Hindawi
Security and Communication Networks
Volume 2017, Article ID 8547876, 7 pages
https://doi.org/10.1155/2017/8547876

Research Article

A Novel Design of Membership Authentication and Group Key Establishment Protocol

Lein Harn[1] and Ching-Fang Hsu[1,2]

[1]*Department of Computer Science Electrical Engineering, University of Missouri-Kansas City, Kansas City, MO 64110, USA*
[2]*Computer School, Central China Normal University, Wuhan 430079, China*

Correspondence should be addressed to Ching-Fang Hsu; cherryjingfang@gmail.com

Received 20 April 2017; Revised 11 July 2017; Accepted 27 July 2017; Published 30 August 2017

Academic Editor: Ángel Martín Del Rey

A new type of authentication, called *group authentication*, has been proposed recently which can authenticate all users belonging to the same group at once in a group communication. However, the group authentication can only detect the existence of nonmembers but cannot identify who are the nonmembers. Furthermore, in a group communication, it needs not only to authenticate memberships but also to establish a group key among all members. In this paper, we propose a novel design to provide both membership authentication and group key establishment. Our proposed membership authentication can not only detect nonmembers but also identify who are the nonmembers. We first propose a basic membership authentication and key establishment protocol which can only support one-time group communication. Then, we extend the basic protocol to support multiple group communications. Our design is unique since tokens of users issued by a group manager (GM) during registration are used for both membership authentication and group key establishment.

1. Introduction

User authentication and key establishment are two primary security functions in most secure communications. User authentication is the process of determining whether someone is, in fact, who it is declared to be. Key establishment is the process of distributing a secret communication key to all users. The key can be used to protect the secrecy or integrity of exchange messages in the communication.

The trend of communication research has been moved from peer-to-peer communication into group communication in which more than two users participated in the communication session. Although conventional peer-to-peer authentication [1, 2] can be used in group communication to authenticate participants in a straightforward manner the complexity of using this approach is $O(n^2)$, where n is the number of users involved in the group communication. In a recent paper [3], a new type of authentication, called *group authentication*, has been proposed which is specially designed for the group communications. The complexity of using a group authentication is $O(1)$ in which it authenticates participants all at once. However, the group authentication can only detect the existence of nonmembers but cannot identify who are the nonmembers. Furthermore, in a group communication, it needs not only to authenticate memberships but also to establish a group key among all members.

Centralized group key establishment protocols [4, 5] are the most widely used group key management protocols due to their efficiency. The centralized group key has a mutually trusted KGC to select a group key and then transport the group key to group members secretly. For example, the IEEE 802.11i standard [6] has an online server to select a group key and transport it to each group member. Laih et al. [7] proposed the first group key protocol using a (t, n) secret sharing scheme. Harn and Lin [8] proposed an authenticated group key transfer protocol based on a secret sharing scheme. The advantage of using a secret sharing scheme is its efficiency. However, the limitation of using a centralized group key establishment is due to its requirement of a trusted KGC. In some applications, such as in an ad hoc network, a trusted KGC may not be available.

The most commonly used public-key agreement protocol is the Diffie-Hellman (DH) key exchange protocol [9, 10]. Harn and Lin [11] proposed a group DH protocol using the secret sharing scheme. Recently, Wu et al. [12] proposed a new approach which is a hybrid of group key agreement and public-key broadcast encryption. Their scheme is built from public-key based bilinear groups. The main disadvantage of the group DH key exchange is due to its computational and communication complexity since the group key is determined by all group members so each member needs to compute DH keys and exchange information to other members in the process.

In this paper, we propose a novel design to provide both membership authentication and group key establishment. Our proposed membership authentication can not only detect nonmembers but also identify who are the nonmembers. In our protocols, members can accomplish membership authentication and key establishment by themselves without needing any other trusted KGC. We first propose a basic membership authentication and key establishment protocol which can only support one-time group communication. Then, we extend the basic protocol to support multiple group communications. Our design is unique since tokens of users issued by a group manager (GM) during registration are used for both membership authentication and group key establishment.

Here, we summarize contributions of our paper.

(i) We propose protocols to provide both membership authentication and group key establishment. Our protocols do not need a trusted KGC in real-time to provide authentication and key establishment.

(ii) The membership authentication can not only detect nonmembers but also identify who are nonmembers.

(iii) Tokens of members obtained during registration can not only be used for membership authentication but also be used to establish a pairwise shared key between any pair of members.

(iv) All exchange information between members can be encrypted using pairwise shared keys.

The rest of paper is organized as follows. In Section 2, we provide some preliminaries, including bivariate polynomials and membership authentication and objectives of our proposed protocols. The basic protocol of membership authentication and group key establishment for one-time group communication is proposed in Section 3. The extended protocol for multiple group communications is presented in Section 4. The conclusion is given in Section 5.

2. Preliminaries

2.1. Bivariate Polynomials. Shamir's (t, n) SS [13] 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 [14–16] using bivariate polynomials. A bivariate polynomial with degree $t-1$ can be represented as $F(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy + a_{2,0}x^2 + a_{0,2}y^2 + a_{1,2}xy^2 + a_{2,1}x^2y + a_{2,2}x^2y^2 + \cdots + a_{t-1,t-1}x^{t-1}y^{t-1} \bmod p$, where $a_{i,j} \in \mathrm{GF}(p)$, $\forall i, j \in [0, t-1]$. 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, $F(x_i, y) \bmod p$, $i = 1, 2, \ldots, n$, 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.

2.2. Membership Authentication and Key Establishment. In this section, we describe membership authentication proposed in this paper. Motivated by the group authentication [3] which authenticates users all at once with complexity $O(1)$, we extend its capability of group authentication such that our protocol can not only detect the existence of nonmembers but also identify nonmembers. In our protocols, the GM is in charge of registering all members initially. GM selects a secret and hides the secret in a polynomial. GM issues tokens which are coordinate points on the polynomial to members initially.

Later, in real-time operation, members can accomplish membership authentication and key establishment by themselves without the assistance of any trusted KGC. We need to point out that both GM and KGC must be trusted parties; but GM is needed only during initialization and KGC is needed during real-time implementation. Members present their tokens to be authenticated. Nonmembership detection process is first executed. If all released tokens are valid tokens, the secret can be recovered successfully and all users are members; otherwise the recovered secret is invalid so there exist nonmembers. Thus, the detectability of our protocol is guaranteed if there are a sufficient number of tokens available to recover the secret. In other words, the minimal number of tokens needed is determined by the degree of polynomial used to generate tokens initially.

After nonmember being detected, nonmembership identification process is executed. The protocol first needs to identify a set of tokens which can recover the valid secret. The token holders are all members. Then, the set of valid tokens can be used as a base to check each remaining token to determine its validity. In this approach, nonmembers can be identified one at a time gradually. Thus, the identifiability of our protocol is guaranteed if there exists at least a set of valid tokens which can be used to recover the real secret.

In the membership authentication, the GM is in charge of registering all members initially. GM knows all members; but each member does not need to know other members. This unique feature is especially suitable for some applications. For example, after an earthquake, the Department of Homeland Security may dispatch a responsive team which involves agents from different agencies, such as Department of Defense and Department of Health and Human Services, to form a mobile ad hoc network and uses the network to

exchange sensitive information. In such network, there is a GM to register members initially; but each member does not need to know other members. The GM issues tokens to members before deploying them to the disaster site. In forming such a secure ad hoc network, all members can follow the membership authentication protocol without the assistance of the GM. If all users are legitimate members, the outcome of the membership authentication can authenticate users all at once; otherwise, the membership authentication can further identify nonmembers. Finally, a group key is shared among all members.

During system setup, the GM follows a (t, n) SS to select a univariate polynomial, $f(x)$, with degree $t-1$ and $f(0) = s$, where s is the secret. The GM generates tokens, $f(x_i)$, $i = 1, 2, \ldots, n$, for members, where x_i is the public information associated with each member U_i. The GM sends each token s_i to each member $U_i \in U$ secretly. The GM makes $H(s)$ publicly known, where $H(s)$ is a one-way function of the secret. In a membership authentication which involves μ (i.e., $t \leq \mu \leq n$) users, for example, P_{v_i}, $i = 1, 2, \ldots, \mu$, each user uses his token to compute, c_{v_i}, as his released value. Each c_{v_i} will be encrypted using a pairwise shared key and send it to each other user separately. After decrypting and collecting all released values, c_{v_i}, $i = 1, 2, \ldots, \mu$, each member can compute $F(c_{v_1}, c_{v_2}, \ldots, c_{v_\mu})$, where F is a public function. There is a nonmembership detection algorithm, GA, which allows each user to determine whether all users are members based on their released values. That is,

$$\mathrm{GA}\left(H(s) \stackrel{?}{=} H\left(F\left(c_{v_1}, c_{v_2}, \ldots, c_{v_\mu}\right)\right)\right) = \begin{cases} 0 \longrightarrow \left\{\exists P_{v_i} \notin U\right\}; \\ 1 \longrightarrow \left\{\forall P_{v_i} \in U\right\}. \end{cases} \quad (1)$$

Furthermore, if there are nonmembers, a nonmembership identification algorithm can identify nonmembers.

In a secure group communication, it needs not only membership authentication but also a group key establishment to distribute a group key to all members. The group key is used to protect exchange messages. One unique feature of our proposed protocols is that tokens of members generated by GM initially can not only be used to authenticate membership but also be used to establish pairwise keys between any pair of members. Therefore, in our protocols, all exchange information between members is encrypted by pairwise shared keys and thus the recovered secret is not available to nonmembers. We propose using the recovered secret as the group key for secure communication. This proposed key establishment is accomplished efficiently.

2.3. Objectives of Our Protocols

2.3.1. Security Objective. In our protocols, we consider two types of adversaries: insider and outsider.

Inside Attacker. Inside attacker is a legitimate member who owns a token generated by GM. But inside attacker may try to recover other member's token. After obtaining other members' tokens, the inside attacker is able to recover the secret of GM and forge tokens for attackers. We will also consider attack imposed by colluded inside attackers.

Outside Attacker. Outside attacker is an attacker who does not own any token generated by GM and may try to impersonate a legitimate member or to recover the secret group key.

2.3.2. Performance Objective. The objectives of membership authentication are not only to detect the existence of nonmembers but also to identify nonmembers. The following two properties are associated with our proposed protocols.

Detectability. This property means the ability of membership authentication to detect the existence of nonmembers.

Identifiability. This property means the ability of membership authentication to identify who are nonmembers.

In Section 3.2, we will examine conditions which will limit these two properties.

3. Basic Protocol of Membership Authentication and Key Establishment

In our design, the GM uses a bivariate polynomial to generate tokens for members. The tokens can be used not only to establish pairwise keys between any pair of members but also to achieve membership authentication and group key establishment.

3.1. Algorithm

Basic Protocol of Membership Authentication and Key Establishment

Token Generation. The GM selects a $t-1$ degree symmetric polynomial:

$$F(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy + a_{2,0}x^2 + a_{0,2}y^2 + a_{1,2}xy^2 + a_{2,1}x^2y + a_{2,2}x^2y^2 + \cdots + a_{t-1,t-1,t-1}x^{t-1}y^{t-1} \bmod p, \quad (2)$$

where $F(0,0) = s$, $a_{i,j} \in \mathrm{GF}(p)$, $a_{i,j} = a_{j,i}$, $\forall i, j \in [0, t-1]$, s is the secret, and p is a prime with $p > s$. The GM computes tokens, $s_i(y) = F(x_i, y) \bmod p$, for group members, U_i, $i = 1, 2, \ldots, n$, where x_i is the public information associated with each group member, U_i. The GM sends each token, $s_i(y)$, to member U_i secretly. The GM makes $H(s)$ publicly known, where $H(s)$ is a one-way function of the secret.

Membership Authentication and Key Establishment. Assume that u (i.e., $t \leq u \leq n$) members, $\{U_{v_1}, U_{v_2}, \ldots, U_{v_u}\}$, want to establish a secure group communication.

Step 1. Each member U_{v_i} uses his/her token, $s_{v_i}(y)$, to compute $w_{v_i} = s_{v_i}(0) \prod_{l=1, l\neq i}^{u} (-x_{v_l}/(x_{v_i} - x_{v_l})) \bmod p$.

Step 2. Each member U_{v_i} uses his/her token, $s_{v_i}(y)$, to compute pairwise shared keys, $k_{i,j} = s_{v_i}(x_{v_j}) = F(x_{v_i}, x_{v_j})$, $j =$

$1, 2, \ldots, u,\ j \neq i$, where $k_{i,j}$ is the secret key shared between members, U_{v_i} and U_{v_j}.

Step 3. Each member U_{v_i} computes $c_{i,j} = E_{k_{i,j}}(w_{v_i}),\ j = 1, 2, \ldots, u,\ j \neq i$, where $E_{k_{i,j}}(w_{v_i})$ denotes the conventional encryption of w_{v_i} using the key $k_{i,j}$, each member U_{v_i} $c_{i,j},\ j = 1, 2, \ldots, u,\ j \neq i$, to other members.

Step 4. After receiving ciphertext, $c_{j,i},\ j = 1, 2, \ldots, u,\ j \neq i$, from other members, U_{v_i} computes $w_{v_j} = D_{k_{i,j}}(c_{j,i}),\ j = 1, 2, \ldots, u,\ j \neq i$, where $D_{k_{i,j}}(c_{j,i})$ denotes the decryption of $c_{j,i}$ using the key $k_{i,j}$.

Nonmembership Detection

Step 5. Each member U_{v_i} computes $\sum_{i=1}^{u} w_{v_i} \bmod p = s'$. If $H(s) = H(s')$, all members have been successfully authenticated and s is the group communication key; otherwise, there are nonmembers and continue on next step.

Nonmembership Identification

Step 6. Each member U_{v_i} uses $w_{v_j},\ j = 1, 2, \ldots, u,\ j \neq i$, obtained from Step 4 to compute $s_{v_j}(0) = w_{v_j}(\prod_{l=1, l\neq j}^{u}(-x_{v_l}/(x_{v_j} - x_{v_l})))^{-1} \bmod p,\ j = 1, 2, \ldots, u,\ j \neq i$.

Step 7. Each member U_{v_i} searches for a subset of t values from the set, $\{s_{v_j}(0),\ j = 1, 2, \ldots, u\}$, for example, the subset is $\{s_{v_j}(0),\ j = 1, 2, \ldots, t\}$, and uses them to compute $s' = \sum_{i=1}^{t} s_{v_i}(0)\prod_{l=1, l\neq i}^{t}(-x_{v_l}/(x_{v_i} - x_{v_l})) \bmod p$. If $H(s') = H(s)$, then tokens in this sunset are all valid and they are members; s is the group communication key. Then, this subset is used as a base to test each remaining token one at a time to check whether using this token and all tokens in the subset can still recover the same secret or not. If it is so, the token is valid and the token holder is a member; otherwise, it is invalid and the token holder is a nonmember.

3.2. Analysis

(i) Correctness

Nonmembership Detection. In Step 1, each member U_{v_i} uses his/her token to compute the partial information of the secret, w_{v_i}, and, in Step 2, to compute pairwise secret keys shared with other members. In Step 3, the partial information of the secret w_{v_i} is encrypted using these pairwise shared keys to other members and then, in Step 4, each member U_{v_i} recovers $w_{v_j}\ (j = 1, \ldots, u,\ j \neq i)$, from other members. Finally, in Step 5, since $w_{v_i} = s_{v_i}(0)\prod_{l=1, l\neq i}^{u}(-x_{v_l}/(x_{v_i} - x_{v_l})) \bmod p$ and $s_i(y) = F(x_i, y) \bmod p$, following Lagrange interpolation formula, we have $\sum_{i=1}^{u} w_{v_i} \bmod p = \sum_{i=1}^{u} F(x_{v_i}, 0)\prod_{l=1, l\neq i}^{u}(-x_{v_l}/(x_{v_i} - x_{v_l})) \bmod p = F(0, 0) = s$. It implies that any subset $A = \{U_{v_1}, U_{v_2}, \ldots, U_{v_u}\} \in \Gamma$ with t or more than t members can work together with others to compute $\sum_{i=1}^{u} w_{v_i} \bmod p = s'$. Hence, it holds that $H(s) = H(s')$. On the other hand, if there are nonmembers, then $H(s) \neq H(s')$.

Nonmembership Identification. Following Lagrange interpolation formula, in Step 7 of our proposed protocol, any t members with their valid tokens, for example, the subset of tokens is $\{s_{v_j}(y),\ j = 1, 2, \ldots, t\}$, can use their tokens to recover the secret. This set of valid tokens can be used to test the validity of each remaining token one at a time. The test procedure is just by including this token and all tokens in the set to check whether it can still recover the same secret or not. This process can be used to identify nonmembers.

(ii) Security

Theorem 1 (inside attack). *The proposed basic protocol can resist up to $\lfloor (t-1)/2 \rfloor$ colluded members to recover the secret polynomial $F(x, y)$ of GM.*

Proof. $F(x, y)$ is a symmetric polynomial with $a_{i,j} \in \mathrm{GF}(p)$, $a_{i,j} = a_{j,i},\ \forall i, j \in [0, t-1]$, selected by GM which contains $t(t+1)/2$ different coefficients. In the proposed basic protocol, each token, $s_i(y)$, is a univariate polynomial with degree $t-1$. In other words, each member can use his token to establish t linearly independent equations in terms of the coefficients of the polynomial $F(x, y)$. There are ht linearly independent equations with knowing h tokens. If GM wants to prevent up to h colluded group members from recovering the secret polynomial, $F(x, y)$, it needs $t(t+1)/2 > ht (\Rightarrow t + 1 > 2h)$. Thus, up to $\lfloor (t-1)/2 \rfloor$ colluded members cannot recover the secret polynomial, $F(x, y)$. □

Theorem 2 (outside attack). *The proposed basic protocol can resist any nonmember to obtain the secret.*

Proof. In our proposed protocol, the partial information of the secret is encrypted using pairwise keys shared with other group members. Since nonmember does not own any valid token generated by the GM, nonmembers neither can impersonate any group members nor can decrypt any ciphertext, then, to obtain the partial information of the secret. Thus, after all members are successfully authenticated, the recovered secret can be used as the secret group key since the recovered secret is not available to nonmembers. □

(iii) Performance

Detectability. The nonmembership detection is based on Lagrange interpolation formula. That is, with t or more than t coordinate points of a polynomial can uniquely determine this polynomial and the secret; however, if there is any invalid value in the set of coordinate points, it cannot determine the original polynomial and the secret. Thus, our nonmembership detection can detect the existence of nonmembers. The only condition which limits the detectability is that it requires to have at least t tokens presented in the process.

Identifiability. The nonmembership identification is based on the polynomial and the secret which was used to generate

tokens initially. According to Lagrange interpolation formula, any t valid tokens can recover this original polynomial. Thus, each member needs first to search for a set of t valid tokens which can be used to recover the real secret. The token holders in this set are members. Then, this set of tokens is used as a base to test each remaining token by checking whether with this token and all tokens in the base the same secret can still be recovered or not. If it is so, the token holder is a member; otherwise, the token holder is a nonmember. The only condition which limits the identifiability is that it requires having at least t valid tokens presented in the protocols.

Computational Complexity. In the basic protocol, each token, $s_i(y)$, is a univariate polynomial with degree $t-1$. Thus, each member needs to store t coefficients of a univariate polynomial. The memory storage of each shareholder is $t\log_2 p$ bits, where p is the modulus. In the protocol, there is no interaction among users. Each member U_{v_i} sends ciphertext $c_{i,j} = E_{k_{i,j}}(w_{v_i})$, $j = 1, 2, \ldots, u$, $j \neq i$, to other members. Horner's rule [17] can be used to evaluate polynomials. In the following discussion, we show the cost for computing $w_{v_i} = s_{v_i}(0)\prod_{l=1,l\neq i}^{u}(-x_{v_l}/(x_{v_i} - x_{v_l}))\bmod p$, in Step 1. From Horner's rule, evaluating a polynomial of degree $t-1$ needs $t-1$ multiplications and t additions. Since multiplication takes more time than addition, the performance is only addressed to the number of multiplications needed. The computational cost in Step 1 to compute w_{v_i} is to evaluate one polynomial. The computational cost in Step 2 to compute pairwise shared keys, $k_{i,j} = s_{v_i}(x_{v_j})$, $j = 1, 2, \ldots, u$, $j \neq i$, is to evaluate $u-1$ polynomials, where u is the number of members participating in the secret reconstruction. Overall, the computational cost to reconstruct the secret of each member is to compute ut multiplications.

In our proposed protocol, the main computation is the polynomial evaluation. The modulus in our polynomial computation is much smaller than the modulus (e.g., 1,024 bits) used in most public-key cryptosystems. In addition, not like most conventional user authentication protocol which authenticates one user each time, the proposed protocol authenticates all users at once. After all users are successfully authenticated, there is no computation needed to establish a group key. Thus, the proposed protocol is very efficient in comparing with most communication protocols.

However, if there exist nonmembers, the nonmembership identification is invoked. Since each member needs to search for a subset of t valid tokens from a set containing u users participating in a secure group communication, the complexity of this searching is $O(u!)$, where u is the number of participants in a group communication. We would like to point out that in some practical applications u can be a small integer. Once this subset of valid tokens is determined, Lagrange interpolation formula is executed to test each remaining token one at a time to identify whether it is an invalid token or not.

After user authentication and key establishment, all participating members can recover the secret and the tokens, $s_i(0)$, $i = 1, 2, \ldots, n$, of other members. In other words, the tokens cannot be reused for multiple times since members can impersonate other members participating in different secret group communications. In the next section, we extend the basic protocol to support multiple group communications.

4. Extended Protocol for Multiple Group Communications

In this section, an extended protocol in which tokens obtained from the GM initially can be reused for multiple group communications is presented. The basic idea is that the GM needs to select two large public primes, p and q, such that p divides $q-1$, GF(p) is a unique subgroup of GF(q) with order p, and every g_i is a generator of GF(p). GM follows the same *token generation* procedure as described in Section 3 to select a symmetric polynomial, $F(x, y)$, and generate tokens, $s_i(y) = F(x_i, y)\bmod p$, for group members, U_i, $i = 1, 2, \ldots, n$. In addition, GM computes, $s_i = g_i^s$, $i = 1, 2, \ldots, m$, and makes $\{g_i, H(s_i) \mid i = 1, 2, \ldots, m\}$ publicly known, where m is the number of secure group communications that the protocol can support.

4.1. Algorithm

Extended Protocol for Multiple Group Communications

Group Authentication and Key Establishment. Assume that, at jth round, u (i.e., $t \leq u \leq n$) members, $\{U_{v_1}, U_{v_2}, \ldots, U_{v_u}\}$, want to establish a secure group communication.

Step 1. Each member U_{v_i} uses his/her token, $s_{v_i}(y)$, to compute $w_{v_i} = s_{v_i}(0)\prod_{l=1,l\neq i}^{u}(-x_{v_l}/(x_{v_i} - x_{v_l}))\bmod p$, and $d_{v_i} = g_j^{w_{v_i}} \bmod q$.

Step 2. Each member U_{v_i} uses his/her token, $s_{v_i}(y)$, to compute pairwise shared keys, $k_{i,r} = s_{v_i}(x_{v_r}) = F(x_{v_i}, x_{v_r})$, $r = 1, 2, \ldots, u$, $r \neq i$, where $k_{i,r}$ is the secret key shared between members, U_{v_i} and U_{v_r}.

Step 3. Each member U_{v_i} computes $c_{i,r} = E_{k_{i,r}}(d_{v_i})$, $r = 1, 2, \ldots, u$, $r \neq i$, where $E_{k_{i,r}}(d_{v_i})$ denotes the conventional encryption of d_{v_i} using the key $k_{i,r}$. Each member U_{v_i} sends $c_{i,r}$, $r = 1, 2, \ldots, u$, $r \neq i$, to other members.

Step 4. After receiving ciphertext, $c_{r,i}$, $r = 1, 2, \ldots, u$, $r \neq i$, from other members, U_{v_i} computes $d_{v_r} = D_{k_{i,r}}(c_{r,i})$, $r = 1, 2, \ldots, u$, $r \neq i$, where $D_{k_{i,r}}(c_{r,i})$ denotes the decryption of $c_{r,i}$ using the key $k_{i,r}$.

Nonmembership Detection

Step 5. Each member U_{v_i} computes $\prod_{i=1}^{u} d_{v_i} \bmod q = s_j'$. If $H(s_j) = H(s_j')$, all members have been successfully authenticated and s_j is the group communication key; otherwise, there are nonmembers and continue on next step.

Nonmembership Identification

Step 6. Each member U_{v_i} uses d_{v_r}, $r = 1, 2, \ldots, u$, $r \neq i$, obtained from Step 4 to compute $c_{v_r} = d_{v_r}^{(\prod_{l=1,l\neq r}^{u}(-x_{v_l}/(x_{v_r}-x_{v_l})))^{-1} \bmod p} \bmod q$, $r = 1, 2, \ldots, u$, $r \neq i$.

Step 7. Each member U_{v_i} searches for a subset of t values from the set, $\{c_{v_r}, r = 1, 2, \ldots, u\}$, for example, the subset is $\{c_{v_r}, r = 1, 2, \ldots, t\}$, and uses them to compute $\prod_{i=1}^{t} c_{v_r}^{\prod_{l=1,l\neq i}^{t}(-x_{v_l}/(x_{v_i}-x_{v_l}))\bmod p} \bmod q = s_j'$. If $H(s_j) = H(s_j')$, then tokens in this sunset are all valid and they are members and s_j is the group communication key. Then, this subset is used as a base to test each remaining token one at a time to check whether using this token and all tokens in the subset can still recover the same secret or not. If it is so, the token is valid and the token holder is a member; otherwise, it is invalid and the token holder is a nonmember.

4.2. Analysis

(i) Correctness

Nonmembership Detection. In Step 5, since $w_{v_i} = s_{v_i}(0)\prod_{l=1,l\neq i}^{u}(-x_{v_l}/(x_{v_i} - x_{v_l})) \bmod p$, $d_{v_i} = g_j^{w_{v_i}} \bmod q$ and $s_i(y) = F(x_i, y) \bmod p$, following Lagrange interpolation formula, we have $\prod_{i=1}^{u} d_{v_i} \bmod q = \prod_{i=1}^{u} g_j^{w_{v_i}} \bmod q = g_j^{\sum_{i=1}^{u} w_{v_i} \bmod p} \bmod q = g_j^{\sum_{i=1}^{u} F(x_{v_i},0)\prod_{l=1,l\neq i}^{u}(-x_{v_l}/(x_{v_i}-x_{v_l}))\bmod p} \bmod q = g_j^{F(0,0)} \bmod q = g_j^{s} \bmod q = s_j$. It implies that any subset $A = \{U_{v_1}, U_{v_2}, \ldots, U_{v_u}\} \in \Gamma$ of group members can work together to compute $\prod_{i=1}^{u} d_{v_i} \bmod q = s_j'$. Hence, it holds that $H(s_j) = H(s_j')$. Otherwise, if there are nonmembers, then $H(s_j) \neq H(s_j')$.

Nonmembership Identification. In Step 6, we get $c_{v_r} = d_{v_r}^{(\prod_{l=1,l\neq r}^{u}(-x_{v_l}/(x_{v_r}-x_{v_l})))^{-1} \bmod p} \bmod q = g_j^{w_{v_r}(\prod_{l=1,l\neq r}^{u}(-x_{v_l}/(x_{v_r}-x_{v_l})))^{-1} \bmod p} \bmod q = g_j^{s_{v_r}(0) \bmod p} \bmod q$. Thus, in Step 7, if values in the subset $\{c_{v_r}, r = 1, 2, \ldots, t\}$ are all valid, we should have $\prod_{i=1}^{t} c_{v_r}^{\prod_{l=1,l\neq i}^{t}(-x_{v_l}/(x_{v_i}-x_{v_l}))\bmod p} \bmod q = s_j$.

(ii) Security. In this extended protocol, each member's private value of token, $s_{v_i}(0)$, is protected in the value $d_{v_i} = g_j^{w_{v_i}} \bmod q$, under the discrete logarithm assumption. Similarly, the secret, s, is protected in the public value, $s_i = g_i^{s}$, $i = 1, 2, \ldots, m$, under the discrete logarithm assumption.

(iii) Performance. The modular exponentiation takes more computational time than multiplication and addition. So, we only consider the modular exponentiation in the following discussion. In this extended protocol, each member needs to compute only one modular exponentiation if all users are members. However, if there are nonmembers, more modular exponentiations are needed to identify nonmembers.

Remark 3. In comparison between algorithms presented in Sections 3 and 4, tokens generated during initiation can only be used for one group communication in the basic algorithm but tokens can be used for multiple group communications in the extended algorithm. Furthermore, only polynomial evaluations are needed in the basic algorithm but modular exponentiations are needed in the extended algorithm. According to Horner's rule [17], each polynomial evaluation needs t modular multiplications. But, each modular exponentiation with two large moduli, p and q (say p is 160 bits and q is 1024 bits), needs $1.5\log_2 p$ modular multiplications. Since t is much smaller than p, computational speed in the basic algorithm is much faster than computational speed in the extended algorithm.

5. Conclusion

We propose two efficient protocols of membership authentication and key establishment. The basic protocol can support a one-time communication in which each member needs only to perform polynomial evaluation. The extended protocol can support multiple communications in which each member needs to perform modular exponentiations. Both protocols are noninteractive.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Authors' Contributions

Lein Harn and Ching-Fang Hsu contributed equally to this work.

References

[1] T. W. Chim, S.-M. Yiu, V. O. K. Li, L. C. K. Hui, and J. Zhong, "PRGA: Privacy-preserving recording & gateway-assisted authentication of power usage information for smart grid," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 85–97, 2015.

[2] Z. Sitova, J. Sedenka, Q. Yang et al., "HMOG: New Behavioral Biometric Features for Continuous Authentication of Smartphone Users," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 877–892, 2016.

[3] L. Harn, "Group authentication," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 62, no. 9, pp. 1893–1898, 2013.

[4] J. Li, M. Wen, and T. Zhang, "Group-based authentication and key agreement with dynamic policy updating for MTC in LTE-A networks," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 408–417, 2016.

[5] J. L. T. Woo and M. V. Tripunitara, "Composing kerberos and multimedia internet keying (MIKEY) for authenticated-transport of group keys," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 4, pp. 898–907, 2014.

[6] IEEE CS, "802.1X, IEEE standard for local and metropolitan area networks, port-based network access control," The Inst. of Electrical and Electronics Engineers, Inc., 2004.

[7] C. S. Laih, J. y. Lee, and 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, 1989.

[8] L. Harn and C. Lin, "Authenticated group key transfer protocol based on secret sharing," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 59, no. 6, pp. 842–846, 2010.

[9] W. Diffie, W. Diffie, and M. E. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[10] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," *Journal of Cryptology. The Journal of the International Association for Cryptologic Research*, vol. 20, no. 1, pp. 85–113, 2007.

[11] L. Harn and C. Lin, "Efficient group Diffie-Hellman key agreement protocols," *Computers and Electrical Engineering*, vol. 40, no. 6, pp. 1972–1980, 2014.

[12] Q. Wu, B. Qin, L. Zhang, J. Domingo-Ferrer, and J. A. Manjón, "Fast transmission to remote cooperative groups: A new key management paradigm," *IEEE/ACM Transactions on Networking*, vol. 21, no. 2, pp. 621–633, 2013.

[13] A. Shamir, "How to share a secret," *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612-613, 1979.

[14] R. Kumaresan, A. Patra, and C. P. Rangan, "The round complexity of verifiable secret sharing: The statistical case," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6477, pp. 431–447, 2010.

[15] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 52–61, October 2003.

[16] D. Liu, P. Ning, and L. I. Rongfang, "Establishing pairwise keys in distributed sensor networks," *ACM Transactions on Information and System Security*, vol. 8, no. 1, pp. 41–77, 2005.

[17] D. E. Knuth, *The art of computer programming. Vol. 2: Seminumerical algorithms*, Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont, 1969.