Attribute-Based Encryption from Identity-Based Encryption

Chun-I Fan*, Yi-Fan Tseng, and Chih-Wen Lin

Abstract—Ciphertext-policy attribute-based encryption (CP-ABE) is an access control mechanism where a data provider encrypts a secret message and then sends the ciphertext to the receivers according to the access policy which she/he decides. If the attributes of the receivers match the access policy, then they can decrypt the ciphertext. This manuscript shows a relation between ABE and identity-based encryption (IBE), and presents a bi-directional conversion between an access structure and identities. By the proposed conversion, the ABE scheme constructed from an IBE scheme will inherit the features, such as constant-size ciphertexts and anonymity, from the IBE scheme, and vice versa. It turns out that the proposed conversion also gives the first ABE achieving access structures with wildcard and constant-size ciphertexts/private keys.

Index Terms—Attribute-based Encryption, Identity-based Encryption, Constant-size Ciphertexts/keys, Hidden Access Policies, Wildcard.

1 Introduction

In an attribute-based encryption (ABE) scheme, if the attributes of users satisfy the access policy (also called access structure) which is decided by other users, then they can decrypt the ciphertext. The first ABE scheme was proposed by Sahai and Waters [30], which is an extended concept from identity-based encryption (IBE). In such a scheme, an encryptor can send the ciphertext to many users by indicating the attributes about the expected receivers, and those users who possess the attributes matching the attributes assigned by the encryptor can successfully decrypt the ciphertext.

We discover an interesting relation between ABE and IBE. The discovery inspires us to present a new generic construction of ABE and IBE. We can construct an ABE scheme from an IBE scheme by the proposed method, and vice versa. The main ideal of our method is to convert an AND-gate only access structure into an identity, and vice versa. Moreover, we also design two algorithms for converting an access structure in DNF into a set of identities, and vice versa. By adopting these two algorithms above, we can construct an ABE scheme from an identity-based broadcast encryption (IBBE) scheme, and vice versa. The proposed conversion method would preserve features, such as constant-size ciphertexts, anonymity, wildcards, etc. Furthermore, our conversion method gives the first ABE achieving hidden access structures with wildcard and constant-size ciphertexts/private keys. It may also imply some impossibility. For example, we can prove that one can never achieve hidden access structures and constant-size ciphertexts simultaneously in an ABE supporting access structures in DNF.

2 Preliminary

In this section, we first give the definition for two access structures and use them in our proposed method. Then we provide the definitions and security models associated with CP-ABE, IBE, and IBBE.

2.1 Access Structures

There are two types of access structures in the proposed method as follows.

Definition 2.1. (AND-gate-only Access Structure) The universe of attributes is denoted by $\mathcal{U}$ and the size of the universe is $|\mathcal{U}|$. We can use an AND-gate-only access structure $\mathbb{A}$ such as (att_1 AND ... AND att_n), where $1 \le n \le |\mathcal{U}|$. It also can be written as a set of attributes, e.g. $\mathbb{A} = \{att_1, att_2, ..., att_n\}$. Let $S = \{X_1, ...X_n\}$, where $1 \le n \le |\mathcal{U}|$, be an attribute set of a user. We say that S satisfies the access structure $\mathbb{A}$ if and only if $att_i = X_i$, for all $1 \le i \le n$, denoted as $S \vDash \mathbb{A}$. (Note that the AND-gate-only access structure is non-monotone.)

Definition 2.2. (Generic Access Structure [4]) Let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^{\{P_1,P_2,...,P_n\}}$ is monotone if $\forall B, C$: if $B \in \mathbb{A}$ and $B \subseteq C$ then $C \in \mathbb{A}$. An access structure (respectively, monotone access structure) is a collection (respectively, monotone collection) $\mathbb{A}$ of non-empty subsets of $\{P_1, P_2, ..., P_n\}$, i.e., $\mathbb{A} \subseteq 2^{\{P_1,P_2,...,P_n\}} \backslash \{\varnothing\}$. The sets in $\mathbb{A}$ are called the authorized sets, and the sets not in $\mathbb{A}$ are called the unauthorized sets. We can also represent the generic access structure as a disjunction of conjunctive clauses, i. e. disjunctive normal form (DNF).

- *C.-I. Fan is with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan.*
  E-mail: cifan@mail.cse.nsysu.edu.tw (The corresponding author)*
- *Y.-F. Tseng is with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan.*
  E-mail: yftseng1989@gmail.com
- *C.-W. Lin is with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan.*
  E-mail: ywenywen220@gmail.com

In our context, the role of the parties is taken by the attributes. Thus, the access structure $\mathbb{A}$ will contain the authorized sets of attributes. In this work, we restrict our attention to monotone access structures.

In our conversion method, we will use another access structure as well, called "and-gate with wildcard." It means that there are "don't care" attributes in an access structure, denoted by symbols "$*$".

2.2 Definition

2.2.1 Ciphertext-Policy Attribute-Based Encryption

A CP-ABE scheme includes the following four algorithms:
- **Setup(1^l):** The private key generator (PKG) takes a security parameter l as an input. Then it outputs a master secret key MK and a public key PK.
- **KeyGen(PK, MK, U):** The PKG takes the master secret key MK, the attribute set of user U, and the public key PK as inputs. It outputs the private key SK_U.
- **Encrypt($M, PK, \mathbb{A}$):** The encryptor takes a message $M \in \{0,1\}^*$, the public key PK, and the access structure $\mathbb{A}$ as inputs. It outputs a ciphertext $CT_{\mathbb{A}}$.
- **Decrypt($CT_{\mathbb{A}}, SK_U$):** The decryptor takes the ciphertext $CT_{\mathbb{A}}$ and the private key SK_U as inputs. It outputs a message M.

These algorithms must satisfy the correctness condition, i. e. for $SK_U \leftarrow$ **KeyGen(PK, MK, U)** and $CT_{\mathbb{A}}$ $\leftarrow$**Encrypt($M, PK, \mathbb{A}$)**, then we can decrypt the ciphertext from **Decrypt($CT_{\mathbb{A}}, SK_U$)** = M if $U \vDash \mathbb{A}$.

2.2.2 Identity-based Encryption

An identity-based encryption (IBE) scheme includes the following four algorithms:
- **Setup(1^l):** The PKG takes a security parameter l as an input. Then it outputs a master secret key MK and a public key PK.
- **KeyGen(PK, MK, ID):** The PKG takes the public key PK, the master secret key MK, and the identity $ID \in \{0,1\}^l$ as inputs. It outputs the private key SK_{ID}.
- **Encrypt(M, PK, ID):** The encryptor takes the identity $ID \in \{0,1\}^l$, the public key PK, and a message $M \in \{0,1\}^*$ as inputs. It outputs a ciphertext CT_{ID}.
- **Decrypt(CT_{ID}, SK_{ID}):** The decryptor takes the ciphertext CT_{ID} and the private key SK_{ID} as inputs. It outputs a message M.

These algorithms must satisfy the correctness condition, i. e. for SK_{ID} $\leftarrow$ **KeyGen(PK, MK, ID)** and CT_{ID} $\leftarrow$**Encrypt(M, PK, ID)**, then we can decrypt the ciphertext from **Decrypt(CT_{ID}, SK_{ID})** = M if $ID \in SK_{ID} = ID \in CT_{ID}$.

2.2.3 Identity-based Broadcast Encrytion

We slightly modify the algorithms *Encrypt* and *Decrypt* from an traditional IBBE scheme. The modified IBBE scheme includes the following four algorithms:
- **Setup(1^l):** The PKG takes a security parameter l as an input. Then it outputs a master secret key MK and a public key PK.
- **KeyGen(PK, MK, ID):** The PKG takes the public key PK, the master secret key MK, and the identity $ID \in \{0,1\}^l$ as inputs. It outputs the private key SK_{ID}.
- **Encrypt(M, PK, S):** The encryptor takes a message $M \in \{0,1\}^*$, the public key PK, and a set of identities $S = \{ID_1, ...ID_n\}$ of receivers as inputs. It outputs a ciphertext CT_S.
- **Decrypt(CT_S, SK_{ID}):** The decryptor takes the ciphertext CT_S and the private key SK_{ID} as inputs. It outputs the message M.

These algorithms must satisfy the correctness condition, i. e. for SK_{ID} $\leftarrow$ **KeyGen(PK, MK, ID)** and CT_S $\leftarrow$**Encrypt(M, PK, S)**, then we can decrypt the ciphertext from **Decrypt(CT_S, SK_{ID})** = M if $ID \in S$.

3 Our Construction

3.1 The relationship between IBE and AND-gate-only ABE

In this section, we discuss the relationship between IBE and ABE. Under certain conditions, IBE and ABE will be equivalent through some transformation. Such relationship can bring some interesting results. For instance, if we consider an AND-gate-only ABE, then our transformation gives the first ABE supporting hidden access policy, constant-size ciphertexts and private keys.

3.1.1 Conversion between access structures and identities

Consider an ABE supporting AND gates only. Note that, in an AND-gate-only ABE scheme, an access structure can be viewed as a non-empty set of attributes for simplicity. Therefore, in the rest of this section, we represent an access structure $\mathbb{A}$ as an attribute set. For such a scheme, we now propose a method to uniquely relate an access structure $\mathbb{A}$ to an identity $ID_{\mathbb{A}}$, whose length equals to $|\mathcal{U}|$, i.e. the size of the universe $\mathcal{U}$. Roughly speaking, given an access structure $\mathbb{A}$, for $i = 1$ to $|\mathcal{U}|$, if an attribute X_i is in $\mathbb{A}$, then set the i-th bit of $ID_{\mathbb{A}}$ as 1; otherwise set it to be 0. For instance, if $\mathcal{U} = \{A, B, C, D\}$ and $\mathbb{A} = A$ AND B AND $D = \{A, B, D\}$, then we can use the above method to construct an identity $ID_A = 1101$. The transformation mentioned above can be inverted, i.e. an identity can also be uniquely converted to an access structure. We give the transformation more precisely as follows and shown in Figure 1.

```
Input: an access structure $\mathbb{A} = \{X_1, ..., X_n\}$, where
       $1 \le n \le |\mathcal{U}|$, a universe $\mathcal{U}$
Output: an identity $ID_A$
Let $ID_A[i]$ be the $i$-th bit of $ID_A$
for $i = 1$ to $|\mathcal{U}|$ do
    if $X_i \in \mathbb{A}$ then
        $ID_A[i] = 1$;
    else
        $ID_A[i] = 0$;
    end
end
Return $ID_A$;
```

Algorithm 1: Algorithm - Γ

```
Input: an identity $ID_A$, a universe $\mathcal{U}$
Output: Output: an access structure
        $\mathbb{A} = \{X_1, ..., X_n\}$, where $1 \le n \le |\mathcal{U}|$
Let $ID_A[i]$ be the $i$-th bit of $ID_A$, and $\mathbb{A}$ be a null
  set.
for $i = 1$ to $|\mathcal{U}|$ do
    if $ID_A[i] = 1$ then
        $\mathbb{A} \leftarrow \mathbb{A} \cup \{X_i\}$;
    end
end
Return $\mathbb{A}$;
```

Algorithm 2: Algorithm - Γ^{-1}

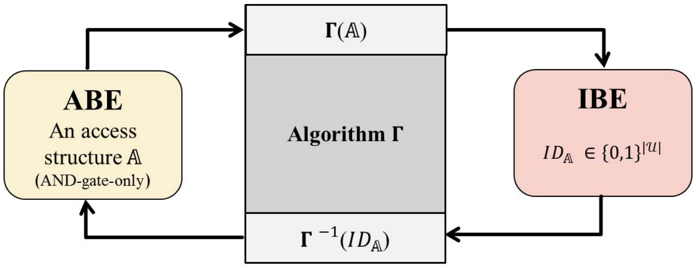
$\Gamma(\mathbb{A})$
ABE
An access structure $\mathbb{A}$
(AND-gate-only)
Algorithm Γ
IBE
$ID_{\mathbb{A}} \in \{0,1\}^{|\mathcal{U}|}$
$\Gamma^{-1}(ID_{\mathbb{A}})$

Fig. 1. The algorithm - Γ

3.1.2 ABE from IBE

In this section, we discuss about the generic construction of an ABE scheme, which supports AND gates only, from an IBE scheme. In such an ABE scheme, the access structure may look like as follows,

School: XYZ **AND** (*Position*: Student **AND** *Grade*: College).

And as mentioned above, we view an access structure as a set of attributes, i.e. School: XYZ, Position: Student, Grade: College . Assume that IBE is an identity-based encryption scheme with four algorithms: **Setup**, **KeyGen**, **Encrypt**, and **Decrypt**. We construct an ABE scheme as follows.

- *Setup*(1^l): Taking a security parameter l as an input, this algorithm runs

  $$(IBE.MK, IBE.PK) \leftarrow IBE.\textbf{Setup}(1^l).$$

  and then sets the master secret key MK and the public key PK of the system as

  $$(MK, PK) = (IBE.MK, IBE.PK).$$

  It outputs the master secret key MK and the public key PK.

- *KeyGen*(PK, MK, U): Taking the master secret key MK, a set of attributes U, and the public key PK as inputs, this algorithm converts the set of attributes U to an identity $ID_U \in \{0,1\}^{|\mathcal{U}|}$ by running the algorithm - Γ, and gets the private key as follows,

  $$IBE.SK_{ID_U} \leftarrow IBE.\textbf{KeyGen}(PK, MK, \Gamma(U)).$$

  It outputs the private key $SK_U = IBE.SK_{ID_U}$.

- *Encrypt*$(M, PK, \mathbb{A})$: Taking a message M, the public key PK, and an access structure $\mathbb{A}$ as inputs, this algorithm converts the access structure $\mathbb{A}$ to an identity $ID_{\mathbb{A}} \in \{0,1\}^{|\mathcal{U}|}$ by running the algorithm - Γ, and gets the ciphertext as follows,

  $$IBE.CT \leftarrow IBE.\textbf{Encrypt}(M, PK, \Gamma(\mathbb{A})).$$

  It outputs a ciphertext $CT = IBE.CT$.

- *Decrypt*(CT, SK_U): Taking the ciphertext CT and the private key SK_U as inputs, this algorithm gets the plaintext by running the decrypt algorithm as follows,

  $$IBE.M \leftarrow IBE.\textbf{Decrypt}(CT, SK_U).$$

  It outputs a message $M = IBE.M$.

3.1.3 IBE from ABE

In this section, we discuss the generic construction of an IBE scheme from an ABE scheme supporting AND gates only.

Assume that ABE is an attribute-based encryption scheme with four algorithms: **Setup**, **KeyGen**, **Encrypt**, and **Decrypt**. We construct an IBE scheme from an ABE scheme as follows.

- *Setup*(1^l): Taking a security parameter l as an input, this algorithm runs

  $$(ABE.MK, ABE.PK) \leftarrow ABE.\textbf{Setup}(1^l).$$

  and then sets the master secret key MK and the public key PK of the system as

  $$(MK, PK) = (ABE.MK, ABE.PK).$$

  It outputs the master secret key MK and the public key PK.

- *KeyGen*(MK, ID_U): Taking the master secret key MK and an identity $ID_U \in \{0,1\}^{|\mathcal{U}|}$ as inputs, this algorithm converts the identity ID_U to the set of attributes U by running the algorithm - Γ^{-1}, and gets the private key as follows,

  $$ABE.SK_U \leftarrow ABE.\textbf{KeyGen}(PK, MK, \Gamma^{-1}(ID_U)).$$

  It outputs the private key $SK_{ID_U} = ABE.SK_U$.

- *Encrypt*(M, PK, ID): Taking a message M, the public key PK, and an identity $ID \in \{0,1\}^{|\mathcal{U}|}$ as inputs, this algorithm converts the identity ID to an access structure $\mathbb{A}$ by running the algorithm - Γ^{-1}, and gets the ciphertext as follows,

  $$ABE.CT \leftarrow ABE.\textbf{Encrypt}(M, PK, \Gamma^{-1}(ID)).$$

  It outputs a ciphertext $CT = ABE.CT$.

- *Decrypt*(CT, SK_{ID_U}): Taking the ciphertext CT and the private key SK_{ID_U} as inputs, this algorithm gets the plaintext by running the decrypt algorithm as follows,

$$ABE.M \leftarrow ABE.\textbf{Decrypt}(CT, SK_{ID_U}).$$

It outputs a message $M = ABE.M$.

3.1.4 Discussion

By transforming an AND-gate-only access structure into an identity, and vice versa, we realize the conversion between ABE and IBE. One can observe that, the features of the encryption scheme may be inheritable through the conversion. For instance, if we use an IBE with receiver anonymity to construct an ABE, then we will have an ABE with hidden access policy. Therefore, we can realize an AND-gate-only ABE with constant-size ciphertexts/private keys and hidden access policy from an anonymous IBE [6], [14].

3.2 The relationship between IBBE and ABE with DNF

In this section, we give a conversion between an IBBE and an ABE with access structures in DNF. Note that the formal definition of an access structure we use here is equivalent to a DNF formula, as mentioned in Definition 2.2. Since every clause in a DNF formula contains only AND gates, we can use the algorithm Γ to transform each clause into an identity. Thus a DNF formula implies a set of identities, which can be viewed as the receiver set in an IBBE scheme. Also, the concept allows us to convert an identity set into an access structure. Following the concept above, we propose a generic construction of ABE from IBBE. Our conversion method gives many interesting results. By adopting the conversion, we can construct the first ABE achieving access structures with wildcard and constant-size ciphertexts/private keys. Our conversion method may also imply some impossibilities. For instance, through our method, we can prove that, if an ABE supports access structures in DNF, then it will never achieve hidden access structures and constant-size ciphertexts simultaneously.

3.2.1 Conversion between access structures in DNF and a set of identities

Consider an ABE with supporting boolean functions in DNF. For such a scheme, we now propose a method to uniquely relate an access structure $\mathbb{A}$ to a set of identities $S = \{ID_1, ...ID_n\}$ for some integer n. We give the transformation more precisely below and shown in Figure 2.

```
Input: an access structure $\mathbb{A} = \{A_1, A_2, \ldots, A_n\} \subseteq 2^{\mathcal{U}}$,
       where $\mathcal{U}$ is the universe
Output: Output: a receiver set $S = \{ID_1, ...ID_n\}$
Let $S$ be a null set
for $i = 1$ to $n$ do
    $ID_i \leftarrow \Gamma(A_i)$;
    $S \leftarrow S \cup \{ID_i\}$;
end
Return $S$;
```

Algorithm 3: Algorithm - Ψ

```
Input: a receiver set $\{ID_1, ...ID_n\}$
Output: Output: an access structure
        $\mathbb{A} = \{A_1, A_2, \ldots, A_n\} \subseteq 2^{\mathcal{U}}$
Let $\mathbb{A}$ be a null set.
for $i = 1$ to $n$ do
    $A_i \leftarrow \Gamma^{-1}(ID_i)$;
    $\mathbb{A} \leftarrow \mathbb{A} \cup \{A_i\}$;
end
Return $\mathbb{A}$;
```

Algorithm 4: Algorithm - Ψ^{-1}

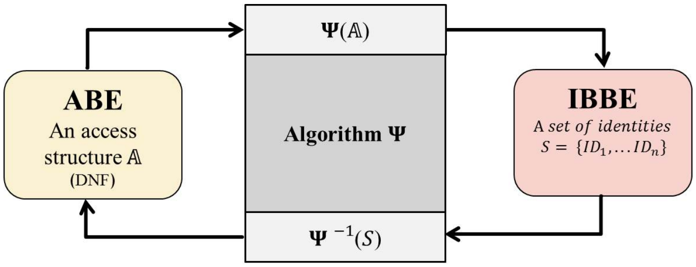

Fig. 2. The algorithm - Ψ

3.2.2 ABE from IBBE

In this section, we discuss the generic construction of an ABE scheme, which supports access structures in DNF, from an IBBE scheme. Assume that $IBBE$ is an identity-based broadcast encryption scheme with the four algorithms: **Setup**, **KeyGen**, **Encrypt**, and **Decrypt**. We construct an ABE scheme from an IBBE scheme as follows.

- *Setup*(1^l): Taking a security parameter l as an input, this algorithm runs

  $$(IBBE.MK, IBBE.PK) \leftarrow IBBE.\textbf{Setup}(1^l).$$

  and then sets the master secret key MK and the public key PK of the system as

  $$(MK, PK) = (IBBE.MK, IBBE.PK).$$

  It outputs the master secret key MK and the public key PK.

- *KeyGen*(PK, MK, U): Taking the public key PK, the master secret key MK, and the set of attributes U as inputs, this algorithm converts the set of attributes U to an identity $ID_U \in \{0,1\}^{|\mathcal{U}|}$ by running the algorithm - Γ mentioned in Algorithm 1, and gets the private key as follows,

  $$IBBE.SK_{ID_U} \leftarrow IBBE.\textbf{KeyGen}(PK, MK, \Gamma(U)).$$

  It outputs the private key $SK_U = IBBE.SK_{ID_U}$.

- *Encrypt*$(M, PK, \mathbb{A})$: Taking a message M, the public key PK, and an access structure $\mathbb{A}$ as inputs, this algorithm converts the access structure $\mathbb{A}$ to a set of identities $S = \{ID_1, ...ID_n\}$ of receivers by running the algorithm - Ψ, and gets the ciphertext as follows,

$$IBBE.CT \leftarrow IBBE.\textbf{Encrypt}(M, PK, \Psi(\mathbb{A})).$$

It outputs a ciphertext $CT = IBBE.CT$.

- *Decrypt(CT, SK_U)*: Taking the ciphertext CT and the private key SK_U as inputs, this algorithm gets the plaintext by computing,

$$IBBE.M \leftarrow IBBE.\textbf{Decrypt}(CT, SK_U).$$

It outputs a message $M = IBBE.M$.

3.2.3 IBBE from ABE

Using the algorithm Ψ^{-1}, we can also give a generic construction of IBBE from ABE. Assume that ABE is an attribute-based encryption scheme with the four algorithms: **Setup**, **KeyGen**, **Encrypt**, and **Decrypt**. We construct an IBBE scheme from an ABE scheme as follows.

- *Setup(1^l)*: Taking a security parameter l as an input, this algorithm runs

$$(ABE.MK, ABE.PK) \leftarrow ABE.\textbf{Setup}(1^l).$$

and then sets the master secret key MK and the public key PK of the system as

$$(MK, PK) = (ABE.MK, ABE.PK).$$

It outputs the master secret key MK and the public key PK.

- *KeyGen(PK, MK, ID_i)*: Taking the public key PK, the master secret key MK, and the identity $ID_U \in \{0,1\}^{|\mathcal{U}|}$ as inputs, this algorithm converts the identity ID_U to the set of attributes U by running the algorithm - Γ^{-1} mentioned in Algorithm 2, and gets the private key as follows,

$$ABE.SK_U \leftarrow \\ ABE.\textbf{KeyGen}(PK, MK, \Gamma^{-1}(ID_U)).$$

It outputs the private key $SK_{ID_U} = ABE.SK_U$.

- *Encrypt(M, PK, S)*: Taking a message M, the public key PK, and a set of identities $S = \{ID_1, ...ID_n\}$ of receivers as inputs. This algorithm converts the set of identities S to the the access structure $\mathbb{A}$ by running the algorithm - Ψ^{-1}, and gets the ciphertext as follows,

$$ABE.CT \leftarrow ABE.\textbf{Encrypt}(M, PK, \Psi^{-1}(S)).$$

It outputs a ciphertext $CT = ABE.CT$.

- *Decrypt(CT, SK_{ID_U})*: Taking the ciphertext CT and the private key SK_{ID_U} as inputs, this algorithm gets the plaintext by computing,

$$ABE.M \leftarrow ABE.\textbf{Decrypt}(CT, SK_{ID_U}).$$

It outputs a message $M = ABE.M$.

3.2.4 Discussion

In this section, we discuss the effect about the transformation between ABE and IBBE. According to the method for converting an access structure in DNF into a set of identities, and vice versa, as mentioned above, we can realize a generic construction of an ABE scheme from an IBBE scheme, and vice versa. Furthermore, this conversion method will bring some interesting results as follows.

- We can obtain an ABE with hidden access policies from an IBBE with receiver anonymity, and vice versa.
- We can use an IBBE with constant-size ciphertexts/private keys to construct an ABE with constant-size ciphertexts/private keys, and vice versa.
- We can realize an AND-gate-only ABE with wildcard.
  The conversion method is shown below. Consider an AND-gate-only ABE scheme with wildcard from an IBBE. It means that there are "don't care" attributes in an access structure. Let the symbol "$*$" denote wildcard, e.g. an attribute a^* is a "don't care" attribute in access structure $\mathbb{A}$. For such a scheme, given the access structure $\mathbb{A}$, if there is a "don't care" attribute X_i^* in $\mathbb{A}$, then we will obtain a pair of identities (ID_A, ID_B) by our converted method, where the value of the i-th bit in ID_A is 1 and the value of the i-th bit in ID_B is 0. For instance, if $\mathcal{U} = \{a, b, c, d\}$ and $\mathbb{A} = \{a, c^*, d\}$, then we can obtain two different identities, $ID_A = 1011$ and $ID_B = 1001$, by applying the above method. And the ciphertext is generated by the encryption algorithm of IBBE with the receiver set $S = \{ID_A, ID_B\}$. Moreover, if we take advantage of an IBBE with constant-size ciphertexts/private keys [10], [39], we can obtain the first AND-gate-only ABE with wildcard supporting constant-size ciphertexts/private keys.
- In 2012, Kiayias and Samari [19] have proved that the size of a ciphertext in an anonymous broadcast encryption is at least of linear size in the number of receivers. Following their result, we can use our transformation technique to prove that there is no ABE supporting access structures in DNF that can achieve hidden access structures and constant-size ciphertexts simultaneously. This is because that if there exist such schemes, we can use the proposed method to obtain an anonymous IBBE with constant-size ciphertexts, which will go against the result of [19].

For the results above, we conclude that if there is an IBE scheme with some features, then the ABE scheme will inherit those features from the IBE by our conversion methods.

4 Conclusion

In this manusceipt, we have proposed the algorithms for the transformation between access structures and identities. Generic constructions of ABE and IBE are given in the manuscript as well. Our conversion methods bring some interesting results in constant-size ciphertexts, anonymity, wildcards, etc. The ABE scheme will inherit from the properties of the underlying IBE/IBBE scheme, and vice versa. In the future, we will provide the proofs for the uniqueness of the proposed conversion methods and the CCA security proofs for confidentiality and anonymity to demonstrate the security of the proposed conversion methods.

References

[1] A. Beimel. Secure schemes for secret sharing and key distribution. PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[2] X. Boyen and B. Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *Advances in Cryptology-CRYPTO 2006*, pages 290–307. Springer, 2006.

[3] C. Delerablée. Identity-based broadcast encryption with constant size ciphertexts and private keys. In *Advances in Cryptology–ASIACRYPT 2007*, pages 200–215. Springer, 2007.

[4] C. Gentry. Practical identity-based encryption without random oracles. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 445–464. Springer, 2006.

[5] A. Kiayias and K. Samari. Lower bounds for private broadcast encryption. In *International Workshop on Information Hiding*, pages 176–190. Springer, 2012.

[6] A. Sahai and B. Waters. Fuzzy identity-based encryption. In *Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques*, EUROCRYPT'05, pages 457–473, 2005.

[7] L. Zhang, Y. Hu, and Q. Wu. Adaptively secure identity-based broadcast encryption with constant size private keys and ciphertexts from the subgroups. *Mathematical and computer Modelling*, 55(1):12–18, 2012.

Chun-I Fan received the M.S. degree in computer science and information engineering from the National Chiao Tung University, Hsinchu, Taiwan, in 1993, and the Ph.D. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 1998. From 1999 to 2003, he was an Associate Researcher and a Project Leader with Telecommunication Laboratories, Chunghwa Telecom Company, Ltd., Taoyuan, Taiwan. In 2003, he joined the faculty of the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, where has been a Full Professor since 2010. His current research interests include applied cryptology, cryptographic protocols, and information and communication security. Prof. Fan is the Deputy Chairman of the Chinese Cryptology and Information Security Association, and the Chief Executive Officer (CEO) of "Aim for the Top University Plan" Office at National Sun Yat-sen University. He was the recipient of the Best Student Paper Awards from the National Conference on Information Security in 1998, the Dragon Ph.D. Thesis Award from Acer Foundation, the Best Ph.D. Thesis Award from the Institute of Information and Computing Machinery in 1999, and the Engineering Professors Award from Chinese Institute of Engineers - Kaohsiung Chapter in 2016. Prof. Fan is also an Outstanding Faculty in Academic Research in National Sun Yat-sen University.

Yi-Fan Tseng was born in Kaohsiung, Taiwan. He received the MS degree in computer science and engineering from National Sun Yat-sen University, Taiwan, in 2014, and now is a PhD student in the same department. His research interests include cloud computing and security, network and communication security, information security, cryptographic protocols, and applied cryptography.

Chih-Wen Lin was born in Kaohsiung, Taiwan. She now is a MS student in computer science and engineering from National Sun Yat-sen University, Kaohsiung, Taiwan, Her research interests include cloud computing and cloud storage, network and communication security, and applied cryptography.