K_e = h(K_{1,m}, (1, r_1), (m, r_m))$, is different in every session.

(d) ***Forward secrecy of group keys:*** If a sensor has been captured by the attacker, knowing the keys of the captured sensor cannot access the content of future communications since the group key, K, can only be computed by sensors involved in the path of a secure communication.

(e) ***Backward secrecy of group keys:*** If a new sensor added to the WSN, the new sensor cannot access the content of any past communications since the group key, K, can only be computed by sensors involved in the path of a secure communication.

B. Performance Evaluation

Our proposed protocol can provide both authentication and key establishment simultaneously. Furthermore, our authentication has complexity $O(n)$, where n is the number of sensors in a path, which is different from most existing authentication schemes which are one-to-one authentication with complexity $O(n^2)$.

Each sensor needs to store keys, $\{s_{j,1}(x), s_{j,2}(x), \ldots, s_{j,m-1}(x)\}$, which are $m - 1$ univariate polynomials. Thus, the memory storage of each sensor is $(m - 1)(k + 1)$ coefficients from Z_N.

In Step 2 routing path establishment stage, to compute the group key, $K = s_{i,1}(i_1) \cdot s_{i,2}(i_2) \cdot \ldots \cdot s_{i,m-1}(i_{m-1}) \bmod N$, needs to evaluate $m - 1$ polynomials. Horner's rule [18] can be used to evaluate polynomials. From Horner's rule, evaluating a polynomial of degree k needs k multiplications and $k + 1$ additions. The computational cost to establish a group key with size m consists of the cost of evaluating $m-1$ polynomials. Overall, the computational cost to compute the group key, K, each sensor needs to evaluate $(m - 1)k$ multiplications and $(m - 1)(k + 1)$ additions. In addition, each sensor needs to generate one authentication response and to verify $(m - 1)$ authentication responses. Since each authentication response is a key-hash output, each sensor needs to compute m key-hash outputs. Finally, there is one more key-hash output to compute the authentication key of ciphertext by each sensor.

There is one major advantage in using our proposed path key in comparing with using pairwise shared keys. An end-to-end communication is connected by a path which involves multiple links. Thus, data encryption and data decryption is needed in every link using pairwise shared key. However, in our proposed path key, data encryption is needed only at the source sensor and data encryption is only needed at the destination sensor. The ciphertext authentication is only computed at the source sensor and is verified by intermediate sensors. Our proposed protocol can significantly reduce data processing time of intermediate sensors. Table 1 shows the comparison of our proposed protocol with all existing end-to-end routing protocols based on random key distribution.

TABLE I
COMPARISON

	Our proposed protocol	**Protocols based on random key distribution**
Connectivity	Guaranteed	Probability
Type of data protection	End-to-end data protection based on a path key	Hop-to-hop data protection based on pairwise shared keys
Encryption/decryption	Source sensor encrypts data and destination sensor decrypts ciphertext	Every sensor except the destination in the path needs to encrypt and every sensor except the source in the path needs to decrypt data
Ciphertext authentication generation	Source sensor generates ciphertext authentication	Every sensor except the destination in the path needs to generate ciphertext authentication
Ciphertext verification	Every sensors except the source in the path needs to authenticate the ciphertext	Every sensors except the source in the path needs to authenticate the ciphertext

The communication of authentication is performed completely in the broadcast channel. Total communication time is to transmit m identities and random integers, $\{(1, r_1), i = 1, 2, \ldots, m\}$, and m authentication responses, $\{AS_i, i = 1, 2, \ldots, m\}$, of all sensors. There is no additional communication in order to establish the path and the path key.

VI. CONCLUSION

We have proposed a novel design of a secure end-to-end routing protocol which can provide authentication to establish a routing path and to establish a path key. All existing routing protocols use pairwise shared keys to protect routing data. Our proposed protocol uses path key to protect routing data by removing encryption and decryption in intermediate sensors. We have included the security analysis and performance evaluation in the paper.

REFERENCES

[1] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.

[2] I. Downnard, "Public-key cryptography extensions into kerberos," *IEEE Potentials*, vol. 21, no. 5, pp. 30–34, Dec. 2002.

[3] L. Harn and J. Ren, "Generalized digital certificate for user authentication and key establishment for secure communications," *IEEE Trans. Wireless Commun.*, vol. 10, no. 7, pp. 2372–2379, Jul. 2011.

[4] W.-C. Ku, "Weaknesses and drawbacks of a password authentication scheme using neural networks for multiserver architecture," *IEEE Trans. Neural Netw.*, vol. 16, no. 4, pp. 1002–1005, Jul. 2005.

[5] H.-A. Park, J. W. Hong, J. H. Park, J. Zhan, and D. H. Lee, "Combined authentication-based multilevel access control in mobile application for daily life service," *IEEE Trans. Mobile Comput.*, vol. 9, no. 6, pp. 824–837, Jun. 2010.

[6] K. Ren, S. Yu, W. Lou, and Y. Zhang, "Multi-user broadcast authentication in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 8, pp. 4554–4564, Oct. 2009.

[7] J. Yan, A. Blackwell, R. Anderson, and A. Grant, “Password memorability and security: Empirical results,” *IEEE Security Privacy*, vol. 2, no. 5, pp. 25–31, Sep./Oct. 2004.

[8] L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in *Proc. ACM Conf. CCS*, 2002, pp. 41–47.

[9] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in *Proc. IEEE Symp. SP*, 2003, pp. 197–213.

[10] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, “A key management scheme for wireless sensor networks using deployment knowledge,” in *Proc. IEEE INFOCOM*, Mar. 2004, pp. 586–597.

[11] S. Ruj, A. Nayak, and I. Stojmenovic, “Pairwise and triple key distribution in wireless sensor networks with applications,” *IEEE Trans. Comput.*, vol. 62, no. 11, pp. 2224–2237, Nov. 2013.

[12] R. Blom, “Non-public key distribution,” in *Advances in Cryptology—Proceedings of Crypto*, D. Chaum, R. L. Rivest, and A. T. Sherman, Eds. New York, NY, USA: Plenum, 1982, pp. 231–236.

[13] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, “Perfectly-secure key distribution for dynamic conferences,” in *Advances in Cryptology*, E. F. Brickell, Ed. Berlin, Germany: Springer-Verlag, 1992, pp. 471–486.

[14] E. Khan, E. Gabidulin, B. Honary, and H. Ahmed, “Matrix-based memory efficient symmetric key generation and pre-distribution scheme for wireless sensor networks,” *IET Wireless Sensor Syst.*, vol. 2, no. 2, pp. 108–114, 2012.

[15] L. Harn and C. F. Hsu, “Predistribution scheme for establishing group keys in wireless sensor networks,” *IEEE Sensors J.*, vol. 15, no. 9, pp. 5103–5108, Sep. 2015.

[16] S. M. G, R. J. D’Souza, and G. Varaprasad, “Digital signature-based secure node disjoint multipath routing protocol for wireless sensor networks,” *IEEE Sensors J.*, vol. 12, no. 10, pp. 2941–2949, Oct. 2012.

[17] R. Azarderakhsh, K. U. Jarvinen, and M. Mozaffari-Kermani, “Efficient algorithm and architecture for elliptic curve cryptography for extremely constrained secure applications,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 4, pp. 1144–1155, Apr. 2014.

[18] W. Gu, N. Dutta, S. Chellappan, and X. Bai, “Providing end-to-end secure communications in wireless sensor networks,” *IEEE Trans. Netw. Service Manage.*, vol. 8, no. 3, pp. 205–218, Sep. 2011.

[19] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[20] D. E. Knuth, *The Art of Computer Programming: Seminumerical Algorithms*, vol. 2. Reading, MA, USA: Addison-Wesley, 1981.

Lein Harn received the B.S. degree from National Taiwan University in 1977, the M.S. degree from the State University of New York–Stony Brook in 1980, and the Ph.D. degree from the University of Minnesota in 1984, all in electrical engineering. He joined the Department of Electrical and Computer Engineering, University of Missouri-Columbia, in 1984, as an Assistant Professor, and in 1986, he moved to the Computer Science and Telecommunication Program University of Missouri-Kansas City (UMKC). While at UMKC, he was on development leave to work with the Racal Data Group in Florida for a year. He has authored a number of papers on digital signature design and applications, wireless, and network security. He has written two books on security. His research interests are cryptography, network security, and wireless communication security. He is investigating new ways of using digital signature in various applications. In 2015, he was appointed as a Chu-Tian Researcher by the School of Computer Science and Technology, Hubei University of Technology, China.

Ching-Fang Hsu was born in Hubei, China, in 1978. She received the M.Eng. and Ph.D. degrees in information security from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2010, respectively. From 2010 to 2013, she was a Research Fellow with the Huazhong University of Science and Technology. She is currently an Assistant Professor with Central China Normal University, Wuhan. Her research interests are in cryptography and network security, especially in secret sharing and its applications.

Ou Ruan received the Ph.D. degree in information security from the Huazhong University of Science and Technology, Wuhan, China. He is currently teaching at the Hubei University of Technology, Wuhan. His research interests are in cryptography and network security, especially in secure multiparty computation.

Mao-Yuan Zhang received the Ph.D. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China. He is currently a Professor with Central China Normal University, Wuhan. His research interests are in computer networks and network security.