On Efficient Ciphertext-Policy Attribute Based Encryption and Broadcast Encryption

Zhibin Zhou and Dijiang Huang

Arizona State University

Abstract. Ciphertext Policy Attribute Based Encryption (CP-ABE) enforces an expressive data access policy, which consists of a number of attributes connected by logical gates. Only those decryptors whose attributes satisfy the data access policy can decrypt the ciphertext. CP-ABE is very appealing since the ciphertext and data access policies are integrated together in a natural and effective way.
However, all existing CP-ABE schemes incur very large ciphertext size, which increases linearly with respect to the number of attributes in the access policy. Large ciphertext prevents CP-ABE from being adopted in the communication constrained environments. In this paper, we proposed a new construction of CP-ABE, named Constant-size CP-ABE (denoted as CCP-ABE) that significantly reduces the ciphertext to a constant size for an AND gate access policy with any given number of attributes. Each ciphertext in CCP-ABE requires only 2 elements on a bilinear group.
Based on CCP-ABE, we further proposed an Attribute Based Broadcast Encryption (ABBE) scheme. Compared to existing Broadcast Encryption (BE) schemes, ABBE is more flexible because a broadcasted message can be encrypted by an expressive access policy, either with or without explicit specifying the receivers. Moreover, ABBE significantly reduces the storage and communication overhead to the order of $O(\log N)$, where N is the system size. Also, we proved, using information theoretical approaches, ABBE attains minimal bound on storage overhead for each user to construct all possible subgroups in the communication system.

1 Introduction

Research in Ciphtertext Policy Attribute-Based Encryption (CP-ABE) has been a very active area in recent years [1, 10, 5, 19]. Under the construction of CP-ABE, an attribute is a descriptive string assigned to (or associated with) an entity and each entity may be tagged with multiple attributes. Many entities may share common attributes, which allow message encryptors to specify a secure data access policy by composing multiple attributes through logical operators such as "AND", "OR", etc. To decrypt the message, the decryptor's attributes need to satisfy the access policy. These unique features of CP-ABE solutions make them appealing in many systems that require the expressive data access control for a large numbers of users.

Apart from the promising features provided by CP-ABE solutions, there is a major problem of the existing CP-ABE schemes, which usually incur large, linearly increasing ciphertext. In the CP-ABE schemes reported in [1, 5, 19], the size of a ciphertext increases linearly with respect to the number of included attributes. For example, the message size in BSW CP-ABE [1] starts at about 630 bytes, and each additional attribute adds about 250-300 bytes.

In this paper, we propose a novel CP-ABE construction, named *Constant-size Ciphertext Policy Attribute Based Encryption* (CCP-ABE), which incurs constant-size of ciphertext, regardless of the number of attributes in a logical AND data access policy with wildcards. Besides the encrypted message and encoded access policy, each ciphertext only requires 2 bilinear group elements, which are bounded by 300 bytes in total. Moreover, due to the new construction of CCP-ABE, we can prove that the CCP-ABE is CPA secure. To the best of our knowledge, this is the first few such constructions that achieve these properties [1].

Based on presented CCP-ABE, we further provide a new construction named as *Attribute Based Broadcast Encryption* (ABBE) that supports efficient Broadcast Encryption (BE). In existing BE schemes, e.g., [3], a broadcaster encrypts a message for an specified set of receivers who are listening on a broadcast channel. Each receiver in the specified set can decrypt the message while all other receivers that are not in the specified set cannot decrypt even though they collude together. However, in a system with large number of users, identifying every decryptor may be impractical. For example, to broadcast a message to all CS students, the encryptor needs

[1] We note that Herranz et al. has independently proposed more general construction of CP-ABE with constant ciphertext. More comprehensive comparison between our construction and this construction will be reported in Section 2

to query a central directory to get the contact information from every CS student in the roaster, in which the operation could be very expensive and time consuming.

Using ABBE, an encryptor has the flexibility to encrypt the broadcasted data using CCP-ABE, either with or without the information of each intended receiver. For example, Alice can specify the access policy: "CS" **AND** "Student" to restrict the broadcast message to all CS students without specifying the receivers explicitly. Accordingly, Bob, who has attributes {"EE" , "Faculty"}, cannot decrypt the data while Carol, who has attributes {"CS" , "Student" } can access the data. On the other hand, to send a message to an arbitrary set of receivers, such as Bob and Carol, Alice specifies them explicitly as intended receivers and encrypts the broadcasted message. ABBE also significantly reduces the storage overhead compared to many existing BE schemes, whose cryptographic key materials required by encryption or decryption is can be linear or sublinear depending on the number of receivers. For example, in BGW scheme [3], the public key size is $O(N)$ or $O(N^{1/2})$, where N is the number of users in the system. ABBE addresses this key storage overhead problem by optimizing the organization of attribute hierarchy to minimize the storage requirement for each user. In a system with N users, the storage overhead is $O(\log N + m)$, where m is a constant number and $m \ll N$. We also proved from information theoretical perspective that ABBE achieves storage lower bound to satisfy all possible group/subgroup formations in a given system. As a result, ABBE requires minimal-level of stored key materials for each user, and thus it can be applied to storage constrained systems.

The most significant nature of this research article is that we present a fundamental and unified Attribute Based solution considering constraints on both communication and storage, and our solution is provable secure. In a summary, the main contributions of this research are presented as follows:

- *CCP-ABE*: We construct an efficient Constant Ciphertext Policy Attribute Based Encryption (CCP-ABE) scheme that can encrypt a message with an AND-gate access policy with wildcards. Moreover, CCP-ABE supports non-monotonic data access control policy. To the best of our knowledge, this is the first construction that achieves these properties.
- *ABBE*: Based on CCP-ABE, we present an Attribute Based Broadcast Encryption (ABBE) scheme. Compared with existing BE schemes, ABBE is flexible as it uses both descriptive and non-descriptive attributes, which enables a user to specify the decryptors based on different abstraction levels, with or without exact information of intended receivers. Moreover, ABBE demands less storage overhead compared to existing BE schemes. We proved that our construction requires minimal storage to support all the possible user group formations for BE applications.

The rest of this paper is organized as follows. We first summarize the related work in Section 2. Then, in Section 3, we present system models used in this paper. We give detailed CCP-ABE construction in Section 4. In Section 5, we present the construction of ABBE and the storage performance analysis using an information theoretical approach. In Section 6, the performance of ABBE is presented through both theoretical analysis y and experimental studies. Finally, we conclude our work in Section 7.

2 Related Works

The first fully functional Identity Based Encryption (IBE) scheme was proposed in [6]. In IBE, an identity or ID is an string one-to-one mapped to each user. A user can acquire a private key corresponding to his/her ID in an off-line manner from trusted authority and the ID is used as public key. The ciphertext encrypted by a particular ID can only be decrypted by the user with corresponding private key, i.e., the encryption is one-to-one.

Attribute Based Encryption (ABE) was first proposed as a fuzzy version of IBE in [31], where an identity is viewed as a set of descriptive attributes. The private key for an identity w can decrypt the message encrypted by the identity w' if and only if w and w' are closer to each other than a pre-defined threshold in terms of set overlap distance metric. In the paper [28], the authors further generalize the threshold-based set overlap distance metric to expressive access policies with AND and OR gates. There are two main variants of ABE proposed so far, namely Key Policy Attribute Based Encryption (KP-ABE [17]) and Ciphertext Policy Attribute Based Encryption (CP-ABE [1]). In KP-ABE, each ciphertext is associated with a set of attributes and each user's private key is embedded with an access policy. Decryption is enabled only if the attributes on the ciphertext satisfy the access policy of the user's private key. In CP-ABE [1, 10, 13, 19, 26, 16, 36, 21], each user has a set of attributes that associate with user's private key and each ciphertext is encrypted by an access policy. To decrypt the message, the attributes in the user private key need to satisfy the access policy. CP-ABE is more appealing since it is conceptually closer to the Role Based Access Control (RBAC) [32] model.

Although ABE schemes have shown their strong capability to construct a flexible data access control model, existing ABE schemes suffers from large ciphertext size problem. We must note that, in [13], the authors proposed a CP-ABE scheme with constant ciphertext size. However, their scheme does not support wildcards (or do-not-care) in its access policy, which makes the the number of access policy increase exponentially [2]. Moreover, to decrypt a ciphertext, the decryptor's attributes needs to be identical to the access policy. In other words, the model is still one-to-one, i.e., an access policy is satisfied by one attribute list or ID. Thus, their scheme can be simply implemented using IBE schemes with same efficiency by using each user's attribute list as his/her ID. We also note that Herranz et al. has recently proposed more general construction of CP-ABE with constant ciphertext independently. Their proposed scheme achieves constant ciphertext with any monotonic threshold data access policy, e.g. n-of-n (AND), 1-of-n (OR) and m-of-n.

ABE can be used as a perfect cryptographic building block to realize Broadcast Encryption (BE), which was introduced by Fiat and Naor in [14]. In BE, a broadcaster encrypts a message for some set of users who are listening to a broadcasting channel and use their private keys to decrypt the message. Compared with traditional one-to-one encryption schemes, BE is very efficient. Instead of sending messages encrypted with each individual recipient's public key, the broadcast encryptor broadcast one encrypted message to be decrypted by multiple eligible recipients with their own private keys.

The encrypter in the existing BE schemes need to specify the receiver list for a particular message. In many scenarios, it is very hard to know the complete receiver list and it is desirable to be able to encrypt without exact knowledge of possible receivers. Also, existing BE schemes can only support simple receiver list. It is hard to support flexible, expressive access control policies. An broadcast encryption with attribute based mechanism was proposed in [23], where expressive attribute based access policy replaces the flat receiver list. Also, in [9, 10], the authors proposed to use CP-ABE [1, 10] and flat-table [8] mechanism to minimize the number of messages and support expressive access policy. Compared with these works, our proposed scheme significantly reduce the size of ciphertext from linear to constant.

Based on different tradeoffs between storage and communication overhead, existing BE schemes can be generally categorized into the following classes: (1) constant ciphertext, linear public and/or private key on number of total receivers [3, 12]; (2) linear ciphertext on number of revoked receivers, constant (or logarithm) public and/or private key, [12, 25, 2, 20, 22]; (3) sublinear ciphertext, sublinear public and/or private key [3, 4]. If we denote the number of excluded or revoked receivers as r and total number of receivers as N, class (1) is more suitable for the case $(N - r) \ll N$; class (2) is more efficient when $r \ll N$ and class (3) can be used in most cases with balanced performance.

Although existing class (1) BE schemes feature constant ciphertext size, the number of public/private key each user needs to perform encryption/decryption is linearly proportional to the max number of non-colluding users in the system. In the case of the fully collusion-resistant BE systems, the number of public key each user needs to store equals to the number of users in the system. In a system with N users, where N is a large number, e.g., 2^{32}, the set of public keys $\{PK_i | i = 1 \dots N\}$ is huge and is impossible for each user to pre-load all public keys. Although it is possible to follow a PKI manner to issue certificate for each user, the encrypter needs to contact each each recipient to acquire the certificate or the encrypter needs to download the public keys from a centralized server, which is very costly and greatly undermined efficiency of BE. Although class (3) schemes tried to reduce the complexity of storing public keys to sublinear, the size of ciphertext is also increased to sublinear, which can still be huge in large system. As for the class (2) BE schemes, they are very efficient when $r \ll N$. However, as the increase of r, the efficiency of class (2) schemes drops linearly. In this work, we proposed a new construction of ABBE scheme to address the deficiency of all 3 class existing works. Particularly, ABBE supports any arbitrary number of receivers with much lower complexity of storage and communication.

3 System and Models

In this Section, we first describe how to use attributes to form a data access policy, then we present the bilinear map, which is the building block of ABE schemes. Finally, we present the complexity assumption, which will be used for our security proof.

[2] In a system with n attributes, the number of attribute combinations is 2^n. Without wildcard, we need 2^n access policies to express all combinations. On the other hand, one can use a single access policy with wildcards to express all combinations of attributes.

3.1 Attributes and Policy

Let $U = \{A_1, A_2, \cdots, A_k\}$ be the *Universe* of attributes in the system. Each A_i has three values: $\{A_i^+, A_i^-, A_i^*\}$. When a user u joins the system, u is tagged with an attribute list defined as follows:

Definition 1 *A user's attribute list is defined as* $L = \{A_1^{+/-}, A_2^{+/-}, \ldots, A_k^{+/-}\}$, *where* $A_i^{+/-} \in \{A_i^+, A_i^-\}$ *and* k *is the number of attributes in the universe.* $L = L^+ \cup L^-$. $L^+ = \{A_i^+ | \forall i \in \{1 \cdots k\}\}$ *and* $L^- = \{A_i^- | \forall i \in \{1 \cdots k\}\}$. *Also, we have* $L^+ \bigcap L^- = \emptyset$. □

Intuitively, A_i^+ denotes the user has A_i; A_i^- denotes the user does not have A_i or A_i is not a proper attribute of this user. For example, suppose $U = \{A_1 = \text{CS}, A_2 = \text{EE}, A_3 = \text{Faculty}, A_4 = \text{Student}\}$. Alice is a student in CS department; Bob is a faculty in EE department; Carol is a faculty holding a joint position in EE and CS department. Their attribute lists are illustrated in the following table:

Attributes	A_1	A_2	A_3	A_4
Description	CS	EE	Faculty	Student
Alice	A_1^+	A_2^-	A_3^-	A_4^+
Bob	A_1^-	A_2^+	A_3^+	A_4^-
Carol	A_1^+	A_2^+	A_3^+	A_4^-

The AND-gate access policy is defined in below:

Definition 2 *Let* $W = \{A_1, A_2, \cdots, A_k\}$ *be an AND-gate access policy, where* $A_i \in \{A_i^+, A_i^-, A_i^*\}$. *We use the notation* $L \models W$ *to denote that the attribute list* L *of a user satisfies* W, *as:*

$$L \models W \Longleftrightarrow W \subset L \bigcup \{A_1^*, A_2^*, \cdots A_k^*, \}.$$

□

A_i^+ or A_i^- requires the exact same attribute in user's attribute list. As for A_i^*, it denotes a wildcard value, which means the policy does not care the value of attribute A_i. Effectively, each user with either A_i^+ or A_i^- have the fulfills A_i^*. For example, to specify an access policy W_1 for all CS Student and an access policy W_2 for all CS people:

Attributes	A_1	A_2	A_3	A_4
Description	CS	EE	Faculty	Student
W_1	A_1^+	A_2^-	A_3^-	A_4^+
W_2	A_1^+	A_2^-	A_3^*	A_4^*

3.2 Bilinear Maps

Pairing is a bilinear map function $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_1$, where $\mathbb{G}_0$ and $\mathbb{G}_1$ are two multiplicative cyclic groups with large prime order p. The Discrete Logarithm Problem on both $\mathbb{G}_0$ and $\mathbb{G}_1$ are hard. Pairing has the following properties:

- *Bilinearity*:
$$e(P^a, Q^b) = e(P, Q)^{ab}, \quad \forall P, Q \in \mathbb{G}_0, \forall a, b \in \mathbb{Z}_p^*.$$
- *Nondegeneracy*:
$e(g, g) \neq 1$ where g is the generator of $\mathbb{G}_0$.
- *Computability*:
There exist an efficient algorithm to compute the pairing.

3.3 Complexity Assumption

The security of our proposed constructions is based on a complexity assumption called the Bilinear Diffie-Hellman Exponent assumption (BDHE) [2].

Let $\mathbb{G}_0$ be a bilinear group of prime order p. The K-BDHE problem in $\mathbb{G}_0$ is stated as follows: given the following vector of $2K+1$ elements (Note that the $g^{\alpha^{K+1}}$ is not in the list):

$$(h, g, g^{\alpha}, g^{(\alpha^2)}, \cdots, g^{\alpha^K}, g^{\alpha^{K+2}}, \cdots, g^{\alpha^{2K}}) \in \mathbb{G}_0^{2K+1}$$

as the input and the goal of the computational K-BDHE problem is to output $e(g,h)^{\alpha^{(K+1)}}$. We can denote the the set as:

$$Y_{g,\alpha,K} = \{g^{\alpha}, g^{(\alpha^2)}, \cdots, g^{\alpha^K}, g^{\alpha^{K+2}}, \cdots, g^{\alpha^{2K}}\}.$$

Definition 3 *(Decisional K-BDHE) The decisional K-BDHE assumption is said to be hold in $\mathbb{G}_0$ if there is no probabilistic polynomial time adversary who is able to distinguish*

$$< h, g, Y_{g,\alpha,K}, e(g,h)^{\alpha^{(K+1)}} >$$

and

$$< h, g, Y_{g,\alpha,K}, e(g,h)^{R} >$$

with non-negligible advantage, where $\alpha, R \in \mathbb{Z}_p$ and $g, h \in \mathbb{G}_0$ are chosen independently and uniformly at random. □

4 Constant Ciphertext Policy Attribute Based Encryption Construction

In this section, we present our construction of CCP-ABE scheme.

4.1 CCP-ABE Construction Overview

The CCP-ABE scheme consists of four fundamental algorithms:

- **Setup**($\mathbf{k}$)
  The **Setup** algorithm takes input k as the number of attributes in the system. It returns public key PK and master key MK. The public key is used for encryption while the master key is used for private key generation.
- **KeyGen**($\mathbf{PK}, \mathbf{MK}, \mathbf{L}$)
  The **KeyGen** algorithm takes the public key PK, the master key MK and the user's attribute list L as input . It outputs the private key of the user.
- **Encrypt**($\mathbf{PK}, \mathbf{W}, \mathbf{M}$)
  The **Encrypt** algorithm takes the public key PK, the specified access policy W and the message M as input. The algorithm outputs ciphertext CT such that only a user with attribute list satisfying the access policy can decrypt the message. The ciphertext also associates the access policy W.
- **Decrypt**($\mathbf{PK}, \mathbf{SK}, \mathbf{CT}$)
  The **Decrypt** algorithm decrypts the ciphertext when the user's attribute list satisfies the access policy specified in the ciphertext. It takes the public key PK, the private key SK of the user and the ciphertext CT as input. It returns the plaintext M if $L \models W$, where L is the user's attribute list and W is the access policy.

Boneh et al. proposed a broadcast encryption construction with constant ciphertext size in [3], where the broadcast encryptor use the public key list corresponding to intended receivers to perform encryption. To make the ciphertext constant, each receiver's public key is multiplied together, assuming a multiplicative group structure. Thus, the result ciphertext is still an element on the group, i.e., the ciphertext is constant size. We use a similar strategy to achieve constant ciphertext in our proposed scheme.

In our construction, each public key is mapped to an attribute value, including A_i. To encrypt a message, the encryptor specify an access policy W by assigning an attribute value ($A_i \in \{1, 0, *\}$) for each of the n attributes in the Universe and encrypt the message using public keys of the attribute values in the W. Each decryptor is generated a set of private key components corresponding to his/her attribute list L. All the private key components of the same user are tied together by a common random factor to prevent collusion attacks.

4.2 Setup

Assuming there are k attributes $\{A_1, A_2, \cdots, A_k\}$ in the system, we have $K = 3k$ attributes values since each attribute A_i has 3 values: $\{A_i^+, A_i^-, A_i^*\}$. For ease of presentation, we map $\{A_1^+, A_2^+, \cdots, A_k^+\}$ to $\{1, \cdots, k\}$, $\{A_1^-, A_2^-, \cdots, A_k^-\}$ to $\{k+1, \cdots, 2k\}$ and k wildcards $\{A_1^*, A_2^*, \cdots, A_k^*\}$to $\{2k+1, \cdots, 3k\}$ as in the following table:

Table 1. Mapping attribute values to numbers.

Attributes	A_1	A_2	A_3	$\cdots$	A_k
$+$	1	2	3	$\cdots$	k
$-$	$k+1$	$k+2$	$k+3$	$\cdots$	$2k$
$*$	$2k+1$	$2k+2$	$2k+3$	$\cdots$	$3k$

Let $\mathbb{G}_0$ be the bilinear group of prime order p. Trusted Authority (TA) first picks a random generator $g \in \mathbb{G}_0$ and a random $\alpha \in \mathbb{Z}_p$. It computes $g_i = g^{(\alpha^i)}$ for $i = 1, 2, \cdots, K, K+2, \cdots, 2K$ where $K = 3k$. Next, TA picks a random $\gamma \in \mathbb{Z}_p$ and sets $v = g^\gamma \in \mathbb{G}_0$. The public key is:

$$PK = (g, g_1, \ldots, g_K, g_{K+2}, \ldots, g_{2K}, v) \in \mathbb{G}_0^{2K+1}.$$

The master key $MK = \{\gamma, \alpha\}$ is guarded by the TA.

4.3 Key Generation

Each user u is tagged with the attribute list $L_u = L_u^+ \cup L_u^-$ when joining the system. We have $L_u^+ \subset \{1, \cdots, k\}$, $L_u^- \subset \{k+1, \cdots, 2k\}$. We also have $L^* = \{2k+1, \cdots, 3k\}$. The TA first selects k random numbers $\{r_1, r_2, \cdots, r_k\}$ from $\mathbb{Z}_p$ and calculate $r = \sum_{i=1}^k r_i$.

The TA computes $D = g^{\gamma r} = v^r$. For every $i \in L_u^+$, TA calculates $D_i = g^{\gamma(\alpha^i + r_{i'})}$ where $i' = i$; for every $i \in L_u^-$, TA calculates $D_i = g^{\gamma(\alpha^i + r_{i'})}$ where $i' = i - k$; for every $i \in L^*$, TA calculates $F_i = g^{\gamma(\alpha^i + r_{i'})}$ where $i' = i - 2k$.

The private key for user u is computed as:

$$SK_u = (D,\ \{D_i | \forall i \in L_u^+\},\ \{D_i | \forall i \in L_u^-\},\ \{F_i | \forall i \in L^*\}).$$

4.4 Encryption

The encrypter picks a random t in $\mathbb{Z}_p$ and sets the one-time symmetric encryption key $Key = e(g_K, g_1)^{kt}$. Suppose AND-gate policy is W with k attributes. Each attribute is either positive/negative or wildcards.

The encryptor first encrypts the message using symmetric key Key as $\{M\}_{Key}$. The encryptor also sets $C_0 = g^t$. Then, it calculates $C_1 = (v \prod_{j \in W} g_{K+1-j})^t$. The ciphertext is:

$$\begin{aligned} CT &= (W, \{M\}_{Key}, g^t, (v \prod_{j \in W} g_{K+1-j})^t) \\ &= (W, \{M\}_{Key}, C_0, C_1). \end{aligned}$$

4.5 Decryption

The decryptor u needs to checks whether $L_u \models W$ when receiving the ciphertext. If not, u returns $\perp$.

Then for $\forall i \in W$, u calculates the following terms:

$$\begin{aligned} e(g_i, C_1) &= e(g^{\alpha^i}, g^{t(\gamma + \sum_{j \in W} \alpha^{K+1-j})}) \\ &= e(g, g)^{t\gamma\alpha^i + t\sum_{j \in W} \alpha^{K+1-j+i}} \end{aligned}$$

and

$$
\begin{aligned}
& e(C_0, D_i \cdot \prod_{j \in W, j \neq i} g_{K+1-j+i}) \\
&= e(g^t, g^{\gamma(\alpha^i + r_{i'}) + \sum_{j \in W, j \neq i} \alpha^{K+1-j+i}}) \\
&= e(g,g)^{t\gamma(\alpha^i + r_{i'}) + t\sum_{j \in W, j \neq i} \alpha^{K+1-j+i}}.
\end{aligned}
$$

Then, we calculate

$$
\begin{aligned}
& e(g_i, C_1)/e(C_0, d_i \cdot \prod_{j \in S, j \neq i} g_{K+1-j+i}) \\
&= e(g,g)^{-t\gamma r_{i'} + t\alpha^{K+1}}.
\end{aligned}
$$

After we calculate all k terms, we make a production of all the quotient terms and get:

$$
e(g,g)^{-t\gamma(r_1 + r_2 + \cdots + r_k) + kt\alpha^{k+1}} = e(g,g)^{-t\gamma r + kt\alpha^{K+1}}.
$$

We calculate:

$$
e(D, C_0) = e(g,g)^{t\gamma r}.
$$

Then, we produce these two terms and get $Key = e(g,g)^{kt\alpha^{K+1}} = e(g_K, g_1)^{kt}$ and decrypt the message.

4.6 Security Analysis

We reduce Chosen Plaintext Attack (CPA) security of our proposed scheme to decisional K-BDHE assumption. We first define the decryption proxy to model collusion attackers.

Security Game for CCP-ABE

A CP-ABE scheme is considered to be secure against chosen CPA if no probabilistic polynomial-time adversaries have non-negligible advantages in this game.

Init: The adversary choose the challenge access policy W and give it to challenger.

Setup: The challenger runs the Setup algorithm and gives adversary the PK.

Phase 1: The adversary submits L for a KeyGen query, where $L \nvDash W$. The challenger answers with a secret key SK for L. This can be repeated adaptively

Challenge: The challenger runs Encrypt algorithm to obtain $\{< C_0, C_1 >, Key\}$. Next, the challenger picks a random $b \in \{0,1\}$. It sets $Key_0 = Key$ and picks a random Key_1 with same length to Key_0. It then gives $\{< C_0, C_1 >, Key_b\}$ to the adversary.

Phase 2: Same as Phase 1.

Guess: The adversary outputs its guess $b' \in \{0,1\}$ and it wins the game if $b' = b$.

Note that the adversary may make multiple secret key queries both before and after the challenge, which result in the collusion resistance in our proposed scheme. We also point out this CPA security game is called as selective ID security, because the adversary must submit a challenge access structure before the setup phase.

Theorem 1 *If a probabilistic polynomial-time adversary wins the CPA game with non-negligible advantage, then we can construct a simulator that distinguish a K-DBHE tuple with non-negligible advantage.* □

Proof 1 *Please see Appendix A.* □

5 Attribute Based Broadcast Encryption

Based on our construction of CCP-ABE, we construct an efficient and flexible Broadcast Encryption (BE) scheme– Attribute Based Broadcast Encryption (ABBE), where the size of a ciphertext is still constant.

Compared to existing BE schemes, using ABBE, encryptor does not need to store a large number of key materials, i.e., public key and private key. By carefully organizing the attributes in the system, we will show that the storage overhead of each user can be reduced from $O(N)$ to $O(\log N + m)$, where N is the number of users in the system and $m \ll N$ is the number of descriptive attributes in the system.

Also, in ABBE, an encryptor enjoys the flexibility of encrypting broadcast data using either a specific list of decryptors or an access policy without giving an exact list of decryptors.

5.1 ABBE Setup

In ABBE with N users, each user is issued an n-bit binary ID $b_0 b_1 \cdots b_n$, where b_i represents the i'th bit in the user's binary ID, where $n = \log N$. Accordingly, we can define n bit-assignment attributes $\{B_1, B_2, \cdots, B_n\}$. Each user is assigned n bit-assignment attribute values according to his/her ID. If the $b_i = 1$, he/she is assigned the B_i^+, if the $b_i = 0$, he/she is assigned the B_i^-. For example, in a system with 8 possible users, each user is assigned 3 bit-assignment attributes to represent the bit values in their ID, as illustrated in Figure 1 and Figure 2:

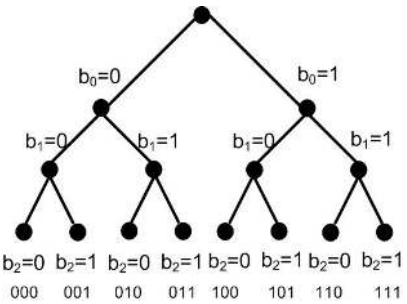

Fig. 1. An illustration of ID distribution.

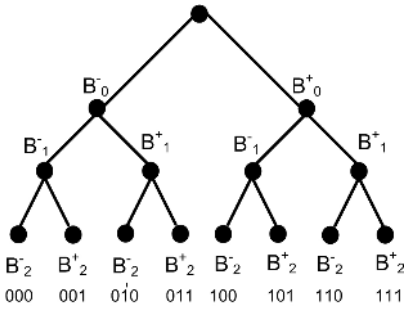

Fig. 2. An illustration of bit-assignment attributes.

Given the $n = \log N$ the bit-assignment attributes, TA generates $3n$ attributes values, i.e., bit-assigment attribute B_i has $\{B_i^+, B_i^-, B_i^*\}$ values.

In addition to the bit-assignment attributes, the TA also chooses m descriptive attributes for the system. These descriptive attributes present the real properties or features of an entity, which can be used to describe the decryptors' social or role features, e.g., "CS", "EE", "Student", "Faculty", etc. Each of the m descriptive attributes has $\{1, 0, *\}$ values.

With the $3n + 3m$ attribute values, the authority runs $\mathbf{Setup(n + m)}$ algorithm and generate public keys and private keys.

5.2 Broadcast Encryption

In order to control the access to the broadcasted message, the sender needs to specify an access policy using either the descriptive attributes or bit-assignment attributes. For example in the Table 5.2, if Alice wants send message to all CS students, she can specify the descriptive policy W_1 in the following table. Or she wants to send message to Bob and Carol, whose ID are 100 and 101 respectively, she can use the bit-assignment policy W_2, which is equivalent to enumerate every receivers.

	A_1=CS	A_2=EE	A_3=Student	A_4=Faculty	B_0	B_1	B_2
W_1	A_1^+	A_2^-	A_3^+	A_4^-	B_0^*	B_1^*	B_2^*
W_2	A_1^*	A_2^*	A_3^*	A_4^*	B_0^+	B_1^-	B_2^*

Here, we focus on how an encryptor can specify the list of receivers explicitly using n bit-assignment attributes. We first define some of the terms used in the following presentations:

- *Literal*: A variable or its complement, e.g., b_1, $\overline{b_1}$, etc.
- *Product Term*: Literals connected by AND, e.g., $\overline{b}_2 b_1 \overline{b}_0$.
- *Sum-of-Product Expression (SOPE)*: Product terms connected by OR, e.g., $\overline{b}_2 b_1 b_0 + b_2$.

Given the set of receivers S, the membership functions $f_S()$, which is in the form of SOPE, specifies the list of receivers:

$$f_S(b_1^u, b_2^u, \ldots, b_n^u) = \begin{cases} 0 & \text{iff } u \in S, \\ 1 & \text{iff } u \notin S. \end{cases}$$

For example, if the subgroup $S = \{000, 001, 011, 111\}$, then $f_S = \overline{b}_0\overline{b}_1\overline{b}_2 + \overline{b}_0\overline{b}_1 b_2 + \overline{b}_0 b_1 b_2 + b_0 b_1 b_2$.

Then, the broadcast encryptor runs the Quine-McCluskey algorithm [24] to reduce f_S to minimal SOPE f_S^{min}. The reduction can consider *do not care* values $*$ on those IDs that are not currently assigned to any receiver to further reduce number of product terms in the membership function. For example, if $S = \{000, 001, 011, 111\}$, $f_S^{min} = \overline{b}_0\overline{b}_1 + b_1 b_2$.

Since f_S^{min} is in the form of SOPE, encryption is performed on each product term. That is, for each product term E in f_S^{min}, the encryptor specifies an AND-gate access policy W using the following rules:

1. For positive literal $b_i \in f_S^{min}$, set B_i^+ in the access policy W.
2. For negative literal $\overline{b}_i \in f_S^{min}$, set B_i^- in the access policy W.
3. Set B_i^* for the rest of bit-assignment attributes.

For each W, the encryptor uses $\mathbf{Encrypt}(\mathbf{PK}, \mathbf{W}, \mathbf{M})$ algorithm to encrypt the message. The total number of encrypted message equals to the number of product terms in f_S^{min}.

For example, if $S = \{000, 001, 011, 111\}$, $f_S^{min} = \overline{b}_0\overline{b}_1 + b_1 b_2$. The access policies W_1 and W_2 are shown in the following table:

	A_1 =CS	A_2 =EE	A_3 =Student	A_4 =Faculty	B_0	B_1	B_2
W_1	A_1^*	A_2^*	A_3^*	A_4^*	B_0^-	B_1^-	B_2^*
W_2	A_1^*	A_2^*	A_3^*	A_4^*	B_0^*	B_1^+	B_2^+

We can find that f_S^{min} contains 2 product terms. the message M for S can be encrypted into 2 ciphertexts with W_1 and W_2 respectively.

5.3 Information Theoretical Optimality

In this section, we present the optimality of ABBE through an information theoretical approach similar to the models in [29]. In Section 5.3, we proved that ABBE attains information theoretical lower bound of storage requirements with $O(\log N)$ bit-assignment attributes. In Section 5.3, we also compared the BGW [3] BE scheme [3] and ABBE from information theoretical perspective.

Prefix Free Bit-Assignment Attributes Assignment To be uniquely identified, each user's ID should not be prefix of any other user's. For example, suppose a user u' is issued an ID 00, which is prefix of u_1 with ID 000 and u_2 with ID 001. When an encryptor tries to reach u_1 and u_2, the minimized membership function is $f = \overline{x}_0\overline{x}_1$, which is also satisfied by u'. Thus, it is also imperative that a user's bit-assignment attributes should not be a prefix of any other user's.

Theorem 2 *If we denote the number of bit-assignment attributes (or number of bits in the ID) for a user u_i by l_i. For an attribute based encryption system with N users and the attribute lists of users satisfy the prefix-free condition, the set $\{l_1, l_2, \ldots, l_N\}$ satisfies the Kraft inequality:*

$$\sum_{i=1}^{N} d^{-l_i} \leq 1.$$

□

Proof 2 *The proof is available in [11].* □

The prefix free condition is necessary and sufficient condition for addressing any user with their bit-assignment attributes.

For a message addressed to one particular user, we use p_i to denote the possibility that a user u_i is the target. Note that the ability to address to any one of the users is the necessary condition for a functioning broadcast encryption. To reach a receiver u_i, the encryptor needs l_i bit-assignment attributes, i.e., storage overhead of l_i. From the sender's perspective, we model the storage overhead as:

$$\sum_{i=1}^{N} p_i l_i. \tag{1}$$

Intuitively, this formation argues that the storage overhead from a sender's perspective is the average number of bit-assignment attributes required to address to a particular users. Thus, an optimization problem is formulated to minimize the storage overhead for a broadcast encryption system:

$$\min_{l_i} \sum_{i=1}^{N} p_i l_i$$

s.t.

$$\sum_{i=1}^{N} d^{-l_i} \leq 1.$$

This problem can be further rewritten as a Lagrangian optimization problem as:

$$\min_{l_i} \{\sum_{i=1}^{N} p_i l_i + \lambda(\sum_{i=1}^{N} d^{-l_i} - 1)\}, \tag{2}$$

where λ is the Lagrangian multiplier. The optimization problem is identical to the optimal codeword-length selection problem [11] in information theory. Before giving the solution to this optimization problem, we define the entropy of targeting one user in the broadcast encryption system:

Definition 4 *The entropy H of targeting a user in the broadcast encryption system is*

$$H = -\sum_{i=1}^{N} p_i \log p_i.$$

□

Theorem 3 *For an broadcast encryption system of N users with prefix free distribution of bit-assignment attributes, the optimal (i.e., minimal) average number of attributes required for a sender to address a receiver, written as $\sum_{i=1}^{N} p_i l_i$ is given by the d-ary entropy*

$$H_d = -\sum_{i=1}^{N} p_i \log p_i.$$

□

Proof 3 *The theorem is equivalent to to optimal codeword-length selection problem and proof is available in [11].* □

Since the average number of attributes required for addressing one particular receiver is given by the entropy of targeting a user, we now try to derive the upper and lower bounds of the entropy:

$$\max_{p_i} -\sum_{i=i}^{N} p_i \log p_i$$

and

$$\min_{p_i} -\sum_{i=i}^{N} p_i \log p_i$$

s.t.

$$\sum_{i=1}^{N} p_i = 1.$$

The upper bound $H_{max} = -\sum_{i=1}^{N} \frac{1}{N} \log N = \log N$ is yielded when $p_i = 1/N$, $\forall i \in \{1, 2, \ldots, N\}$, when each user has equal possibility to be addressed as the receiver. When there is no apriori information about the possibility distribution of targeting one of the users, $l = H_{max} = \log_d N$ correspond to the optimal strategy to minimize the average number of attributes required for each user. On the other hand, the lower bound $H_{min} = 0$ is achieved when $p_i = 1$ for $\exists i \in \{1, 2, \ldots, N\}$, which is an extreme case where there is no randomness and only one user is reachable.

Compare with BGW BE scheme If we denote our optimal bit-assignment attributes assignment to be minimalist, which requires the least number of bit-assignment attributes to identity each user. We can refer BGW scheme in [3] as maximalist. In BGW scheme, for a system with N users, each user is mapped to a unique public key. Given all N public keys, the number of combinations is $2^N - 1$, which equals to the number of receiver subsets in the system. Thus, each encryptor needs maximal number of public keys to perform broadcast encryption.

To compare the minimalist and maximalist storage strategy, we can treat each attribute or public key as an binary variable $v \in \{1, 0\}$. We denote $p = P_{v=1}$ as the percentage of totals users who have this attributes or public key and $1 - p = P_{v=0}$ as the percentage of totals users who do not have this attributes or public key, given that $P_{(v=1)} + P_{(v=0)} = 1$.

Definition 5 *The entropy of an attribute or a public key is defined as:*

$$H(v) = p \log p^{-1} + (1 - p) \log(1 - p)^{-1}.$$

□

Based on the Definition 5 , we see the entropy of each attribute in minimalist strategy as $H_a(1/2) = 1$ since ,for each particular attribute, exact half of the users have it while the other half do not have it. On the other hand, the entropy of public key in maximalist strategy is $H_a(1/N) = (1/N) \log(N) + ((N-1)/N) \log(N/(N-1)) < 1$. Hence, we can conclude that minimalist strategy attains maximal binary entropy while the maximalist strategy attains minimal binary entropy.

6 Performance Analysis

In this section, we analyze the performance of ABBE and compare it with several related solutions: subset-difference broadcast encryption scheme (Subset-Diff) [15], BGW broadcasting encryption [7], NNL [25], DPP [12], BW [4], LT [22], access control polynomial (ACP) scheme [39] and FT implemented using CP-ABE (FT-ABE) [9]. We also compared some works in tree-based multicast group key distribution domain where a group controller remove some group members by selectively multicasting key update messages to all remaining members. Those solution can be broadly divided into 2 categories: Flat-Table (FT) scheme [8, 38] and Non-Flat-Table schemes, including OFT [34], LKH [37], ELK [27].

The performance is assessed in terms of communication overhead (number and size of messages), storage overhead (system data stored on the users and system centers), and computation overhead (number of cryptographic operations needed in encryption and decryption operations) when a user talks to any given subgroup of users in the system. We denote the group size be N.

6.1 Communication Overhead

The complexity analysis of communication overhead for various schemes is summarized in Table 2. In Subset-Diff scheme, the communication overhead is $O(t^2 \cdot log^2 t \cdot \log N)$, with t as maximum number of colluding users to compromise the ciphertext. For BGW scheme, the message size is $O(N^{\frac{1}{2}})$ as reported in [7]. In ACP scheme, the size of message depends on the degree of access control polynomial, which equals to the number of current receivers. Thus, the message size is $O(N)$.

For Non-flat-table tree-based multicast key distribution schemes such as OFT [34], LKH [37], ELK [27], etc., the communication overhead for removing members depends on the number of keys in the tree that need to be updated [35, 27]. In the case of removing a single member, $O(\log N)$ messages are required since the center needs to update $\log N$ auxiliary keys distributed to the removed member. Some tree-based schemes tried to optimize the number of messages to update all the affected keys in the case of multiple *leaves*. In ELK [27], which is known to be one of the most efficient tree-based schemes, the communication overhead for multiple *leaves* is $O(a - l)$, where $a \approx l \log N$ is the number of affected keys and l is the number of leaving members. Thus, the complexity can be written as $O(l \log N)$.

For flat-table tree-based scheme [8], the complexity of removing a single member is also $O(\log N)$. The main benefit of flat-table, however, is the minimal number of messages for batch removing multiple members. In fact, our scheme requires the same number of messages compared to flat-table scheme, thus they both achieved

Table 2. Comparison of communication overhead and Storage overhead in different broadcast encryption schemes and group key management schemes.

Scheme	Communication Overhead		Storage Overhead	
	single receiver	multiple receivers	Center	User
ABBE	$O(1)$	$\approx O(\log N)$	N/A	$O(\log N + m)$
Subset-Diff	$O(t^2 \cdot log^2 t \cdot \log N)$	$O(t^2 \cdot log^2 t \cdot \log N)$	$O(N)$	$O(t \log t \log N)$
BGW_1	$O(1)$	$O(1)$	N/A	$O(N)$
BGW_2	$O(N^{\frac{1}{2}})$	$O(N^{\frac{1}{2}})$	N/A	$O(N^{\frac{1}{2}})$
NNL_1	N/A	$O(t \log(N/t))$	N/A	$O(\log N)$
NNL_2	N/A	$O(t)$	N/A	$O(\log^2 N)$
DPP_1	O(1)	$O(1)$	N/A	$O(N)$
DPP_2	N/A	$O(t)$	N/A	$O(1)$
BW	$O(N^{\frac{1}{2}})$	$O(N^{\frac{1}{2}})$	N/A	$O(N^{\frac{1}{2}})$
LT	N/A	$O(t)$	N/A	$O(\log N)$
ACP	$O(N)$	$O(N)$	$O(N)$	$O(1)$
Flat-Table	$O(\log N)$	$\approx O(\log N)$	$O(\log N)/O(N)$	$O(\log N)$
Flat-Table-ABE	$O(\log N)$	$\approx O(\log^2 N)$	$O(\log N)/O(N)$	$O(\log N)$
Non-Flat-Table-Tree	$O(\log N)$	$O(l \cdot \log N)$	$O(N)$	$O(\log N)$

N: the number of group members; l: the number of leaving members; t: maximum number of colluding users to compromise the ciphertext.

information theoretical optimality. However, flat-table is vulnerable to collusion attacks [38]. In [9], the authors proposed to implement flat-table using CP-ABE [1] to counter collusion attacks.

To control a set of receivers S using ABBE, the number of messages depends on the number of product terms in the f_S^{min}. In [33], the authors derived an upper bound and lower bound on the average number of product terms in a minimized SOPE. Experimentally, the average number of message required is $\approx \log N$ [9].

Number of Messages: Worst Cases We examine some cases when maximal number of messages is required to reach multiple receivers.

Lemma 1 (multiple receivers worst case) *The worst case of reaching multiple receivers happens when both of following conditions hold: 1) the number of distinct receivers is $N/2$; 2) the Hamming distance between IDs of any two receivers is at least 2. In the worst case, the number of key updating messages is $N/2$.* □

Proof 4 *Please refer to [8] for complete proof.* □

In this case, the number messages is $N - N/2 = N/2$ using ABBE. However, we can see that the worst cases happens in extremely low probability:

Lemma 2 (worst case possibility) *When communicating all subgroups with uniform opportunity, the worst case scenario happens with probability $\frac{1}{2^{N-1}}$.* □

Proof 5 *In the worst case, the Hamming distance of IDs of $N/2$ receivers should be at least 2. As shown in the Karnaugh table in Figure 3, each cell represents an ID. For any cell marked 0 and any cell marked 1, the Hamming distance is at least 2. Thus, the worst cases happens in two cases: (1) the encryptor wants to reach $N/2$ receivers marked 1 in Figure 3; (2) the encryptor wants to reach $N/2$ receivers marked 0 in Figure 3.*

□

We also have the worst case for communicating the majority of users.

Lemma 3 (Worst case of reaching N-2 receivers) *When reaching $N-2$ receivers, the maximal number of messages required is $n = \log N$, when the Hamming distance between 2 non-receivers is n.* □

Proof 6 *Please refer to [8].* □

b_0b_1 \ b_2b_3	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	1	0	1	0
10	0	1	0	1

Fig. 3. Worst cases of broadcast encryption to $N/2$ receivers

Number of Messages: Average Case To investigate the average case, we simulated ABBE in a system with 512 users and 1024 users, and the number of messages required are shown in Figure 4 and Figure 5 respectively. In the simulation, we consider the cases of 0%, 5%, 25%, 50% IDs are not assigned (i.e., *do not care* value). For each case, different percentages of receivers are randomly selected from the group. We repeat 100 times to average the results. From the result, we can see that ABBE performs achieves roughly $O(\log N)$ complexity, where the constant factor is about 9 for the 512-member group and 18 for the 1024-member group.

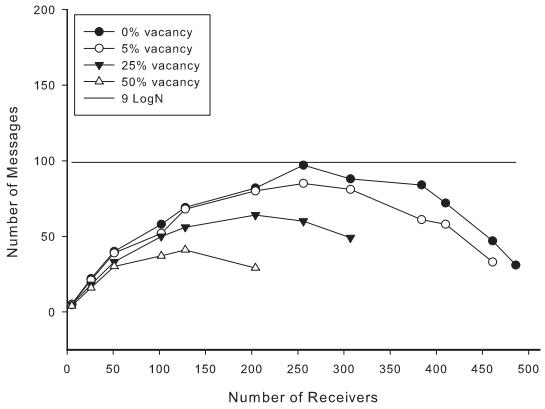

Fig. 4. Number of messages in a system with 512 users.

Total Message Size Finally, we look into the message size of ABBE, with comparison to FT-CP-ABE[9]. As mentioned in [9], in FT-CP-ABE, the size of ciphertext grows linearly based on the increase of the number of attributes in the access policy [9, 1]. Experimentally, the message size in FT-CP-ABE starts at about 630 bytes, and each additional attribute adds about 300 bytes. In a system with 10 bit ID or 1024 users, the number of attributes using FT-CP-ABE ciphertext is at most 10 and the message size may be as large as $630+9\cdot300 = 3330$ bytes. Since the number of attributes in the access policy is bounded by $\log N$, we can conclude that the communication overhead of FT-CP-ABE is in the order of $O(\log^2 N)$. In ABBE, every ciphertext contains exactly 2 group member on $\mathbb{G}_0$. Empirically, the size of one element on $\mathbb{G}_0$ is about 128 bytes. Thus, the ciphertext in ABBE is bounded within 300 bytes, which is significantly smaller than the ciphertext size reported in FT-CP-ABE [9]. Moreover, since the component C_0 in the ciphertext can be shared by multiple messages, we can further reduce the message size of ABBE with efficient communication protocol design.

6.2 Storage Overhead

In ABBE, there are $6\log N + 1$ elements on $\mathbb{G}_0$ in the PK. Also, a user needs to store $m \ll N$ descriptive attributes. Thus, the storage overhead is $O(\log N + m)$, assuming a user does not store any IDs of other users. Although the broadcast encryptor may need the list of receivers' IDs along with the list of *do not care* IDs to perform boolean function minimization, we can argue that this does not incur extra storage overhead.

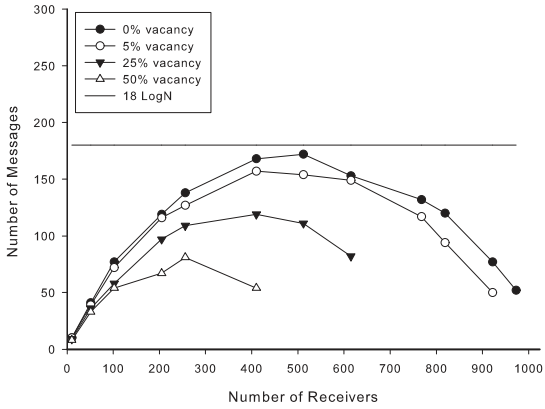

Fig. 5. Number of messages in a system with 1024 users.

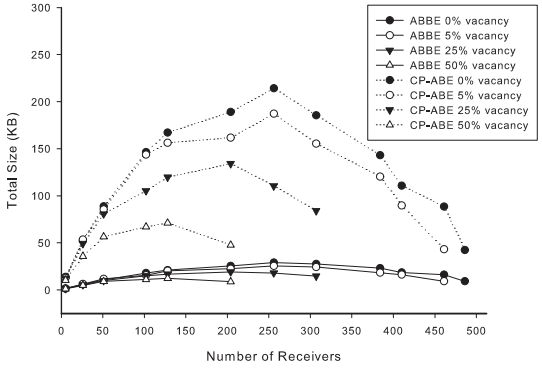

Fig. 6. Total Size of messages in a system with 512 users.

- The encryptors do not need to store the receiver's IDs after the broadcast; thus, the storage space can be released.
- The TA can periodically publish the minimized SOPE of all *do not care* IDs, which can be used by encryptors to further reduce number of messages.
- If IDs are assigned to users sequentially, i.e., from low to high, TA can simply publish the lowest unassigned IDs to all users, who can use the all higher IDs as *do not care* values.
- Even if a user needs to store N IDs, the space is merely $N \log N$ bits. If $N = 2^{20}$.
- If a broadcast encryptor cannot utilize *do not care* values to further reduce the membership function in SOPE form, the communication overhead might be a little higher. As shown in Figure 4and Figure 5, the curve of 0% vacancy can also be used as number of messages required if a broadcast encryptor does not know the *do not care* IDs.

6.3 Computation Overhead

In this section, we compare the computation overhead of those asymmetric key based schemes and the summarized results are presented in Table 3. In ACP scheme, the author reports that the encryption needs $O(N^2)$ finite field operations when the sub-group size if N; in the BGW scheme, the encryption and decryption require $O(N)$ operations on the bilinear group, which are heavier than finite field operations [18, 30]. In ABBE, each encryption requires $\log N$ operations on the $\mathbb{G}_0$, and the decryption requires $2 \log N + 1$ pairings and $\log N(\log N - 1) + \log N$ operations on $\mathbb{G}_0$ and $\log N$ operations on $\mathbb{G}_1$. Thus, the complexities of encryption and decryption are bounded by $O(\log N)$. Although the problem of minimizing SOPE is NP-hard, efficient approximations are widely known. Thus, ABBE is much more efficient than ACP and BGW when group size is large.

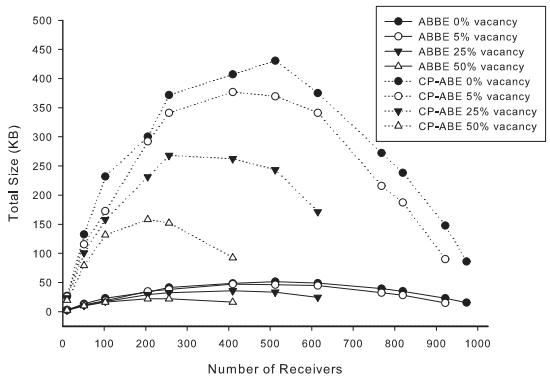

Fig. 7. Total Size of messages in a system with 1024 users.

Table 3. Comparison of computation complexity in different broadcast encryption schemes.

Scheme	Computation Overhead	
	Encryption	Decryption
ABBE	$O(\log N)$	$O(\log N)$
BGW	$O(M)$	$O(M)$
ACP	$O(M^2)$	$O(1)$

N: the number of group members; M: the number of receivers.

7 Conclusion

In this paper, a Constant Ciphertext Policy Attribute Based Encryption (CCP-ABE) was proposed. Compared with existing CP-ABE constructions, CCP-ABE significantly reduces the ciphertext size from linear to constant and supports expressive access policies. Thus, CCP-ABE can be used in many communication constrained environments.

Based on CCP-ABE, we further proposed an Attribute Based Broadcast Encryption (ABBE) scheme that attains information theoretical minimal storage overhead. Thus, a storage restricted user can easily pre-install all required key materials to perform encryption and decryption. Through theoretical analysis and simulation, we compared ABBE with many existing BE solutions and we showed that ABBE achieve better trade-offs between storage and communication overhead.

The security of CCP-ABE is based on selective-ID attackers. One open problem is constructing constant CP-ABE that is secure against adaptive adversaries. Another limitation of this paper is the CCP-ABE is constructed and proved following BGW [3] model. We are looking for new constructions with equal or stronger security level. Also, in this paper, we only proved ABBE is minimalist in terms of storage overhead. We are working on more information theoretical analysis that takes into account both storage-communication overhead in BE schemes.

References

1. J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-Policy Attribute-Based Encryption. *Proceedings of the 28th IEEE Symposium on Security and Privacy (Oakland)*, 2007.
2. D. Boneh, X. Boyen, and E.J. Goh. Hierarchical identity based encryption with constant size ciphertext. *Advances in Cryptology–EUROCRYPT 2005*, pages 440–456, 2005.
3. D. Boneh, C. Gentry, and B. Waters. Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys. *Advances in Cryptology-Crypto 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, 2005.
4. D. Boneh and B. Waters. A fully collusion resistant broadcast, trace, and revoke system. In *Proceedings of the 13th ACM conference on Computer and communications security*, page 220. ACM, 2006.
5. D. Boneh and B. Waters. Conjunctive, subset, and range queries on encrypted data. *Theory of Cryptography*, pages 535–554, 2007.
6. Dan Boneh and Matt Franklin. Identity-based encryption from the weil pairing. *SIAM Journal of Computing*, 32(2):586–615, 2003.

7. Dan Boneh, Amit Sahai, and Brent Waters. Fully collusion resistant traitor tracing with short ciphertexts and private keys. pages 573–592, 2006.
8. I. Chang, R. Engel, D. Kandlur, D. Pendarakis, D. Saha, I.B.M.T.J.W.R. Center, and Y. Heights. Key management for secure lnternet multicast using Boolean functionminimization techniques. *INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2, 1999.
9. L. Cheung, J. Cooley, R. Khazan, and C. Newport. Collusion-Resistant Group Key Management Using Attribute-Based Encryption. Technical report, Cryptology ePrint Archive Report 2007/161, 2007. http://eprint. iacr. org.
10. L. Cheung and C. Newport. Provably secure ciphertext policy ABE. In *Proceedings of the 14th ACM conference on Computer and communications security*, page 465. ACM, 2007.
11. T.M. Cover and J.A. Thomas. *Elements of information theory*. wiley, 2006.
12. C. Delerablée, P. Paillier, and D. Pointcheval. Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys. *Pairing-Based Cryptography–Pairing 2007*, pages 39–59.
13. Keita Emura, Atsuko Miyaji, Akito Nomura, Kazumasa Omote, and Masakazu Soshi. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length. In *ISPEC '09: Proceedings of the 5th International Conference on Information Security Practice and Experience*, pages 13–23, Berlin, Heidelberg, 2009. Springer-Verlag.
14. A. Fiat and M. Naor. Broadcast Encryption, Advances in Cryptology-Crypto93. *Lecture Notes in Computer Science*, 773:480–491, 1994.
15. A. Fiat and M. Naor. Broadcast Encryption, Advances in Cryptology-Crypto93. *Lecture Notes in Computer Science*, 773:480–491, 1994.
16. V. Goyal, A. Jain, O. Pandey, and A. Sahai. Bounded ciphertext policy attribute based encryption. *Automata, Languages and Programming*, pages 579–591.
17. 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*, pages 89–98, 2006.
18. D.R. Hankerson, S.A. Vanstone, and A.J. Menezes. *Guide to Elliptic Curve Cryptography*. Springer, 2004.
19. J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. *Advances in Cryptology-EUROCRYPT*, 4965:146–162, 2008.
20. J. Lee, Y. Hwang, and P. Lee. Efficient public key broadcast encryption using identifier of receivers. *Information Security Practice and Experience*, pages 153–164.
21. X. Liang, Z. Cao, H. Lin, and D. Xing. Provably secure and efficient bounded ciphertext policy attribute based encryption. In *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, pages 343–352. ACM, 2009.
22. Y.R. Liu and W.G. Tzeng. Public key broadcast encryption with low number of keys and constant decryption time. *Public Key Cryptography–PKC 2008*, pages 380–396.
23. D. Lubicz and T. Sirvent. Attribute-based broadcast encryption scheme made efficient. *Progress in Cryptology–AFRICACRYPT 2008*, pages 325–342.
24. E.J. McCluskey. Minimization of Boolean functions. *Bell System Technical Journal*, 35(5):1417–1444, 1956.
25. D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. *Lecture Notes in Computer Science*, pages 41–62, 2001.
26. R. Ostrovsky and B. Waters. Attribute-based encryption with non-monotonic access structures. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 195–203. ACM New York, NY, USA, 2007.
27. A. Perrig, D. Song, and J. Tygar. ELK, A New Protocol for Efficient Large-Group Key Distribution. *IEEE SYMPOSIUM ON SECURITY AND PRIVACY*, pages 247–262, 2001.
28. M. Pirretti, P. Traynor, P. McDaniel, and B. Waters. Secure attribute-based systems. In *Proceedings of the 13th ACM conference on Computer and communications security*, page 112. ACM, 2006.
29. R. Poovendran and JS Baras. An information-theoretic approach for design and analysis ofrooted-tree-based multicast key management schemes. *IEEE Transactions on Information Theory*, 47(7):2824–2834, 2001.
30. Archana Ramachandran, Zhibin Zhou, and Dijiang Huang. Computing Cryptographic Algorithms in Portable and Embedded Devices. *Portable Information Devices, 2007. PORTABLE07. IEEE International Conference on*, 25-29:1–7, 2007.
31. A. Sahai and B. Waters. Fuzzy Identity-Based Encryption. *Advances in Cryptology–Eurocrypt*, 3494:457–473.
32. RS Sandhu, EJ Coyne, HL Feinstein, and CE Youman. Role-based access control models. *Computer*, 29(2):38–47, 1996.
33. T. Sasao. Bounds on the average number of products in the minimum sum-of-products expressions for multiple-value input two-valued output functions. *Computers, IEEE Transactions on*, 40(5):645–651, May 1991.
34. A.T. Sherman and D.A. McGrew. Key Establishment in Large Dynamic Groups Using One-Way Function Trees. *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, pages 444–458, 2003.
35. J. Snoeyink, S. Suri, and G. Varghese. A lower bound for multicast key distribution. *Computer Networks*, 47(3):429–441, 2005.
36. B. Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. *ePrint report*, 290, 2008.
37. C.K. Wong, M. Gouda, and SS Lam. Secure group communications using key graphs. *Networking, IEEE/ACM Transactions on*, 8(1):16–30, 2000.

38. Zhibin Zhou and Dijiang Huang. An Optimal Key Distribution Scheme for Multicast Group Communication. In *To Appear Proceedings of The 29th Conference on Computer Communications (INFOCOM Mini-Conference)*, 2010.
39. X. Zou, Y.S. Dai, and E. Bertino. A Practical and Flexible Key Management Mechanism For Trusted Collaborative Computing. *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 538–546, 2008.

A Security Proof of Theorem 1

We reduce CPA security of our proposed scheme to decisional K-BDHE assumption. We first define the decryption proxy to model collusion attackers.

Definition 6 *(Decryption Proxy) In order to model the collusion attacks, we define* $2k$ *decrypting proxies in the security game. Each decrypting proxy* $p_i(r) = g^{\gamma(\alpha^i+r)}$*, where* $r \in \mathbb{Z}_p$ *and* $i \in \{1, \cdots, 2k\}$*, i.e., a private key component corresponding to a particular attribute value.*

In collusion attacks against access policy W, a user with attribute list $L \nvDash W$ collude with $x \leq k$ decryption proxies to attack the ciphertext. We call the colluding with x decryption proxy as x-collusion. Intuitively, x-collusion means the attacker needs x attributes values, say $\{i_1, i_2, \cdots, i_x\}$ to add to his attribute list L such that $L \cup \{i_1, i_2, \cdots, i_x\} \models W$. Note that 0-collusion means no decryption proxy is used and user does not collude.
Proof of Theorem 1 :
Suppose that an adversary $\mathcal{A}$ wins the selective game for CCP-ABE with the advantage ε. Then, we can construct a Simulator $\mathcal{B}$ that breaks decisional K-BDHE assumption with the advantage $\max\{\varepsilon/2, (1-q/p)^l\varepsilon/2, (1-(1-(1-q/p)^l)^m)\varepsilon/2\} = \varepsilon/2$. The simulator $\mathcal{B}$ takes an input a random decisional K-BDHE challenge

$$< h, g, Y_{g,\alpha,K}, Z >,$$

where Z is either $e(g,h)^{\alpha^{(K+1)}}$ or a random element on $\mathbb{G}_0$. $\mathcal{B}$ now plays the role of challenger in the pre-defined CPA game:

Init*:* $\mathcal{A}$ sends to $\mathcal{B}$ the access policy W that $\mathcal{A}$ wants to be challenged.
Setup*:* $\mathcal{B}$ runs the **Setup** algorithm to generate PK. $\mathcal{B}$ chooses random $d \in \mathbb{Z}_p$ and generates:

$$v = g^d(\prod_{j\in W} g_{K+1-j})^{-1} = g^{d-\sum_{j\in W}\alpha^{K+1-j}} = g^{\gamma}.$$

The $\mathcal{B}$ outputs the PK as:

$$PK = (g, Y_{g,\alpha,K}, v) \in \mathbb{G}_0^{2K+1}.$$

Phase 1*:* The adversary $\mathcal{A}$ submits an attribute list L for a private key query, where $L \nvDash W$. There must exist a j in L such that: either $j \in \{1, \cdots, k\}$ and $j + k \in W$ or $j \in \{k+1, \cdots, 2k\}$ and $j - k \in W$.

The simulator $\mathcal{B}$ first selects k random numbers $r_i \in \mathbb{Z}_p$ for $i = 1 \ldots k$ and set $r = r_1 + \cdots + r_k$. Then, $\mathcal{B}$ generates

$$\begin{aligned} D &= (g^d \prod_{j\in W} (g_{K+1-j})^{-1})^r \\ &= g^{(d-\sum_{j\in W}\alpha^{K+1-j})r} \\ &= g^{\gamma r}. \end{aligned}$$

Then, for all $i \in L^+$ and $i + k \in W$: $\mathcal{B}$ generates:

$$D_i = g_i^d \prod_{j\in W} (g_{K+1-j+i})^{-1} g^{ur_{i'}} \prod_{j\in W} (g_{K+1-j})^{-r_{i'}},$$

where $i' = i$.

For all $i \in L^-$ and $i - k \in W$: $\mathcal{B}$ generates:

$$D_i = g_i^d \prod_{j\in W} (g_{K+1-j+i})^{-1} g^{ur_{i'}} \prod_{j\in W} (g_{K+1-j})^{-r_{i'}},$$

where $i' = i - k$.

For all $i \in L^*$ and $i \notin W$: $\mathcal{B}$ generates:

$$F_i = g_i^d \prod_{j \in W} (g_{K+1-j+i})^{-1} g^{ur_{i'}} \prod_{j \in W} (g_{K+1-j})^{-r_{i'}},$$

where $i' = i - 2k$.

Note that each for each D_i or F_i is valid since:

$$D_i = (g^d (\prod_{j \in W} g_{K+1-j})^{-1})^{(\alpha^i + r_{i'})} = g^{\gamma(\alpha^i + r_{i'})},$$

and

$$F_i = (g^d (\prod_{j \in W} g_{K+1-j})^{-1})^{(\alpha^i + r_{i'})} = g^{\gamma(\alpha^i + r_{i'})}.$$

Challenge: The simulator $\mathcal{B}$ sets $< C_0, C_1 >$ as $< h, h^d >$. It then gives the challenge $\{< C_0, C_1 >, Z^k\}$ to $\mathcal{A}$.

To see the validity of challenge, $C_0 = h = g^t$ for some unknown t. Then:

$$\begin{aligned} h^d &= (g^d)^t \\ &= (g^d \prod_{j \in W} (g_{K+1-j})^{-1} \prod_{j \in W} (g_{K+1-j}))^t \\ &= (v \prod_{j \in W} (g_{K+1-j}))^t, \end{aligned}$$

and if $Z = e(g,h)^{\alpha^{(K+1)}}$, then $Z^k = Key$.

Phase 2: Repeat as ***Phase 1***.

Guess: The adversary $\mathcal{A}$ output a guess b' of b. When $b' = 0$, $\mathcal{A}$ guesses that $Z = e(g,h)^{\alpha^{(K+1)}}$. When $b' = 1$, $\mathcal{A}$ guesses Z is a random element.

If Z is a random element, then the $\mathbf{Pr}[\mathcal{B}(h, g, Y_{g,\alpha,K}, Z) = 0] = \frac{1}{2}$.

Before considering the case when $Z == e(g,h)^{\alpha^{(K+1)}}$, we explain how we use decryption proxy in the proof. Each decryption proxy $p_i(r)$ simulates a legal private key component embedded with random number r. When calling $p_i(r)$, $\mathcal{A}$ passes a random r as a guess of the $r_{i'}$, which is the random number embedded in the D_i or F_i, where $i \in W$. As a matter of fact, the procedure of calling decryption proxy mimics the collusion of multiple users, who combine their private key components.

Lemma 4 (probability of collision with 1 decryption proxy) *Suppose the $\mathcal{A}$ has issued q private queries and there is only 1 attribute $i \notin W$, $\mathcal{A}$ queries $p_i(r)$ l times. The possibility that the none of the queries returns a legal private key component of any q is $(1 - q/p)^l$.* □

Proof 7 *The possibility that the one query does not return a legal private key component of any q is $1 - q/p$. Thus, if none of the l query succeed, the probability $\mathbf{Pr}[r \neq r_{i'}] = (1 - q/p)^l$, where r is the random number in decryption proxy, $r_{i'}$ is the random number embedded in the private key, q is the number of private key queries in phase 1 and phase 2, l is the number of calling decryption proxy with different r, and p is the order of $\mathbb{Z}_p$.* □

Lemma 5 (probability of collision with m decryption proxy) *Suppose the $\mathcal{A}$ has issued q private queries and there is m attributes violate the W, $\mathcal{A}$ queries each of the m decryption proxy $p_{i_1}(r_1), p_{i_2}(r_2), \cdots, p_{i_m}(r_m)$ l times. The possibility that the none of the queries returns a legal private key component of any q is $(1 - (1 - q/p)^l)^m$.* □

Proof 8 *The probability that 1 decryption proxy fails is $\mathbf{Pr}[r \neq r_{i'}] = (1 - q/p)^l$. The probability that all the m decryption proxy successfully return legal components is $(1 - (1 - (q/p)^l))^m$. In the case of not all m succeed, the probability is $\mathbf{Pr}[r_{i_j} \neq r_{i'_j}, \exists j \leq m] = 1 - (1 - (1 - q/p)^l)^m$.* □

If $Z == e(g,h)^{\alpha^{(K+1)}}$, we consider the following cases:

– *0-Collusion*: If no decryption proxy is used, $\mathcal{A}$ has at least $\varepsilon/2$ advantage in breaking our scheme, then $\mathcal{B}$ has at least ε advantage in breaking K-BDHE, i.e.,

$$|\mathbf{Pr}[\mathcal{B}(h, g, Y_{g,\alpha,K}, Z) = 0] - \frac{1}{2}| \geq \varepsilon/2.$$

– *1-collusion* If 1 decryption proxy, say $p_i(r)$ is used, $\mathbf{Pr}[r \neq r_{i'}] = (1 - q/p)^l$, where r is the random number in decryption proxy, $r_{i'}$ is the random number embedded in the private key, q is the number of private key queries in phase 1 and phase 2, l is the number of calling decryption proxy with different r, and p is the order of $\mathbb{Z}_p$. Note that if $r = r_{i'}$, $\mathcal{A}$ can use $p_i(r)$ as a valid private key component to compromise the ciphertext. If the $\mathcal{A}$ has at least ε advantage in breaking our scheme, then $\mathcal{B}$ has at least $(1 - q/p)^l\varepsilon/2$ advantage in breaking K-BDHE.
– *m-collusion* If m decryption proxies, say

$$p_{i_1}(r_1), p_{i_2}(r_2), \cdots, p_{i_m}(r_m)$$

are used. The possibility that $\mathbf{Pr}[r_{i_j} \neq r_{i'_j}, \exists j \leq m] = (1 - (1 - (q/p)^l))^m$, where r_m is the random number in m decryption proxy $p_{i_m}(r_{i_m})$ for the private key component i_m, $r_{i'_m}$ is the random number generated for the $\mathcal{A}$, q is the number of private key queries in phase 1 or phase 2, l is the number of calling m decryption proxies with different r's, p is the order of $\mathbb{Z}_p$.
If the $\mathcal{A}$ has at least ε advantage in breaking our scheme, then $\mathcal{B}$ has at least $(1 - (1 - (1 - q/p)^l)^m)\varepsilon/2$ advantage in breaking K-BDHE.