Privacy-aware Attribute-based Encryption with User Accountability*

Jin Li[1], Kui Ren[1], Bo Zhu[2], and Zhiguo Wan[3]

[1] Department of ECE, Illinois Institute of Technology, USA
{jinli,kren}@ece.iit.edu
[2] Canada Concordia University, Canada
zhubo@ciise.concordia.ca
[3] Tsinghua University, China
wanzhiguo@tsinghua.edu.cn

Abstract. As a new public key primitive, attribute-based encryption (ABE) is envisioned to be a promising tool for implementing fine-grained access control. To further address the concern of user access privacy, privacy-aware ABE schemes are being developed to achieve hidden access policy recently. For the purpose of secure access control, there is, however, still one critical functionality missing in the existing ABE schemes, which is user accountability. Currently, no ABE scheme can completely prevent the problem of illegal key sharing among users. In this paper, we tackle this problem by firstly proposing the notion of accountable, anonymous, and ciphertext-policy ABE (CP-A^{3}BE, in short) and then giving out a concrete construction. We start by improving the state-of-the-art of anonymous CP-ABE to obtain shorter public parameters and ciphertext length. In the proposed CP-A^{3}BE construction, user accountability can be achieved in black-box model by embedding additional user-specific information into the attribute private key issued to that user, while still maintaining hidden access policy. The proposed constructions are provably secure.

Keywords: Access control, Anonymity, Attribute-based, Ciphertext-policy, Accountability

1 Introduction

Today's computing and electronic technology innovations have unprecedentedly enabled ubiquitous information generation, processing, and distribution in both volume and speed. Vast amounts of information resources are made available and readily accessible to individuals and organizations through various computer systems and the Internet. This trend, however, also poses new challenges in designing suitable secure access control mechanisms. Generally, among the

* An extended abstract of this paper been accepted by the Information Security Conference 2009, Italy.

various requirements, today's access control schemes should at least meet the following ones: 1) fine-grained access policy, 2) protection of user privacy, and 3) assurance of user accountability.

Recently, the notion of ABE, which was proposed by Sahai and Waters [27], has attracted much attention in the research community to design flexible and scalable access control systems. For the first time, ABE enables public key based one-to-many encryption. Therefore, it is envisioned as a highly promising public key primitive for realizing scalable and fine-grained access control systems, where differential yet flexible access rights can be assigned to individual users. To address complex and general access policy, two kinds of ABE have been proposed : key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE). In KP-ABE, access policy is assigned in attribute private key, whereas, in CP-ABE, the access policy is specified in the ciphertext.

Besides fine-grained access policy, there is an increasing need to protect user privacy in today's access control systems. To address this problem, anonymous ABE was introduced in [20, 30] and further improved by [24]. Anonymous ABE has a wide range of applications. For example, in some military circumstances, the access policy itself could be sensitive information. Therefore, to share resources with users possessing certain attribute-policy, anonymous ABE scheme can be applied to encrypt the resources while keeping the access policy specified in the ciphertext hidden.

Although the anonymous ABE can provide secure anonymous access control, before its widely deployment, another important security aspect, user accountability, has to be formally addressed. In particular, the problem of key abuse, *i.e.*, illegal key sharing among users, should be prevented. This problem is extremely important as in an ABE-based access control system, the attribute private keys directly imply users' privileges to the protected resources. The dishonest users may share their attribute private keys with other users, who do not have these privileges. They can just directly give away part of their original or transformed keys such that nobody can tell who has distributed these keys. Consequently, it renders the system useless. To the best of our knowledge, the issue of user accountability in access control system based on ABE is quite new in the literature and has not been solved yet. Such key abuse problems exist in all current access control schemes constructed from ABE as the attribute private keys assigned to users are never designed to be linked to any user specific information except the commonly shared user attributes. This is the reason why attribute private key can be abused by users without being detected.

To construct privacy-aware fine-grained ABE with user accountability, in this paper, the notion of accountable and anonymous CP-ABE (CP-A^{3}BE) is proposed. This is achieved by binding user identity in the attribute private key. CP-A^{3}BE can be applied to prevent the key sharing among users based on the following observation. If the user shares his attribute private key, the user's identity will be detected from the pirate device embedded with the shared private key. In normal encryption of CP-A^{3}BE, the message is encrypted with respect to some ciphertext-policy, in which the identity part is for all users. Any users can decrypt the ciphertext as long as their attribute private keys satisfy this policy. In tracing encryption, a message is encrypted to users with some ciphertext-policy, in which the identity part is for the suspicious users. In this algorithm, only the suspicious users with attribute private keys that satisfy this ciphertext-policy can decrypt the ciphertext. Due to the anonymity of CP-A^{3}BE, the tracing encryption algorithm and normal encryption algorithm are indistinguishable from the viewpoint of any user. Specifically, given a pirate device and the detected attributes embedded, the attribute center, who is in charge of the attribute private key issuing, can find the suspicious identity list of users possessing these attributes. To pinpoint the identity of the user sharing the attribute private key in the pirate device, the attribute center applies the tracing algorithm to encrypt a message with respect to the attributes and identities in the suspicious list. Computing in this way, the identity could be found if the ciphertext for some specific identity can be decrypted by this pirate device.

1.1 Related Work

Since the introduction of ABE in implementing fine-grained access control systems, a lot of works have been proposed to design flexible ABE schemes. There are two methods to realize the fine-grained access control based on ABE: KP-ABE and CP-ABE. They were both mentioned in [18] by Goyal *et al.* In KP-ABE, each attribute private key is associated with an access structure that specifies which type of ciphertexts the key is able to decrypt, and ciphertext is labeled with sets of attributes. In a CP-ABE system, a user's key is associated with a set of attributes and an encrypted ciphertext will specify an access policy over attributes. CP-ABE is different from KP-ABE in the sense that, in CP-ABE, it is the encryptor who assigns certain access policy for the ciphertext. When a message is being encrypted, it will be associated with an access structure over a predefined set of attributes. In CP-ABE, user will only be able to decrypt a given ciphertext if its attributes pass through the corresponding access structure specified in the ciphertext. The first KP-ABE construction [18] realized the monotonic access structures for key policies. To enable more flexible access policy, Ostrovsky *et al.* [25] presented the first KP-ABE system that supports the expression of non-monotone formulas in key policies. However, KP-ABE is less flexible than CP-ABE because the policy is determined once the user's attribute private key is issued. Later, Bethencourt *et al.* [3] proposed the first CP-ABE construction. However, the construction [3] is only proved secure under the generic group model. To overcome this weakness, Cheung and Newport [11] presented another construction that is proved to be secure under the standard model. The construction supports the types of access structures that are represented by AND of different attributes. Later, in [16], the authors gave another construction for more advanced access structures based on number theoretic assumption. To further achieve receiver-anonymity, Boneh and Waters [8] proposed a predicate encryption scheme based on the primitive called Hidden Vector Encryption. The scheme in [8] can also realize the anonymous CP-ABE by using the opposite semantics of subset predicates. Katz, Sahai, and Waters [21] proposed a novel predicate encryption scheme supporting inner product predicates. Their scheme is very general and can achieve both KP-ABE and hidden CP-ABE schemes. However, the constructions of [8, 21] are very inefficient compared to [24]. Recently, several attempts [22, 19, 31] have been made to address the accountability problem in ABE-based access control. In [31], they considered how to defend the key-abuse problem in KP-ABE schemes while only achieving privacy for part of the attributes. In [19], another trusted party was introduced in the protocol and each decryption operation should get assistance from the trusted party. As a result, the third party has to handle a huge amount of load, which greatly limits their application in the real world. The work [22] does not rely on the existence of trusted party. Instead, they used the technique of identity-based wildcard encryption [1] to achieve the accountability for the user. However, a strong assumption of well-formedness decryption key is required in the pirate device. Therefore, the result in [22] is still not practical enough. In our work, these two drawbacks: the introduction of trusted party and strong assumption of white-box, can be avoided. In addition to the accountability, the user privacy, is also considered in our constructions, which cannot be realized in [22, 19].

ORGANIZATION. Some preliminaries are given in Section 2, including the syntax and basic mathematic tools used in the paper. In Section 3, we propose two improved constructions of privacy-aware CP-ABE. In Section 4, the CP-A^{3}BE construction is proposed to realize the fine-grained access control system with user privacy and accountability. This paper ends with concluding remarks.

2 Preliminaries

2.1 Syntax

System Model Before introducing CP-A^{3}BE, we first give the system model for anonymous CP-ABE. In the anonymous CP-ABE architecture, there are two entities: attribute center (AC) and user. AC is in charge of the issue of attribute private key to users requesting them. The user, who wants to access data, should get the attribute private key from AC in advance. The encryptor can specify the ciphertext-policy such that only users whose attribute private keys satisfy the policy are able to decrypt the ciphertext. In addition, the ciphertext-policy is kept hidden. The users with an attribute private key are able to check whether his attributes satisfy the ciphertext-policy or not. In our system model, a binary relation R is defined as part of public parameter according to the concrete requirements of anonymous CP-ABE. We denote it by $R(L, W) = 1$ if the attribute list L satisfies ciphertext-policy W.

Definition 1. *An anonymous CP-ABE system consists of four algorithms, namely, Setup, KeyGen, Encryption, and Decryption, which are defined as follows:*

Setup(1^λ). The setup algorithm, on input security parameter 1^λ, outputs a master secret key sk and public key pk.

KeyGen(L, sk). The key generation algorithm, on input attribute list L and master key sk, outputs sk_L as the attribute private key for L.

Enc(M, W, pk). The encryption algorithm, on input a message M together with ciphertext-policy W, outputs $\mathcal{C}$, as the encryption on M with respect to W.

Dec($\mathcal{C}$, sk_L). The decryption algorithm, on input the ciphertext $\mathcal{C}$ and the attribute private key sk_L, outputs M if $R(L, W) = 1$. Otherwise, it returns $\perp$.

Adversary Model The goal of adversary in anonymous CP-ABE system can be either one of the following 1) Extracting information of plaintext from the ciphertext. Here, the adversary is allowed to control some users and access their attribute private keys that do not match the ciphertext-policy; 2) Distinguishing underlying access-policy in the ciphertext.

The two goals of adversary can be integrated in the indistinguishability against ciphertext-policy and chosen ciphertext attacks (CP-IND-CCA). In this work, a weaker notion, called indistinguishability against selective ciphertext-policy and chosen message attack (sCP-IND-CPA) [3, 11, 18], will be used. The definition is the same with CP-IND-CCA, except in sCP-IND-CPA, the adversary has to submit its challenge attributes before the setup phase. Furthermore, the decryption oracle is not available to the adversary. The formal definition is given based on the following sCP-IND-CPA game involving an adversary $\mathcal{A}$:

Initial. The adversary commits to the challenge ciphertext policies W_0^*, W_1^* before setup algorithm.

Setup. Choose a sufficiently large security parameter 1^λ, and run *Setup* to get a master secret key *sk* and public key *pk*. Retain *sk* and give *pk* to $\mathcal{A}$;

Phase 1. $\mathcal{A}$ can perform a polynomially bounded number of queries to key generation oracle on attributes L, the only restriction on L is that, $R(L, W_0^*) = R(L, W_1^*) = 0$ or $R(L, W_0^*) = R(L, W_1^*) = 1$;

Challenge. $\mathcal{A}$ outputs two messages M_0, M_1 on which it wishes to be challenged with respect to W_0^* and W_1^*. It requires that $M_0 = M_1$ if any attribute private key on L satisfying $R(L, W_0^*) = R(L, W_1^*) = 1$ has been queried. The challenger randomly chooses a bit $b \in \{0, 1\}$, computes $\mathcal{C} = Enc(M_b, W_b^*, pk)$ and sends $\mathcal{C}$ to $\mathcal{A}$;

Phase 2. $\mathcal{A}$ continues to issue queries to the key generation oracle, with the same restriction as before;

Guess. Finally, $\mathcal{A}$ outputs a guess bit b'.

$\mathcal{A}$ wins the game if $b = b'$. The advantage of $\mathcal{A}$ in Game sCP-IND-CPA is defined as the probability that $\mathcal{A}$ wins the game minus $1/2$. This model can be considered to be analogous to the selective-ID model [4] utilized in IBE protocols. In their security model, the adversary should commit to the challenge identity ID before *Setup* phase.

Definition 2. *An anonymous CP-ABE satisfies sCP-IND-CPA if no polynomial time adversary can break the above game.*

In CP-A^{3}BE, as explained, we consider how to achieve user accountability in addition to fine-grained access-policy and user privacy. The system model for CP-A^{3}BE is the same with anonymous CP-ABE, except here the algorithm for tracing is added.

Trace. *This algorithm is applied to trace an attribute private key in black-box to its original holder. It takes as input a pirate device, and outputs identity associated with this attribute private key in the pirate device.*

Because the CP-A^{3}BE is still one kind of anonymous CP-ABE, the adversary model and security requirement of sCP-IND-CPA are defined in the same way as anonymous CP-ABE. The only difference lies in the ciphertext-policy where it is defined by two parts $W = W' \vee \overline{W}$: The first part is the same as in the anonymous CP-ABE while the second part is for the identity. That is, $\overline{W}$ could be $*$ or specific ID. Accordingly, the challenge ciphertext would be $W_0^* = W_{0,1}^* \| W_{0,2}^*$ and $W_1^* = W_{1,1}^* \| W_{1,2}^*$. This kind of security implies that if a user has an attribute private key on attributes L for identity ID, it cannot decrypt the ciphertext encrypted for the ciphertext-policy W if $R(L\|ID, W) = 0$. Additionally, to trace the identity who shares the attribute private key, the tracing algorithm should be indistinguishable with the normal encryption algorithm to avoid detection by the pirate device.

2.2 Basic Mathematic Tools

We give a brief review on the property of pairings and some candidates of hard problem from pairings. Let $\mathbb{G}_1, \mathbb{G}_2$ be cyclic groups of prime order p, writing the group action multiplicatively. Let g be a generator of $\mathbb{G}_1$, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a map with the following properties. *Bilinearity*: $\hat{e}(g_1^a, g_2^b) = \hat{e}(g_1, g_2)^{ab}$ for all $g_1, g_2 \in \mathbb{G}_1$, and $a, b \in_R \mathbb{Z}_p$; *Non-degeneracy*: There exist $g_1, g_2 \in \mathbb{G}_1$ such that $\hat{e}(g_1, g_2) \neq 1$. In other words, the map does not send all pairs in $\mathbb{G}_1 \times \mathbb{G}_1$ to the identity in $\mathbb{G}_2$; *Computability*: There is an efficient algorithm to compute $\hat{e}(g_1, g_2)$ for all $g_1, g_2 \in \mathbb{G}_1$.

3 Improved Privacy-aware CP-ABE Constructions

3.1 Anonymous CP-ABE with Short Public Parameters

First, we give a construction of anonymous CP-ABE with short public parameters. In this work, the ciphertext-policy has the same fine-grained access structure (ciphertext-policy) with CP-ABE scheme [11]. Details of the access structure in [11] are described below. Assume that the total number of attributes in the system is n and the universal attributes set is $U = \{w_1, w_2, \cdots, w_n\}$. To encrypt a message, it specifies the ciphertext-policy $W = [W_1, W_2, \cdots, W_n]$. The notion of wildcard $*$ in the ciphtertext policies means the value of "don't care". For example, let the ciphertext-policy $W = [1, 0, 1, *]$ when $n = 4$. This ciphertext-policy means that the recipient who wants to decrypt must have the value 1 for

W_1 and W_3, the value 0 for W_2, and any possible values for W_4. Therefore, if the receiver has an attribute private key for $[1, 0, 1, 0]$, it can decrypt the ciphertext because the first three values for W_1, W_2 and W_3 are equivalent to the corresponding values in ciphertext-policy. Moreover, the fourth value 0 in the private key satisfies the ciphertext-policy because $W_4 = *$. If an attribute private key is associated with the attribute list $[1, 1, 1, 0]$, this attribute private key will not match the ciphertext-policy since $W_2 \neq 0$. To be more generalized, given an attribute list $L = [L_1, L_2, \cdots, L_n]$ and a ciphertext-policy $W = [W_1, W_2, \cdots, W_n]$, we say that L matches W if for all $i \in [1, n]$, $L_i \in W_i$, *i.e.*, $L_i = W_i$ or $W_i = *$. In [11], each attribute can take two values 1 and 0. In our construction, we generalize the access structures such that each attribute can take two or more values. More formally, let $S_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,n_i}\}$ be a set of possible values for attribute w_i where n_i is the number of the possible values for w_i. Then the attribute list L for a user is $L = [L_1, L_2, \cdots, L_n]$ where $L_i \in S_i$ for $1 \leq i \leq n$, and the generalized ciphertext policy W is $W = [W_1, W_2, \cdots, W_n]$. The attribute list L satisfies the ciphertext-policy W (that is, $R(L, W) = 1$) if $L_i = W_i$ or $W_i = *$ for $1 \leq i \leq n$.

Main Idea We use $H(i \| v_{i,k_i})$ to denote the k_i-th value v_{i,k_i} for the i-th attribute. Instead, in [24], they used different public keys to denote the universal attributes, which makes the size of public parameters to be $O(N)$, where N is the total number of all attribute values defined in the system. To keep the receiver-anonymity in ciphertext, we cannot just replace the public key pk_{i,k_i} with $H(i \| v_{i,k_i})$ directly. The ciphertext of the v_{i,k_i} is computed by splitting the random value used in encryption into two parts $H(1 \| i \| v_{i,k_i}))$ and $H(0 \| i \| v_{i,k_i})$, together with two different generators g_1 and g_2. The reason for choosing different generators is to prevent the public verifiability of the ciphertext's validity, which achieves hidden policy. User can only check whether his own attribute private key matches the ciphertext-policy. Furthermore, the user cannot check if the ciphertext is valid or not with respect to other attribute list he does not have, which keeps the ciphertext-policy hidden. The four algorithms of our scheme are defined as follows.

Setup Let $\mathbb{G}_1, \mathbb{G}_2$ be cyclic groups of prime order p, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a pairing defined in Section 2. Let g_1, g_2 be random elements from $\mathbb{G}_0$. Define a hash function $H : \{0,1\}^* \to \mathbb{G}_0$. Assume there are n attributes in universe. That is to say, let the universal attributes set be $U = \{\omega_1, \omega_2, \cdots, \omega_n\}$. And, each attribute has multiple values, where S_i is the multi-value set for ω_i and $| S_i |= n_i$. This algorithm also chooses a random number $\alpha \in \mathbb{Z}_p$ and computes $T = \hat{e}(g_1, g_2)^{\alpha}$. The system public parameter is $para = (g_1, g_2, T, H)$. The system master secret key msk is α, which is only known to AC.

KeyGen To generate an attribute private key for user with attribute list $L = [L_1, L_2, \cdots, L_n] = [v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$, AC picks up random $s_1, s_2, \cdots, s_{n-1} \in \mathbb{Z}_p^*$ and computes $s_n = \alpha - \sum_{i=1}^{n-1} s_i \bmod p$. It also chooses n random numbers $\{r_i\}_{1 \leq i \leq n} \in \mathbb{Z}_p^*$ and computes the attribute private key with respect to L as

$sk_L = \{(d_{i0}, d_{i1}, d'_{i0}, d'_{i1})\}$=$\{(g_2^{s_i} H(1\|i\|v_{i,k_i})^{r_i}, g_1^{r_i}, g_1^{s_i} H(0\|i\|v_{i,k_i})^{r'_i}, g_2^{r'_i})\}_{1\le i\le n}$. The validity of $sk_L = \{(d_{i0}, d_{i1}, d'_{i0}, d'_{i1})\}_{1\le i\le n}$ can be verified through the following equation: $\prod_{i=1}^{n} \frac{\hat{e}(d_{i0},g_1)\hat{e}(d'_{i0},g_2)}{\hat{e}(d_{i1},H(1\|i\|v_{i,k_i}))\hat{e}(d'_{i1},H(0\|i\|v_{i,k_i}))} = T$.

Enc To encrypt a message $M \in \mathbb{G}_2$ under ciphertext-policy $W = [W_1, W_2, \cdots, W_n]$, pick up a random value $z \in \mathbb{Z}_p$ and compute $C_0 = MT^z$. For each $1 \le i \le n$ and $1 \le t_i \le n_i$,

† if $v_{i,t_i} \in W_i$, choose $z_{i,t_i} \in \mathbb{Z}_p^*$ and compute $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$
$= ((H(1\|i\|v_{i,t_i}))^{z_{i,t_i}}, g_1^{z_{i,t_i}}, (H(0\|i\|v_{i,t_i}))^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}})$;
‡ if $v_{i,t_i} \notin W_i$, choose randomly z_{i,t_i}, $z'_{i,t_i} \in \mathbb{Z}_p^*$ and compute
$(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$=$((H(1\|i\|v_{i,t_i}))^{z_{i,t_i}}, g_1^{z_{i,t_i}}, (H(0\|i\|v_{i,t_i}))^{z'_{i,t_i}}, g_2^{z'_{i,t_i}})$.

Finally, output the ciphertext as $C = (C_0, \{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})\}$ for $1 \le t_i \le n_i$ and $1 \le i \le n$.

Dec Assume a user has an attribute private key $sk_L = \{(d_{i0}, d_{i1}, d'_{i0}, d'_{i1})\}_{1\le i\le n}$ on attribute list $L = [v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$. To decrypt the ciphertext $C = (C_0, \{\{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})\}_{1\le t_i\le n_i}\}_{1\le i\le n})$ without the information of ciphertext-policy W, the user first computes $C' = \prod_{i=1}^{n} \frac{\hat{e}(C_{i,k_i,1},d_{i0})\hat{e}(C'_{i,k_i,1},d'_{i0})}{\hat{e}(C_{i,k_i,0},d_{i1})\hat{e}(C'_{i,k_i,0},d'_{i1})}$ and then decrypts the ciphertext as $M = C_0/C'$.

To check whether the decryption is correct or not, redundancy can be added in the plaintext such that the user knows if his attribute private key matches the ciphertext-policy. There are many ways to add redundancy, such as appending 0^λ to the message for security parameter λ. After decryption, the user can verify the correctness of decryption by checking whether the first λ is 0^λ.

3.1.1 Security Result

Before giving security result for the anonymous CP-ABE, we show definitions of the following problems and assumptions based on the bilinear groups.

DBDH Problem. The Decision Bilinear Diffie-Hellman (DBDH) problem is that, given g, g^x, g^y, $g^z \in \mathbb{G}_1$ for unknown random $x, y, z \in \mathbb{Z}_p^*$, $T \in \mathbb{G}_2$, to decide if $T = \hat{e}(g, g)^{xyz}$.

We say that a polynomial-time adversary $\mathcal{A}$ has advantage ϵ in solving the DBDH problem in groups $(\mathbb{G}_1, \mathbb{G}_2)$ if $\mid Pr[\mathcal{A}(g, g^x, g^y, g^z, \hat{e}(g, g)^{xyz}) = 1] - Pr[\mathcal{A}(g, g^x, g^y, g^z, \hat{e}(g, g)^r) = 1] \mid \ge 2\epsilon$, where the probability is taken over the randomly chosen x, y, z, r and the random bits consumed by $\mathcal{A}$. (t, ϵ)-DBDH assumption holds in $(\mathbb{G}_1, \mathbb{G}_2)$ if no t-time algorithm has the probability at least ϵ in solving the DBDH problem for non-negligible ϵ.

D-Linear Problem. Let $z_1, z_2, z_3, z_4, z \in \mathbb{Z}_p$ be chosen at random and $g \in \mathbb{G}_1$ be a generator. The D-Linear problem is that given $g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, g^{z_2 z_4}, T$, to decide if $T = g^{z_3+z_4}$.

We say that a polynomial-time adversary $\mathcal{A}$ has advantage ϵ in solving the D-Linear Problem in groups $(\mathbb{G}_1, \mathbb{G}_2)$ if $\mid Pr[\mathcal{A}(g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, g^{z_2 z_4}, T] - Pr[g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, g^{z_2 z_4}, g^{z_3+z_4}] \mid \geq 2\epsilon$, where the probability is taken over the randomly chosen z_1, z_2, z_3, z_4 and the random bits consumed by $\mathcal{A}$. (t, ϵ)-D-Linear assumption holds in $(\mathbb{G}_1, \mathbb{G}_2)$ if no t-time algorithm has the probability at least ϵ in solving the D-Linear problem for non-negligible ϵ. The D-Linear assumption was first proposed in [5] and one of its variants will be used in the proof. We have the following security result for the above construction:

Theorem 1. *The Anonymous CP-ABE construction is secure in* sCP-IND-CPA *model, under the* DBDH *and D-Linear assumption.*

Proof. The proof is given in Appendix B.

To achieve IND-sCP-CCA security in the standard model, we can use the technique of simulation-sound NIZK proofs [26]. The most efficient transformation from IND-sCP-CPA to IND-sCP-CCA is to use the Fujisaki-Okamoto technique [13], which adds only a little computation overhead on the original scheme. So, the resulted IND-sCP-CCA anonymous CP-ABE construction is very efficient.

3.2 Anonymous CP-ABE with Shorter Ciphertext

To further reduce the ciphertext size of the above scheme, we propose another construction by expressing the attribute values as bit pattern. The ciphertext-policy W_i can be only one value or $*$. This technique, together with the above construction, will be applied to design the CP-A^{3}BE scheme in the next Section.
Main Idea The value set S_i for each attribute ω_i is expressed using bit pattern. Suppose the length of $\mid S_i \mid$ is ρ_i. Instead of computing the ciphertext for each value in S_i, we encrypt the message with respect to 0 or 1 for each bit by using the anonymous CP-ABE technique above. It is indistinguishable that some bit is encrypted for $0, 1$, or $*$, from the viewpoint of users. Without loss of generality, the values in set S_i can be mapped to $\{1, 2, \cdots, \mid S_i \mid\}$ with some injective function. As a result, the ciphertext size can be reduced from $O(\mid S_i \mid)$ to $O(\log \mid S_i \mid)$. Here, for each i, the ciphertext policy W_i can be some v_{i,k_i} in S_i or $*$. To encrypt a message, it specifies the ciphertext-policy $W = [W_1, W_2, \cdots, W_n]$ with AND gate as above.

Setup Assume there are n attributes in universe denoted by $U = \{\omega_1, \omega_2, \cdots, \omega_n\}$. Each attribute has multiple values. Let S_i be the multi-value set for ω_i and $\mid S_i \mid = n_i$. Assume the length of $\mid S_i \mid$ is ρ_i. The system public parameter is the same as the above scheme $para = (g_1, g_2, T, H)$. The system master secret key msk is α.

KeyGen To generate an attribute private key for user with attribute list $L = [L_1, L_2, \cdots, L_n] = [v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$, AC picks up random $s_1, s_2, \cdots, s_{n-1} \in \mathbb{Z}_p^*$ and computes $s_n = \alpha - \sum_{i=1}^{n-1} s_i \bmod p$. For each $1 \leq i \leq n$, the following steps are taken:

1. AC picks up $s_{i,1}, s_{i,2}, \cdots, s_{i,\rho_i} \in Z_p^*$ such that $s_i = \sum_{k=1}^{\rho_i} s_{i,k} \bmod p$;
2. For each $1 \le t_i \le \rho_i$, AC chooses random numbers (r_{i,t_i}, r'_{i,t_i}) from $\mathbb{Z}_p^*$. Assume $v_{i,k_i} = (I_{i,1}, I_{i,2}, \cdots, I_{i,\rho_i}) \in \{0,1\}^{\rho_i}$. AC computes the attribute private key for v_{i,k_i} as

$$\begin{aligned} D_i &= \{(d_{i,t_i,0}, d_{i,t_i,1}, d'_{i,t_i,0}, d'_{i,t_i,1})\}_{1\le t_i \le \rho_i} \\ &= (g_2^{s_{i,t_i}} H(1\|i\|t_i\|I_{i,t_i})^{r_{i,t_i}}, g_1^{r_{i,t_i}}, g_1^{s_{i,t_i}} H(0\|i\|t_i\|I_{i,t_i})^{r'_{i,t_i}}, g_2^{r'_{i,t_i}})_{1\le t_i\le\rho_i}. \end{aligned}$$

The validity of $sk_L = \{D_i\}_{1\le i\le n}$ can be also verified in a similar way as the construction in Section 3.1.

Enc To encrypt a message $M \in \mathbb{G}_2$ under ciphertext-policy $W = [W_1, W_2, \cdots, W_n]$, pick up a random value $z \in \mathbb{Z}_p$ and compute $C_0 = MT^z$. For each $1 \le i \le n$,

1. If $W_i = v'_{i,k'_i} (= (I'_{i,1}, I'_{i,2}, \cdots, I'_{i,\rho_i}))$, choose $\{(z_{i,t_i}, z'_{i,t_i}, \bar{z}_{i,t_i})\}_{1\le t_i\le\rho_i} \in \mathbb{Z}_p$. For $1 \le t_i \le \rho_i$, if $I'_{i,t_i} = 1$, compute $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}) = (H(1\|i\|t_i\|1)^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|t_i\|1)^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}})$ and $(\hat{C}_{i,t_i,0}, \hat{C}_{i,t_i,1}, \hat{C}'_{i,t_i,0}, \hat{C}'_{i,t_i,1}) = (H(1\|i\|t_i\|0)^{z'_{i,t_i}}, g_1^{z'_{i,t_i}}, H(0\|i\|t_i\|0)^{\bar{z}_{i,t_i}}, g_2^{\bar{z}_{i,t_i}})$; otherwise, compute $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}) = (H(1\|i\|t_i\|1)^{z'_{i,t_i}}, g_1^{z'_{i,t_i}}, H(0\|i\|t_i\|1)^{\bar{z}_{i,t_i}}, g_2^{\bar{z}_{i,t_i}})$, $(\hat{C}_{i,t_i,0}, \hat{C}_{i,t_i,1}, \hat{C}'_{i,t_i,0}, \hat{C}'_{i,t_i,1}) = (H(1\|i\|t_i\|0)^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|t_i\|0)^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}})$.
2. If $W_i = *$, choose $\{(z_{i,t_i}, z'_{i,t_i})\}_{1\le t_i\le\rho_i}$ from $\mathbb{Z}_p$. For $1 \le t_i \le \rho_i$, compute $\{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})\} = \{H(1\|i\|t_i\|1)^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|t_i\|1)^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}}\}$, $(\hat{C}_{i,t_i,0}, \hat{C}_{i,t_i,1}, \hat{C}'_{i,t_i,0}, \hat{C}'_{i,t_i,1}) = (H(1\|i\|t_i\|0)^{z'_{i,t_i}}, g_1^{z'_{i,t_i}}, H(0\|i\|t_i\|0)^{z''_{i,t_i}}, g_2^{z''_{i,t_i}})$, where $z'_{i,t_i} + z''_{i,t_i} = z$.

The ciphertext is $C = (C_0, \{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}), (\hat{C}_{i,t_i,0}, \hat{C}_{i,t_i,1}, \hat{C}'_{i,t_i,0}, \hat{C}'_{i,t_i,1})\}$ for $1 \le t_i \le \rho_i$ and $1 \le i \le n$.

Dec Assume a user has an attribute private key $sk_L = \{D_i\}_{1\le i\le n}$ for attribute list $L = [v_{1,t_1}, v_{2,t_2}, \cdots, v_{n,t_n}]$. To decrypt the ciphertext $C = (C_0, \{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}), (\hat{C}_{i,t_i,0}, \hat{C}_{i,t_i,1}, \hat{C}'_{i,t_i,0}, \hat{C}'_{i,t_i,1})\}_{1\le t_i\le\rho_i}\}_{1\le i\le n})$ without knowing ciphertext-policy W, the user first computes

$$C' = \prod_{i=1}^{n} \left(\prod_{t_i=1}^{\rho_i} \frac{\hat{e}(\widetilde{C}_{i,t_i,1}, d_{i,t_i,0})\hat{e}(\widetilde{C}'_{i,t_i,1}, d'_{i,t_i,0})}{\hat{e}(\widetilde{C}_{i,t_i,0}, d_{i,t_i,1})\hat{e}(\widetilde{C}'_{i,t_i,0}, d'_{i,t_i,1})} \right).$$

– If $I_{i,t_i} = 1$, $(\widetilde{C}_{i,t_i,b}, \widetilde{C}'_{i,t_i,b}) = (C_{i,t_i,b}, C'_{i,t_i,b})$ for $b \in \{0,1\}$;
– If $I_{i,t_i} = 0$, $(\widetilde{C}_{i,t_i,b}, \widetilde{C}'_{i,t_i,b}) = (\hat{C}_{i,t_i,b}, \hat{C}'_{i,t_i,b})$ for $b \in \{0,1\}$.

Finally, the user decrypts the ciphertext as $M = C_0/C'$.

The method given in Section 3.1 can be used here to check the correctness of decryption. We have the following security result for the construction:

Theorem 2. *The Anonymous CP-ABE construction is secure in* sCP-IND-CPA *model, under the* DBDH *and D-Linear assumption.*

Proof. The construction is similar to the construction in Section 3.1. The difference here is that the message is encrypted with respect to each bit, other than each value of the attribute. Therefore, the proof is easy to be derived from the proof for Theorem 1.

4 CP-A^{3}BE: Privacy-aware Attribute-based Encryption with User Accountability

In this Section, we propose a CP-A^{3}BE construction, that is, the anonymous CP-ABE with user accountability, which is based on the anonymous CP-ABE scheme in Section 3.1. In fact, to construct CP-A^{3}BE, the technique can be also easily applied to the anonymous CP-ABE scheme [24].
Main Idea In this scheme, user is issued an attribute private key for $L\|ID$, where L is an attribute list and ID is the user's identity. In a normal encryption algorithm, a message is encrypted under ciphertext-policy $W = W'\|*$ such that any user with $L\|ID$ satisfying $R(L\|ID, W) = 1$ is able to decrypt, regardless of the user's identity ID. This holds because the second part in the ciphertext-policy is "don't care" (This technique is used here to keep the one-to-many property in ABE, even though different identities have been inserted in the attribute private keys). In tracing algorithm, a message is encrypted with $W'\|ID^*$ to test whether the identity in the pirate device is ID^*. Due to the anonymity in CP-A^{3}BE, the ciphertext is indistinguishable from other ciphertext under ciphertext-policy $W = W'\|*$. In this case, only user with private key on $L\|ID$ satisfying $R(L\|ID, W'\|ID^*) = 1$ can decrypt the ciphertext. As a result, the identity ID^* can be determined in the pirate device. There are five algorithms of our CP-A^{3}BE scheme, which are defined as follows:

Setup Let $\mathbb{G}_1, \mathbb{G}_2$ be cyclic groups of prime order p, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a pairing defined in Section 2. Let g_1, g_2 be random elements from $\mathbb{G}_0$. Define a hash function $H : \{0,1\}^* \to \mathbb{G}_0$. Assume there are n attributes in universe. That is to say, let the universal attributes set be $U = \{\omega_1, \omega_2, \cdots, \omega_n\}$. Each attribute has multiple values, where S_i is the multi-value set for ω_i and $\mid S_i \mid= n_i$. This algorithm also chooses a random number $\alpha \in \mathbb{Z}_p$ and computes $T = \hat{e}(g_1, g_2)^{\alpha}$. The system public parameter is $para = (g_1, g_2, T, H)$. The system master secret key msk is α, which is only known to AC.

KeyGen To generate an attribute private key for user with $ID = (I_1, I_2, \cdots, I_\rho) \in \{0,1\}^\rho$ for attribute list $L = [L_1, L_2, \cdots, L_n] = [v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$, AC picks up random $s_1, s_2, \cdots, s_n \in \mathbb{Z}_p^*$ and computes $s_{n+1} = \alpha - \sum_{i=1}^{n} s_i \bmod p$. AC also chooses $n+1$ numbers $\{r_i\}_{1\le k\le n} \in \mathbb{Z}_p^*$ and ρ numbers $\{s_{n+1,k}\}_{1\le k\le \rho}$

such that $s_{n+1} = \sum_{k=1}^{\rho} s_{n+1,k}$. Finally, it computes the attribute private key on L as

$$sk_L = \{\{(d_{i0}, d_{i1}, d'_{i0}, d'_{i1})\}_{1\le i\le n}, \{(d_{n+1,k,0}, d_{n+1,k,1}, d'_{n+1,k,0}, d'_{n+1,k,1})\}_{1\le k\le \rho}\}$$
$$= \{(g_2^{s_i} H(1\|i\|v_{i,k_i})^{r_i}, g_1^{r_i}, g_1^{s_i} H(0\|i\|v_{i,k_i})^{r'_i}, g_2^{r'_i})\}, \{(g_2^{s_{n+1,k}} H(1\|n+1\|k\|I_k)^{r_{n+1,k}},$$
$$g_1^{r_{n+1,k}}, g_1^{s_{n+1,k}} H(0\|n+1\|k\|I_k)^{r'_{n+1,k}}, g_2^{r'_{n+1,k}})\}, 1 \le i \le n \wedge 1 \le k \le \rho.$$

Enc To encrypt a message $M \in \mathbb{G}_2$ under ciphertext-policy $W = [W_1, W_2, \cdots, W_n] \vee W_{n+1}$ where $W_{n+1} = *$, this algorithm picks up a random value $z \in \mathbb{Z}_p$ and computes $C_0 = MT^z$.

1. For each $1 \le i \le n$,
   † if $v_{i,t_i} \in W_i$, choose $z_{i,t_i} \in Z_p^*$ and compute $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$ $= (H(1\|i\|v_{i,t_i})^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|v_{i,t_i})^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}})$;
   ‡ if $v_{i,t_i} \notin W_i$, choose randomly $z_{i,t_i}, z'_{i,t_i} \in Z_p^*$ and compute $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$=$(H(1\|i\|v_{i,t_i})^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|v_{i,t_i})^{z'_{i,t_i}}, g_2^{z'_{i,t_i}})$.
2. For $i = n+1$, this algorithm selects $z_{n+1,k}, z'_{n+1,k}$ from $\mathbb{Z}_p^*$. Then, for each $1 \le k \le \rho$, it computes

$$(C_{n+1,k,0}, C_{n+1,k,1}, C'_{n+1,k,0}, C'_{n+1,k,1}) = (H(1\|n+1\|k\|1)^{z_{n+1,k}}, g_1^{z_{n+1,k}}, H(0\|n+1\|k\|1)^{z-z_{n+1,k}}, g_2^{z-z_{n+1,k}})$$

$$(\hat{C}_{n+1,k,0}, \hat{C}_{n+1,k,1}, \hat{C}'_{n+1,k,0}, \hat{C}'_{n+1,k,1}) = (H(1\|n+1\|k\|0)^{z'_{n+1,k}}, g_1^{z'_{n+1,k}}, H(0\|n+1\|k\|0)^{z-z'_{n+1,k}}, g_2^{z-z'_{n+1,k}})$$

Finally, the ciphertext is computed as $C = (C_0, \{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})\}$ for $1 \le t_i \le n_i$ and $1 \le i \le n$, $\{(C_{n+1,k,0}, C_{n+1,k,1}, C'_{n+1,k,0}, C'_{n+1,k,1})$, $(\hat{C}_{n+1,k,0}, \hat{C}_{n+1,k,1}, \hat{C}'_{n+1,k,0}, \hat{C}'_{n+1,k,1})\}_{1\le k\le\rho})$.

Dec Assume a user has an attribute private key

$$sk_L = \{\{(d_{i0}, d_{i1}, d'_{i0}, d'_{i1})\}_{1\le i\le n}, \{(d_{n+1,k,0}, d_{n+1,k,1}, d'_{n+1,k,0}, d'_{n+1,k,1})\}_{1\le k\le \rho}\}$$

for $L = [v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$. To decrypt the ciphertext C without knowing ciphertext-policy W, he computes

$$C' = \prod_{i=1}^{n} \frac{\hat{e}(C_{i,k_i,1}, d_{i0})\hat{e}(C'_{i,k_i,1}, d'_{i0})}{\hat{e}(C_{i,k_i,0}, d_{i1})\hat{e}(C'_{i,k_i,0}, d'_{i1})} \prod_{k=1}^{\rho} \frac{\hat{e}(\widetilde{C}_{n+1,k,1}, d_{n+1,k,0})\hat{e}(\widetilde{C}'_{n+1,k,1}, d'_{n+1,k,0})}{\hat{e}(\widetilde{C}_{n+1,k,0}, d_{n+1,k,1})\hat{e}(\widetilde{C}'_{n+1,k,0}, d'_{n+1,k,1})}.$$

1. If $I_k = 1$, $(\widetilde{C}_{n+1,k,b}, \widetilde{C}'_{n+1,k,b})$=$(C_{n+1,k,b}, C'_{n+1,k,b})$ for $b \in \{0, 1\}$;
2. If $I_k = 0$, $(\widetilde{C}_{n+1,k,b}, \widetilde{C}'_{n+1,k,b})$=$(\hat{C}_{n+1,k,b}, \hat{C}'_{n+1,k,b})$ for $b \in \{0, 1\}$.

Finally, decrypt and output the ciphertext as $M = C_0/C'$.

Trace Suppose a given pirate device can decrypt the ciphertext under ciphertext-policy W. AC extracts part of the attribute list $(L_{i_1}, L_{i_2}, \cdots, L_{i_k})$ out of W. The values in other positions except $\{i_1, i_2, \cdots, i_k\}$ in W are $*$. AC checks the issuing record of attribute private key and determines the suspicious users set S, who have the attributes $(L_{i_1}, L_{i_2}, \cdots, L_{i_k})$. There are two ways to pinpoint the exact identity from S: If the size of set S is not huge, then, AC just encrypts some message with respect to ciphertext-policy W for each $ID \in S$ until the identity is found. To make the trace algorithm and encryption algorithm indistinguishable, the technique used in Section 3.2 is applied here.

AC picks up a random value $z \in \mathbb{Z}_p$ and computes $C_0 = MT^z$ to encrypt a message $M \in \mathbb{G}_2$ under ciphertext-policy $W = [W_1, W_2, \cdots, W_n] \vee W_{n+1}$ where $W_{n+1} = ID$,

1. For each $1 \le i \le n$,

† if $v_{i,t_i} \in W_i$, AC picks $z_{i,t_i} \in Z_p^*$ and computes

$$(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}) = (H(1\|i\|v_{i,t_i})^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|v_{i,t_i})^{z-z_{i,t_i}}, g_2^{z-z_{i,t_i}});$$

‡ if $v_{i,t_i} \notin W_i$, AC chooses $z_{i,t_i}, z'_{i,t_i} \in \mathbb{Z}_p^*$ and computes

$$(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1}) = (H(1\|i\|v_{i,t_i})^{z_{i,t_i}}, g_1^{z_{i,t_i}}, H(0\|i\|v_{i,t_i})^{z'_{i,t_i}}, g_2^{z'_{i,t_i}}).$$

2. For $i = n+1$, assume $ID{=}(I_1, I_2, \cdots, I_\rho))$. AC chooses $\{(z_{n+1,k}, z'_{n+1,k}, \bar{z}_{n+1,k})\}$ for $1 \le k \le \rho$,

† if $I_k = 1$, AC computes

$$\begin{aligned}(C_{n+1,k,0}, C_{n+1,k,1}, C'_{n+1,k,0}, C'_{n+1,k,1}) = (&H(1\|n+1\|k\|1)^{z_{n+1,k}}, g_1^{z_{n+1,k}},\\ &H(0\|n+1\|k\|1)^{z-z_{n+1,k}}, g_2^{z-z_{n+1,k}})\end{aligned}$$

$$\begin{aligned}(\hat{C}_{n+1,k,0}, \hat{C}_{n+1,k,1}, \hat{C}'_{n+1,k,0}, \hat{C}'_{n+1,k,1}) = (&H(1\|n+1\|k\|0)^{z'_{n+1,k}}, g_1^{z'_{n+1,k}},\\ &H(0\|n+1\|k\|0)^{\bar{z}_{n+1,k}}, g_2^{\bar{z}_{n+1,k}})\end{aligned}$$

‡ if $I_k = 0$, AC computes

$$\begin{aligned}(C_{n+1,k,0}, C_{n+1,k,1}, C'_{n+1,k,0}, C'_{n+1,k,1}) = (&H(1\|n+1\|k\|1)^{z'_{n+1,k}}, g_1^{z'_{n+1,k}},\\ &H(0\|n+1\|k\|1)^{\bar{z}_{n+1,k}}, g_2^{\bar{z}_{n+1,k}})\end{aligned}$$

$$(\hat{C}_{n+1,k,0}, \hat{C}_{n+1,k,1}, \hat{C}'_{n+1,k,0}, \hat{C}'_{n+1,k,1}) = (H(1\|n+1\|k\|0)^{z_{n+1,k}}, g_1^{z_{n+1,k}}, \\ H(0\|n+1\|k\|0)^{z-z_{n+1,k}}, g_2^{z-z_{n+1,k}})$$

It can be easily seen that the user is able to decrypt the ciphertext only when his identity is ID and he has the attribute list L=$(L_{i_1}, L_{i_2}, \cdots, L_{i_k})$.

If $|S|$ is too huge, the tracing algorithm works in the following way: First, AC tries an attribute value L_j from the position j where $W_j = *$. Then, it encrypts a message as the normal encryption algorithm with respect to W' such that all positions are set to be $*$, except the positions of $\{i_1, i_2, \cdots, i_k, j\}$ are set to be $L' = L \cup L_j$. The ciphertext is sent to the pirate device. If the ciphertext can be decrypted correctly, AC knows one of the users with L' shares his attribute private key. The suspicious user set is of course not greater than $|S|$. AC continues the above procedure until the suspicious set $|S|$ is not too huge. Finally, the technique for small $|S|$ can be applied and the identity in the pirate device can be pinpointed. To verify the correctness of the decryption, we also use the method described in Section 3.1 by adding redundancy in the plaintext. Actually, based on the tracing algorithm, the scheme is secure against collusion attack, in which users with different attributes can collude to generate a pirate device. The tracing algorithm still works and at least one of the illegal users will be detected from the pirate device. Our definition and construction of tracing requires that the adversary produces a perfect pirate decoder device, namely a decoder that correctly decrypts all well-formed ciphertexts. In reality, the pirate has a decoder that may work only a fraction of the time. When interact with such a decoder, just repeat the tracing algorithm for each suspicious identity such that the error-rate is lower than some predefined number. The tracing algorithm is indistinguishable from the normal encryption algorithm because of the anonymous CP-ABE. We have following security result for the construction of CP-A^{3}BE:

Theorem 3. *The CP-A^{3}BE construction is secure in* sCP-IND-CPA *model, under the* DBDH *and D-Linear assumptions.*

Proof. This construction is based on the construction in Section 3.1, with the technique of anonymous CP-ABE in Section 4.1. Therefore, the proof is easy to be derived from the proof for Theorem 1 and Theorem 2, and is omitted here.

5 Conclusion

Three requirements are desired in many secure access control systems, that is, 1) Fine-grained access policy, 2) User privacy, and 3) User accountability. ABE schemes are promising in providing fine-grained access policy, but no existing ABE schemes can achieve user accountability to prevent illegal key sharing while still maintaining user privacy. In this paper, we solved this problem by proposing the notion of accountable and anonymous CP-ABE (CP-A^{3}BE). We started by giving two improvements of privacy-aware CP-ABE. In the first improvement of anonymous CP-ABE, the size of public parameter is only $O(1)$, instead of $O(N)$ required in [24], where N denotes the number of attributes in universe. In the second improvement, the size of public parameter and ciphertext is $O(1)$ and $O(\log(N))$, respectively, while in [24], they are both $O(N)$. Based on the improvements, we presented a CP-A^{3}BE construction. The user accountability can be achieved in black-box model by embedding additional user-specific information into the attribute private key, while still maintaining hidden access policy. The construction of CP-A^{3}BE is provably secure.

Acknowledgement

This work was supported in part by the US National Science Foundation under grant CNS-0831963 and the National Sciences and Engineering Research Council of Canada under Grant RGPIN/356059-2008. Thanks to Shucheng Yu, Cong Wang and Qian Wang for their helpful comments on this work.

References

1. Michel Abdalla, Dario Catalano, Alexander W. Dent, John Malone-Lee, Gregory Neven, and Nigel P. Smart. *Identity-Based Encryption Gone Wild*. ICALP'06, LNCS 4052, pp. 300-311, Springer, 2006.
2. Joonsang Baek, Willy Susilo, and Jianying Zhou. *New Constructions of Fuzzy Identity-Based Encryption*. ASIACCS'07, pp. 368-370, ACM, 2007.
3. John Bethencourt, Amit Sahai, and Brent Waters. *Ciphertext-Policy Attribute-Based Encryption*. IEEE Symposium on Security and Privacy'07, pp. 321-334, IEEE, 2007.
4. Dan Boneh and Xavier Boyen. *Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles*. EUROCRYPT'04, LNCS 3027, pp. 223-238, Springer, 2004.
5. Dan Boneh, Xavier Boyen, Shacham. *Short Group Signatures*.CRYPTO'04. LNCS, vol. 3152, pp. 41-55, Springer, 2004.
6. Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. *Chosen-Ciphertext Security from Identity-Based Encryption*. SIAM J. Comput. 36(5): 1301-1328, 2007.
7. Dan Boneh and M. Franklin. *Identity-Based Encryption from The Weil Pairing*, Crypto'01, LNCS 2139, pp.213-229, Springer-Verlag, 2001.
8. Dan Boneh and Brent Waters. *Conjunctive, Subset, and Range Queries on Encrypted Data*. TCC'07. LNCS 4392, pp. 535-554. Springer, 2007.
9. Xavier Boyen, Qixiang Mei, and Brent Waters. *Direct Chosen Ciphertext Security from Identity-Based Techniques*. CCS'05, pp. 320-329, ACM press, 2005. Full version at http://eprint.iacr.org/2005/288.
10. Melissa Chase. *Multi-Authority Attribute Based Encryption*. TCC'07, LNCS 4392, pp. 515-534, Springer, 2007.
11. Ling Cheung and Calvin Newport. *Provably Secure Ciphertext Policy ABE*. In CCS'07, Proceedings of the 14th ACM conference on Computer and communications security, pages 456-465, ACM, 2007.
12. Benny Chor, Amos Fiat, and Moni Naor. *Tracing Traitor*. CRYPTO'94, LNCS 839, pp. 257-270, Springer, 1994.

13. Eiichiro Fujisaki and Tatsuaki Okamoto. *Secure Integration of Asymmetric and Symmetric Encryption Schemes*. CRYPTO'99, LNCS 1666, pp. 537-554, Springer, 1999.
14. Craig Gentry. *Practical Identity-Based Encryption without Random Oracles*. EUROCRYPT'06. LNCS, vol. 4004, pp. 445-464. Springer, 2006.
15. Vipul Goyal. *Reducing Trust in the PKG in Identity Based Cryptosystems*. CRYPTO'07, LNCS 4622, pp. 430-447, 2007. Extension is available at http://eprint.iacr.org/2007/368.
16. Vipul Goyal, Abhishek Jain, Omkant Pandey, and Amit Sahai. *Bounded Ciphertext Policy Attribute Based Encryption*. ICALP'08. LNCS 5126, pp. 579-591, 2008.
17. Vipul Goyal, Steve Lu, Amit Sahai, and Brent Waters. *Black-Box Accountable Authority Identity-Based Encryption*. CCS'08, USA.
18. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. *Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data*. CCS'06, pp. 89-98, ACM, 2006.
19. M. Jason Hinek, Shaoquan Jiang, Reihaneh Safavi-Naini, and Siamak Fayyaz Shahandashti. *Attribute-Based Encryption with Key Cloning Protection*. Available at http://eprint.iacr.org/2008/478.
20. Apu Kapadia, Patrick P. Tsang, and Sean W. Smith. *Attribute-based Publishing with Hidden Credentials and Hidden Policies*. In Proc. of Network and Distributed System Security Symposium (NDSS), pp. 179-192, 2007.
21. Jonathan Katz, Amit Sahai, and Brent Waters. *Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products*. EUROCRYPT'08, LNCS 4965, pp. 146-162, Springer, 2008.
22. Jin Li, Kui Ren, and Kwangjo Kim. *A^2BE: Accountable Attribute-Based Encryption for Abuse Free Access Control*. Available at http://eprint.iacr.org/2009/118.
23. Jin Li, Kui Ren, Bo Zhu, and Zhiguo Wan. *Privacy-aware Attribute-based Encryption with User Accountability*. Full version. Available at http://eprint.iacr.org/2009/284.
24. Takashi Nishide, Kazuki Yoneyama, and Kazuo Ohta. *ABE with Partially Hidden Encryptor-Specified Access Structure*. ACNS'08, LNCS 5037, pp. 111-129, Springer, 2008.
25. Rafail Ostrovsky, Amit Sahai, and Brent Waters. *Attribute-based Encryption with Non-Monotonic Access Structures*. CCS'07, pp. 195-203, ACM, 2007.
26. Amit Sahai. *Non-Malleable Non-Interactive Zero Knowledge and Adaptive Chosen Ciphertext Security*. IEEE Symp. on Foundations of Computer Science, 1999.
27. Amit Sahai and Brent Waters. *Fuzzy Identity-Based Encryption*. EUROCRYPT'05, LNCS 3494, pp. 457-473, Springer, 2005.
28. Adi Shamir. *How to Share a Secret*. vol. 22, pp. 612-613, ACM, 1979.
29. Adi Shamir. *Identity-Based Cryptosystems and Signature Schemes*. CRYPTO'84, LNCS 196, pp. 47-53, Springer, 1984.
30. Shucheng Yu, Kui Ren, and Wenjing Lou. *Attribute-Based Content Distribution with Hidden Policy*. NPSEC'08, pp. 39-44, 2008.
31. Shucheng Yu, Kui Ren, Wenjing Lou, and Jin Li. *Defending Against Key Abuse Attacks in KP-ABE Enabled Broadcast Systems*. Accepted by SECURECOMM'09. Also available at http://eprint.iacr.org/2009/295.
32. Brent Waters. *Efficient Identity-Based Encryption Without Random Oracles*. EUROCRYPT'05, LNCS 3494, pp. 114-127, Springer, 2005.

Appendix A: Correctness of Anonymous CP-ABE Scheme in Section 3.1

The correctness of the attribute private key has been described in KeyGen algorithm. We only show the correctness of the decryption from the following equations:

$$
\begin{aligned}
C' &= \prod_{i=1}^{n} \frac{\hat{e}(C_{i,k_i,1}, d_{i0})\hat{e}(C'_{i,k_i,1}, d'_{i0})}{\hat{e}(C_{i,k_i,0}, d_{i1})\hat{e}(C'_{i,k_i,0}, d'_{i1})} \\
&= \prod_{i=1}^{n} \frac{\hat{e}(g_1^{z_{i,k_i}}, g_2^{s_i}(H(1\|i\|v_{i,k_i}))^{r_i})\hat{e}(g_2^{z-z_{i,k_i}}, g_1^{s_i}H(0\|i\|v_{ik_i})^{r'_i})}{\hat{e}(H(1\|i\|v_{i,k_i})^{z_{i,k_i}}, g_2^{r_i})\hat{e}(H(0\|i\|v_{i,k_i})^{z-z_{i,k_i}}, g_2^{r'_i})} \\
&= \prod_{i=1}^{n} \hat{e}(g_2^{s_i}, g_1^{z_{i,k_i}})\hat{e}(g_1^{s_i}, g_2^{z-z_{i,k_i}}) \\
&= \prod_{i=1}^{n} \hat{e}(g_1^{s_i}, g_2^{z_{i,k_i}})\hat{e}(g_1^{s_i}, g_2^{z-z_{i,k_i}}) \\
&= \prod_{i=1}^{n} \hat{e}(g_1^{s_i}, g_2^{z}) = \hat{e}(g_1^{\sum_{i=1}^{n} s_i}, g_2^{z}) = \hat{e}(g_1^{\alpha}, g_2^{z}) \\
&= T^z
\end{aligned}
$$

Appendix B: Proof of Theorem 1

We prove the scheme is sCP-IND-CPA under the DBDH assumption and the D-Linear assumption. The adversary commits to the challenge ciphertext policies $W_0^* = [W_{0,1}^*, W_{0,2}^*, \cdots, W_{0,n}^*]$ and $W_1^* = [W_{1,1}^*,\ W_{1,2}^*,\ \cdots,\ W_{1,n}^*]$ in advance. Based on the sCP-IND-CPA security model, we use a sequence of hybrid games to prove that the adversary cannot win the original security game denoted by G with non-negligible probability. We begin by slightly modifying the game G into a game G_0. The definitions of Games G and G_0 are the same except the computation of challenge ciphertext. In game G_0, if the adversary did not obtain the sk_L with respect to attribute list L satisfying the condition of $R(L, W_0^*) = 1 \wedge R(L, W_1^*) = 1$, just compute C_0 as a random group element in $\mathbb{G}_1$, while computing the other parts of ciphertext in normal way. On the other hand, if the adversary obtained an attribute private key sk_L on attribute list L satisfying the condition of $R(L, W_0^*) = 1 \wedge R(L, W_1^*) = 1$, the challenge ciphertext will be generated correctly. In this case, we have $G = G_0$.

In general, we will prove this theorem in two steps: First, under the DBDH assumption, we will prove that the difference of advantage of $\mathcal{A}$ in game G and game G_0 is negligible for any polynomial time adversary $\mathcal{A}$. Then, the game G_0 is modified by changing the ciphertext parts $(C_0, \{\{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})\}_{1 \le t_i \le n_i}\})$

for $1 \leq i \leq n$. We define a sequence of games as follows. For $v_{i,t}$ such that $(v_{i,t_i} \in W_{0,i} \wedge v_{i,t_i} \in W_{1,i})$ or $(v_{i,t_i} \notin W_{0,i} \wedge v_{i,t_i} \notin W_{1,i})$, the $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$ are computed as in the real scheme through the sequence of all the games. In case that there is v_{i,t_i} such that $(v_{i,t_i} \in W_{0,i} \wedge v_{i,t_i} \notin W_{1,i})$ or $(v_{i,t_i} \notin W_{0,i} \wedge v_{i,t_i} \in W_{1,i})$, $(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$ in game $G_{\ell-1}$ are replaced with the random values in the new modified game G_ℓ regardless of the random coin b. This process is repeated until there is no component satisfying $(v_{i,t_i} \in W_{0,i} \wedge v_{i,t_i} \notin W_{1,i})$ or $(v_{i,t_i} \notin W_{0,i} \wedge v_{i,t_i} \in W_{1,i})$. At the end of the sequence game, the advantage of the adversary is zero because the adversary is given a ciphertext chosen from the same distribution regardless of the random coin b. By replacing the well-formed ciphertext components in $G_{\ell-1}$ with the random values in G_ℓ in this way, we can prove that if there is a distinguisher of $G_{\ell-1}$ and G_ℓ, the D-Linear assumption would hold.

Given a DBDH challenge instance $(g, A = g^x, B = g^y, C = g^z, Z)$, where Z is either $\hat{e}(g,g)^{xyz}$ or random with equal probability, the simulator $\mathcal{S}$ computes as follows. At first, $\mathcal{A}$ gives $\mathcal{S}$ two challenge chiphertext policies $W_0^* = [W_{0,1}^*, W_{0,2}^*, \cdots, W_{0,n}^*]$ and $W_1^* = [W_{1,1}^*, W_{1,2}^*, \cdots, W_{1,n}^*]$ before setup algorithm. Then $\mathcal{S}$ flips a random coin $b \in \{0,1\}$. $\mathcal{S}$ also chooses randomly $b' \in \{0,1\}$ and sets $g_{b'} = B$, without loss of generality, assume $g_2 = B$ and $g_1 = g^a$ for some $a \in Z_p$. Let $T = \hat{e}(g_1, g_2)^x = \hat{e}(A^a, B)$, which implies $\alpha = x$. For each attribute v_{i,t_i}, if $v_{i,t_i} \notin W_{b,i}^*$, then, $H(1\|i\|v_{i,t_i}) = g_2^{\delta_{i,t_i}} g^{b_i}$ and $H(0\|i\|v_{i,t_i}) = g_2^{\delta_{i,t_i}} g^{b'_i}$ by choosing $b_i, b'_i \in \mathbb{Z}_p^*$. Otherwise, if $v_{i,t_i} \in W_{b,i}^*$, $H(1\|i\|v_{i,t_i}) = g^{b_i}$ and $H(0\|i\|v_{i,t_i}) = g^{b'_i}$, where b_i, b'_i are chosen from $\mathbb{Z}_p^*$. During key generation phase, $\mathcal{A}$ submits an attribute list $L = (L_1, \cdots, L_n)$ requesting private key. We consider only the case where $R(L, W_0^*)$=0 and $R(L, W_1^*)$=0. This is because according to definition of our games, if $R(L, W_0^*)$=1 and $R(L, W_1^*)$=1, the challenge messages M_0, M_1 will be equal. As a result, the games G and G_0 are the same and there is no difference of advantage of $\mathcal{A}$ in G and G_0. Therefore, $\mathcal{S}$ simply aborts and takes a random guess. When $R(L, W_0^*)$=0 and $R(L, W_1^*)$=0, there must be a $k \in \{1, \cdots, n\}$ such that $L_k \notin W_{b,k}^*$. It then chooses randomly s_i for $i \neq k$, and assigns $s_k = x - \sum_{i=1}^{k-1} s_i$ (Actually, the simulator does not know s_k because x is unknown). For the private key components $i \neq k$, they can be easily computed because s_i is known. For the k-th private key component, the simulator chooses r'_k, $\hat{r}_k$ from Z_p^* and computes the private key as $(g_2^{s_k} H(1\|i\|v_{k,t_k})^{r_k}$, $g_1^{r_k}$, $g_1^{s_k} H(0\|i\|v_{k,t_k})^{r'_k}$, $g_2^{r'_k}$, $)$=$(B^{\hat{r}_k \delta_{k,t_k} - b_k/\delta_{k,t_k} - \sum_{i=1,i\neq k}^{n} s_i}\ g^{b_k r_k}$, $B^a g^{-a\sum_{i=1,i\neq k}^{n} s_i}$, $B^{r'_k}$, $g^{r'_k})$. It is correct by just letting $r_k = -x/\delta_{k,t_k} + \hat{r}_k$.

Finally, $\mathcal{A}$ submits two challenge messages M_0 and M_1. The simulator $\mathcal{S}$ computes the challenge ciphertext as $(C_0, \{(C_{i,t_i,0}, C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})_{1\leq t_i \leq n_i}\}_{1\leq i\leq n})$ with $C_0 = Z^a M_b$. When $v_{i,t_i} \in W_{b,t_i}^*$, $\mathcal{S}$ can compute $(C_{i,t_i,0}$, $C_{i,t_i,1}$, $C'_{i,t_i,0}$, $C'_{i,t_i,1}) = (C^{b_i} g^{-b_i z_i}$, $C^a g^{-a z_i}$, $g^{b'_i z_i}$, $B^{z_i})$. The ciphertext is simulated correctly from the viewpoint of $\mathcal{A}$ because $H(1\|i\|v_{i,t_i})$=g^{b_i}, $H(0\|i\|v_{i,t_i})$=$g^{b'_i}$. Therefore, $H(1\|i\|v_{i,t_i})^{z_i}$=$C^{b_i} g^{-b_i z'_i}$, $g_1^{z_i} = C^a g^{-a z'_i}$, $H(0\|i\|v_{i,t_i})^{z-z_i}$=$g^{b'_i z'_i}$, and $g_2^{z-z_i} = B^{z'_i}$ by choosing $z'_i \in \mathbb{Z}_p^*$ and letting $z_i = z - z'_i$. When $v_{i,t_i} \notin W_{b,t_i}^*$, $(C_{i,t_i,0}$, $C_{i,t_i,1}, C'_{i,t_i,0}, C'_{i,t_i,1})$ can be easily computed just by simply choosing z_i, z'_i at random according to the real encryption algorithm. The above private key queries are repeated. Finally, $\mathcal{A}$ outputs a guess b^* of b. If $b^* = b$, $\mathcal{S}$ outputs 1 and otherwise outputs 0. By our assumption, the probability that $\mathcal{A}$ guesses b correctly in game G has a non-negligible difference from that of guessing b correctly in G_0. When $Z = \hat{e}(g,g)^{xyz}$, $\mathcal{A}$ is in game G and when Z is random, $\mathcal{A}$ is in game G_0. Therefore the simulator $\mathcal{S}$ has the same advantage as the adversary's in the DBDH game.

Next, we prove that under the D-Linear assumption, the difference of advantage of $\mathcal{A}$ in game $G_{\ell-1}$ and game G_ℓ is negligible for any polynomial time adversary $\mathcal{A}$. Otherwise, we can build a simulator $\mathcal{S}$ that breaks the D-Linear assumption with advantage ϵ. As mentioned in Section 4, the ciphertext components are generated as in the real scheme in G_0, the components are random regardless of the random coin b. Assume that $(v_{i_\ell,t_\ell} \in W^*_{1,i} \wedge v_{i_\ell,t_\ell} \notin W^*_{0,i})$ without loss of generality. The simulator $\mathcal{S}$ runs $\mathcal{A}$. $\mathcal{A}$ gives $\mathcal{S}$ two challenge ciphertext policies $W^*_0 = [W^*_{0,1}, W^*_{0,2}, \cdots, W^*_{0,n}]$ and $W^*_1 = [W^*_{1,1}, W^*_{1,2}, \cdots, W^*_{1,n}]$. Then $\mathcal{S}$ flips a random coin $b \in \{0,1\}$. If $b=0$, $\mathcal{S}$ aborts and takes a random guess. The reason for this is by our definition if $b=0$ where $(v_{i_\ell,t_\ell} \in W^*_{1,i} \wedge v_{i_\ell,t_\ell} \notin W^*_{0,i})$, we have $G_{\ell-1} = G_\ell$ because the distribution of the challenge ciphertext in game $G_{\ell-1}$ is the same as that of game G_ℓ, so there is no difference of advantage of $\mathcal{A}$ in $G_{\ell-1}$ and G_ℓ. Assume $b=1$, given a D-Linear challenge $(g, g^{z_1}, g^{z_2}, Z, g^{z_1 z_2}, g^{z_2 z_4}, g^{z_3+z_4})$ where Z is either $g^{z_1 z_3}$ or random with equal probability, the simulator $\mathcal{S}$ computes as follows. This variant D-Linear assumption is similar to that one used in the proof of [24]. $\mathcal{S}$ sets $g_1 = g^{z_1}, g_2 = g^{z_2}$, $\alpha = w \in \mathbb{Z}^*_p$ and $T = \hat{e}(g_1,g_2)^\alpha$. $\mathcal{A}$ submits an attribute list $L = [L_1, L_2, \cdots, L_n]$=$[v_{1,k_1}, v_{2,k_2}, \cdots, v_{n,k_n}]$ in a secret key query. $\mathcal{S}$ computes the private key for the attribute list because w is known. During this phase, assume $\mathcal{S}$ has simulated the random oracle queries as $H(1\|i\|v_{i,t_i}) = g_1^{b_{i,t_i}}$ and $H(0\|i\|v_{i,t_i}) = g_2^{b'_{i,t_i}}$ if $v_{i,t_i} \in W^*_{b,i}$. Otherwise, if $v_{i,t_i} \notin W^*_{b,i}$, $H(1\|i\|v_{i,t_i}) = g^{b_{i,t_i}}$ and $H(0\|i\|v_{i,t_i}) = g^{b'_{i,t_i}}$. $\mathcal{A}$ submits two challenge messages M_0 and M_1. Let $z = z_3 + z_4$. If $R(L, W^*_0) = 0$ and $R(L, W^*_1) = 0$ for every queried $L = [L_1, L_2, \cdots, L_n]$, $\mathcal{S}$ sets C_0 to be random and otherwise sets $C_0 = M_b \hat{e}(g^{z_1 z_2}, g^{z_3+z_4})^w$. It is correct because $T^z = \hat{e}(g_1,g_2)^{\alpha z} = \hat{e}(g^{z_1 z_2}, g^{z_3+z_4})^w$. $\mathcal{S}$ also computes for W^*_b that the ciphertext components $\{(C_{i,t_i,0},\ C_{i,t_i,1},\ C'_{i,t_i,0},\ C'_{i,t_i,1}\ )_{1\le t_i\le n_i}\}_{1\le i\le n}$ with the exception that the components $(C_{i_\ell,t_{i_\ell},0},\ C_{i_\ell,t_{i_\ell},1},\ C'_{i_\ell,t_{i_\ell},0},\ C'_{i_\ell,t_{i_\ell},1})$ are computed as $(Z^{b_{i,t_i}}, Z, g^{b'_{i,t_i} z_2 z_4}, g^{z_2 z_4})$. The ciphertext is correct by just letting $z_{i,t_i} = z_3$. Because the simulation of random oracles, we have $H(1\|i\|v_{i,t_i}) = g_1^{b_{i,t_i}}$ and $H(0\|i\|v_{i,t_i}) = g_1^{b'_{i,t_i}}$. Finally, the above ciphertext can be rewritten as $(H(1\|i\|v_{i,t_i})^{z_{i,t_i}},\ g_1^{z_{i,t_i}},\ H(0\|i\|v_{i,t_i})^{z-z_{i,t_i}},\ g_2^{z-z_{i,t_i}})$.

Appendix C: Correctness of CP-A^{3}BE in Section 4

The correctness of the decryption can be verified as follows:

$$
\begin{aligned}
C' &= \prod_{i=1}^{n} \frac{\hat{e}(C_{i,k_i,1}, d_{i0})\hat{e}(C'_{i,k_i,1}, d'_{i0})}{\hat{e}(C_{i,k_i,0}, d_{i1})\hat{e}(C'_{i,k_i,0}, d'_{i1})} \prod_{k=1}^{\rho} \frac{\hat{e}(\widetilde{C}_{n+1,k,1}, d_{n+1,k,0})\hat{e}(\widetilde{C}'_{n+1,k,1}, d'_{n+1,k,0})}{\hat{e}(\widetilde{C}_{n+1,k,0}, d_{n+1,k,1})\hat{e}(\widetilde{C}'_{n+1,k,0}, d'_{n+1,k,1})} \\
&= \prod_{i=1}^{n} \frac{\hat{e}(g_1^{z_{i,k_i}}, g_2^{s_i} H(1\|i\|v_{i,k_i})^{r_i})\hat{e}(g_2^{z-z_{i,k_i}}, g_1^{s_i}(H(0\|i\|v_{ik_i}))^{r'_i})}{\hat{e}(H(1\|i\|v_{i,k_i})^{z_{i,k_i}}, g_2^{r_i})\hat{e}(H(0\|i\|v_{i,k_i})^{z-z_{i,k_i}}, g_2^{r_i})} \\
&\quad \prod_{k=1}^{\rho} \hat{e}(g_2^{s_{n+1}}, g_1^{z_{n+1,k}})\hat{e}(g_1^{s_{n+1}}, g_2^{z-z_{n+1,k}}) \\
&= \prod_{i=1}^{n} \hat{e}(g_2^{s_i}, g_1^{z_{i,k_i}})\hat{e}(g_1^{s_i}, g_2^{z-z_{i,k_i}}) \prod_{k=1}^{\rho} \hat{e}(g_2^{s_{n+1}}, g_1^{z_{n+1,k}})\hat{e}(g_1^{s_{n+1}}, g_2^{z-z_{n+1,k}}) \\
&= \prod_{i=1}^{n+1} \hat{e}(g_1^{s_i}, g_2^{z_{i,k_i}})\hat{e}(g_1^{s_i}, g_2^{z-z_{i,k_i}}) \\
&= \prod_{i=1}^{n+1} \hat{e}(g_1^{s_i}, g_2^{z}) = \hat{e}(g_1^{\sum_{i=1}^{n} s_i}, g_2^{z}) = \hat{e}(g_1^{\alpha}, g_2^{z}) \\
&= T^{z}
\end{aligned}
$$