# Efficient and Secure Group Key Management Based on EBS and Attribute Encryption

Yanli Chen
College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, China
Key Lab of Network Security and Cryptology, Fujian Normal University, FuZhou, China
Email:chenyl@njupt.edu.cn

Geng Yang
Department of technology, Nanjing University of Posts and Telecommunications, Nanjing , China
Email:yangg@njupt.edu.cn

*Abstract*—**Exclusion Basis Systems (EBS) is a combinatorial optimization methodology for key management of group communication. The EBS approach proves to be very scalable for large networks and enables great flexibility in network management. But it is highly vulnerable to collusion attacks. In this paper, a novel secure group management scheme based on EBS and attribute encryption is proposed. Our proposed scheme provides group forward/backward secrecy, and it is resilience to colluding attacks. Moreover, compared to some previous approaches, performance evaluation shows that our scheme is more efficient in communication (*O*(*m*) for single leave), storage (*O*(*k*) for each group member), and computation (*O*(*k*) for each group member), where *k and m* are the number of the attributes that each member holds or not. As *k* and *m* could be relatively small even in large-scale systems, so as to our new scheme works well in large-scale applications.**

***Keywords-group key management; exclusion basis systems; attribute-based encryption; collusion resistant***

## I. INTRODUCTION

Multicast communication refers to the transmission of a message from one sender to multiple receivers or from multiple senders to multiple receivers. Nowadays multicast services have been widely deployed for applications such as video conference, real-time information services, pay per view, and distance learning[1]. One of the most important issues in multicast security is the group key management. There are various proposed schemes for group key management with their own advantages and disadvantages.

Mohamed Eltoweissy et al.(2004) proposed Exclusion Basis Systems (EBS)[2]. EBS outperforms other key management schemes in terms of storage and communication overheads. But it suffers from the collusion problem, which means a small number of members may collude and collectively reveal all the network keys. In order to resolve the problem, EBS-based schemes with collusion-resistance have been proposed recently in references [3]-[8]. But as far as we know, these schemes still cannot solve the problem completely, only to reduce the possibility of the collusion attack. Younis et al.[3] presented a collusion resisted EBS-based scheme called SHELL. It performs location-based key assignment to decrease the number of keys revealed by the collusion of attackers. Mohamed Eltoweissy et al.[4] proposed another EBS-based scheme called LOCK which uses key polynomials to improve network resilience to collusion instead of location-based key assignment as in SHELL. Ma et al.[5] proposed a Location-aware and secret share based dynamic key management scheme supported by EBS, to replace the compromised central node and enhance the security level of the network. WANG et al. [6] presented a EBS-based group key management scheme, which distributes the administrative keys based on the hamming distance and EBS to prevent the adversary uncover all the administrative keys by a few colluding members. Kong et al.[7] presented a discrete particle swarm optimization algorithm for EBS collusion problem. Zhang' et al. [8] proposed a wireless sensor network security architecture based on LOCK. In this architecture, wireless sensor network security upper layer is based on an ID-based public key management algorithm.

The concept of Attribute-Based Encryption (ABE) was introduced by Sahai and Waters[9]. Two variants of ABE were subsequently proposed. In the ciphertext policy variant (CP-ABE)[10-13], each user is associated with a set of attributes and he receives a secret key based on that set. The ciphertext is associated with the access tree and the message sender determines the policy under which the data can be decrypted. A user can decrypt if and only if his attribute set satisfies this access structure. In the key policy variant (KP-ABE)[14], the situation is reversed. Based on CP-ABE, Cheung et al.[15] and Zhou et al.[16] proposed collusion resistant group key management schemes respectively. The schemes enhance the flat table (FT) group key management schemes[17], which are vulnerable to collusion attacks by utilizing the basic construction of CP-ABE. Motivated by them, we propose a secure and efficient key management scheme based on EBS and CP-ABE. If even all evicted users collude, they get no information about the encrypted communications in the group. To the best of our knowledge, this is the first work on EBS-based group key management which can completely resolve collusion problem. Moreover, our scheme is efficient in terms of communication, storage and computation overheads. As the overheads is independent to the group size, so as to our proposed scheme works well in large-scale applications.

The rest of this paper is organized as follows: Section Ⅱ briefly describe the technique preliminaries on which our proposed scheme is designed. Section III presents the specifics of our proposed new group key management scheme. Section IV evaluates the performance of the scheme. Finally, Section Ⅴ concludes the paper and outlines further research.

## II. PRELIMINARIES

**978-1-4244-8728-8/11/$26.00 ©2011 IEEE**

### A. Exclusion Basis Systems (EBS)

An EBS Γ of dimension (*n,k,m*) represents a situation in a secure group. A set of (*k*+*m*) administrative key are used to support a set of *n* members. Each group member (GM) is randomly assigned distinct combination of *k* keys out of the total of *C*(*k*+*m*,*k*). To evict the compromised member, the *m* keys unknown to the member are used to perform rekeying. For detailed description of EBS, please refer to Ref. [2].

Assume that there are 8 members in the group with keys as Table 1.

Table 1. Canonical matrix of EBS(8,3,2)

| | $M_1$ | $M_2$ | $M_3$ | $M_4$ | $M_5$ | $M_6$ | $M_7$ | $M_8$ | $M_9$ | $M_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $K_1$ | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| $K_2$ | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| $K_3$ | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| $K_4$ | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| $K_5$ | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |

Suppose member $M_1$ has been compromised. The following messages will be generated for rekeying:

Message 1: $E_{K_4}(S', E_{K_1}(K_1')),\ E_{K_2}(K_2'),\ E_{K_3}(K_3'))$

Message 2: $E_{K_5}(S', E_{K_1}(K_1')),\ E_{K_2}(K_2'),\ E_{K_3}(K_3'))$

where $E_{K_i}(x)$ denotes encryption of $x$ by key $K_i$, $K_i'$ represents the replacement key for the old key $K_i$ and $S'$ represents the new group key.

Although EBS scheme is efficient, a drawback of the basic EBS-based solution is that a small number of members may collude and collectively reveal all the network keys. In the example above, if $M_1$ and $M_6$ collude, then they may collude to obtain the new group key, no matter how it is encrypted.

### B. Bilinear maps

Our design is based on some facts about groups with efficiently computable bilinear maps.

Let $G_0$, $G_1$ be two multiplicative groups of prime order $p$ and let $g$ be a generator of $G_0$ and $\hat{e}$ be a bilinear map, $\hat{e}$: $G_0 \times G_0 \rightarrow G_1$. The map must satisfy the following properties:

(1)Bilinear: we say that a map $\hat{e}$: $G_0 \times G_0 \rightarrow G_1$ is bilinear if $\hat{e}(u^a,v^b) = \hat{e}(u,v)^{ab}$ for all $u,v \in G_0$ and all $a, b \in Z_p$.

(2)Non-degenerate: $\hat{e}(g,g) \neq 1$,The map does not send all pairs in $G_0 \times G_0$ to the identity in $G_1$.

(3)Computable: there is an efficient algorithm to compute $\hat{e}(u,v)$ for any $u,v \in G_0$.

### C. Ciphertext-Policy Attribute-Based Encryption

An CP-ABE scheme consists of four fundamental algorithms: Setup, Encrypt, KeyGen and Decrypt. We describe these algorithms at an abstract level, for detailed description of CP-ABE, please refer to Ref. [10].

**Setup** This algorithm takes as input the security parameter $\kappa$ and returns a public key PK and a master secret key MK.

**KeyGen** This algorithm takes as input the public key PK, the master key MK, and a set of attributes *S* with the user. It returns a secret key SK associated with *S*.

**Encrypt** This algorithm takes as input the public key PK, a message *M* and an access structure *T*. It returns a ciphertext CT

**Decrypt** This algorithm takes as input a ciphertext CT and a secret key SK. It returns the message *M* if *S* satisfies *T*, where *S* is the attribute set used to generate SK.

## III. THE PROPOSAL SCHEME

This section presents our new proposal for group key management. We assume there is a group controller (GC) in the network that is the initiator of every multicast group. The GC is a trusted party. Each group member (GM) is neither trustworthy to the GC nor between themselves. The underlying concept of our proposal is EBS. But we rely on CP-ABE to carry out rekey operations. The main idea is to associate each GM with *k* attributes such that the group key is encrypted under certain attributes and only those who own the intended attributes are able to decrypt it. These attributes allow the GC to distinguish current GMs from former/leaving GMs. The GC computes an access structure that is satisfied by the attribute set of every current GM, but not by that of any former/leaving GM. Although two users may share the same attributes, the private key components associated with each attribute are distinct. Moreover, private key components belongs to different user is incompatible and thus cannot be used together. Thus, collusion attacks are prevented.

### A. System Setup

On system setup, the GC will execute the following steps:

(1)Select two cyclic groups $G_0$ and $G_1$ of prime order $p$ as well as a bilinear map $\hat{e}$: $G_0 \times G_0 \rightarrow G_1$. Let $g$ be the generator of $G_0$. Choose a random $\alpha \in Z_p$ and a hash function $H$:$\{0,1\}^* \rightarrow G_0$.

The public key is PK=$\{g, H, \hat{e}(g, g)^{\alpha}\}$ and the master key is MK=$\{\alpha\}$. MK is only known to the GC.

(2)According to the group size, the GC decides on the suitable parameters *k* and *m*. Then it constructs EBS(n,k,m) employing a canonical enumeration of all possible ways of forming subsets of *k* objects from a set of *k* + *m* objects. Thus, each group member has a valid bit string ID=$X_{k+m}X_{k+m-1}...X_1$, with *k* 1's and *m* 0's and a attribute set $S=\{A_i=\text{“}K_i\text{”} \mid \forall\ X_i=1, i \in Z_{k+m}\}$ stored at it. For example, showed in Fig.1 new member $M_1$ with ID= “00111” possesses a set of attributes $S=\{A_1,A_2,A_3\}$.

(3)The GC constructs the private key for each group member according to his attribute set *S*. It first chooses a random $r \in Z_p$, and random $r_i \in Z_p$ for each attribute $i \in S$. Then it computes the private key SK= $<D = g^{\alpha - r}, \forall\ i \in S: D_i = g^r \bullet H(i)^{r_i}, D_i' = g^{r_i}>$.

### B. Join

When a set of members, denoted by *J* want to join the communication group, the following steps are executed:

(1)The joining members first establish a secure unicast connection with the GC who checks whether each member is authorized to join.

(2)If the checks succeed, according the theorem in Ref. [2], the GC create a new column in the EBS table,

assigns each accepted member $a \in J$ a valid bit string ID=$X_{k+m}X_{k+m-1}...X_1$, with $k$ 1's and $m$ 0's and a attribute set $S_a$={$A_i$="$K_i$"| $\forall$ $X_i$=1, $i \in Z_{k+m}$}, distinct from the former ones.

(3)To preserve group backward secrecy, the GC first selects a new group key $K' \in G_1$ at random and multicasts $\{K'\}_K$. Upon receiving the multicast message, current GMs decrypt and update the new group key $K'$.

(4)The GC constructs the private key $SK_a$= $<D = g^{\alpha - r}, \forall i \in S_a$: $D_i = g^r \bullet H(i)^{r_i}$, $D_i' = g^{r_i} >$ for each accepted member $a \in J$ according to his attribute set $S_a$.

(5)Finally, the GC sends the group key $K'$ and the private key to each GM $a \in J$ through a secure channel.

### C. Leave

Let $L$ denote the set of leaving members, and $l$= $|L|$ denote the number of leaving members. The GC first constructs the access structure $T$. If $l$ is small, a three-level access structure $T$ is constructed as follows: the top level is an AND gate, and the number of children is $l$. Each child representing a leaving member is a two-level subtree connected by an OR gate, whose leaves are $m$ attributes not held by the leaving member. For example the leaving members are $M_1$ with ID '00111' and $M_6$ with ID '11001', the attributes not held by the leaving members are $\{A_4,A_5\}$ and $\{A_2,A_3\}$. The access structure $T$ is showed in Fig. 1(a).

On the other hand, if there is a large number of leaving members, a three-level access structure $T$ is constructed as follows: the top level is an OR gate and the number of children is $n$, where $n$ is the number of remaining GMs. Each child representing a remaining member is a two-level subtree connected by an AND gate, whose leaves are $k$ attributes held by the remaining member. For example the remaining members are $M_1$ with ID '00111' and $M_6$ with ID '11001', the attributes held by each remaining member are $\{A_1,A_2,A_3\}$ and $\{A_1,A_4,A_5\}$. The access structure $T$ is showed in Fig. 1(b).

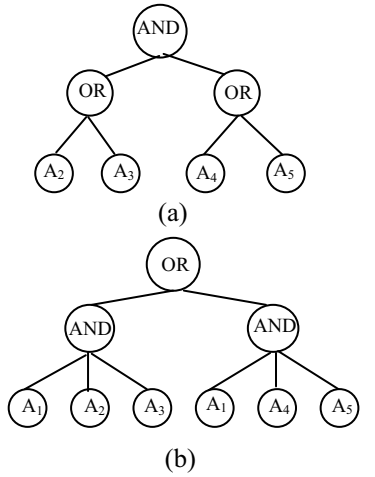


Figure1. Three-level access structure $T$

Then the GC select at random a new group key $K' \in G_1$, encrypt the new group key $K'$ under the tree access structure $T$. As the access structure used in our scheme is restricted to AND, OR gates only, we present a new algorithm for realizing efficient CP-ABE without using Shamir's threshold secret sharing. Referred to the scheme of Luan Ibraimi et al.[11], each node $x$ of the tree $T$ has a secret share value $q_x$. Starting with root node $R$, the algorithm chooses a random element $s \in Z_p$ and sets $q_R$=$s$, mark all non-leaf nodes as un-assigned. Recursively, for each un-assigned non-leaf node do the following:

(a)If the node $x$ is an AND gate, we use a unanimous consent control by modular addition scheme to assign a value to each child node. To do that, for each child node except the last one, set the secret share value $q$ of the child node a random value $s_i$ where $1 \le s_i \le p$-1. To the last child node $t$ assign the value $q_t$=$q_x$- $\sum_{i=1}^{t_x-1} s_i$ mod $p$, where $t_x$ is the number of children of the node $x$. Mark this node $x$ assigned.

(b)If the node $x$ is an OR gate, set the secret share value $q$ of each child node to be $s_x$. Mark this node assigned.

In Fig. 2, we show an example of assigning secret shares to the access tree $T$ showed in Fig. 1(a). Let $Y$ be the set of leaf attributes in $T$. The ciphertex is then constructed. CT=$<T, \tilde{C}=K\hat{e}(g,g)^{\alpha s}$, $C = g^s$, $\forall j \in Y : C_y = g^{q_j}$, $C_y' = H(j)^{q_j} >$.

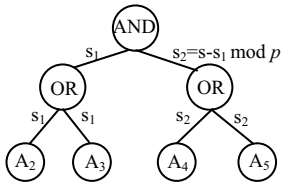


Figure2. Assigning secret shares

On receiving CT, GMs run the decryption algorithm if the attribute set $S$ satisfy $T$. First, chooses the smallest set $S' \subseteq S$ (we assume that this can be computed efficiently by the decryptor). For every attribute $j \in S'$, compute

$$\prod_{j \in S'} \frac{\hat{e}(D_j, C_j)}{\hat{e}(D_j', C_j')}$$
$$= \prod_{j \in S'} \frac{\hat{e}(g^r \cdot H(j)^{r_j}, g^{q_j})}{\hat{e}(g^{r_j}, H(j)^{q_j})}$$
$$= \prod_{j \in S'} \hat{e}(g,g)^{rq_j}$$
$$= \hat{e}(g,g)^{rs}$$

Then compute

$$\hat{e}(C_0, D_0) \bullet \hat{e}(g,g)^{rs}$$
$$= \hat{e}(g^s, g^{\alpha - r}) \bullet \hat{e}(g,g)^{rs} = \hat{e}(g,g)^{\alpha s}$$

Now the new group key $K'$ is computed as follows:

$$\tilde{C} / \hat{e}(g,g)^{\alpha s}$$
$$= K' \ \hat{e}(g,g)^{\alpha s} / \hat{e}(g,g)^{\alpha s}$$
$$= K'$$

### D. Private key update

For perfect backward and forward secrecy, we should update all the remaining GMs' private keys after GMs

leave or at regular intervals. The GC chooses a random $\alpha' \in Z_p$. The public key and master key are updated as $PK'=\{g,H,\hat{e}(g, g)^{\alpha'}\}$, and $MK'=\{\alpha'\}$. Then the GC multicasts message $\{g^{(\alpha'-\alpha)}\}_K$. Each current GM can decrypt the message to update the secret key because he knows $K$. The new secret key is calculated as $SK'=<D\cdot g^{(\alpha'-\alpha)} = g^{(\alpha'-r)}, \forall i \in S_a: D_i = g^r \bullet H(i)^{r_i}, D_i' = g^{r_i}>$. The old key SK is securely erased.

Note that only the first component of SK is changed, therefore $SK'$ is a valid secret key in the new system for the same attribute set. Moreover, for $\alpha' \neq \alpha$, $SK'$ is not a valid secret key in the system parameterized by $\alpha$. Thus, SK' cannot be used to decrypt past rekey messages.

## IV. SCHEME EVALUTAION

### A. Security analysis

Our proposed scheme provides the following security properties:

(1)Collusion Resistance

The encryption of the private key is provably secure under the DBDH assumption. For each GM, $r$ is randomly and independently selected from $Z_p$. Although two users may share the same attributes, the private key components associated with each attribute are distinct. The secret key from one GM does not give the other GMs any help in terms of computing $e(g, g)^{rs}$. This turns out that the adversary is not able to decrypt the ciphertext and get the new group key unless he owns the intended attribute. Thus, collusion attacks are prevented.

(2)Forward Secrecy

When GMs leave the group, we have seen that leaving GMs cannot recover the new group key $K'$, even if they collude. Also as is described in the previous section, our proposed scheme is able to update the private key for legitimate members. In the private key update procedure above, $\alpha'$ is generated randomly. Therefore leaving GMs cannot decrypt future encrypted messages since the decrypting parameter $D$ is changed. Therefore, perfect forward secrecy is satisfied.

(3) Backward Secrecy

When new GMs join the group, a new random group key $K'$ is encrypted $\{K'\}_K$, and then distributed. Also as a private key update procedure will be performed after GMs leave, the private keys of joining GMs are generated under new versions of $\alpha'$. Given 1) randomness of $K'$ and $\alpha'$; 2) security of symmetric encryption and CP-ABE, new GMs cannot decrypt either type of messages from the past; hence group backward secrecy is satisfied.

### B. Performance Evaluation

This section evaluates the performance of our proposed scheme in terms of storage, communication and computation overheads. We denote the maximum number of GMs to be $N$, the number of current GMs to be $n$, the number of leaving GMs to be $l$. In original EBS schemes, $k$ is the number of administrative keys that each member is assigned, and $m$ is the number of keys unknown to the member, which are used to perform rekeying. In our new scheme, $k$ and $m$ are the number of attributes that each member holds or not.

**(1)Storage Overhead**

In our proposed scheme, the GC stores (i) group key (ii) public key and master key (iii) the list of current GMs. Since the public key and master key are of constant size, the GC's storage is dominated by $n$. Each GM stores a group key and its secret key which consists of $k$ secret key component on $G_0$.Thus the GM's storage overhead is $O(k)$.

**(2)Communication Overhead**

Our discussion focuses on the complexity of leave operation. In our proposed scheme, the size of message linearly depends on the number of leafs, i.e., attributes in the access control policy tree. We construct different policy trees according to the number of leaving members to reduce the number of leaves in the access control tree. If the number of leaving numbers is small, the message complexity is $O(l\cdot m)$. Otherwise the complexity e is $O(n\cdot k)$. For single leave, the complexity is $O(m)$.

(3)**Computation Overhead**

Our discussion also only focuses on the complexity of leave operation. The encryption computation overhead is dominated by the number of attributes in the access control policy tree, and the decryption computation overhead is dominated by the number of attributes that each member holds. The GC requires $m\cdot l$ or $k\cdot n$ operation on $G_0$. GMs requires $2k+1$ pairing operations, $2k+2$ operations on $G_1$. Thus, the complexities of encryption is $O(m\cdot l)$ or $O(k\cdot n)$, and the complexities of decryption is bounded by $O(k)$.

The summary of performance assessment is presented in Table 1. We compare our proposed scheme against the original EBS scheme and several previous solutions: tree based schemes (e.g., LKH [18], OFT[19] and ELK[20]), flat table scheme using CP-ABE(FTABE) [15,16], and ID-based key distribution(IBKD) scheme[21]. As $k$, $m$ is adjustable and far smaller than $n$ *and* $N$, we can conclude our proposed scheme is not only more flexible, but also more efficient when the size of group is large and the leaving numbers are small.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a novel collusion resistant group key management scheme based on the exclusion basis Systems (EBS) and ciphertext policy attribute encryption. In this approach, each GM is identified by a set of attributes based on bits in the IDs following EBS concept, and is given a secret key that corresponds to his attribute set. As the access tree in our scheme is an n-ary tree represented only by OR and AND gates, we use an efficient CP-ABE to encrypt the new data key to all (and only) remaining members.

The analysis shows that compared with applying exists, our scheme can resolve EBS collusion problem completely. In addition, our scheme is more efficient in terms of communication and computation overhead when the group size is large. So our proposal can be well controlled even in the case of large-scale application scenarios. The future work of this paper would be considered in the following directions: this work is subjected to single failure problem when GC fails, we can investigate decentralized and distributed group management infrastructure to improve the robustness of the scheme.

Table 1. COMPARISON OF PERFORMANCE IN DIFFERENT SCHEMES

| Scheme | Compromise Resistance | Storge | | CommunicationOverhead | Computation Overhead | |
|---|---|---|---|---|---|---|
| | | *GC* | *GM* | *leave* | *GC* | *GM* |
| Proposed scheme | Yes | $O(n+k+m)$ | $O(k)$ | Min($O(l\cdot m)$, $O$(n·k)) | Min($O(l\cdot m)$, $O$(n·k)) | $O(k)$ |
| EBS | No | $O(n+k+m)$ | $O(k)$ | $O(l\cdot m)$ | Symmetric Computation | |
| Tree-based | Yes | $O(n)$ | $O(\log n)$ | $O(l\cdot\log n)$ | Symmetric Computation | |
| FT ABE | Yes | $O(n)$ | $O(\log N)$ | $\approx O(n)$ | $O$(logN) | $O$(logN) |
| IBKD | Yes | $O(n)$ | $O(1)$ | $O(n)$ | $O$(n) | $O$(n) |


## ACKNOWLEDGMENT

This work was supported by the Foundation of Key Lab of Fujian Province University Network Security and Cryptology under Grant No. 5319456069, the National Natural Science Foundation of China under Grant No. 60873231, the Natural Science Foundation of Jiangsu Province under Grant No. BK2009426 and the Major State Basic Research Development Program of China under Grant No.2011CB302903.


## References


[1]S. C. Yu, K. Ren and W. J. Lou, "Attribute-based on-demand multicast group setup with membership anonymity", Computer Networks,Vol. 54,No.3,pp.377–386, 2010.

[2]M. Eltoweissy, M. H. Heydari, L. H. Morales, et al, "Combinatorial optimization of group key management", Journal of Network and System Management, Vol.12,No.1, pp. 33-50,2004.

[3]M. F. Younis, K. Ghumman, M. Eltoweissy, "Location- aware combinatorial key management scheme for clustered sensor networks", IEEE Trans. on Parallel and Distributed Systems. Vol.17, No.8, pp. 865−882,2006.

[4]M. Eltoweissy, M. Moharrum, R. Mukkamala, "Dynamic key management in sensor networks", IEEE Communications Magazine, Vol.44, No.3, pp. 122−130,2006.

[5]C. G. Ma, G. N. Geng and H. Q. Wang, "A location-aware and secret-share based dynamic key management scheme for heterogeneous sensor Networks", Journal of Networks Vol.5, No.4, pp. 500-507, 2010.

[6]W. Wang, W. H. Zhao, F. H. Li, et al., "EBS-based efficient and secure group key management in wireless sensor networks". Journal on Communications,Vol.30, No.9, pp.76-82,2009.

[7]F. R. Kong, C. W. Li, Q. Q. Ding, et al, "Collusion Problem of the EBS-Based Dynamic Key Management Scheme", Journal of Software, Vol.20, No.9, pp. 2531−2541, 2009.

[8]J. Q. Zhang, V. Varadharajan, "A New Security Scheme for Wireless Sensor Networks", Global Telecommunications Conference, 2008. IEEE GLOBECOM, New Orleans, LA, USA, Nov. 30-Dec. 4, pp. 1-4,2008.

[9]A. Sahai, B. Waters, "Fuzzy Identity Based Encryption", Advances in Cryptology, Eurocrypt 2005.volume 3494 of LNCS pages, Springer, pp. 457-473, 2005.

[10]J. Bethencour, A. Sahai, B. Waters, "Ciphertext-policy attribute-based encryption", IEEE Symposium on Security and Privacy, Berkeley, CA , USA, May 20-23, pp. 321–334, 2007.

[11] L. Ibraimi, Q. Tang, P. Hartel and W. Jonker, "Efficient and provable secure ciphertext-policy attribute-based encryption schemes," Lecture Notes in Computer Science. Berlin, Germany: Springer, vol. 5451, pp. 1-12, 2009.

[12]B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization", Cryptology ePrint Archive, Report 2008/290. 2008,http://eprint.iacr.org/.

[13]V. Goyal, A. Jain, O. Pandey, et al. "Bounded ciphertext policy attribute-based encryption", Reykjavik, Iceland, July 6–13 July, pp. 579-591, 2008.

[14]V. Goyal, O. Pandey, A. Sahai, B. Waters, "Attribute Based Encryption for Fine-Grained Access Control of Encrypted Data", ACM conference on Computer and Communications Security (ACM CCS),Alexandria,VA,USA, Oct 30-Nov 3, pp. 89-98, 2006.

[15]L. Cheung, J. A. Cooley, R. Khazan, and C. Newport, "Collusion resistant group key management using attribute-based encryption," Cryptology ePrint Archive, Report 2007/161, 2007, http://eprint.iacr.org/.

[16]Z. Zhou and D. Huang, "BGKM: An Efficient Secure Broadcasting Group Key Management Scheme," Cryptology ePrint Archive: Report 2008/436, http://eprint.iacr.org/.

[17]I. Chang, R. Engel, D. Kandlur, D. Pendarakis, D. et al. Key management for secure lnternet multicast using Boolean function minimization techniques. IEEE INFOCOM , New York, NY, USA, PP. 689-698, Mar. 21-25 , 1999.

[18] D. Wallner, E. Harder, R. Agee, "Key management for multicast: issues and architectures", RFC 2627

[19] D. McGrew, A. Sherman, "Key establishment in large dynamic groups using one way function trees", Technical Report 0755, TIS Labs at Network Associates, 1998

[20] A. Perrig, D. Song, J. Tygar, "ELK: a new protocol for efficient large-group key distribution. Proceedings of the IEEE Symposium on Security and Privacy". Oakland, California, USA, pp. 247-262, May 14 – 16, 2001.

[21]G. Yang, J. T. Wang, H. B. Cheng, et al, "An Identity-Based Encryption Scheme for Broadcasting", IFTP International Conference on Network and Parallel Computing, Dalian, China, pp. 123-126, Sep. 18-21, 2007.