Symmetric-bivariate-polynomial-based Lightweight Authenticated Group Key Agreement for Industrial Internet of Things

Shan Wu[1,2], Chingfang Hsu[3], Zhe Xia[4], Jinlong Zhang[1], Di Wu[3]

[1] School of Management, Huazhong University of Science and Technology, China
[2] Research Center for mathematical modeling, Wuhan Technology and Business University, China
[3] Computer School, Central China Normal University, China
[4] Department of Computer Science, Wuhan University of Technology, China
hariny@163.com, cherryjingfang@gmail.com, xiazhe@yahoo.com, jlzhang@hust.edu.cn, wud@mails.ccnu.edu.cn

Abstract

Nowadays, mobile and embedded cyber-physical systems are ubiquitous and can be found in many industrial applications, ranging from industrial control systems, modern vehicles, to critical infrastructure. These smart mobile devices consistently generate, process and exchange a large amount of security-critical and privacy-sensitive data, which makes them as attractive targets of cyber attacks. The prevention of these cyber attacks against smart mobile devices in Industrial Internet of Things (IIoT) systems is very crucial, as they may cause physical damage or even threaten human lives. Moreover, group-oriented communications are playing an important role in IIoT and they have been widely used in various industrial areas for data gathering and area monitoring. However, due to the open nature of wireless channels and resource-constrained feature of sensor nodes, how to guarantee that the sensitive data collected by the sensors is only accessible by valid group members becomes a critical challenge in the IIoT environment. Recently, secure and efficient group communications for IIoT systems have attracted more and more attentions from both the academia and the industry. Membership authentication ensures that all users are legitimate group members, and group key agreement enables a group of users to negotiate a session key so that the group-oriented communications can be protected using cryptographic primitives thereafter. In this paper, we propose a novel solution for the above problem using a symmetric bivariate polynomial, in which membership authentication and group key establishment can be achieved simultaneously. Each member just needs to store a univariate polynomial, and they can generate pairwise keys without interaction. Then, each member mixes his/her input with the pairwise keys with other members and broadcasts the encrypted value. After collecting all these released values, each member can compute the group key. Our proposed scheme is more efficient in computations and communications, compared with the existing solutions in the literature. This design is suitable for efficient membership authenticated group key establishment in IIoT.

Keywords: Symmetric bivariate polynomial, Lightweight, Membership authentication, Group key agreement, Industrial Internet of Things

1 Introduction

The emerging of Internet of Things has inspired many attractive applications, such as health care, environment monitoring, smart cities [1-2], industrial control systems, modern vehicles, and etc. These applications are providing various services for our modern society. In the past decades, several related technologies, including classical production engineering, automation, and intelligent computation systems, have merged into the Industrial Internet of Things (IIoT). Moreover, the number of components that are integrated into industrial control systems, production systems, and factories is steadily increasing. Programmable logic controllers are replaced by more advanced cyber physical systems (CPS). Although CPS is typically communicating over closed industrial communication networks, it is also often connected to the Internet.

In the manufacturing industry, it is a difficult task for the factory management to keep on tracking of the deployed smart sensing devices that produce the real-time data of the environment. Moreover, these smart sensing devices are deployed in a hostile environment in the manufacturing industry, and hence, accessing such smart sensing devices by an unauthorized user/industrial professional is always viable. Under such scenario, it is crucial to design a real-time data transmission system that can efficiently take care of the activities performed by these devices to enrich the

*Corresponding Author: Chingfang Hsu; E-mail: cherryjingfang@gmail.com

DOI: 10.3966/160792642020122107011

manufacturing industry system's security. Using this, the real time data collected by the smart sensing devices can be monitored remotely and the accuracy of working of the machines can be evaluated by a remote user (e.g., a manager). However, the real time data is transmitted over the public channel, i.e., the Internet, and the data exchanged is sensitive as well as private in nature. Therefore, an illegal access by an adversary to the transmitted data should not be revealed any sensitive information related to the manufacturing industry [3-5]. It is necessary to design a lightweight non-interactive computational-efficient membership authentication and group key establishment protocol in the IIoT environment. IIoT networks have very restricted requirements in terms of computational power and execution time. Unfortunately, regular authenticated group messaging protocols consume too much resource for the typical IIoT device. Accordingly, we have to find a way to achieve the desired security properties with limited resources. For example, how to realize secure and efficient membership authentication and group key establishment for driverless vehicles in automotive industries? Obviously, in this specific application the group members usually are not too many. The object of our paper is to design new lightweight secure protocols for IIoT environment.

Group-oriented applications in IIoT motivate the needs for secure group communications over open and insecure networks. The negotiation of group session keys among different group members is a fundamental security service for group communication [6-10]. In secure communications, a session key needs to be distributed to all users beforehand. This session key is then used to encrypt all the exchanged messages. The objective of key distribution is to distribute this session key to the users in a secure and authenticated way. Among the security requirements, key confidentiality ensures that the session key is only known by legitimate users but not by any attacker, while key authentication ensures that the session key is sent by a legitimate authority but not by any impersonator.

Membership authentication and key agreement are two fundamental security services in secure communications for IIoT. Member authentication is the process of determining whether someone is who it claims to be, and key agreement is the process of distributing a secret session key to all users. The key can be used to protect both secrecy and integrity of exchange messages afterwards.

Many key distribution schemes have been proposed in the literature to distribute pairwise keys for conventional one-to-one communications. For example, the most well-known Diffie-Hellman public-key distribution scheme [11] enables two users to establish a pairwise secret by exchanging some public information. The first quantum key distribution scheme proposed by Bennett and Brassard in 1984 [12] (also called the BB84 scheme) relies its security on quantum physics. The BB84 scheme employs two pairs of states. Each pair conjugates to the other pair, and the two states within a pair are orthogonal to each other. But the BB84 scheme is only capable of establishing pairwise shared keys but not group keys. A bivariate polynomial has been used to distribute pairwise keys for wireless sensor networks (WSNs) [13-14]. A key generation center first selects a symmetric/asymmetric bivariate polynomial and then generates tokens for the sensor nodes. Each token can be used to establish a unique pairwise key with every other token without interaction. This type of key distribution is called *deterministic key distribution scheme* since it ensures that a pairwise key can be established between every pair of sensor nodes. In WSNs, random key distribution [15-16] is another common technique to generate tokens for sensor nodes. However, random key distribution is probabilistic and it cannot guarantee that a pairwise key exists between every pair of sensor nodes.

Many research papers [17-19] have tried to extend the original Diffie-Hellman scheme into one that can establish a group key among multiple users. In 2004, Joux [20] devised a simple three-party Diffie-Hellman group key exchange scheme using bilinear pairings. Since then, a number of works have devoted to group key distribution using bilinear pairings [21-22]. In quantum cryptography, similar ideas have been applied to extend BB84 scheme to a group key distribution for multiple users. In 2010, Chong *et al.* [23] proposed a quantum group key distribution scheme based on BB84 in which the key is formed by the agreement of all participants. Recently, Chou et al. [24] proposed a dynamic multi-party group key distribution scheme which is able to achieve arbitrary number of groups and users under the same resources. Many papers have also tried to extend polynomial-based approach to key distribution for more than two users. In 1992, Blundo et al. [25] proposed a non-interactive *k*-secure *m*-conference scheme based on a multivariate polynomial. Their scheme can establish a conference key among *m* participants. But the storage space of each user is exponentially proportional to the size of conference, which makes this scheme inefficient in real-world applications when the group is with large size. Laih et. al [26] proposed the first group key distribution scheme based on the secret sharing scheme. During the registration phase, each group member obtains a token from the group manager. The group manager can distribute a group key to all participated members through broadcasting transmission. There are many published papers based on this idea [27-28]. Also, there are several research papers on group key distribution using bivariate polynomials [29-30].

Recently, Cheng et.al [31] proposed a group key establishment protocol using a multivariate polynomial over an RSA modulus, where for a group of m members, the storage space of each member is (m-1) univariate polynomials' coefficients which is linearly proportional to the size of group communication and in order to compute the group key, each member needs to evaluate (m-1) univariate polynomials. Then, another group key agreement protocol based on an asymmetric bivariate polynomial was constructed [32], where each user still needs to store two univariate polynomials with t-1 degree in x and h-1 degree in y. After that, a new group-key distribution scheme based on pairwise keys has been proposed by Harn et al. [33]. This scheme can be applied on top of any pairwise-key distribution schemes. However, this scheme can only provide confidentiality of the group key; but not provide authentication of the group key. In other words, an attacker can impersonate to be an initiator to distribute a group key to other group members without being detected.

Because IIoT consists of a large number of devices, and these devices are heterogeneous with limited capabilities in terms of storage, computation, communication and energy, one of the main challenges faced by IIoT is how to secure the communications among these heterogeneous devices. The conventional protocols are not suitable for IIoT, since group key agreement in this environment requires lightweight communications and computations.

In this paper, we propose a new construction which not only achieves membership authentication and group key establishment simultaneously, but also enjoys advantages in computations and communications because of two reasons: (1) each member is only required to store a univariate polynomial; (2) the pairwise keys can be established without any interaction. During the registration phase, each member receives a "token" from the membership registration center (MRC), which are generated by a bivariate polynomial and each token is a univariate polynomial. These tokens can serve for three purposes: (a) membership authentication; (b) pairwise keys distribution and (c) group key establishment. As follows, each member mixes his/her input with the pairwise secrets shared with the other users, and then uses his/her pairwise shared keys to encrypt the computed value. Finally, the member sends this value to other members. After collecting all values from the other members, each member can compute the group key. Recall that most of the existing solutions to the same problem need additional membership authentication and shared keys distribution, and they require interactions among the members and complex computations for encryption and decryption. Our proposed scheme is especially suitable for IIoT for its simplicity and lightweight in communications and computations.

In summary, we list the contributions of this paper below.

- A lightweight membership authenticated group key establishment for IIoT is proposed. Our scheme is very efficient since there is no need for additional membership authentication, pairwise shared key distribution and each member just stores a univariate polynomial to generate pairwise shared keys.
- Tokens generated by a symmetric bivariate polynomial initially can be used for membership authentication and pairwise shared key establishment.
- Our protocol is secure against both inside and outside attackers. The desirable security properties, such as confidentiality, authentication, freshness, forward secrecy and backward secrecy of group key, can be achieved.
- One unique feature of our group key establishment is that only additional operation is required. Hence, the computational overheads are very low.

The organization of this paper is as follows. In Section 2, we provide some preliminaries about bivariate polynomials. In Section 3, we present the model of our protocols including the system model, the adversary model and security definitions. Our proposed protocol consists of three phases (a) token generation, (b) membership authentication and (c) group key establishment. The detailed description of our protocol is given in Section 4. In Section 5, we analyze its security and performance. The conclusion is given in Section 5.

2 Preliminaries

Shamir's (t, n) SS [34] is based on a univariate polynomial, $f(x)$, with $f(0) = s$, where s is the secret. The dealer selects this polynomial with degree $t-1$ and uses it to generate shares, $f(x_i) \bmod p, i =$ $1, 2, \ldots, n$, for shareholders, where p is a prime with $p > s$, and x_i is the public information associated with each shareholder,

There are many (t, n) verifiable secret sharing schemes [3, 35-40] using bivariate polynomials. A bivariate polynomial with degree $t-1$ can be represented as $F(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy + a_{2,0}x^2 + a_{0,2}y^2 + a_{1,2}xy^2 + a_{2,1}x^2y + a_{2,2}x^2y^2 + \cdots + a_{t-1}x^{t-1}y^{t-1}\ nod\ p$, where $a_{i,j} \in GF(p), \forall i, j \in [0, t-1]$. If the coefficients satisfy $a_{i,j} = a_{j,i}, \forall i, j \in [0, t-1]$, it is a symmetric polynomial.

The dealer can use a symmetric bivariate polynomial, $f(x, y)$, to generate shares, $f(x_i, y) \bmod p, i =$ $1, 2, ..., n$, for shareholders. Each share, $F(x_i, y)$, is a univariate polynomial with degree $t-1$. Note that since $F(x_i, x_j) = F(x_j, x_i), \forall i, j \in [0, t-1]$, a pairwise key, $F(x_i, x_j) = F(x_j, x_i)$, can be established between shareholders, U_i and U_j. Thus, using a symmetric bivariate polynomial can enable two users to establish a pairwise shared key.

3 Model of Our Proposed Protocol

In this section, we describe the model of our proposed new lightweight membership authenticated group key agreement protocol for industrial IoT including the network model description and security model, which gives the adversary and security properties of our proposed protocol.

3.1 Network Model Description

In the network model of our proposed protocol, there has a membership registration center (MRC) and n users, $\{U_1, U_2, ..., U_n\}$. Each user needs to register at the MRC initially and obtain secret token. The MRC selects a symmetric bivariate polynomial and generates tokens. Token of each user is a univariate polynomial. The detailed network model is described in Figure 1.

In order to establish a secure group communication involving m (i.e., $2 \le m < n$) members, it requires to execute a membership authentication first in which all participated users interact with each other to prove that they belong to the same group. In the membership authentication, each member needs to broadcast a random integer. After receiving all random integers, each member needs to use his secret tokens to compute pairwise shared keys and then compute a hash output as his *authentication response*. Members can use this authentication response to authenticate his membership. This membership authentication can also identify non-members. At the end of membership authentication, each member knows exactly the memberships of users participated in the secure group communication. Then, by using addition operation function, each member mixes his/her input with pairwise shared keys, and after that, uses his/her pairwise shared keys to encrypt the computed value, and next, sends this value to other members. After collecting all values from other members, each member can compute the group key, that is, a secret group session key is obtained by each member individually. There is no interaction with other members to compute the group key. Thus, our proposed protocol is very efficient in both membership authentication and group key establishment since there is only broadcast transmission. Furthermore, the computation of each member needs only polynomial evaluation, addition computation and hash function which are much faster than most public-key computations. We will give detail discussion for its performance evaluation in Section 5.

Figure 1. Network model

3.2 Security Model

In security model of our proposed protocol, there are some possible adversaries and security features a group key agreement protocol need to satisfy. We describe them as follows.

3.2.1 Type of Adversaries

We consider two types of attacks: inside and outside attacks as shown in Figure 2.

(1) The inside attackers are legitimate members who have obtained valid tokens from MRC initially. From inside attack, colluded members try to recover MRC's secret polynomial used to generate tokens for members and then use these uncovered tokens to obtain group keys which they are not authorized to access.

(2) The outside attackers are illegitimate members who try to generate valid tokens of members and use them to impersonate members in a secure group communication or to recover secret group keys which they are not authorized to access.

Figure 2. Type of adversaries

3.2.2 Security Features of Proposed Protocol

Our membership authenticated group key agreement protocol needs meet the correctness and security features as shown in Figure 3.

(a) ***Correctness***: The protocol can successfully authenticate memberships of all participated users and then establish a secret group key among all members, finally each member can successfully authenticate the group key he/she computed is equal to other members' group key.

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

(c) ***Freshness of group keys:*** The secret group key generated by members in the key establishment can only be used for one time communication. This feature can prevent attackers to reuse previously compromised group keys to gain access to other secure communications.

(d) ***Freshness of the group key authentication:*** The group key authentication messages generated by members in the group key establishment can only be used for one time. This feature can prevent replay attack in which attackers replay recorded group key authentication messages to fail the group key authentication.

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

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

Figure 3. Security features

4 Our Proposed Protocol

In this paper, we propose a membership authenticated group key agreement protocol using a symmetric bivariate polynomial and addition operation function. The protocol is illustrated in Figure 4.

Tokens generation

The MRC selects a $t-1$ degree symmetric polynomial over $GF(p)$, $F(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy + a_{2,0}x^2 + a_{0,2}y^2 + a_{1,2}xy^2 + a_{2,1}x^2y + a_{2,2}x^2y^2 + \cdots + a_{t-1,t-1}x^{t-1}y^{t-1}$ *nod* p, where p is a large prime and $a_{i,j} \in GF(p), \forall i, j \in [0, t-1]$ and the coefficients satisfy $a_{i,j} = a_{j,i}, \forall i, j \in [0, t-1]$. The MRC computes tokens, $s_i(y) = F(x_i, y) \bmod p$, for users, U_i, $i = 1, 2, ..., n$, where, $x_i \notin \{0, 1\}$ is the public information associated with each user, U_i. The MRC sends each token, $s_i(y)$, to user U_i secretly.

Membership authentication

We assume that m (i.e., $2 \le m < n$) users, for example $\{U_{v_1}, U_{v_2}, ..., U_{v_m}\}$, want to engage in a group key establishment in industrial IoT application.

Step 1. Each member U_{v_i} broadcasts a random integer, $\gamma_1 \in GF(p)$, to all other members.

Step 2. Each member U_{v_i} uses his token, $s_{v_i}(y)$, to compute pairwise shared keys, $k_{i,j} = s_{v_i}(x_{v_i}) = F(s_{v_i}, s_{v_j})$, $j = 1, 2, ..., m$, $j \ne i$, where $k_{i,j}$ is the secret key shared between shareholders, U_{v_i} and U_{vj}.

Step 3. Each member U_{v_i} computes authentication responses, $Auth_{i,j} = h(k_{i,j} \| r_j)$, $j = 1, 2, ..., m$, $j \ne i$, where $h(k_{i,j} \| r_j)$ is a one-way hash output with $k_{i,j}$ and r_j as inputs. Each $Auth_{i,j}$ is sent to member U_{vj} publicly for authentication.

Step 4. After receiving $Auth_{i,j} = h(k_{i,j} \| r_j)$, from member U_{v_i}, the member U_{vj} ses his computed pairwise shared key, $k_{j,i} = s_{v_j}(x_{v_i}) = F(x_{v_j}, x_{v_i})$, in Step 2 to check whether $Auth_{i,j} \overset{?}{=} h(k_{j,i} \| r_j)$. If the checking is successful, member U_{v_i} has been authenticated; otherwise, member U_{v_i} has not been authenticated. Repeat this process for all other members $U_{v_i}, i = 1, 2, ..., m, i \ne j$.

Group Key Agreement and Authentication

Let us assume that at the end of membership authentication, all m members, $\{U_1, U_2, ..., U_m\}$, have been successfully authenticated. Then, members follow an addition operation algorithm to complete the group key establishment process. However, all exchange information among members is encrypted under the pairwise shared keys, $k_{i,j} = 1, 2, ..., m, j \ne i$, in the Step 2of membership authentication.

Step 1. Each member U_{v_i} need to select a secret input $s_i \in GF(p)$. At the same time, he broadcasts $g^{s_i} \bmod p$, to all other members, where $i = 1, 2, ..., m$ and g is a given generator over $GF(p)$.

Step 2. Each member U_{v_i} uses his pairwise shared keys with other members to compute $q_{v_i} = s_i + \sum_{j=1, j\ne 1}^{m} (-1)^{\alpha} k_{i,j} \bmod p$,

where $\begin{cases} if\ i < j, then\ a = 0; \\ if\ i > j, then\ a = 1. \end{cases}$

Step 3. Each member U_{v_i} uses his computed pairwise shared keys, $k_{i,j}, j = 1, 2, ..., m, j \ne i$, in the Step 2 of membership authentication to encrypt q_{v_i} as $u_{i,j} = E_{k_{i,j}}(q_{v_i}), j = 1, 2, \ldots, m = j \ne i$. Member U_{v_i} sends each $u_{i,j}$ to member U_{v_j}.

Step 4. After receiving $u_{j,i}$, from other member, member U_{v_i} uses his computed pairwise shared key, $k_{i,j}$, in the Step 2 of membership authentication to decrypt as $(q_{v_i}) = E_{k_{i,j}}(u_{j,i})$. Repeat this process for all $u_{j,i}, j = 1, 2, ..., m, j \ne i$.

Step 5. After obtaining $q_{v_j}, j = 1, 2, ..., m, j \ne i$, from all other members, member U_{v_i} computes $\sum_{i=1}^{m} q_{v_i} \bmod p = \sum_{i=1}^{m} s_i \bmod p = K_i$, $i = 1, 2, ..., m$.

Step 6. Each member U_{v_i} checks if $g^{K_i} \bmod p = \prod_{i=1}^{m} g^{s_i} \bmod p$. If the checking is successful, the group key has been authenticated, $K_i = k$ is the secret group communication key; otherwise, the group key has not been authenticated. Repeat this process for all group members $U_{v_i}, i = 1, 2, ..., m$.

Figure 4. Membership authentication and group key agreement

5 Analysis

In this section, we give the security and performance analysis of our protocol respectively.

5.1 Security Analysis

Our protocol is secure against possible adversaries, they are inside attackers and outside attackers, which are described in Section 3. The confidentiality, authentication, freshness, forward secrecy and backward secrecy of group key also can be achieved.

5.1.1 Security features

First, we discuss correctness and security features of our protocol as described in Section 3, which are illustrated in Figure 5.

(a) Correctness: Membership authentication- If all participated users are members as they claimed in Step 1 of Membership authentication, each member, U_i, in Step 2 should be able to compute the pairwise shared key $k_{i,j}$. Thus, in Step 4 the authentication response, $Auth_{i,j}=h(k_{i,j}\|r_j)$,can be used to verify U_{v_i}'s membership by U_{v_j}. Non-members cannot forge this authentication response since non-members do not know the secret tokens of member, U_i.

Group key establishment- The correctness of this property comes from the rule of addition operation and $q_{v_i}=s_i+\sum_{j=1,j\neq 1}^{m}(-1)^{\alpha}k_{i,j} \bmod p$, where $\begin{cases} if\ i<j, then\ a=0; \\ if\ i>j, then\ a=1. \end{cases}$, thus $\sum_{i=1}^{m} q_{v_i} \bmod p=\sum_{i=1}^{m} s_i \bmod p=K, i=1,2,...,m.$

Group key authentication- If the values $K_1,...,K_m$ computed by all members are identical, the checking, $g^{K_i} \bmod p=\prod_{i=1}^{m} g^{s_i} \bmod p, i=1,2,...,m,$ is successful in Step 6, the group key has been authenticated.

(b) Freshness of authentication response: In Step 3 of Membership authentication, the authentication response, $Auth_{i,j}=h(k_{i,j}\|r_j)$, is a hash output of pairwise shared key and random integer selected by participated member initially. By recording a previously used authentication response cannot impersonate a member since this random integer is different in every session.

(c) Freshness of group keys: In the group key establishment, the group key, $K=\sum_{i=1}^{m} s_i \bmod p$ is the sum of s_i, is determined by U_{v_i}'s secret input s_i initially. This group key is different in every session.

(d) Freshness of the group key authentication: In Step 6 of Group Key Establishment, the group key authentication by checking if $g^{K_i} \bmod p=\prod_{i=1}^{m} g^{s_i} \bmod p$ is determined by each member's secret input s_i. By recording a previously used group key authentication cannot success since the group key authentication by checking if $g^{K_i} \bmod p=\prod_{i=1}^{m} g^{s_i} \bmod p$ is different in every session.

(e) Forward secrecy of group keys: If a member has departed from the group, the departed member cannot access the content of future communications since the any group key, *K* can only be computed by members involved in the secure communication.

(f) Backward secrecy of group keys: If a member joins the group, the new member cannot access the content of any past communications since the any group key, *K* can only be computed by members involved in the secure communication.

Figure 5. Analysis of correctness and security features

5.1.2 Threshold of the Secret

Before we discuss the possible attacks, we need to analyze the threshold of a symmetric bivariate polynomial.

There are two major differences between shares generated by a $t-1$ degree univariate polynomial and by a $t-1$ degree symmetric bivariate polynomial, (a) there are t different coefficients in a $t-1$ degree univariate polynomial but there are $\frac{t(t+1)}{2}$ different coefficients in a $t-1$ degree symmetric bivariate polynomial, and (b) shares by a $t-1$ degree univariate polynomial are integers in *GF(p)*; but shares by a $t-1$ degree symmetric bivariate polynomial is a univariate polynomial having $t-1$ degree. We give the definition of the threshold of a threshold SS.

Definition 1. Threshold of a threshold SS. *The threshold of a threshold SS specifies the minimal number of shares needed to reconstruct the secret.*

It is well-known that the threshold of shares generated by a $t-1$ degree univariate polynomial is t. The following theorem states the threshold of shares generated by a $t-1$ degree symmetric bivariate polynomial.

Theorem 1. The threshold of shares generated by a $t-1$ degree symmetric bivariate polynomial is t.

Proof. In a $t-1$ degree symmetric bivariate polynomial, there are $\frac{t(t+1)}{2}$ different coefficients. In addition, each share is a univariate polynomial having $t-1$ degree. In other words, it can establish t linearly independent equations in terms of coefficients of the bivariate polynomial from each share. At the same time, for h users there are C_2^h pairwise keys. Hence, having enough number of shares (suppose as, h), the total number of linearly independent equations, i.e., $h \cdot t - C_2^h$ needs to satisfy $h \cdot t - C_2^h \geq \left\lceil \frac{t(t+1)}{2} \right\rceil$ in order to recover the bivariate polynomial and then to reconstruct the secret and the tokens. It is easy to compute that for t users, there are C_2^t pairwise keys needs to satisfy $t^2 - C_2^t \geq \left\lceil \frac{t(t+1)}{2} \right\rceil$ in order to recover the bivariate polynomial. As a result, we have $t^2 - C_2^t = t^2 - \frac{t(t-1)}{2} \geq \left\lceil \frac{t(t+1)}{2} \right\rceil$.

This implies that $h = \lceil t \rceil$.

5.1.3 Possible Attacks

In this sub-section, we will discuss that our protocol is secure against inside attackers and outside attackers, which are illustrated in Figure 6.

(1) *Inside attackers*- Inside attackers are legitimate members who own valid tokens from the MRC during registration. From Theorem 1, we obtain that the threshold of shares generated by a $t-1$ degree symmetric bivariate polynomial is $\lceil t \rceil$ Thus, it needs at least t inside attackers to work together to reconstruct the tokens. the proposed protocol can resist up to $\lfloor t-1 \rfloor$ colluded members to recover the secret polynomial $F(x, y)$ of MRC. According the security level requirement, the proper value of t can be determined. For instance, when $n = \lfloor t-1 \rfloor$ all members collusions cannot recover the secret polynomial $F(x, y)$ of MRC. This security is information-theoretic secure.

(2) *Outside attackers*- Outside attackers are illegitimate users who do not own any valid tokens from MRC. The outside attackers may try to impersonate members in the group key establishment to obtain the group key. However, since in the group key establishment, all exchange information of legitimate members is encrypted using pairwise shared keys and outside attackers do not own any valid token to recover any pairwise shared key, so the outside attacker cannot obtain any secret information.

Figure 6. Analysis of possible attacks

5.2 Performance Evaluation

Most of existing schemes can either provides user authentication or group key establishment separately [41-44]. They need additional membership authentication and shared keys distribution, also need interactive communications or complex computations for encryption and decryption. For recent lightweight group key agreement, in scheme [31] the storage space of each member is (m-1) univariate polynomials' coefficients which is linearly proportional to the size of group communication and in order to compute the group key, each member needs to evaluate (m-1) univariate polynomials, and in scheme [32] each user still needs to store two univariate polynomials with t-1 degree in x and h-1 degree in y. We first discuss performance features of our protocol, which are illustrated in Figure 7.

In summary, our proposal is lightweight and storage/computation efficient. The Specific analysis of our protocol in storage, computation and communication cost is as follows.

5.2.1 Stroage Cost

Each member just needs to store a token, $s_i(y)$, which is just a univariate polynomial. Thus, the memory storage of each member is t coefficients from $GF(p)$. The storage requirement for each user is $t \log_2 p$ bits, where p is the modulus. This polynomial-based modulus is far less than public-key-based modulus.

From Figure 6, our protocol can resist up to $\lfloor t-1 \rfloor$ colluded members to recover the secret polynomial $F(x, y)$ of MRC. In the case $n = t-1$, all members collusions cannot recover the secret polynomial $F(x, y)$ of MRC. This security is information-theoretic secure. Here the token of each user is a $t-1$ degree polynomial, and thus each user stores $t = n+1$ elements in *GF(p)*. Compare with symmetric key method, which needs each user to store $n-1$ symmetric keys, we can see that our protocol has the same level of storage cost with the symmetric key approach, but can additionally provide membership authentication and efficient pairwise shared key distribution.

(a) Compare with most of the existing scheme, our protocol can provide both membership authentication and group key establishment simultaneously. By using a symmetric bivariate polynomial, membership authentication and pairwise shared keys distribution are realized at the same time. Each member just needs store a univariate polynomial, which can generate the pairwise shared keys. Then, just by the addition operation, each member mixes his/her input with pairwise shared keys with other members and releases the encrypted value in a broadcast channel. After collecting all released values, each member can compute the group key efficiently. Our proposal is non-interactive, computation-efficient and lightweight.

(b) According to the definition in most communications, "Interactive communications" means acting one upon or with the other. In our group key establishment process, each member computes his/her own values and releases the values to others without "waiting" for other members' inputs. In other words, each member doesn't need waiting time in computing and releasing values to other members. We call this property "non-interactive", which can speed up the communication process significantly. At the same time, there is only broadcast transmission. Thus, our proposed protocol is very fast.

(c) Symmetric key encryption a way that each pair of users shares a symmetric key, but this way only provides confidentiality. Further, key distribution and management is a bottleneck in symmetric key cryptography, which produce huge communication and storage cost. Hence, public key encryption appeared, which can provide confidentiality, authenticity and non-repudiation. However, this way needs high computation cost due to very large modulus and modular exponentiation operations. For instance, RSA modulus is at least 1024 bits. Observe that the latest group key establishment protocols [39,42,43,44] are all based on Bilinear map and complex computational assumptions, which need modular exponentiation, pairing and scalar multiplication operations. Compare with public key operations producing high computation cost. bivariate polynomial-based approach can provide not only authentication and information-theoretic security, but also with lower computation cost. At the same time, compare with symmetric key distribution producing huge communication cost. bivariate polynomial-based approach saves a lot of communication cost. It is really efficient while providing authentication. Furthermore, one unique feature of our group key establishment is that the addition operation is the main computation in group key establishment. It is simple and lightweight.

Figure 7. Performance features

5.2.2 Computation Cost

In Step 2 of membership authentication, to compute pairwise shared key, $k_{i,j} = s_{v_i}(x_{v_j}) = F(x_{v_i}, x_{v_j})$, $j = 1, 2, ..., m,\ \ j \neq i$ each member needs to evaluate a $t-1$ degree univariate polynomial. Horner's rule [45] can be used to evaluate polynomials. From Horner's rule, evaluating a polynomial of degree $t-1$ needs $t-1$ multiplications and t additions. In addition, each member needs to generate one authentication response and to verify $(m-1)$ authentication responses. Since each authentication response is a hash output, each member needs to compute m hash outputs. In Steps of group key establishment, there are all addition operations, symmetric encryption and decryption operations which is very efficient in comparing with all existing protocols. Finally, there is only one exponential operation to authenticate the group session key by each member. The computation load of our proposed protocol is much simpler than most public-key based schemes. For example, the RSA [46] public-key operation requires approximately $1.5\log_2 N$ modulo multiplications (i.e., in RSA, N is at least 1024 bits).

5.2.3 Communication Cost

The communication of membership authentication is performed completely in the broadcast channel. Total communication time is to transmit m random integers, $\{\gamma_1, i=1,2,...,m\}$, and $m(m-1)$ authentication responses for all participated group members. To establish the group key, total communication time is to transmit $m(m-1)$ encrypted messages and m hash outputs to authenticate the group session key for all participated group members. The transmission overhead of each group member is calculated as the bit length of the transmitted data in executing GKA algorithm. In our protocol, this cost is significantly reduced to avoid causing heavy communication cost since all transmitted data are computed on polynomial-based modulus. In addition, our protocols are non-interactive, all released values can be broadcasted simultaneously, it is very efficient.

6 Conclusion

We have proposed a novel construction of a lightweight membership authenticated group key establishment protocol for industrial IoT. Our protocol not only achieves membership authentication and group key establishment simultaneously, but also enjoys advantages in computations and communications because of two reasons: (1) each member is only required to store a univariate polynomial; (2) the pairwise keys can be established without any interaction. We have included the security analysis and performance evaluation in the paper. Our protocol is lightweight in terms of computation and communication, so it is absolutely attractive for secure group communications in industrial IoT.

Acknowledgments

Chingfang Hsu and Shan Wu contributed equally to this work. This work was partially supported by the National Natural Science Foundation of China (Grants No. 61772224), National Social Science Foundation (Grants No. 16ZDA013) and Natural Science Foundation of Hubei Province (Grant No. 2017CFB303).

References

[1] Q. Jiang, N. Zhang, J. Ni, J. Ma, X. Ma, K. R. Choo, Unified Biometric Privacy Preserving Three-factor Authentication and Key Agreement for Cloud-assisted Autonomous Vehicles, *IEEE Transactions on Vehicular Technology*, February 2020. DOI: 10.1109/TVT.2020.2971254.

[2] Q. Jiang, Z. Chen, J. Ma, X. Ma, J. Shen, D. Wu, Optimized Fuzzy Commitment based Key Agreement Protocol for Wireless Body Area Network, *IEEE Transactions on Emerging Topics in Computing*, October 2019. DOI: 10. 1109/TETC.2019.2949137.

[3] A. Sadeghi, C. Wachsmann, M. Waidner, Security and privacy challenges in industrial internet of things, *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, San Francisco, CA, USA, 2015, pp. 1-6. DOI: 10.1145/ 2744769.2747942.

[4] A. K. Das, M. Wazid, N. Kumar, A. V. Vasilakos, J. J. P. C. Rodrigues, Biometrics-based Privacy-preserving User Authentication Scheme for Cloud-based Industrial Internet of Things Deployment, *IEEE Internet of Things Journal*, Vol. 5, No. 6, pp. 4900-4913, December, 2018.

[5] Z. Yang, J. He, Y. Tian, J. Zhou, Faster Authenticated Key Agreement with Perfect Forward Secrecy for Industrial Internet-of-Things, *IEEE Transactions on Industrial Informatics*, Vol. 16, No. 10, pp. 6584-6596, October, 2020. DOI: 10.1109/TII.2019.2963328.

[6] L. Harn, C. F. Hsu, A Practical Hybrid Group Key Establishment for Secure Group Communications, *The Computer Journal*, Vol. 60, No. 11, pp. 1582-1589, November, 2017.

[7] L. Harn, C. F. Hsu, A Novel Design of Membership Authentication and Group Key Establishment Protocol, *Security and Communication Networks*, Vol. 2017, ID 8547876, pp. 1-7, August, 2017.

[8] C. F. Hsu, L. Harn, Y. Mu, M. Zhang, X. Zhu, Computation-Efficient Key Establishment in Wireless Group Communications, *Wireless Networks*, Vol. 23, No. 1, pp. 289-297, January, 2017.

[9] L. Harn, C. F. Hsu, B. Li, Centralized Group Key Establishment Protocol without a Mutually Trusted Third Party, *Mobile Networks and Applications*, Vol. 23, No. 5, pp. 1132-1140, October, 2018.

[10] H. Xiong, Y. Wu, Z. Lu, A Survey of Group Key Agreement Protocols with Constant Rounds, *ACM Computing Surveys (CSUR)*, No. 52, No. 3, pp. 1-32, July, 2019.

[11] W. Diffie, M. Hellman, New Directions in Cryptography, *IEEE Transactions on Information Theory*, Vol. 22, No. 6, pp. 644-654, November, 1976.

[12] C. H Bennett, G. Brassard, Quantum Cryptography: Public Key Distribution and Coin Tossing, *Theoretical Computer Science*, Vol. 560 (Part 1), pp. 7-11, December, 2014.

[13] L. Harn, C. F. Hsu, O. Ruan, M. Y. Zhang, Novel Design of Secure End-to-end Routing Protocol in Wireless Sensor Networks, *IEEE Sensors Journal*, Vol. 16, No. 6, pp. 1779-1785, March, 2016.

[14] L. Harn, C. F. Hsu, Z. Xia, J. Zhou, How to Share Secret Efficiently over Networks, *Security and Communication Networks*, Vol. 2017, Article ID 5437403, pp. 1-6, September, 2017.

[15] H. Chan, A. Perrig, D. Song, Random Key Predistribution Schemes for Sensor Networks, *IEEE Security and Privacy*, Berkeley, CA, USA, 2003. pp. 197-213.

[16] L. Li, G. Xu, L. Jiao, X. Li, H. Wang, J. Hu, H. Xian, W. Lian, H. Gao, A Secure Random Key Distribution Scheme against Node Replication Attacks in Industrial Wireless Sensor Systems, *IEEE Transactions on Industrial Informatics*, Vol. 16, No. 3, pp. 2091-2101, March, 2020.

[17] T. Brecher, E. Bresson, M. Manulis, Fully Robust Tree-Diffie-Hellman Group Key Exchange, *Cryptology and Network Security*, Kanazawa, Japan, 2009, pp. 478-497.

[18] L. Harn, C. Lin, Efficient Group Diffie-Hellman Key Agreement Protocols, *Computers and Electrical Engineering*, Vol. 40, No. 6, pp. 1972-1980, August, 2014.

[19] S. Jarecki, J. Kim, G. Tsudik, Flexible Robust Group Key Agreement, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 22, No. 5, pp. 879-886, May, 2011.

[20] A. Joux, A One Round Protocol for Tripartite Diffie-Hellman, *International Algorithmic Number Theory Symposium*, Leiden, The Netherlands, 2000, pp. 385-393.

[21] F. Li, D. Xia, W. Gao, X. A. Wang, J. Yan, An ID-based Dynamic Authenticated Group Key Agreement Scheme with Optimal Round Complexity from Pairings, *2016 10th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*, Fukuoka, Japan, 2016, pp. 468-472.

[22] E. J. Yoon, K. Y. Yoo, Cryptanalysis of Authenticated Multiple Keys Exchange Protocol Based on Bilinear Pairings, *2011 IEEE 3rd International Conference on Communication Software and Networks*, Xi'an, China, 2011, pp. 321-325.

[23] S.-K Chong, T. Hwang, Quantum Key Agreement Protocol Based on BB84, *Optics Communications*, Vol. 283, No. 6, pp. 1192-1195, March, 2010.

[24] Y. H. Chou, G. J. Zeng, Z. H. Chang, S. Y. Kuo, Dynamic Group Multi-party Quantum Key Agreement, *Scientific Reports*, Vol. 8, Article number: 4633, pp. 1-8, March, 2018.

[25] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung, Perfectly-secure Key Distribution for Dynamic Conferences, in: E. F. Brickell (Ed.), *Advances in Cryptology— Crypto '92*, *Lecture Notes in Computer Science, Vol. 740*, Springer-Verlag, 1993, pp. 471-486.

[26] C. S. Laih, J. Y. Lee, L. Harn, A New Threshold Scheme and its Application in Designing the Conference Key Distribution Cryptosystem, *Information Processing Letters*, Vol. 32, No. 3, pp. 95-99, August, 1989.

[27] L. Harn, C. Lin, Authenticated Group Key Transfer Protocol

Based on Secret Sharing, *IEEE Transactions on Computers*, Vol. 59, No. 6, pp. 842-846, June, 2010.

[28] R. Jiao, H. Ouyang, Y. Lin, Y. Luo, G. Li, Z. Jiang, Q. Zheng, A Computation-efficient Group Key Distribution Protocol Based on a New Secret sharing Scheme, *Information*, Vol. 10, No. 5, 175, May, 2019.

[29] C. F. Hsu, L. Harn, T. He, M. Zhang, Efficient Group Key Transfer Protocol for WSNs, *IEEE Sensors Journal*, Vol. 16, No. 11, pp. 4515-4520, June, 2016.

[30] L. Harn, G. Gong, Conference Key Establishment Protocol Using a Multivariate Polynomial and Its Applications, *Security and Communication Networks*, Vol. 8, No. 9, pp. 1794-1800, June, 2015.

[31] Q. Cheng, C. Hsu, Z. Xia, L. Harn, Fast Multivariate-Polynomial-Based Membership Authentication and Key Establishment for Secure Group Communications in WSN, *IEEE Access*, Vol. 8, pp. 71833-71839, April, 2020. Doi: 10.1109/ACCESS.2020.2987978.

[32] Q. Cheng, C. F. Hsu, L. Harn, Lightweight Noninteractive Membership Authentication and Group Key Establishment for WSNs, *Mathematical Problems in Engineering*, Vol. 2020, Article ID 1452546, pp. 1-9, May, 2020.

[33] L. Harn, C. F. Hsu, Z. Xia, Lightweight Group Key Distribution Schemes Based on Pre-shared Pairwise Keys, *IET Communications*, Vol. 14, No. 13, pp. 2162-2165, August, 2020.

[34] A. Shamir, How to Share a Secret, *Communications of the ACM*, Vol. 22, No. 11, pp. 612-613, November, 1979.

[35] B. Chor, S. Goldwasser, S. Micali, B. Awerbuch, Verifiable Secret Sharing and Achieving Simultaneity in the Presence of Faults, *26th IEEE Symposium on the Foundations of Computer Science (sfcs 1985)*, Portland, OR, USA, 1985, pp. 383-395.

[36] R. Cramer, I. Damgard, S. Dziembowski, M. Hirt, T. Rabin, Efficient Multiparty Computations Secure Against an Adaptive Adversary, *18th Annual IACR EUROCRYPT*, Prague, Czech Republic, 1999, pp. 311-326.

[37] Y. Cheng, Y. Agrawal, An Improved Key Distribution Mechanism for Large-scale Hierarchical Wireless Sensor Networks, *Journal of Ad Hoc Networks*, Vol. 5, No. 1, pp. 35-48, January, 2007.

[38] Y. Desmedt, Y. Frankel, Shared Generation of Authenticators and Signatures, in: J. Feigenbaum (Ed.), *Advances in Cryptology-CRYPTO '91. Lecture Notes in Computer Science, vol. 576*, Springer, 1992, pp. 457-569.

[39] J. Katz, C. Y. Koo, R. Kumaresan, Improving the round Complexity of VSS in Point-to-point Networks, in: L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, I. Walukiewicz (Eds.), *Automata, Languages and Programming. ICALP 2008, Lecture Notes in Computer Science, Vol. 5126*, Springer, 2008, pp. 499-510.

[40] R. Kumaresan, A. Patra, C. P. Rangan, The Round Complexity of Verifiable Secret Sharing: The Statistical Case, in: M. Abe (Ed.), *Advances in Cryptology - ASIACRYPT 2010, Lecture Notes in Computer Science, Vol. 6477*, Springer, 2010, pp. 431-447.

[41] H. Tan, I. Chung, A Secure and Efficient Group Key Management Protocol with Cooperative Sensor Association in WBANs, *Sensors*, Vol. 18, No. 11, 3930, November, 2018.

[42]J. Zheng, Y. Tan, Q. Zhang, X. Zhang, L. Zhu, Q. Zhang, Cross-cluster Asymmetric Group Key Agreement for Wireless Sensor Networks, *Science China Information Sciences*, Vol. 61, No. 4, 048103, April, 2018.

[43] Q. Zhang, Y. Gan, Q. Zhang, R. Wang, Y. Tan, A Dynamic and Cross-domain Authentication Asymmetric Group Key Agreement in Telemedicine Application, *IEEE Access*, Vol. 6, pp. 24064-24074, January, 2018.

[44] Q. Zhang, Y. Gan, L. Liu, X. Wang, X. Luo, Y. Li, An Authenticated Asymmetric Group Key Agreement Based on Attribute Encryption, *Journal of Network and Computer Applications*, Vol. 123, pp. 1-10, December, 2018.

[45] D. E. Knuth, *The Art of Computer Programming, Semi-numerical Algorithms*, Vol. 3, Pearson Addision Wesley Professional, 1998.

[46] R. L. Rivest, A. Shamir, L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, pp. 120-126, February, 1978.

Biographies

Shan Wu was born in Hubei, China, on Oct. 30, 1979. She is currently a Ph.D. candidate in Management Science and Engineering at Huazhong University of Science and Technology. She is a Professor at Wuhan Technology and Business University, Wuhan, China. She is currently studying security and efficiency issues in the context of the industrial Internet.

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

Zhe Xia received the M.Eng. and the Ph.D. degrees in information security from University of Surrey, UK, in 2005 and 2009 respectively. From 2009 to 2013, he was a Research Fellow at University of Surrey, UK. He is currently an Assistant Professor at Department of Computer Science, Wuhan University of Technology, Wuhan, China. His research interests are in cryptography and network security, especially in secret sharing and its applications.

Jinlong Zhang was born in Jiangxi, China, on Feb. 21, 1952. He received the Ph.D. degree in management Science and Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2003. He is currently a Professor at Huazhong University of Science and Technology, Wuhan, China. His main research interests are information management and e-commerce, management innovation and decision-making patterns.

Di Wu received the master's degrees in engineering from the Central China Normal University, Wuhan, China, in 2020. He is currently an IT system engineer at Bank of Communication, ShangHai, China. His research interests are in cloud computing and network security.