Oriented User Group Verifiable Attribute-Based Encryption Scheme

Xueyan Liu^{1, 2} and Zhanming Li¹

¹*College of Telecom Engineering and Information Engineering, Lanzhou University of Technology, Lanzhou, 730050 China*

²*College of Mathematics and Statistics, Northwest Normal University, Lanzhou 730070 China*

liuxy@nwnu.edu.cn; 37599487@qq.com

Abstract

In this paper, we construct an oriented user group verifiable attribute-based encryption (OG-ABEv) scheme. Our scheme not only maintains on the security of previous ABE schemes, but also enhances two functions: (1) It integrates user group into the ABE scheme, thereby strengthening the confidentiality of data sharing and reducing excessive centralization of user rights; (2) It provides a verification method to supervising the central authority (CA). It also allows users to immediately check the correctness of the attribute keys and allows semi-trusted or untrusted CA be in our system. We formalize the concept of OG-ABEv and prove our scheme.

Keywords: Information Security; Oriented User Group; Attribute-based Encryption; Secret Sharing; Verifiable Secret Sharing

1. Introduction

Attribute-based encryption (ABE), introduced by Sahai and Waters [1] in EUROCRYPT 2005, is based on public key encryption and an extension of identity based encryption. In ABE scheme, the identity is described as a set of attributes; ciphertext and secret key are generated based on the set of attributes. In this way, the set of attributes and access structure can be combined and access control of shared data is realized conveniently.

ABE has developed rapidly since the concept was proposed [1]. In [2], Goyal *et al.* expounded the concept and significance of ABE and divided the ABE scheme into two types: ciphertext-policy ABE (CP-ABE) and key-policy ABE (KP-ABE). Cheung *et al.* [3] and Bethencourt *et al.* [4] constructed a ciphertext policy ABE scheme respectively, in which attributes were used to describe a user's credentials, and a data owner (DO) determined a policy for who could perform decryption. In [5], Waters presented a new methodology for realizing CP-ABE under concrete and non-interactive cryptographic assumptions in the standard model. Emura [6] and Chen *et al.* [7] considered constant-size ciphertext attribute encryption scheme, but the encryption policy can only support threshold access structure. Attrapadung and Libert [8] constructed an ABE scheme supporting general access structure. The length of the ciphertext and the encryption or decryption cost were constant in this scheme. Based on LWE problem, Agrawal *et al.* [9] presented a new ABE scheme, in which access policy was the threshold; the study also discussed the difficulty of general access structure based on lattice. Chase *et al.* [10, 11] solved a single authority corrupted problem by employing multiple authorities and prevented the collusion between users by adopting a global identity GID for each user. Lewko *et al.* [12] presented another multi-authority scheme in which any party could become an authority and any central authority (CA) was not needed.

In light of the rich expression ability of attribute set and flexible access control ability in ABE, it has good application prospects in distributed file management, third data storage, pay TV system, and other fields [13]. However, ABE scheme generally has a CA that is responsible for system initialization and key distribution, so dependence on CA is relatively strong and CA must be trusted, which is unlikely in reality; the CA is unsupervised, and the attribute keys are not verified. Hence, CA can easily become the preferred target. Once compromised, CA might distribute the wrong key, which would cause damage to the system. In ABE scheme, a data owner encrypts shared data. One user can obtain full permission if he has sufficient attributes, which will lead to excessive concentration of user's rights and abuse. High security and decentralization of user permission are needed in crucial applications, such as bank network management, opening vault accounts, missile control and launch, and launch of nuclear weapons, all authorized users or most of authorized users to participate in the completion.

In view of the above problems, we'll propose an oriented user group verifiable ABE scheme.

1.1. Our Contribution

To overcome the above-mentioned problems, we propose an oriented user group verifiable ABE scheme. Firstly, we propose the first oriented user group attribute-based encryption mechanism. In this mechanism, decryptor is a user group, which decentralizes user permission, strengthens the safety and integrity of important, sensitive and confidential information, and enables each participant to store only a small amount of data. Secondly, we introduce a verifiable secret sharing scheme to ensure a supervision of CA. This technique can verify the correctness of generated attribute keys and reduce the dependence on CA, even a semi-trusted or untrusted CA is used. At the same time, each participant can verify the honesty of other participants by checking the information they provide. A user group will only be able to decrypt a ciphertext if at least t participants whose attributes pass through the ciphertext access structure and who provide correct information to others, this situation can avoid invalid recovery operation, and enhance practicality. Furthermore, the access tree structure will be simplified (see details in discussion).

1.2. Organization

The remainder of our paper is structured as follows: in Section 2, we provide the preliminaries and definitions of OG-ABEv. In Section 3, we present concrete construction. The proof is presented in Section 4. The discussion and comparison are presented in Section 5. Finally, concludes this paper.

2. Preliminary

We first present the access structure, bilinear maps and cryptographic assumption. Then we give the definition for the oriented user group verifiable CP-ABE scheme. Finally, we present the Schoenmakers' verifiable secret sharing scheme, which is used in our scheme.

2.1. Access Structure and Bilinear Maps

Definition 1 (Access Structure) Let $\{P_1, P_2, \dots, P_n\}$ be a set of parties. A collection $\mathbf{A} \subseteq 2^{\{P_1, P_2, \dots, P_n\}}$ is monotone if B, C : if $B \in \mathbf{A}$ and $B \subseteq C$, then $C \in \mathbf{A}$. An access structure (respectively, monotone access structure) is a collection (respectively, monotone access structure) $\mathbf{A}$ of non-empty subsets of $\{P_1, P_2, \dots, P_n\}$, i.e., $\mathbf{A} \subseteq 2^{\{P_1, P_2, \dots, P_n\}} \setminus \{\Phi\}$. The

sets in $\mathbf{A}$ are called the authorized sets, and the sets not in $\mathbf{A}$ are called the unauthorized sets [4,14].

Let G_0 and G_1 be two multiplicative cyclic groups of prime order q .

Definition 2 (Bilinear Maps) A bilinear pairing is a computable bilinear map between two groups $e : G_0 \times G_0 \rightarrow G_1$. It is the modified Weil pairing or Tate pairing which has the following properties:

1. *Bilinear*: For any $g^a, g^b \in G_0, e(g^a, g^b) = e(g, g)^{ab}$;
2. *Non-degenerate*: There exists $P, Q \in G_0$, such that $e(P, Q) \neq 1$;
3. *Efficient*: There exists an efficient algorithm to compute the map.

Definition 3 (Decisional Bilinear Diffie-Hellman Assumption) Suppose a challenger chooses $a, b, c, \theta \in_R \mathbf{Z}_q$. The Decisional Bilinear Diffie-Hellman (DBDH) Assumption is that no polynomial-time adversary A_d is to be able to distinguish the tuple $\langle g, A = g^a, B = g^b, C = g^c, e(g, g)^{abc} \rangle$ from the tuple $\langle g, A = g^a, B = g^b, C = g^c, e(g, g)^\theta \rangle$ with more than a non-negligible advantage. The advantage of A_d is:

$$|\Pr[A_d(A, B, C, e(g, g)^{abc}) = 0] - \Pr[A_d(A, B, C, e(g, g)^\theta) = 0]|.$$

2.1. Oriented User Group Verifiable ABE Scheme (OG-ABEv)

Definition 4 OG-ABEv is a special ciphertext policy ABE scheme in which the decryptor is a user group, the correctness of attribute key could be verified when the user acquires the key from CA, and the honesty of each participant could be checked before decryption. OG-ABEv includes the following algorithms:

- **Setup**: Given a security parameter, the trusted CA runs a randomized algorithm that outputs a system public key PK and master secret key MK .
- *User Private Key Generation*: The user runs a randomized algorithm. The user private key and corresponding public key are output.
- **Encryption**: The encryption algorithm takes as input the public parameters PK , a message M , and an access structure T over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that at least $t-1$ users in a user group who possess a set of attributes that satisfy the access structure will be able to decrypt the message. We will assume that the ciphertext implicitly contains T .
- **KeyGen**: The authority runs a randomized algorithm. It takes as input the master secret key and n attribute sets of a user group. The outputs are attribute secret keys for each user of the user group, public information, and proof for later verification.
- **Verification**: Each user runs a randomized algorithm. It takes as input the public information and the proof. *True* or *False* are output.
- **DecRecon**: The user carries out two tasks: reconstruction and decryption.
- *Reconstruction*: A user renews his attribute secret key and offers proof for verification and decryption. Each participant verifies the received information and reconstruct the secret share.
- *Decryption*: A user group runs a deterministic algorithm. It takes as input a ciphertext, which is encrypted under access tree T . It outputs M .

2.1.1. Security Model for OP-ABEv : Semantic security under chosen-plaintext attack (CPA) is modeled by an IND-sAtt-CPA game. The game is carried out between a

challenger and an adversary group $A_G = \{Ad_1, Ad_2, \dots, Ad_n\}$, where the challenger simulates the protocol execution and answers queries from A_G . The game is followed.

Init: The adversary group A_G sends the challenge access tree T^* to the challenger.

Setup: The challenger runs *Setup* to generate (PK, MK) and tells the adversary group A_G the public parameters.

Phase 1: The adversary group A_G is allowed to issue queries for verification information and private keys. The only requirement is that for each user group, at least $n - t + 1$ participants must exist whose attributes do not satisfy T^* .

Challenge: The adversary group A_G submits two equal length messages M_0 and M_1 . The challenger flips a random coin b and returns ciphertext $C_b = \text{Encrypt}(M_b, T^*, PK)$.

Phase 2: Phase 1 is repeated.

Guess: The adversary group A_G outputs a guess b' of b .

Definition 5 (OG-ABEv Secure) OG-ABEv scheme is said to be secure against an adaptive CPA if no polynomial-time adversary has a non-negligible advantage in the IND-sAtt-CPA game, where the advantage is defined to be $\epsilon = |\Pr[b' = b] - \frac{1}{2}|$.

2.2. Schoenmakers' Verifiable Secret Sharing Scheme

In 1985, Schoenmaker [15] realized a classical publicly verifiable secret sharing scheme using non-interactive zero knowledge proof algorithm-DLEQ [16] based on the discrete logarithm problem and Diffie-Hellman hard problem. The scheme has the advantages of a simple structure and better safety and performance compared with other solutions. The dealer is D , the set of parties is $U = \{u_1, u_2, \dots, u_n\}$, and t is the system threshold value. Let G denote a group of prime order q , such that computing discrete logarithms in this group is infeasible. Let g, h denote independently selected generators of G . The scheme consists following three phases:

- **Initialization:** The group G and the generators g, h are selected using an appropriate public procedure. Participant u_i generates a private key $x_i \in_R Z_q^*$ and registers $y_i = h^{x_i}$ as its public key.

- **Distribution and Verification:** The protocol consists of two steps:

- **Distribution of the shares.** The dealer selects a random polynomial $f(x)$ of degree $t - 1$ with coefficients in Z_q and sets $r = a_0$, $f(x) = \sum_{j=0}^{t-1} a_j x^j$. The dealer keeps this polynomial secret but publishes the related commitments $C_j = g^{a_j}$ ($j = 0, 1, \dots, t - 1$). The dealer also publishes the encrypted shares $Y_i = y_i^{f(i)}$ ($i = 1, 2, \dots, n$), using the public keys of the participants. Finally, let $X_i = \prod_{j=0}^{t-1} C_j^{i^j}$. The dealer shows that the encrypted shares are consistent by producing proof of knowledge of the unique $f(i)$ ($i = 1, 2, \dots, n$), satisfying $X_i = g^{f(i)}$, $Y_i = y_i^{f(i)}$. The dealer selects $\omega_i \in Z_q$, and computes $a_{1i} = g^{\omega_i}$, $a_{2i} = y_i^{\omega_i}$. The public commitment is

$c = H(X_1 \parallel \dots \parallel X_n \parallel Y_1 \parallel \dots \parallel Y_1 \parallel a_{11} \parallel \dots \parallel a_{1n} \parallel a_{21} \parallel \dots \parallel a_{2n} \parallel \dots)$, the commitment of each u_i is $r_i = \omega_i - f(i)c \bmod q, i = 1, 2, \dots, n$. The proof is

$$PROOF_D = (c, r_1, r_2, \dots, r_n)$$

- *Verification of the shares.* Each participant can compute $X_i = \prod_{j=0}^{t-1} C_j^{t_j}$ from C_j . Using public information: $y_i, X_i, Y_i, r_i, (i = 1, 2, \dots, n)$, the verifier computes a_{1i}, a_{2i} as

$$a_{1i} = g^{r_i} X_i^c, a_{2i} = y_i^{r_i} Y_i^c$$

and checks that the hash of $X_i, Y_i, a_{1i}, a_{2i}, (i = 1, 2, \dots, n)$ matches c .

• **Reconstruction:** The protocol consists of two steps:

- *Decryption of the shares.* Using its private key x_i , each participant finds the share $S_i = Y_i^{x_i^{-1} \bmod q}$. The non-interactive protocol $DLEQ(h, y_i, S_i, Y_i, x_i)$ is carried out and publishes S_i and a proof $Proof_{u_i}$ that can prove the consistency of S_i , that is to say, the participant can prove that he holds x_i without leaking of it, and $y_i = h^{x_i}$, $Y_i = S_i^{x_i}$.
- *Pooling shares.* Suppose that at least t participants $u_i (i \in B \subseteq \{1, 2, \dots, n\}, |B| = t)$ produce correct values for $S_i, i \in B$. The secret h^r is obtained by Lagrange interpolation:

$$h^r = \prod S_i^{C_{Bi}} = \prod h^{f(i)C_{Bi}} = h^{\sum f(i)C_{Bi}}$$

where $C_{Bi} = \prod \frac{i}{j-i} \bmod q$ is a Lagrange coefficient.

3. Oriented User Group Verifiable ABE Scheme (OG-ABEv)

To show how to construct an oriented user group verifiable ABE scheme, we begin by describing the structure of the access policy. We use the same notation as [4] to describe the access trees.

Access tree T . Let T be a tree representing an access structure. Each non-leaf node of the tree represents a threshold gate, described by its children and a threshold value. If num_x is the number of children of a node x and k_x is its threshold value, then $0 < k_x \leq num_x$. When $k_x = 1$, the threshold gate is an OR-gate and when $k_x = num_x$, it is an AND-gate. Each leaf node x of the tree is described by an attribute and a threshold value $k_x = 1$. We denote the parent of the node x in the tree by $parent(x)$. The function $attr(x)$ is defined only if x is a leaf node and denotes the attribute associated with the leaf node x in the tree. The access tree T also defines an ordering between the children of every node, that is, the children of a node are numbered from 1 to num . The function $index(x)$ returns such a number associated with the node x . Where the index values are uniquely assigned to nodes in the access structure for given key in an arbitrary manner.

Satisfying an access tree. Let T be an access tree with root R . Denote by T_x the subtree of T rooted at a node x . Hence T is the same as T_R . If a set of attributes γ satisfies the access tree T_x , we denote it as $T_x(\gamma) = 1$. We compute $T_x(\gamma)$ recursively as

follows. If x is a non-leaf node, evaluate $\mathbf{T}_{x'}(\gamma)$ for all children x' of node x . $\mathbf{T}_x(\gamma)$ returns 1 if and only if at least k_x children return 1. If x is a leaf node, then $\mathbf{T}_x(\gamma)$ returns 1 if and only if $attr(x) \in \gamma$.

3.1. Main Construction

Our scheme consists of DO, CA and a user group $\mathbf{U} = \{u_1, u_2, \dots, u_n\}$.

Let G_0, G_1 be bilinear groups of prime order q , g and g_1 are two different generators of G_0 . In addition, let $e: G_0 \times G_0 \rightarrow G_1$ denote the bilinear map. We also define the Lagrange coefficient $\Delta_{i,s}$ for $i \in Z_q$ and a set S , the elements in Z_q : $\Delta_{i,s}(x) = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$. We will additionally employ a hash function $H: \{0,1\}^* \rightarrow G_0$ that will model as a random oracle. The function will map any attribute described as a binary string to a random group element. The proposed oriented user group verifiable ABE scheme is shown as follows:

• Setup

- **Authority:** The authority CA selects $\alpha, \beta \in_R Z_q$, computers $h = g^\beta, Y = e(g, g)^\alpha$, publishes public key $PK = (G_0, G_1, q, g, g_1, h, Y)$, and the master key is $MK = (\beta, g^\alpha)$.
- **User u_i :** The user u_i selects $x_i \in_R Z_q^*$ as his private key and publishes his public key $y_i = g_1^{x_i}$.

• **Encryption:** DO selects a polynomial $q(x)$ for each node x . These polynomials are chosen in a top-down manner starting from the root node R . For each node x in the tree, the degree d_x of the polynomial q_x is set to be less than the threshold value k_x of the node, that is $d_x = k_x - 1$. Starting with the root node R , DO chooses $s \in Z_q$ and sets $q_R(0) = s$. Then, he chooses d_R other points randomly to define it completely. For any other node x , he sets $q_x(0) = q_{parent(x)}(index(x))$ and chooses d_x other points randomly to completely define q_x . Let Y_L be the set of leaf nodes in $\mathbf{T}$. Then the ciphertext is:

$$CT = \langle \mathbf{T}, \tilde{C} = MY^s, C' = h^s, \forall y \in Y_L : C_y = g^{q_y(0)}, C_y' = H(attr(y))^{q_y(0)} \rangle.$$

• **KeyGen:** CA selects a polynomial $f(x) = \sum_{j=0}^{t-1} a_j x^j$, $a_j, r, r_{ij} \in Z_q$, $a_0 = r (j=1, 2, \dots, t-1)$. Commitment $C_j = g_1^{a_j} (j=0, 1, \dots, t-1)$ and the private key shares $X_i = g^{f(i)}, Y_i = y_i^{f(i)} (i=1, 2, \dots, n)$. Then, CA selects $\omega_i \in Z_q$ randomly and computes $a_{1i} = g^{\omega_i}, a_{2i} = y_i^{\omega_i}, c = H(X_1 \parallel \dots \parallel X_n \parallel Y_1 \parallel \dots \parallel Y_n \parallel a_{11} \parallel \dots \parallel a_{1n} \parallel a_{21} \parallel \dots \parallel a_{2n})$. The response for each u_i is $r_i = \omega_i - f(i)c \bmod q$.

CA publishes $(C_j, Y_i, PROOF_{au} = (c, r_1, \dots, r_n))$, sends $SK_i = (D = g^{(\alpha+r)/\beta}, \forall j \in A_i, D_{ij} = g^{f(i)} \cdot H(j)^{r_{ij}}, D_{ij}' = g^{r_{ij}} / g_1^{f(i)})$ to u_i , A_i is the attribute set of u_i .

• **Verification:** Each participant u_i computes X_i with C_j , using y_i, Y_i, r_i, c and $a_{1i} = g^{r_i} X_i^c$, $a_{2i} = y_i^{r_i} Y_i^c$ ($1 \leq i \leq n$), to verify $c = H(X_1 \parallel \dots \parallel X_n \parallel Y_1 \parallel \dots \parallel Y_n \parallel a_{11} \parallel \dots \parallel a_{1n} \parallel a_{21} \parallel \dots \parallel a_{2n} \parallel)$. If it returns *True*, then the private key is correct.

• **Decrypt:**

Reconstruction: u_i update $d_{ij} = D_{ij} \cdot S_i = g^{r_{ij}}$, where $S_i = Y_i^{x_i^{-1} \bmod q} = g_1^{f(i)}$. If x is a leaf node, then $DecryptNode(CT, SK_i, x) = \frac{e(D_{ij}, C_x)}{e(d_{ij}, C_x)} = e(g, g)^{f(i)q_x(0)}$, if x is a non-leaf node, we recursively compute the $DecryptNode(CT, SK_i, x)$ until output $PA_i = \prod e(g, g)^{f(i)q_x(0)\Delta_{i,S}(0)} = e(g, g)^{f(i)q_R(0)} = e(g, g)^{f(i)s}$, where $i \in S \subseteq \{0, 1, \dots, num_R\}$. $DLEQ(g_1, y_i, S, Y_i, x_i)$ is carried out, S_i and $proof_{U_i}$ are published, and then S_i is verified to ensure that it is consistent.

Decryption: If at least t participants u_i produce correct values for S_i and obtain PA_i , then $A = \prod PA_i^{\Delta_{i,S}(0)} = e(g, g)^{rs}$ ($i \in S' \subseteq \{1, 2, \dots, n\}, |S'| = t$), $M = \tilde{C} / (e(C', D) / A)$.

3.2. Correctness

Now given the correctness verification:

$$d_{ij} = D_{ij} \cdot S_i = g^{r_{ij}};$$

$$DecryptNode(CT, SK_i, x) = \frac{e(D_{ij}, C_x)}{e(d_{ij}, C_x)} = \frac{e(g^{f(i)} \cdot H(j)^{r_{ij}}, g^{q_x(0)})}{e(g^{r_{ij}}, H(attr(x))^{q_x(0)})} = e(g, g)^{f(i)q_x(0)};$$

$$PA_i = \prod e(g, g)^{f(i)q_x(0)\Delta_{i,S}(0)} = e(g, g)^{f(i)q_R(0)} = e(g, g)^{f(i)s};$$

$$A = \prod PA_i^{\Delta_{i,S}(0)} = \prod e(g, g)^{f(i)s\Delta_{i,S}(0)} = e(g, g)^{rs};$$

$$\tilde{C} / (e(C', D) / A) = MY^s / (e(h^s, g^{(\alpha+r)/\beta}) / e(g, g)^{rs}) = M.$$

4. Proof of Security

Theorem 1. An OG-ABEv scheme is verifiable if a PPT algorithm exists to generate secret keys and verification information, all shares are correct, and the user group could reconstruct the secret with probability 1.

Proof: In OG-ABEv, Schoenmakers' scheme, a classical non-interactive PVSS scheme, security of this scheme was proved and discussed in [15].

In OG-ABEv, CA runs *Setup* to generate secret keys SK_i and verification information $(C_j, Y_i, PROOF_{au})$, $i = 1, 2, \dots, n$. The private keys are verified by computing $c = H(X_1 \parallel \dots \parallel X_n \parallel Y_1 \parallel \dots \parallel Y_n \parallel a_{11} \parallel \dots \parallel a_{1n} \parallel a_{21} \parallel \dots \parallel a_{2n} \parallel)$, and each user obtains the correct share. Then, $e(g, g)^{f(i)s}$ is calculated by each participant through interpolation. In what follows, t users provide their shares to calculate $e(g, g)^{rs}$ and decrypt the right message.

Theorem 2. Suppose the DBDH assumption holds. Then no polynomial adversary group can break the scheme of Section 3.1 with a challenge access tree T^* .

We can reduce the DBDH problem to the problem of breaking our encryption scheme, that is to say we are given $\langle A = g^a, B = g^b, C = g^c \rangle$ and asked to distinguish $e(g, g)^{abc}$ from $e(g, g)^\theta$ for a random $\theta \in Z_q$. We assume that an adversary group exists, which can break the security properties of our OG-ABE scheme as defined in Section 2.2. We want to show that, even given a challenge encryption, $M_b = \tilde{C} / e(g, g)^{\alpha s}$ is indistinguishable from a random message. We are basing our reduction on the DBDH assumption. Therefore, we implicitly set $\alpha s = abc$. We need to output $e(g, g)^\alpha$ and h^s as part of public key. Thus, we implicitly set $\alpha = ab$ and $s = c$. The adversary group is allowed to request secret keys for a given user group as long as at least $n - t + 1 (t > \frac{n}{2} + 1)$ users whose attributes do not satisfy T^* . That means, in the worst case, only $t - 1$ users in the user group will be able to compute d_{ij} and $PA_i = e(g, g)^{f(i)s}$. We need $(e(C', D) / \prod PA_i^{\Delta_{i,s}(0)}) = e(g, g)^{\alpha s}$ to be something that the adversary cannot compute.

Proof: Suppose the adversary group A_G can win the IND-sAtt-CPA game with a non-negligible advantage ε . We show how to use the adversary group A_G to build a simulator B that is able to solve the DBDH assumption with advantage $\frac{\varepsilon}{2}$. Before the game starts, the challenger sets the groups G_0 and G_1 with an efficient bilinear map e and generator g of group G_0 , and selects $a, b, c, \theta \in Z_q$. The challenger flips a fair binary coin μ and sets $Z_\mu = e(g, g)^{abc}$ if $\mu = 0$ and $Z_\mu = e(g, g)^\theta$ otherwise. The challenger gives to simulator B the DBDH challenge: $(g, A, B, C, Z_\mu) = (g, g^a, g^b, g^c, Z_\mu)$ for random a, b, c, θ . B will act as A_G 's challenger in the IND-sAtt-CPA game as follows:

Init. The adversary group $A_G = \{Ad_1, Ad_2, \dots, Ad_n\}$ chooses the challenge access tree T^* and gives it to the simulator.

setup. The simulator implicitly sets $\alpha = ab + a'$ by letting $e(g, g)^\alpha = e(g, g)^{ab+a'} = e(A, B)e(g, g)^{a'}$, $g_1 = g, h = g^\beta = g^b = B$.

Phase 1.

Private Key Query: The adversary group A_G ensures that any private key request for the Ad_i does not satisfy T^* . On each request, the challenger chooses a random variable $k_i \in Z_q^*$ and sets $y_i = g_1^{k_i} = g^{k_i}$.

Attribute Secret Key Query: The adversary group A_G ensures that attribute secret key requests for any attribute sets $attr_i = \{a_{ij} \mid a_{ij} \in \Omega\}$ with the restriction that $a_{ij} \notin T^*$. For all $a_{ij} \in attr_i$, it chooses random $t_j, r_{ij} \in Z_q^*$ and set $H(attr(a_{ij})) = g^{t_j}$. It chooses random $v \in Z_q^*$ and implicitly sets $r = f(0) = ab + av$, chooses $t - 1$ random points v_j and $D = g^{(2ab+a'+av)/b}$. Since $D = g^{(2ab+a'+av)/b} = g^{(ab+a'+ab+av)/b} = g^{(\alpha+r)/\beta}$. If $a_{ij} \in T^*$, $f(i)$ is computable and $D_{ij} = g^{f(i)} \cdot g^{t_j r_{ij}}$, $D'_{ij} = g^{r_{ij}} / g^{f(i)}$; If $a_{ij} \notin T^*$, it will implicitly set $f(i) = av_i$ and $D_{ij} = g^{av_i} \cdot g^{t_j r_{ij}}$, $D'_{ij} = g^{r_{ij}} / g^{av_i}$. The simulator B sends to Ad_i : $SK_i = (D, \forall a_{ij} \in attr_i : D_{ij}, D'_{ij})$.

Challenge: The adversary group A_G submits two challenge messages M_0, M_1 . The simulator flips a fair binary coin b and returns the encryption of M_b . M_b is encrypted as follows:

$$C' = h^s = h^c, \tilde{C} = M_b Y^s = M_b e(g, g)^{as} = M_b Z_\mu e(g^c, g^{a'}),$$

$$\text{For } a_{ij} \in \mathbf{T}^*, \text{ the root is } p(0) = s = c, \lambda_j \in \mathbf{T}_q^*: C_{ij} = g^{\lambda_j}, C'_{ij} = g^{t_j \lambda_j}.$$

The ciphertext $CT^* = (\mathbf{T}^*, \tilde{C}, C', \forall a_{ij} \in \mathbf{T}^*: C_{ij}, C'_{ij})$ is sent to A_G .

Phase 2: Ad_i acts exactly as it did in Phase 1.

Guess: A_G outputs a guess $b' \in \{0, 1\}$.

If $b' = b$, the simulator B will guess that $\mu = 0$ and $Z_\mu = e(g, g)^{abc}$; otherwise, it will guess that $\mu = 1$ and $Z_\mu = e(g, g)^\theta$. When $Z_\mu = e(g, g)^{abc}$ the simulator B provides the perfect simulation, and CT^* is a valid ciphertext. Therefore, the advantage of the adversary is $\Pr[b' = b \mid Z_\mu = e(g, g)^{abc}] = \frac{1}{2} + \varepsilon$.

If $\mu = 1$, then $Z_\mu = e(g, g)^\theta$ and CT^* is random ciphertext for the adversary and the adversary does not gain information about M_b . Hence we have: $\Pr[b' \neq b \mid Z_\mu = e(g, g)^\theta] = \frac{1}{2}$. Since the simulator B guesses $\mu' = 0$ when $b' = b$ and $\mu' = 1$ when $b' \neq b$, the overall advantage of B to solve DBDH assumption is: $\frac{1}{2} \Pr[\mu' = \mu \mid \mu = 0] + \frac{1}{2} \Pr[\mu' = \mu \mid \mu = 1] - \frac{1}{2} = \frac{\varepsilon}{2}$.

5. Discussion and Comparison

5.1. Discussion

The most important property that every CP-ABE scheme must have is to resist collusion. In our scheme, this means that different users in one group cannot combine their secret keys and decrypt an encrypted message. To prevent collusion, the *KeyGen* algorithm of our scheme generates a different value $f(i)$ for each user, keys generated for different users cannot be combined since they are different. To decrypt the message the attacker must know how to recover $e(g, g)^{rs}$, to do that the attacker must first recover $e(g, g)^{f(i)s}$, which would require the attacker to have the secret share blinded with the same value $f(i)$.

In the ABE scheme, a key verification mechanism is introduced. Constructing verifiable ABE scheme is important. In the verifiable ABE system, CA generates the keys and key authentication information, then distributes to users. After a user receives the attribute keys, he can verify the correctness of the received keys based on the authentication information; the user can also verify the correctness of the master key. Thus, a supervision mechanism on CA is established. The mechanism ensures that users can effectively inspect the cheating of CA or error caused by communication. Meanwhile, when recovering the shared secret information, each participant can check the right secret shares of others and can verify whether u_i has x_i while not gaining x_i himself. In the verification process, the use of non-interactive zero knowledge method reduces the communication cost between CA and users as well as between users.

In the **KeyGen** phase, the secret value r whereas $R = g_1^r$ is shared among the user group. The user receives $Y_i = y_i^{f(i)}$ encrypted by his private key x_i instead of $g_1^{f(i)}$.

Moreover, the scheme is homomorphic, anyone can check the consistent of γ_i . All these enhance the security of system. With Lagrange interpolation used as mathematical basis, the structure is simple, and the application is convenient. Moreover, the amount of storage of each participant is very small. Aside from the private key x_i and attribute key sk_i , no other storage information is added.

In our scheme, the access tree structure will be changed into an n -ary tree with only one level. Each leaf node represents an attribute, the interior nodes represent the AND, OR operators. If the symbol is $\wedge$ and its child nodes are unassigned, we assign a random value s_i , $1 \leq s_i \leq q-1$, and to the last child node, we assign the value $s_l = s - \sum_{i=1}^{l-1} s_i$, this node is then marked as assigned. If the symbol is $\vee$, then the values of each node are set to be s , this node is then marked as assigned. Thus, in the encryption and decryption processes, the computation cost of secret s is reduced as it is shared or reconstructed. Obviously, the scheme reduces algorithm complexity, reduces space consumption, and improves efficiency.

5.2. Comparison

In the general ABE scheme, the correctness of private keys is solved as follows: Upon receiving a private key, a user encrypts a sample plaintext with a predicate that is satisfied by the user's attributes, and decrypts it with the private key he received. If it is decrypted successfully, then he can be sure that the key is correct. By contrast, encryption and decryption do not need to be carried out in our scheme, which not only saves unnecessary computational load but also verifies the correctness of various keys through calculation.

Tangs[17] also proposed a verifiable ABE scheme. They constructed two verifiable ABE schemes, one with a single authority and the other with authorities. We then compare Tang's scheme and our proposed scheme in different aspects.

- **Confidentiality and security.** In our scheme, we introduced a user group that offers high security and confidentiality, especially in highly confidential applications. Our scheme can address problems such as excessive power concentration, whereas Tang's scheme is unable to do so.

- **Check the correctness.** Tang's approach can only check the correctness of the attribute key. By contrast, our scheme can verify both the correctness of the attribute key and the honesty of the participants.

- **Verification.** In the **verification** phase, we need only $4n$ modular exponentiation in our scheme, n is the user number in user group U . The user group is used to decrypt highly sensitive information, users who obtain permissions are rare, which is why n will not be great. However, Tang's scheme used the access tree. Therefore, verification efficiency was low. Suppose that an access tree exists, in this access tree, the level number is L , the mid node number is N_{mid} and the leaf node number is N_{leaf} . Therefore, from the leaf node to the root node, one user must carry out at least $N_{mid} + N_{leaf} - 1$ bilinear pairings, the number of n users' is $n(N_{mid} + N_{leaf} - 1)$. Therefore, the computation cost of Tang's scheme is larger than that of our scheme.

6. Conclusion

We give the definition of OG-ABEv and introduce a user group in this scheme, which enhances confidentiality of shared data, weakens user permissions and prevents abuse of rights, particularly in confidential applications, for example: missile control and emission, because of the tremendous damage, one manager own cannot do it, so there must be at least t required managers.

OG-ABEv also allows the user to check the correctness of the attribute key and verify the honesty of other participants. Performing such a verification reduces dependence on CA, so we can select a semi-trusted or untrusted CA for our scheme. We then develop a proper security model and present the concrete construction of OG-ABEv. Finally, we prove the security of OG-ABEv under the security model.

Acknowledgments

The work was supported by the NNSF of China under grant (61262057, 71263045); The Ministry of education of Humanities and social science research projects (12YJCZH282); The Ministry of education program for New Century Excellent Talents (ncet-12-0620); The Northwest Normal University plans of enhancing the research of young teachers (NWNLU-LKQN-14-1).

References

- [1] A. Sahai, B. Waters, "Fuzzy identity based encryption", Advances in Cryptology (EUROCRYPT 2005). Lecture Notes in Computer Science (3494), Berlin: Springer-Verlag, (2005), pp.457-473.
- [2] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data", Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, ACM, (2006), pp.89-98.
- [3] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE", Proceedings of the 14th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, ACM, (2007), pp.456-465.
- [4] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute-based encryption", IEEE Symposium on Security and Privacy, 2007- SP'07, IEEE, (2007), pp.321-334.
- [5] B. Waters, "Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization", Public Key Cryptography-PKC 2011, Springer Berlin Heidelberg, (2011), pp.53-70.
- [6] K. Emura, A. Miyaji, A. Nomura, K. Omote and M. Soshil, "A ciphertext-policy attribute-based encryption scheme with constant ciphertext length", Information Security Practice and Experience. Springer Berlin Heidelberg, (2009), pp.13-23.
- [7] C. Cheng, Z. Zhang, D. Feng, "Efficient ciphertext policy attribute-based encryption with constant-size ciphertext and constant computation-cost", Provable Security, Springer Berlin Heidelberg, (2011), pp.84-101.
- [8] N. Attrapadung, B. Libert and E. De Panafieu, "Expressive key-policy attribute-based encryption with constant-size ciphertexts", Public Key Cryptography-PKC 2011, Springer Berlin Heidelberg, (2011), pp. 90-108.
- [9] S. Agrawal, X. Boyen, V. Vaikuntanathan, P. Voulgaris and H. Wee, "Functional encryption for threshold functions (or fuzzy IBE) from lattices", Public Key Cryptography-PKC 2012, Springer Berlin Heidelberg, (2012), pp.280-297.
- [10] M. Chase, "Multi-authority attribute based encryption", Theory of Cryptography, Springer Berlin Heidelberg, (2007), pp.515-534.
- [11] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption", Proceedings of the 16th ACM Conference on Computer and Communications Security, ACM, (2009), pp. 121-130.
- [12] A. Lewko and B. Waters, "Decentralizing attribute-based encryption", Advances in Cryptology-EUROCRYPT 2011, Springer Berlin Heidelberg, (2011), pp.568-588.
- [13] M. Piretti, P. Traynor, P. McDaniel, P. McDaniel and B. Waters, "Secure attribute-based systems", ACM Conference on Computer and Communications Security-CCS 2006, (2006), pp. 99-112.
- [14] A. Beumel, "Secure Schemes for Secret Sharing and Key Distribution", ThD thesis, Israel Institute of Technology, Technion, Haifa, Israel, (1996).
- [15] J. B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting", Advances in Cryptology-Crypto'99 Proceedings, Berlin: Springer-Verlag, (1999), pp.148-164.
- [16] J. D. Chaum and T. Pedersen, "Wallet data based with observers", CRYPTO'92, Berlin Heidelberg: Springer-Verlag, (1992), pp.89-105.
- [17] Q. Tang and D. Ji, "Verifiable Attribute Based Encryption", International Journal of Network Security, Vol.10, No.2, (2010), pp.114-120.

Authors

Xueyan LIU, she received the B.S. and M.S. degrees in Northwest Normal University, in 2001 and 2004, respectively. During 2001-2012, she stayed in nwnu to work. She now study in Lanzhou University of technology. Now, the main research direction for the cryptographic and information security.

Zhanming LI, he graduated in 1984 from the Department of electronic engineering, Xi'an Jiao Tong University. Obtained M.S. degree in the UK STRATHCLYDE University in 1987 August. 1987 September so far in telecommunications, control theory and control engineering of Lanzhou University of Technology, engaged in teaching and scientific research. At present, the main research direction for the modeling and control of complex system.