doi:10.3772/j.issn.1006-6748.2019.02.002

A novel approach to authenticated group key transfer protocol based on AG codes①

Yuan Dezhai(袁德砦)②[*,**], Peng Xingyi[*,**], Liu Ting[*,**], Cui Zhe[*]
([*] Chengdu Institute of Computer Applications, Chinese Academy of Sciences, Chengdu 610041, P. R. China)
([**] School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, P. R. China)

Abstract

Group key management technique is a fundamental building block for secure and reliable group communication systems. In order to successfully achieve this goal, group session key needs to be generated and distributed to all group members in a secure and authenticated manner. The most commonly used method is based on Lagrange interpolating polynomial over the prime field $\mathbb{F}_p = \{0,1,2,\cdots,p-1\}$. A novel approach to group key transfer protocol based on a category of algebraic-geometry code is presented over the infinite field $GF(2^m)$. The attractive advantages are obvious. Especially, the non-repeatability, confidentiality, and authentication of group key transfer protocols are obtained easily. Besides, a more generalized and simple mathematical construction model is proposed which also can be applied perfectly to related fields of information security.

Key words: group key transfer protocol, erasure code, AG codes, non-repeatability, confidentiality, authentication

0 Introduction

With the development of group-oriented and collaborative applications, group communication is increasingly becoming an attractive research subject in network and communication fields. A critical challenge in designing group communication systems is how to provide confidentiality and authentication. Therefore, group session key needs to be employed and shared with communication parties for meeting the fundamental requirements. The most widely used protocol is Diffie-Hellman (DH) key agreement protocol[1]. It is merely applied to two communication entities, however, a group key transfer protocol is needed when a communication involves more than two entities. Under this background, several approaches have been proposed which can be divided into three main categories.

(1) Centralized group key management protocols. A controlling entity is responsible for managing the whole group members independently including the group key management, without any auxiliary entity.

(2) Decentralized group key management protocols. A large group is usually split into some small subgroups, which can be regarded as a hybrid group key management protocol.

(3) Distributed group key management protocols. There is no specified group key management center in distributed group key management protocols, each member of the group is a peer-to-peer entity. All of them are engaged in the management of the group key.

The most commonly used protocol is the centralized group key management protocol. For example, Ref. [2] proposed an authenticated group key transfer protocol using secret sharing scheme. However, this scheme is based on the Lagrange interpolating polynomial, which needs high computation overhead with communication entities size increase.

Generally, erasure codes have been widely used for error detection and correction in communication and storage fields, which includes Reed-Solomon (RS) codes[3], BCH codes[4,5], low-density parity-check codes[6], etc. Algebraic-geometry (AG) codes[7] can be regarded as a class of more generalized RS codes. Consequently, a new family of group key transfer protocol based on AG codes has been put forward. Our analysis shows that the proposed protocol has attractive advantages over many previously existing group key transfer protocols for secure group communication systems:

(1) Non-repeatability of group key generation is achieved, which basically means that forward and

① Supported by the National Natural Science Foundation of China (No. 61501064) and Sichuan Technology Support Program (No. 2015GZ0088).
② To whom correspondence should be addressed. E-mail: yuandezhai12@mails.ucas.ac.cn
Received on June 22, 2018

backward secrecy is provided against information leakage of group communication systems.

(2) The confidentiality of group key distribution is information-theoretically secure, without any dependence on computational complexity of mathematical assumptions.

(3) The authentication of group key reconstruction is also obtained by computing and broadcasting a single authentication message to all group members.

(4) A more generalized and simple mathematical construction model is proposed in contrast to Lagrange interpolating polynomial.

This paper is organized as follows. In Section 1, some relevant terminologies, group communication model, and security goals will be firstly introduced. Related work is introduced in Section 2. In Section 3, the detailed preliminaries used in the construction of authenticated group key transfer protocol will be proposed. Section 4 describes the proposed scheme via AG codes. Section 5 provides the performance and security analysis of the proposed scheme. Finally, conclusions are given in Section 6.

1 Coding model and terminologies

1.1 Terminologies

The key generation center (KGC) is a mutually trusted entity responsible for initialization and user registration including generation and distribution of the group key initially.

All participated users are required to register at KGC for subscribing the group key distribution service. Once registered, participated users will be considered as group members having the permission to receive the group key.

The inside attackers are group members who collude with other group members to recover the private secret k_i of one group member shared with the KGC.

The outside attackers are unregistered users who impersonate group members for requesting a group key service.

1.2 Model

The group key transfer protocol consists of two types of entities: the KGC and users. The former entity is primarily used to generate and distribute the group key, and the latter is to subscribe the key distribution service. Moreover, the KGC also takes the charge of tracing all group members in case of their dynamic joining and leaving. The proposed protocol uses erasure codes to replace the commonly used secret sharing scheme and cryptosystem.

1.3 Security goals

Non-repeatability The generated group key should be random and unpredictable, and has never been repeated before. Thus, forward and backward secrecy is provided to prevent a previously leaved group member from continuing obtaining any information of this current group communication and a newly joined group member from accessing any information of the previous group communication before it joined the group.

Confidentiality It ensures that the generated group key should only be reconstructed by registered group members rather than by any unregistered member.

Authentication It ensures that the group key is sent by the KGC without any tampering and forging by attackers during the transmission. Thus, authentication mechanism can effectively prevent attackers from impersonating the KGC for generating and distributing the fake group key.

2 Related work

According to Ref. [8], a brief overview of three kinds of commonly used group key management protocols will be discussed.

Centralized group key management protocols: Ref. [9] introduced new theoretical measures for the qualitative and quantitative assessment of encryption schemes designed for broadcast transmissions. A set of secret sharing-based group key transfer protocols have been proposed in Refs [10-12]. Ref. [2] proposed an authenticated group key transfer protocol using secret sharing scheme. Ref. [13] proposed an improved authenticated group key transfer protocol that resists against both insider and outside attacks based on Ref. [2]. Ref. [14] designed a secure and efficient group key transfer protocol based on securely outsourcing interpolation computation method in cloud computing. Ref. [15] proposed a new group key transfer protocol in wireless sensor networks using a linear secret sharing scheme and factoring assumption.

Decentralized group key management protocols: Ref. [16] proposed a framework for scalable secure multicasting with a hierarchy of agents that splits the large group into small subgroups. Ref. [17] proposed a dual encryption protocol that suggests splitting the whole group into hierarchical subgroups where a subgroup manager controls each subgroup. Ref. [18] proposed an intra region group key management protocol composed of many local area groups. In this protocol, there is a domain key distributor and many area key distributors re-

sponsible for each local area group. Ref. [19] proposed Hydra, a decentralized group key management scheme, which divides the large group into smaller subgroups and a Hydra server that controls each subgroup. Ref. [20] proposed a novel decentralized group key management scheme that solves the trusted third party problem of secret data leakage to the intermediate proxies.

Distributed group key management protocols: Most distributed group key management protocols can be either divided into the natural generation of DH key agreement protocol, where the group members use DH protocol to generate group key, or non-DH key agreement protocols. In the former case, Ref. [21] proposed the group DH key exchange that supports group operations instead of DH key exchange protocol. Ref. [22] proposed a provably secure authenticated group DH key exchange with a formal model and security definitions, as well as methods. In the latter case, Ref. [23] proposed a conference key agreement protocol based on a combining function and a one-way Hash function. Ref. [24] investigated a distributed protocol based on logical key hierarchy tree. Ref. [25] proposed a secure fault-tolerance conference key agreement protocol based on discrete logarithm against malicious participants. Ref. [26] revised the definitions of security for group key exchange protocols.

3 Preliminaries

3.1 A brief introduction to erasure codes

The fundamental concept of erasure codes is to encode the k original data blocks into n encoded data blocks. When t pieces of them are lost, the original data blocks can be reconstructed from the rest of any $n-t$ pieces, such kind of erasure code is called the (n, k) coding model. If $t = n - k$, this means that it can provide optimal storage efficiency as shown in Fig. 1.

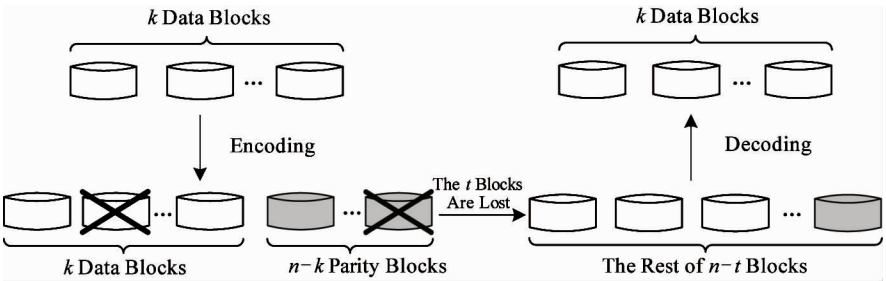

Fig. 1 The (n, k) coding model

3.2 AG codes

The generator matrix and the parity-check matrix will be given. Next, definitions and proposition of the two specific matrices will be given.

Definition 1 Let $n+1 \leqslant q \leqslant 2^m$, $GF(q)$ be a subfield of $GF(2^m)$. A $(n-k) \times n$ parity-check matrix $\boldsymbol{H}_{(n-k)\times n}$ is defined by

$$\boldsymbol{H}_{(n-k)\times n} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ h_1 & h_2 & h_3 & \cdots & h_n \\ h_1^2 & h_2^2 & h_3^2 & \cdots & h_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_1^{n-k-1} & h_2^{n-k-1} & h_3^{n-k-1} & \cdots & h_n^{n-k-1} \end{pmatrix} \tag{1}$$

where g is a primitive element of $GF(q)$, and $h_i = g^i$, for $i = 1, 2, \cdots n$ are n distinct nonzero elements of $GF(q)$. Obviously, $\boldsymbol{H}_{(n-k)\times n}$ is defined by Vendermonde matrix. Generally, the corresponding generator matrix can be easily constructed based on $\boldsymbol{H}_{(n-k)\times n}$.

Definition 2 The $k \times n$ generator matrix $\boldsymbol{G}_{k\times n}$ is defined by

$$\boldsymbol{G}_{k\times n} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & d_{1,k+1} & d_{1,k+2} & \cdots & d_{1,n} \\ 0 & 1 & 0 & \cdots & 0 & d_{2,k+1} & d_{2,k+2} & \cdots & d_{2,n} \\ 0 & 0 & 1 & \cdots & 0 & d_{3,k+1} & d_{3,k+2} & \cdots & d_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & d_{k,k+1} & d_{k,k+2} & \cdots & d_{k,n} \end{pmatrix} \tag{2}$$

where $\boldsymbol{G}_{k\times n}$ consists of two submatrices, a $k \times k$ identity matrix on the left, and another $k \times (n-k)$ submatrix on the right.

Proposition 1 For a given $(n-k) \times n$ parity-check matrix $\boldsymbol{H}_{(n-k)\times n}$, there exists a corresponding $k \times n$ generator matrix $\boldsymbol{G}_{k\times n}$ such that $\boldsymbol{G}_{k\times n} \cdot \boldsymbol{H}^T_{(n-k)\times n} = \boldsymbol{O}_{k\times(n-k)}$, where $\boldsymbol{H}^T_{(n-k)\times n}$ is the transpose of $\boldsymbol{H}_{(n-k)\times n}$ and $\boldsymbol{O}_{k\times(n-k)}$ is a $k \times (n-k)$ zero matrix.

Proof Apparently, the rank of $\boldsymbol{H}_{(n-k)\times n}$ is $n-k$ for being the Vandermonde matrix. In this case, the $n-k$ row vectors of $\boldsymbol{H}_{(n-k)\times n}$ are linearly independent, and they can make a $n-k$ dimensional liner subspace $\boldsymbol{R}_{n-k}$ of n dimensional linearapace over $GF(q)$ accordingly.

Therefore, there exists a linear subspace $\boldsymbol{R}_k$ with dimensional k, called orthogonal complement subspace, in such a way that $\boldsymbol{R}_{n-k} \perp \boldsymbol{R}_k$ holds over $GF(q)$. It is obvious that the orthogonal relation of $\boldsymbol{R}_{n-k} \perp \boldsymbol{R}_k$ holds over $GF(2^m)$ as well. Let's choose k basis vectors, $\alpha_1, \alpha_2, \cdots, \alpha_k$, from $\boldsymbol{R}_k$, and regard them as the k row vectors of a matrix $\boldsymbol{A}_{k\times n} = (\alpha_1, \alpha_2, \cdots, \alpha_k)^{\mathrm{T}}$. It follows that $\boldsymbol{A}_{k\times n} \cdot \boldsymbol{H}^{\mathrm{T}}_{(n-k)\times n} = \boldsymbol{O}_{k\times(n-k)}$. Then, the elementary row transformation of $\boldsymbol{A}_{k\times n}$ will be carried out gradually until it becomes a matrix $\boldsymbol{G}_{k\times n}$ with the first k columns of $\boldsymbol{G}_{k\times n}$ being an identity submatrix on the left. The transformed row vectors of $\boldsymbol{G}_{k\times n}$ are in the subspace R_k due to the elementary row transformation. Thus, $\boldsymbol{G}_{k\times n} \cdot \boldsymbol{H}^{T}_{(n-k)\times n} = \boldsymbol{O}_{k\times(n-k)}$ is true over $GF(2^m)$. The proof is completed, and please see Ref. [27] for more details.

Let $\boldsymbol{u} = (u_1, u_2, \cdots, u_k)$ be the message to be encoded as input over $GF(2^m)$. Then the output of $\boldsymbol{u}$ is a longer sequence $\boldsymbol{v} = (v_1, v_2, \cdots, v_n)$, called codeword of $\boldsymbol{u}$ and $\boldsymbol{v}$ can be defined by the matrix product of $\boldsymbol{u}$ and $\boldsymbol{G}_{k\times n}$ as follows:

$$\boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{G}_{k\times n} \tag{3}$$

Clearly, $\boldsymbol{v}$ is linearly combined by the rows of generator matrix $\boldsymbol{G}_{k\times n}$. It can be easily found from Proposition 1 that $\boldsymbol{G}_{k\times n} \cdot \boldsymbol{H}^{T}_{(n-k)\times n} = \boldsymbol{O}_{k\times(n-k)}$, then the corresponding recovery of $\boldsymbol{u}$ from $\boldsymbol{v}$ is given as below:

$$\boldsymbol{v} \cdot \boldsymbol{H}^{T}_{(n-k)\times n} = \boldsymbol{o} \tag{4}$$

where $\boldsymbol{o}$ is an all-zero $(n-k)$-tuple. Consequently, $\boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{G}_{k\times n}$ is called encoding, and solving $\boldsymbol{u}$ from $\boldsymbol{v} \cdot \boldsymbol{H}^{T}_{(n-k)\times n} = \boldsymbol{o}$ is called decoding. For more details about arithmetic operations of elements of $GF(2^m)$, please see Ref. [28].

4 Scheme based on AG codes

The proposed scheme consists of four phases: initialization of KGC, user registration, the generation and distribution of group key and the reconstruction and authentication of group key.

4.1 The initialization of KGC

Suppose that there are n users, denoted by $\boldsymbol{U} = \{U_1, U_2, \cdots, U_n\}$, participating in the group communication. KGC needs to initialize the generator matrix $\boldsymbol{G}_{(n+1)\times(2n+1)}$ and parity-check matrix $\boldsymbol{H}_{n\times(2n+1)}$ based on the rules of AG codes. Furthermore, the one-variable one-way Hash function $H_1(x)$ and multi-variable one-way Hash function $H_2(x)$ are also expected to be included. All of them should be made known to the public, and their computations are performed over $GF(2^m)$.

4.2 User registration

Each user $U_i \in \boldsymbol{U}$ is required to register at KGC at the beginning of the group communication. Once registered successfully, user U_i will be considered as a legitimate group member for requesting group session key services. KGC shares a mutually different private secret k_i with each user U_i in a secure channel, where $k_i \in GF(2^m)$. Moreover, KGC also needs to supervise all users, for example, trace legitimate group members and remove illegal users.

4.3 The generation and distribution of group key

KGC needs to randomly select a group session key and transfer it to all group members safely after receiving a group key generation request from any initiator. All communications between KGC and group members are in broadcast channel. However, KGC must distribute the generated group session key to all group members securely, and the group members can also reconstruct and authenticate the group session key likewise. The generation and distribution of group key contains the following four steps.

(1) The initiator sends a key generation request of the group communication to KGC with a list of n group members $\boldsymbol{U} = \{U_1, U_2, \cdots, U_n\}$.

(2) KGC broadcasts a random number R to the n group members $\boldsymbol{U} = \{U_1, U_2, \cdots, U_n\}$.

(3) Each group member $U_i \in \boldsymbol{U}$ also needs to send a random challenge $R_i \in GF(2^m)$, for $i = 1, 2, \cdots, n$, to the KGC as a response.

(4) With n private secret k_i shared with n group members, KGC computes the n Hash values $H_1(k_i)$, for $i = 1, 2, \cdots, n$ and randomly selects an unused group session key $k \in GF(2^m)$, and then regard $(k, H_1(k_1), H_1(k_2), \cdots, H_1(k_n))$ as the encoded message u of AG codes. Message u will be used to generate codeword v with generator matrix $\boldsymbol{G}_{(n+1)\times(2n+1)}$, defined by $\boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{G}_{(n+1)\times(2n+1)}$. Since the left part of $\boldsymbol{G}_{(n+1)\times(2n+1)}$ is a $(n+1)\times(n+1)$ identity matrix, the $n+1$ components in the left of $\boldsymbol{v}$ are identical to $\boldsymbol{u}$, so codeword $\boldsymbol{v}$ can be denoted by $\boldsymbol{v} = (k, H_1(k_1), H_1(k_2), \cdots, H_1(k_n), v_1, v_2, \cdots, v_n)$. KGC computes authentication message $AM = H_2(k, H_1(k_1), H_1(k_2), \cdots, H_1(k_n), R_1, R_2, \cdots, R_n)$, and then deletes the same $n+1$ components, $k, H_1(k_1), H_1(k_2), \cdots, H_1(k_n)$. Finally, KGC broadcasts the remaining n components, $v_1, v_2, \cdots, v_n$, and authentication message AM as public information to all group members $\boldsymbol{U}$.

4.4 The reconstruction and authentication of group key

After receiving the broadcasted n components, $v_1, v_2, \cdots, v_n$, and the authentication message AM from KGC, all of the group members $\boldsymbol{U} = \{U_1, U_2, \cdots, U_n\}$ need to reconstruct and authenticate the group session key with the received public information and shared private secrets. For each group member U_i, possessing the Hash value $H_1(k_i)$ with private secret k_i and broadcasted n components, $v_1, v_2, \cdots, v_n$, the group session key k can be reconstructed easily.

For the convenience of better description, let $\boldsymbol{v} = (k', H'_1(k_1), H'_1(k_2), \cdots, H_1(k_i), \cdots, H'_1(k_n), v_1, v_2, \cdots, v_n)$, where $H_1(k_i), v_1, v_2, \cdots, v_n$ are $n+1$ known components, and the other n components $k', H'_1(k_1), H'_1(k_2), \cdots, H'_1(k_{i-1}), H'_1(k_{i+1}), \cdots, H'_1(k_n)$ are unknown. By the generation of group key, it holds that $\boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{G}_{(n+1)\times(2n+1)}$. According to the proposition given in Section 4, every row vector of the parity-check matrix $\boldsymbol{H}_{n\times(2n+1)}$ is orthogonal to the linear subspace spanned by the row vectors of the generator matrix $\boldsymbol{G}_{(n+1)\times(2n+1)}$, that is, equation $\boldsymbol{v} \cdot \boldsymbol{H}^T_{n\times(2n+1)} = \boldsymbol{o}$ holds. Hence, a group of following equations can be got:

$$\begin{cases} k' + H_1'(k_1) + H_1'(k_2) + \cdots + v_n = 0 \\ h_1 k' + h_2 H_1'(k_1) + h_3 H_1'(k_2) + \cdots + h_{2n+1} v_n = 0 \\ h_1^2 k' + h_2^2 H_1'(k_1) + h_3^2 H_1'(k_2) + \cdots + h_{2n+1}^2 v_n = 0 \\ \vdots \\ h_1^{n-1} k' + h_2^{n-1} H_1'(k_1) + h_3^{n-1} H_1'(k_2) + \cdots + h_{2n+1}^{n-1} v_n = 0 \end{cases} \tag{5}$$

where $H_1(k_i), v_1, v_2, \cdots, v_n$ are $n+1$ known terms, and the other n terms k', $H_1'(k_1)$, $H_1'(k_2), \cdots, H_1'(k_{i-1})$, $H_1'(k_{i+1}), \cdots, H_1'(k_n)$ are unknown. By simplifying, a group of n linear equations with n unknown terms can be got, denoted by $\boldsymbol{v}' \cdot \boldsymbol{H}'_{n\times n} = \boldsymbol{o}'$, where $\boldsymbol{v}' = (k', H_1'(k_1), H_1'(k_2), \cdots, H_1'(k_{i-1}), H_1'(k_{i+1}), \cdots, H_1'(k_n))$ is an unknown vector with n components, $\boldsymbol{o}'$ is known, and

$$\boldsymbol{H}_{n\times n}' = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 \\ h_1 & h_2 & h_3 & \cdots & h_{i-1} & h_{i+1} & \cdots & h_{n+1} \\ h_1^2 & h_2^2 & h_3^2 & \cdots & h_{i-1}^2 & h_{i+1}^2 & \cdots & h_{n+1}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ h_1^{n-1} & h_2^{n-1} & h_3^{n-1} & \cdots & h_{i-1}^{n-1} & h_{i+1}^{n-1} & \cdots & h_{n+1}^{n-1} \end{pmatrix} \tag{6}$$

is an $n \times n$ Vandermonde matrix with n different elements. Therefore, the group of equations $\boldsymbol{v}' \cdot \boldsymbol{H}'_{n\times n} = \boldsymbol{o}'$ has a unique solution, and the n terms which include the group key k' can be recovered.

Meanwhile, U_i also needs to compute the Hash value $H_2(k', H'_1(k_1), H'_1(k_2), \cdots, H'_1(k_n), R_1, R_2, \cdots, R_n)$ with the recovered terms, and check if this Hash value is identical to AM. If the two values are the same, then U_i can authenticate that the group key is sent from KGC without any tampering and forging by attackers during the transmission. Fig. 2 gives an outline of our proposed scheme.

Phases		KGC	Users
Initialization		Initialize $G_{(n+1)\times(2n+1)}, H_{n\times(2n+1)}, H_1(x)$ and $H_2(x)$	
Registration		Share secret key k_i	U_i
Generation and Distribution	1	Group key request $U = \{U_1, U_2, \ldots, U_n\}$	Initiator
	2	R	U
	3	$R_1, R_2, \ldots, R_n$	U
	4	The KGC computes $H_1(k_1), H_1(k_2), \ldots, H_1(k_n)$ and selects group key k, and then generatcs $v_1, v_2, \ldots, v_n$ and AM by $v = u \cdot G_{(n+1)\times(2n+1)}$. $v_1, v_2, \ldots, v_n$ and AM	U
Reconstruction and Authentication		Each participated user U_i needs to reconstruct $k', H_1'(k_1), H_1'(k_2), \ldots, H_1'(k_{i-1}), H_1'(k_{i+1}), \ldots, H_1(k_n)$, and then checks whether $AM = H_2(k', H_1'(k_1), H_1'(k_2), \ldots, H_1'(k_n), R_1, R_2, \ldots, R_n)$.	

Fig. 2 The group key transfer protocol based on AG codes

5 Performance and security analysis

In this section, the performance and security of our proposed scheme will be discussed. The efficiency will be evaluated including non-repeatability, confidentiality, and authentication.

5.1 Performance analysis

For evaluating the efficiency of storage, computation, and communication overhead, the performance analysis through the qualitative and quantitative approach will be given. The scheme based on AG codes, however, provides a more general form and diversified techniques available that can improve efficiency of storage, computation, and communication overhead.

Storage overhead: In our scheme, the storage of the received public information based on AG codes model is usually less than the storage of message based on traditional Lagrange interpolating polynomial model. Each group member U_i needs to store n broadcasted components, $v_1, v_2, \cdots, v_n$ and authentication message AM from KGC. In addition, each group member U_i also needs to store its shared private secret k_i, and KGC also needs to store n shared private secrets, $k_1, k_2, \cdots, k_n$, and group key k. Hence, for each group member, the total storage cost is $n + 2$; for KGC, the total storage cost is $2n + 2$.

Computation overhead: The basic arithmetic operation of our scheme is implemented over $GF(2^m)$ instead of traditional Lagrange interpolating polynomial over the prime field $\mathbb{F}_p$. It is inescapably clear that our proposed protocol breaks through the limit of prime field and can be implemented over XOR operations instead of infinite field multiplication operations by converting $GF(2^m)$ into an $m \times m$ matrix over $GF(2)$[29]. Besides, a more practical approach is developed in Ref. [30], where a fast Fourier transform algorithm is employed to reduce significantly the number of expensive finite field multiplications required which provides substantially better performance compared with the standard algorithm. Furthermore, it is also easy to further speed up computation of AG codes by adopting the standard algorithm of a cyclic code and its integrated circuit, because the parity-check matrix in our scheme gives a dual code of RS code in broad sense[31,32], that is to say, a cyclic code.

Communication overhead: In our scheme, five rounds of communication, as shown in Fig. 2, are performed between the two parties, KGC and group members. Obviously, the total communication cost between KGC and group members is $4n + 2$.

5.2 Security analysis

In this subsection, the specific analysis of security goals including non-repeatability, confidentiality, and authentication will be given in detail.

Non-repeatability The non-repeatability of group key generation is ensured by KGC since the generated group key is randomly selected by KGC for group session key service request. In addition, the parity-check matrix $\boldsymbol{H}_{n\times(2n+1)}$ used to reconstruct the group key and the corresponding generator matrix $\boldsymbol{G}_{(n+1)\times(2n+1)}$ used to distribute the group key based on the parity-check matrix $\boldsymbol{H}_{n\times(2n+1)}$ are securely selected by KGC according to the rules of AG codes.

Confidentiality The confidentiality of group key distribution is provided due to the security features of the proposed AG codes. KGC generates and broadcasts n components, $v_1, v_2, \cdots, v_n$, as public information to all group members U. For each authorized group member, U_i, it possesses the Hash value $H_1(k_i)$ with private secret k_i and n broadcasted components, $v_1, v_2, \cdots, v_n$. Thus, any authorized group member is able to reconstruct the group session key k. For each unauthorized member, although it accesses the public information, it doesn't possess the Hash value $H_1(k_i)$ with private secret k_i. Thus, any unauthorized group member can know nothing about the group session key k. This kind of security property is information-theoretically secure without relying on any computational complexity of mathematical assumptions.

Authentication The authentication of group key reconstruction is provided by computing and broadcasting a single authentication message AM to all group members, where AM is the multi-variable one-way Hash function output with the group key, the one-variable one-way Hash function outputs, and all group members' random challenges as inputs. Unauthorized group members cannot forge the authentication message AM since the group key is merely available to authorized group members and KGC. Any inside group members cannot forge a group key without being detected since the group key is a component of codeword containing n Hash values $H_1(k_i)$, for $i = 1, 2, \cdots, n$, with the private secret shared with each group members and KGC.

Theorem 1 The proposed protocol achieves the following security requirements: 1) forward secrecy, and 2) backward secrecy.

Proof The group key will be re-newly generated and distributed as soon as any group member leaves or joins the group. Therefore forward secrecy is provided to prevent a previously leaved group member from con-

tinuing accessing group's communication messages after it leaves the group. Meanwhile, backward secrecy is achieved to prevent a newly joined group member from accessing messages exchanged before it joins the group as well.

Theorem 2 The proposed protocol achieves the following possible attacks: (1) outside attack, and (2) inside attack.

Proof (1) Assume that an outside attacker can impersonate a group member for requesting a group key generation service to KGC without being detected successfully. Then KGC will respond by generating and broadcasting the group key public information accordingly. The group key, however, can only be reconstructed by any authorized group member who shares private secret with KGC. Thus, the outside attacker can know nothing about the group key.

Assume that the outside attacker can reuse a compromised group key by replaying the previously broadcasted group key public information from KGC. But the authentication message AM is a multi-variable one-way Hash function $H_2(x)$ with each group member's random challenge as input. Such an attack cannot succeed in sharing this compromised group key with any group member. So this compromised group key cannot be reused if each group member selects a random challenge for each group sessions. Thus, the outside attacker cannot share a group key with any group member.

(2) Assume the applied AG codes are still secure after the protocol has been used successfully several times. For a group key generation service request, KGC generates and broadcasts n components, $v_1, v_2, \cdots, v_n$, as public information to all group members $\boldsymbol{U}$. For each authorized group member U_i, possessing the Hash value $H_1(k_i)$ with private secret k_i and broadcasted n components, $v_1, v_2, \cdots, v_n$. Thus, any authorized group member is able to reconstruct the session key k and $n-1$ Hash values $H_1(k_1)$, $H_1(k_2), \cdots, H_1(k_{i-1})$, $H_1(k_{i+1}), \cdots, H_1(k_n)$ with other $n-1$ members' private secret, $k_1, k_2, \cdots, k_n$, successfully. Meanwhile, it is commonly believed that it is computationally infeasible to obtain k_i for any given $H_1(k_i)$, Thus, private secret k_i of each authorized group member shared with KGC still remains unknown to outside attackers.

6 Conclusions

In this paper, an efficient approach to the secure group key transfer protocol is proposed. In addition, a way for the group members to authenticate the integrity of group key and verify whether the sender is really KGC is also presented. The confidentiality is information-theoretically secure which can resist both insider and outside attacks, and the non-repeatability of group key generation is also achieved which can provide forward and backward secrecy. Furthermore, a more generalized and simple mathematical construction model for group key transfer protocol than traditional Lagrange interpolating polynomial has been given. This kind of model can also be applied easily to related fields of information security perfectly.

References

[1] Diffie W, Hellman M. New directions in cryptography [J]. *IEEE Transactions on Information Theory*, 1976, 22(6): 644-654

[2] Harn L, Lin CL. Authenticated group key transfer protocol based on secret sharing [J]. *IEEE Transactions on Computers*, 2010, 59(6):842-846

[3] Reed I S, Solomon G. Polynomial codes over certain finite fields [J]. *Journal of the Society for Industrial and Applied Mathematic*, 1960,8(2): 300-304

[4] Bose R C, Chaudhuri D K R. On a class of error correcting binary group codes [J]. *Information and Control*, 1960, 3(1):68-79

[5] Hocquenghem A. Codes correcteursd′ erreurs [J]. *Chiffres*,1959, 2:147-156

[6] Gallager R. Low-density parity-check codes [J]. *IEEE Transactions on Information Theory*, 1962,8(1)1:21-28

[7] Blake I, Heegard C, Hoholdt T, et al. Modeling key compromise impersonation attacks on group key exchange protocols [J]. *IEEE Transactions on Information Theory*, 1998, 44(6):2596-2618

[8] Rafaeli S, Hutchison D. A survey of key management for secure group communication [J]. *ACM Computing Surveys*, 2003, 35(3):309-329

[9] Fiat A, Naor M. Broadcast encryption [C]. In: Proceedings of Annual International Cryptology Conference, California, USA, 1993. 480-491

[10] Li C H, Pieprzyk J. Conference key agreement from secret sharing [C]. In: Proceedings of the 4th Australasian Conference on Information Security and Privacy, Wollongong, Australia, 1999. 64-76

[11] Saez G. Generation of key predistribution schemes using secret sharing schemes [J]. *Discrete Applied Mathematics*, 2003, 128(1): 239-249

[12] Katz J, Yung M. Scalable protocols for authenticated group key exchange [J]. *Journal of Cryptology*,2007, 20(1):85-113

[13] Liu Y N, Cheng C, Cao J Y, et al. An improved authenticated group key transfer protocol based on secret sharing [J]. *IEEE Transactionson Computers*, 2013, 62(11):2335-2336

[14] Wu J C, Liu Q, Liao X F. A secure and efficient outsourceable group key transfer protocol in cloud computing [C]. In: Proceedings of the 2nd International Workshop on Security in Cloud Computing, New York, USA, 2014. 43-50

[15] Hsu C F, Harn L, He T T, et al. Efficient group key

transfer protocol for WSNs[J]. *IEEE Sensors Journal*, 2016,16(11): 4515-4520

[16] Mittra S. Iolus: a framework for scalable secure multicasting[C]. In: Proceedings of the ACM SIGCOMM, New York, USA, 1997. 277-288

[17] Dondeti L R, Mukherjee S, Samal A. Scalable secure one-tomany group communication using dual encryption [J]. *Computer Communications*, 2000, 23(17): 1681-1701

[18] DeCleene B, Dondeti L, Griffin S, et al. Secure group communications for wireless networks[C]. In: Proceedings of the 2001 Military Communications Conference, Mclean, USA, 2001. 113-117

[19] Rafaeli S, Hutchison D. Hydra: A decentralised group key management[C]. In: Proceedings of the 11th IEEE International Workshops on Enabling Technologies: frastructure for Collaborative Enterprises, Pittsburgh, USA, 2002. 62-67

[20] Hur J, Shin Y, Yoon H. Decentralized group key management for dynamic networks using proxy cryptography [C]. In: Proceedings of the 3rd ACM workshop on QoS and Security for Wireless and Mobile Networks, Chania, Greece, 2007. 123-129

[21] Steiner M, Tsudik G, Waidner M. Diffie-Hellman key distribution extended to group communication[C]. In: Proceedings of the 3rd ACM Conference on Computer and Communications Security, New Delhi, India, 1996. 31-37

[22] Bresson E, Chevassut O, Pointcheval D. Provably secure authenticated group Diffie-Hellman key exchange[J]. *ACM Transactions on Information and System Security*, 2007, 10(3)10:1-10:45

[23] Boyd C. On key agreement and conference key agreement [C]. In: Proceedings of the 2nd Australasian Conference on Information Security and Privacy, 1997. 294-302

[24] Rodeh O, Birman K, Dolev D. Optimized group rekey for group communications systems[R]. Ithaca, USA: Cornell University, 1999

[25] Tzeng W G. A secure fault-tolerant conference-key agreement protocol [J]. *IEEE Transactions on Computers*, 2002, 21(4):373-379

[26] Gorantla M C, Boyd C, Nieto J M G, et al. Modeling key compromise impersonation attacks on group key exchange protocols[J]. *ACM Transactions on Information and System Security*, 2011, 14(4):28:1-28:24

[27] Wang X J, Yuan Q Z, Cai H L, et al. A new approach to image sharing with high-security threshold structure[J]. *Journal of the ACM*, 2014, 61(6):39:1-39:19

[28] Lidl R, Niederreiter H, Cohn P M. Finite Fields (2nd version.) [M]. Cambridge: Cambridge University Press, 1997

[29] Bloemer J, Kalfane M, Karp R, et al. An XOR-based erasure-resilient coding scheme [R]. Technical Report No. TR-95-048, International Computer Science Institute, Berkeley, California, 1995

[30] Trifonov P. Low-complexity implementation of RAID based on Reed-Solomon codes[J]. *ACM Transactions on Storage*,2005, 11(1):1:1-1:25

[31] Macwilliams F J, Sloane N J A. The Theory of Error-Correcting Codes[M]. New York: North-Holland Publishing Company, 1977

[32] Lin S, Costello D J. Error Control Coding: Fundamentals and Applications[M]. Englewood Cliffs: Prentice-Hall, 2004

Yuan Dezhai, born in 1989. He is currently working toward the Ph. D degree in computer science at University of Chinese Academy of Sciences. He received the B. S. degree in management information systems from Yantai Nanshan University in 2012. His research interests include erasure codes, secret sharing and cryptography.