Title	A ciphertext-policy attribute-based encryption scheme with constant ciphertext length
Author(s)	Emura, Keita; Miyaji, Atsuko; Omote, Kazumasa; Nomura, Akito; Soshi, Masakazu
Citation	International Journal of Applied Cryptography, 2(1): 46-59
Issue Date	2010-07-02
Type	Journal Article
Text version	author
URL	http://hdl.handle.net/10119/9491
Rights	Copyright (C) 2010 Inderscience. Keita Emura, Atsuko Miyaji, Kazumasa Omote, Akito Nomura, Masakazu Soshi, International Journal of Applied Cryptography, 2(1), 2010, 46-59. http://dx.doi.org/10.1504/IJACT.2010.033798
Description	

A Ciphertext-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length

Keita Emura*, Atsuko Miyaji and Kazumasa Omote
School of Information Science, Japan Advanced Institute of Science and Technology, Japan
E-mail: {k-emura, miyaji, omote}@jaist.ac.jp
* Corresponding author

Akito Nomura
Mathematics Section, Division of Innovative Technology and Science, Graduate School of Natural Science and Technology, Kanazawa University, Japan
E-mail: a-nomura@kanazawa-u.ac.jp

Masakazu Soshi
Graduate School of Information Sciences, Hiroshima City University, Japan
E-mail: soshi@hiroshima-cu.ac.jp

Abstract: An Attribute-Based Encryption (ABE) is an encryption scheme, where users with some attributes can decrypt ciphertexts associated with these attributes. The length of the ciphertext depends on the number of attributes in previous ABE schemes. In this paper, we propose a new Ciphertext-Policy Attribute-Based Encryption (CP-ABE) with constant ciphertext length. In our scheme, the number of pairing computations is also constant. In addition, the number of additional bits required from CPA-secure CP-ABE to CCA-secure CP-ABE is reduced by 90% with respect to that of the previous scheme.

Keywords: Ciphertext-Policy Attribute-based encryption; Constant Ciphertext Length.

Reference This paper should be made as follows: Emura, K., Miyaji, A., Nomura, A., Omote, K., and Soshi, M. (2010) 'A Ciphertext-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length', Int. J. Applied Cryptography, Vol**, No*, pp**–**.

Biographical notes: Keita Emura is a Ph.D student at JAIST (Japan Advanced Institute of Science and Technology). His research interests include cryptography and information security. He is a member of the Institute of Electronics, Information and Communication Engineers.

Atsuko Miyaji is a professor at JAIST. Her research interests include the application of projective varieties theory to cryptography and information security. She is a member of the International Association for Cryptologic Research, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, and of the Mathematical Society of Japan.

Akito Nomura is an associate professor at Kanazawa University. His research interests include algebraic number theory. He is a member of the Mathematical Society of Japan.

Kazumasa Omote is a research assistant professor at JAIST. His research interests include applied cryptography and network security. He is a member of the Information Processing Society of Japan.

Masakazu Soshi is an associate professor of Hiroshima City University. His research interests include quantum cryptography and IP traceback. He is a member of the Information Processing Society of Japan.

1 Introduction

A user identity (such as the name, e-mail address and so on) can be used for accessing control of some resources. For example, in Identity-Based Encryption (IBE) schemes such as [9, 12], an encryptor can restrict a decryptor to indicate the identity of the decryptor. An Attribute-Based Encryption (ABE) is an encryption scheme, where users with some attributes can decrypt the ciphertext associated with these attributes. Although IBE schemes have a restriction such that an encryptor only indicates a single decryptor, in ABE schemes, an encryptor can indicate many decryptors by assigning common attributes of these decryptors such as gender, age, affiliation and so on. There are two kinds of ABE, Key-Policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE). KP-ABE [18, 27] are schemes such that each private key is associated with an access structure. CP-ABE [5, 15, 17, 25, 32] are schemes such that each ciphertext is associated with an access structure. An application of KP-ABE is for a biometric system. If an IBE scheme is used to construct the biometric system, then a user's information (such as a finger-print, iris data and so on) is registered as the identity of the user. However, these values are somewhat changed since they depend on a user's condition, on humidity and so on. Therefore, the user is forced to manage secret keys corresponding to all identities. KP-ABE schemes with threshold structures can solve this problem to indicate a threshold value as an error-tolerant value. An application of CP-ABE is for an encrypted storage system. If 1 data is encrypted by using 1 encryption key, then the total number of encryption and decryption keys increases. If plural data are encrypted by using one encryption key, then a fine-grained access control is not achieved. To indicate the set of attributes of a decryptor such as affiliation, the CP-ABE scheme can achieve a fine-grained access control without increasing the number of keys. There are some extended ABE schemes such as ABE schemes with the multi-authority [14, 22], an attribute-based broadcast encryption scheme [23], and a CP-ABE scheme with recipient anonymity [25]. A problem of previous ABE schemes is that the length of the ciphertext depends on the number of attributes. Also, the number of pairing computations depends on the number of attributes. A Predicate Encryption Scheme (PES), where secret keys correspond to predicates, and where ciphertexts are associated with attributes, has been proposed in [11, 21]. It is shown that PES can be regarded as a kind of CP-ABE (see Appendix A and B in [25] for details). Both the [11] and [21] schemes also have the same problems, in that the length of the ciphertext and the number of pairing computations are not constant.

Contribution. In this paper, for the first time we propose a CP-ABE scheme with a constant length of ciphertext and a constant length of the number of pairing computations. The access structure used in our CP-ABE is constructed by AND-gates on multi-valued attributes. This is a subset of the access structures used in [15, 25]. Although previous CP-ABE schemes [5, 15, 17, 25, 32] can complement our access structures, the length of the ciphertext depends on the number of attributes. This means that, until our work, to the best of our knowledge, there has been no scheme that enables a constant ciphertext length with AND-gates on multi-valued attributes. Our scheme enables Chosen Plaintext Attack (CPA) security. In addition, we construct a Chosen Ciphertext Attack (CCA)-secure CP-ABE scheme with constant ciphertext length by using the conversion method proposed in CN07 [15]. This is the main difference between this paper and the previous version [16].

Organization : The paper is organized as follows: Some definitions are presented in Section 2. The previous scheme is introduced in Section 3. Our scheme with CPA security and the CCA-conversion scheme are described in Section 4. The security proof of our scheme is presented in Section 5. Efficiency comparisons are made in Section 6. The security proof of our CCA-conversion scheme is presented in the Appendix.

2 Preliminary

In this section, some definitions are presented. Note that $x \in_R S$ means x is randomly chosen for a set S.

2.1 Bilinear Groups and Complexity Assumption

Definition 1. (**Bilinear Groups**) *Bilinear groups and a bilinear map are defined as follows:*

1. $\mathbb{G}_1$ *and* $\mathbb{G}_T$ *are cyclic groups of prime order* p.
2. g *is a generator of* $\mathbb{G}_1$.
3. e *is an efficiently computable bilinear map* $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ *with the following properties.*
   - *Bilinearity : for all* $u, u', v, v' \in \mathbb{G}_1$, $e(uu', v) = e(u, v)e(u', v)$ *and* $e(u, vv') = e(u, v)e(u, v')$.
   - *Non-degeneracy :* $e(g, g) \neq 1_{\mathbb{G}_T}$ ($1_{\mathbb{G}_T}$ *is the* $\mathbb{G}_T$*'s unit*).

Definition 2. (**DBDH assumption**)
The Decision Bilinear Diffie-Hellman (DBDH) problem in $\mathbb{G}_1$ *is a problem, for input of a tuple* $(g, g^a, g^b, g^c, Z) \in \mathbb{G}_1^4 \times \mathbb{G}_T$, *to decide whether* $Z = e(g, g)^{abc}$ *or not. An algorithm* $\mathcal{A}$ *has advantage* ϵ *in solving the DBDH problem in* $\mathbb{G}_1$ *if* $Adv_{DBDH}(\mathcal{A}) := |\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 0] - \Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^z) = 0]| \geq \epsilon(\kappa)$, *where* $e(g, g)^z \in \mathbb{G}_T \setminus \{e(g, g)^{abc}\}$. *We say that the DBDH assumption holds in* $\mathbb{G}_1$ *if no PPT algorithm has an advantage of at least* ϵ *in solving the DBDH problem in* $\mathbb{G}_1$.

2.2 Definition of Access Structures

Several access structures such as the threshold structure [27], the tree-based access structure [5, 17], AND-gates on positive and negative attributes with wildcards [15], AND-gates on multi-valued attributes with wildcards [25], and the linear access structure [32] are used in previous ABE schemes. In our scheme, we use AND-gates on multi-valued attributes as follows:

Definition 3. *Let $\mathcal{U} = \{att_1, \dots, att_n\}$ be a set of attributes. For $att_i \in \mathcal{U}$, $S_i = \{v_{i,1}, v_{i,2}, \dots, v_{i,n_i}\}$ is a set of possible values, where n_i is the number of possible values for att_i. Let $L = [L_1, L_2, \dots, L_n]$, $L_i \in S_i$ be an attribute list for a user, and $W = [W_1, W_2, \dots, W_n]$, $W_i \in S_i$ be an access structure. The notation $L \models W$ expresses that an attribute list L satisfies an access structure W, namely, $L_i = W_i$ $(i = 1, 2, \dots, n)$.*

The number of access structures is $\prod_{i=1}^{n} n_i$. For each att_i, an encryptor has to *explicitly* indicate a status $v_{i,*}$ from $S_i = \{v_{i,1}, v_{i,2}, \dots, v_{i,n_i}\}$.

2.2.1 Differences between the previous AND-gate structures [15, 25] and ours

If $n_i = 2$ $(i = 1, 2, \dots, n)$, then our structure is the same as the access structures [15] excluding wildcards. In [25], an access structure W is defined as $W = [W_1, W_2, \dots, W_n]$ for $W_i \subseteq S_i$, and $L \models W$ is defined as $L_i \in W_i$ $(i = 1, 2, \dots, n)$. This means that our access structure is a subset of these in [15, 25].

2.2.2 Adequacy of AND-gate structures

If flexible structures can be achieved (e.g., OR-gate, wildcards, tree-based structures, and so on), then we can achieve fine-grained access control. On the contrary, in our scheme, an encryptor must indicate all attribute, explicitly. We insist that AND-gate structures are the most basic usage, namely, an encryptor indicates a concrete set of attributes, and optionally takes advantage of flexible structures. We have only to accept AND-gate structures, then an efficient CP-ABE scheme can be constructed such as to produce a constant ciphertext length and to reduce the number of additional bits required from CPA-secure CP-ABE to CCA-secure CP-ABE. Furthermore, as a difference of secret sharing [28] (in this case, "AND-gate only" means the unanimous structure, namely, the number of access structures is only 1), the number of access structures is $\prod_{i=1}^{n} n_i$. We insist that no redundancy, namely without wild cards, is a reasonable restriction.

2.3 Ciphertext-Policy Attribute-Based Encryption Scheme (CP-ABE)

CP-ABE is described using four algorithms, Setup, KeyGen, Encrypt and Decrypt [15].

Definition 4. *Ciphertext-Policy Attribute-Based Encryption Scheme*

Setup: *This algorithm takes as input the security parameter κ, and returns a public key PK and a master secret key MK.*

KeyGen: *This algorithm takes as input PK, MK and a set of attributes L, and returns a secret key SK_L associated with L.*

Encrypt: *This algorithm takes as input PK, a message M and an access structure W. It returns a ciphertext C with the property that a user with SK_L can decrypt C if and only if $L \models W$.*

Decrypt: *This algorithm takes as input PK, C which was encrypted by W, and SK_L. It returns M if SK_L is associated with $L \models W$.*

2.4 Selective Game for CP-ABE

The selective game for CP-ABE has been defined in [15]. This game captures the indistinguishability of messages and the collusion resistance of secret keys, namely, attackers cannot generate a new secret key by combining their secret keys. To capture the collusion resistance, multiple secret key queries can be issued by the adversary $\mathcal{A}$ after the challenge phase. This means that $\mathcal{A}$ can issue the KeyGen queries L_1 and L_2 such as $(L_1 \not\models W^*) \wedge (L_2 \not\models W^*)$ and $(L_1 \cup L_2) \models W^*$. This collusion resistance is an important property of the CP-ABE scheme, which has not been considered in Hierarchical IBE (HIBE) schemes such as in [8]. A weaker definition of CP-ABE has been considered [20], where an adversary cannot obtain secret keys associated with any att_i such that $att_i \in L \models W^*$. However, we do not use this weaker definition because it does not guarantee collusion resistance. The selective game for CP-ABE under the CCA is defined as follows:

Definition 5. *Selective Game for CP-ABE under the CCA*

Init: *The adversary $\mathcal{A}$ sends the challenge access structure W^* to the challenger.*

Setup: *The challenger runs* Setup *and* KeyGen*, and gives PK to $\mathcal{A}$.*

Phase 1: *$\mathcal{A}$ makes* KeyGen *and* Decryption *queries. Note that these queries can be repeated adaptively.*

KeyGen queries *: $\mathcal{A}$ sends an attribute list L to the challenger for a* KeyGen *query, where $L \not\models W^*$. The challenger answers with a secret key for these attributes.*

Decryption queries *: $\mathcal{A}$ sends a ciphertext C encrypted to W. If C is an invalid ciphertext, then $\mathcal{A}$ loses. The challenger answers the corresponding plaintext M.*

Challenge: *$\mathcal{A}$ sends two equal-length messages M_0 and M_1 to the challenger. The challenger chooses $\mu \in_R \{0,1\}$, and runs $C^* = \mathsf{Encrypt}(PK, M_\mu, W^*)$. The challenger gives the challenge ciphertext C^* to $\mathcal{A}$.*

Phase 2: *Same as Phase 1. $\mathcal{A}$ sends L to the challenger for a KeyGen query. The challenger answers with a secret key for these attributes. Note that $L \not\models W^*$, and these queries can be repeated adaptively.*

Guess: *$\mathcal{A}$ outputs a guess $\mu' \in \{0,1\}$.*

The advantage of $\mathcal{A}$ is defined as $Adv(\mathcal{A}) := |\Pr(\mu' = \mu) - \frac{1}{2}|$.

The selective game for CP-ABE under the CPA is simply defined in the same way as in the above game excluding Decryption queries. Our scheme is proven along with the selective game for CP-ABE under the CPA, and can be converted into the CP-ABE scheme that is proven along with the selective game for CP-ABE under the CCA.

3 Previous CP-ABE

In this section, we summarize a CPA-secure CP-ABE scheme (called the CN07-1 scheme) and a CCA-secure CP-ABE scheme (called the CN07-2 scheme) proposed in [15]. Let $\bar{\mathcal{U}} = \{\neg att_1, \ldots, \neg att_n\}$ be a set of negative attributes for a set of attributes $\mathcal{U}$. We refer to attributes $att_i \in \mathcal{U}$ and their negations $\neg att_i$ as literals. Let $W = \bigwedge_{att_i \in I} \bar{att_i}$ be an access structure, where $I \subseteq \mathcal{U}$ and $\bar{att_i}$ is either att_i or $\neg att_i$. The public key elements T_i, T_{n+i}, T_{2n+i} correspond to the three properties of att_i, namely, *positive*, *negative* and *don't care*.

Protocol 1. *The CPA-secure CP-ABE Scheme [CN07-1] [15]*

Setup(1^κ)**:** *A trusted authority TA chooses a prime number p, a bilinear group $\mathbb{G}_1$ with order p, a generator $g \in \mathbb{G}_1$, $y \in_R \mathbb{Z}_p$ and $t_i \in_R \mathbb{Z}_p$ $(i = 1, 2, \ldots, 3n)$, and computes $Y = e(g,g)^y$ and $T_i = g^{t_i}$ $(i = 1, 2, \ldots, 3n)$. TA outputs $PK = (e, g, Y, T_1, \ldots, T_{3n})$ and $MK = (y, t_1, \ldots, t_{3n})$.*

KeyGen(PK, MK, S)**:** *Every $att_i \notin S$ is implicitly considered to be a negative attribute. TA chooses $r_i \in_R \mathbb{Z}_p$ $(i = 1, 2, \ldots, n)$, sets $r = \sum_{i=1}^{n} r_i$, and computes $\hat{D} = g^{y-r}$. TA computes D_i and F_i as follows:*

$$D_i = \begin{cases} g^{\frac{r_i}{t_i}} & (att_i \in S) \\ g^{\frac{r_i}{t_{n+i}}} & (att_i \notin S) \end{cases},$$

$$F_i = g^{\frac{r_i}{t_{2n+i}}} (att_i \in \mathcal{U})$$

TA outputs $SK = (\hat{D}, \{D_i, F_i\}_{i \in [1,n]})$.

Encrypt(PK, M, W)**:** *Let $W = \bigwedge_{att_i \in I} \bar{att_i}$. An encryptor chooses $s \in_R \mathbb{Z}_p$, and computes $\tilde{C} = M \cdot Y^s$ and $\hat{C} = g^s$. The encryptor computes C_i as follows:*

$$C_i = \begin{cases} T_i^s & (\bar{att_i} = att_i) \\ T_{n+i}^s & (\bar{att_i} = \neg att_i) \\ T_{2n+i}^s & (att_i \in \mathcal{U} \setminus I) \end{cases}$$

The encryptor outputs $C = (W, \tilde{C}, \hat{C}, \{C_i\}_{i \in [1,n]})$.

Decrypt(PK, C, SK)**:** *A decryptor computes the pairing $e(C_i, D_i)$ $(att_i \in I)$ and $e(C_i, F_i)$ $(att_i \notin I)$ as follows:*

$$e(C_i, D_i) = \left\{ \begin{array}{ll} e(g^{t_i \cdot s}, g^{\frac{r_i}{t_i}}) & (\bar{att_i} = att_i) \\ e(g^{t_{n+i} \cdot s}, g^{\frac{r_i}{t_{n+i}}}) & (\bar{att_i} = \neg att_i) \end{array} \right\}$$

$$= e(g,g)^{r_i \cdot s}$$

$$e(C_i, F_i) = e(g^{t_{2n+i} \cdot s}, g^{\frac{r_i}{t_{2n+i}}}) = e(g,g)^{r_i \cdot s}$$

Then $\frac{\tilde{C}}{e(\hat{C}, \hat{D}) \prod_{i=1}^{n} e(g,g)^{r_i \cdot s}} = M \cdot \frac{e(g,g)^{sy}}{e(g,g)^{s(y-r)} e(g,g)^{sr}} = M$ *holds.*

To compute $e(g,g)^{sr}$, the decryptor has to compute either $e(C_i, D_i)$ or $e(C_i, F_i)$ for each i. This means that all C_i are included in a ciphertext, and thus the length of a ciphertext depends on the number of attributes n. Moreover, the CN07-1 scheme does not provide for adding new attributes after Setup. If some attributes are added after Setup, then some users (who have already obtained the secret key) can decrypt a ciphertext which one must not be able to decrypt. For example, let $\mathcal{U} = \{att_1, att_2\}$, and assume that a user U has secret keys of att_1 and att_2, and that a ciphertext C is associated with $W = att_1 \wedge att_2$. Then, U can decrypt a ciphertext associated with $att_1 \wedge att_2 \wedge att_3$ without a secret key of att_3. Concretely, U ignores a part of the ciphertext for att_3. CP-ABE schemes which enable the addition of new attributes after Setup have been proposed in BSW07 [5] and in the 2nd-scheme of NYO08 [25]. If a user wants to decrypt a ciphertext with an access structure including newly added attributes, then the user must once more obtain a new secret key (including newly added attributes) from the trusted authority again. However, the security proof of both schemes contains no reduction, namely, it is proven under the generic group heuristic.

The CN07-1 scheme can be translated into a CCA-secure CP-ABE scheme by using Strongly Existentially Unforgeable (SEU) one-time signatures. This technique is the same as the CHK (Canetti, Halevi and Katz) technique [13] that is a generic construction for a CCA-secure public key encryption using a CPA-secure IBE and an SEU one-time signature. Let SigKeyGen, Sign and Verify be a signature scheme. SigKeyGen is a probabilistic algorithm which outputs a signing/verification key pair $\langle K_s, K_v \rangle$. Sign is a probabilistic algorithm which outputs a signature σ from

K_s and a message M. Verify is a deterministic algorithm which outputs a bit from σ, K_v and M. If Verify outputs 1, this means that σ is a valid signature, and 0 otherwise. The security game of strong existential unforgeability under an adaptive chosen message attack [7] is defined as follows:

Definition 6. Setup: *The challenger runs* SigKeyGen, *and obtains a signing key K_s and a verification key K_v. The adversary $\mathcal{A}$ is given K_v.*

Sign Queries: *$\mathcal{A}$ requests signatures on messages $M_1, M_2, \ldots, M_{q_s} \in \{0,1\}^*$, where q_s is the number of queries. The challenger answers $\sigma_i = \mathsf{Sign}(K_s, M_i)$ for each query. Note that these queries can be repeated adaptively.*

Output: *$\mathcal{A}$ outputs a pair (M^*, σ^*), and wins the game if $(M^*, \sigma^*) \notin \{(M_1, \sigma_1), \ldots, (M_{q_s}, \sigma_{q_s})\}$ and $\mathsf{Verify}(K_v, \sigma^*, M^*) = 1$.*

In the above definition, the forged pair M could have been signed previously. There are techniques which convert (non-strong) existentially unforgeable signatures into strong existentially unforgeable ones [10, 19, 30]. Especially, Huang et. al. [19] have proposed the generic transformation which converts any existentially unforgeable signature into SEU ones by using strong one-time signature schemes. We call this conversion way the HWZ conversion. Note that strong one-time signature schemes can be constructed from any one-way function-based one-time signature. The security game of strong one-time existential unforgeability [19] is simply defined as follows:

Definition 7. Setup: *The challenger runs* SigKeyGen, *and obtains a signing key K_s and a verification key K_v. The adversary $\mathcal{A}$ is given K_v.*

Sign Queries: *$\mathcal{A}$ requests a signature on a message $M \in \{0,1\}^*$. The challenger answers $\sigma = \mathsf{Sign}(K_s, M)$.*

Output: *$\mathcal{A}$ outputs a pair (M^*, σ^*), and wins the game if $(M^*, \sigma^*) \neq (M, \sigma)$ and $\mathsf{Verify}(K_v, \sigma^*, M^*) = 1$.*

Next, we summarize a CCA-secure CP-ABE scheme (called the CN07-2 scheme). Let m be the size of K_v, $K_{v,i}$ be the i-th bit of K_v, and $\mathcal{M} = \{1, \ldots, m\}$. Added to the construction of a CPA-secure scheme, a user has secret keys G_i^0 and G_i^1 ($i \in \mathcal{M}$) associated with $i \in \mathcal{M}$ (this is a secret key of $K_{v,i} = 0$) and $m + i$ (this is a secret key of $K_{v,i} = 1$), respectively.

Protocol 2. *The CCA-secure CP-ABE Scheme [CN07-2] [15]*

Setup(1^κ): *A trusted authority TA chooses a prime number p, a bilinear group $\mathbb{G}_1$ with order p, a generator $g \in \mathbb{G}_1$, $y \in_R \mathbb{Z}_p$, $t_i \in_R \mathbb{Z}_p$ ($i = 1, 2, \ldots, 3n$) and $u_i \in_R \mathbb{Z}_p$ ($i = 1, 2, \ldots, 2m$), and computes $Y = e(g,g)^y$, $T_i = g^{t_i}$ ($i = 1, 2, \ldots, 3n$) and $U_i = g^{u_i}$ ($i = 1, 2, \ldots, 2m$). TA outputs $PK = (e, g, Y, T_1, \ldots, T_{3n}, U_1, \ldots, U_{2m})$ and $MK = (y, t_1, \ldots, t_{3n}, u_1, \ldots, u_{2m})$.*

KeyGen(PK, MK, S): *Every $att_i \notin S$ is implicitly considered to be a negative attribute. TA chooses $r_i \in_R \mathbb{Z}_p$ ($i = 1, 2, \ldots, n$) and $\omega_i \in_R \mathbb{Z}_p$ ($i = 1, 2, \ldots, m$), sets $r = \sum_{i=1}^n r_i + \sum_{i=1}^m \omega_i$, and computes $\hat{D} = g^{y-r}$. TA computes D_i, F_i, G_i^0 and G_i^1 as follows:*

$$D_i = \begin{cases} g^{\frac{r_i}{t_i}} & (att_i \in S) \\ g^{\frac{r_i}{t_{n+i}}} & (att_i \notin S) \end{cases},$$

$$F_i = g^{\frac{r_i}{t_{2n+i}}} (att_i \in \mathcal{U}),\ G_i^0 = g^{\frac{\omega_i}{u_i}},\ G_i^1 = g^{\frac{\omega_i}{u_{m+i}}}$$

TA outputs $SK = (\hat{D}, \{D_i, F_i\}_{i \in [1,n]}, \{G_i^0, G_i^1\}_{i \in [1,m]})$.

Encrypt(PK, M, W): *Let $W = \bigwedge_{att_i \in I} \bar{att}_i$. An encryptor runs* SigGenKey *and obtains a signing/verification key pair $\langle K_s, K_v \rangle$. The encryptor chooses $s \in_R \mathbb{Z}_p$, and computes $\tilde{C} = M \cdot Y^s$ and $\hat{C} = g^s$. The encryptor computes C_i ($i = 1, 2, \ldots, n$) as follows:*

$$C_i = \begin{cases} T_i^s & (\bar{att}_i = att_i) \\ T_{n+i}^s & (\bar{att}_i = \neg att_i) \\ T_{2n+i}^s & (att_i \in \mathcal{U} \setminus I) \end{cases}$$

The encryptor computes E_i ($i = 1, 2, \ldots, m$) as follows:

$$E_i = \begin{cases} U_i^s & (K_{v,i} = 0) \\ U_{m+i}^s & (K_{v,i} = 1) \end{cases}$$

The encryptor computes a signature $\sigma = \mathsf{Sign}(K_s, \langle W, \tilde{C}, \hat{C}, \{C_i\}_{i \in [1,n]}, \{E_i\}_{i \in [1,m]} \rangle)$. The encryptor outputs $C = (W, \sigma, K_v, \tilde{C}, \hat{C}, \{C_i\}_{i \in [1,n]}, \{E_i\}_{i \in [1,m]})$.

Decrypt(PK, C, SK): *A decryptor checks $\mathsf{Verify}(K_v, \sigma, \langle W, \tilde{C}, \hat{C}, \{C_i\}_{i \in [1,n]}, \{E_i\}_{i \in [1,m]} \rangle)$. If σ is valid, then the decryptor computes the pairing $e(C_i, D_i)$ ($att_i \in I$) and $e(C_i, F_i)$ ($att_i \notin I$) as follows:*

$$e(C_i, D_i) = \left\{ \begin{array}{ll} e(g^{t_i \cdot s}, g^{\frac{r_i}{t_i}}) & (\bar{att}_i = att_i) \\ e(g^{t_{n+i} \cdot s}, g^{\frac{r_i}{t_{n+i}}}) & (\bar{att}_i = \neg att_i) \end{array} \right\}$$

$$= e(g,g)^{r_i \cdot s}$$

$$e(C_i, F_i) = e(g^{t_{2n+i} \cdot s}, g^{\frac{r_i}{t_{2n+i}}}) = e(g,g)^{r_i \cdot s}$$

Moreover, for each $i \in \mathcal{M}$, the decryptor computes $e(E_i, G_i^0)$ (when $K_{v,i} = 0$) and $e(E_i, G_i^1)$ (when $K_{v,i} = 1$), and obtains $e(g,g)^{\omega_i \cdot s}$.

Then $\frac{\tilde{C}}{e(\hat{C}, \hat{D}) \prod_{i=1}^n e(g,g)^{r_i \cdot s} \prod_{i=1}^m e(g,g)^{\omega_i \cdot s}} = M \cdot \frac{e(g,g)^{sy}}{e(g,g)^{s(y-r)} e(g,g)^{sr}} = M$ holds.

To enable the CCA-secure scheme, the CN07-2 scheme has to require the additional values $\{U_i\}_{i\in[1,2m]}$ as PK, $\{u_i\}_{i\in[1,2m]}$ as MK, $\{G_i^0, G_i^1\}_{i\in[1,m]}$ as SK, and $(\{E_i\}_{i\in[1,m]}, \sigma, K_v)$ as C. Especially, the overhead of the length of ciphertext is $m|\mathbb{G}_1|$ + signature size + m. If we require the BB short signature [7] as a SEU signature, the verification key[1] is $(u, v) \in \mathbb{G}_1^2$, and the signature is $(\sigma, r) \in \mathbb{G}_1 \times \mathbb{Z}_p$. Therefore, when we evaluate that $m = 161 \times 2 = 322$ bits, the overhead is $322 \times 161 + 321 + 322 = 52485$ bits and the total ciphertext length is $(n+1)|\mathbb{G}_1| + |\mathbb{G}_T| + 52485 = 161(n+1) + 53505$ bits, where $|p| = 160$ bits. Note that any SEU signature scheme can be regarded as a strong one-time signature scheme. Next, we evaluate the overhead when a strong one-time signature is used. In the HWZ conversion [19], Reyzin et. al.'s HORS (Hash to Obtain Random Subset) scheme [26] is recommended as a one-time signature scheme to convert a strong one-time signature scheme. The verification key length is 40960 bits, the signing key length is 61440 bits, and the signature length is 4800 bits in the *Strong* HORS setting recommended in [19]. An efficient strong one-time signature scheme based on a two-tier signature scheme has been proposed in [4]. The verification key length is 480 bits over a 160-bit elliptic curve group. Therefore, from the viewpoint of the length of ciphertext, the BB short signature is the best one.

4 Our construction

In this section, we propose a constant ciphertext length CP-ABE scheme.

4.1 The difficulty

Our main aim is construction of a constant ciphertext length CP-ABE scheme. Here we explain how difficult is the construction of a constant ciphertext length CP-ABE scheme with access structures included wildcard expression. Under the wildcard setting, an encryptor does not expect what secret keys will be used. Concretely, we can construct a CP-ABE scheme such that a ciphertext will be decrypted using secret keys of a correct set of attributes L, and will not be decrypted using secret keys of an illegal set of attributes L', where $L' \cap L \neq L$. However, it is difficult to treat a set of attributes L'' such that $L \subsetneq L''$ (L is a proper subset of L''), since the encryptor cannot expect a redundancy part $L'' \setminus L$. This problem can be solved to admit the attribute depended number of ciphertexts. Here we show how to enable the wildcard expression in previous CP-ABE schemes [15, 25] without constant ciphertext length. For att_i $(i = 1, 2, \ldots, n)$, we can construct a ciphertext C_i which can be decrypted by using a correct secret key of att_i, and cannot be decrypted by using a illegal secret key. This is the same situation of usual public key encryption scheme, and it is used in the (both) CN07 scheme [15]. In the CN07 scheme, an encryptor computes C_i $(i = 1, 2, \ldots, n)$ for att_i by using one of the public key (T_i, T_{n+1}, T_{2n+i}). Let W be an access structure chosen by the encryptor. If $att_i \in W$ (resp. $\neg att_i \in W$), then T_i (resp. T_{n+i}) is used. Otherwise, if $att_i \notin W$ (this means the encryptor does not care about att_i), then T_{2n+i} is used. A user has three kinds of secret keys (for positive or negative attributes and for wildcards), and decrypts C_i by using a "positive key" ($att_i \in W$) or a "negative key" ($\neg att_i \in W$), or a "wildcard key" ($att_i \notin W$). Therefore, every user has $2n$ secret keys, respectively. It is easy to construct a extended CN07 scheme with AND-gates on multi-valued attributes with wildcards. However, every users have $\sum_{i=1}^{n} n_i$ secret keys, respectively. However, this kind of construction requires the number of n ciphertexts. On the other hand, each user has only n secret keys in the NYO08 scheme [25]. However, the number of $\sum_{i=1}^{n} n_i$ ciphertexts is required. An access structure W is defined as $W = [W_1, W_2, \ldots, W_n]$ for $W_i \subseteq S_i$. If $v_{i,t} \in W_i$, then $C_{i,t}$ is correctly computed. Otherwise, if $v_{i,t} \notin W_i$, $C_{i,t}$ is randomly chosen. A user has one state $v_{i,\ell}$ $(\ell \in [1, n_i])$ for each attribute att_i, and can decrypt $C_{i,\ell}$ if $v_{i,\ell} \in W_i$. If $S_i = W_i$, then any user can decrypt a ciphertext corresponding to att_i, since all states of att_i are included in W_i. This means att_i is indicated as a wildcard, since the encryptor does not care about att_i. From the above considerations, the wildcard expression is achieved to provide the number of n (or $\sum_{i=1}^{n} n_i$) ciphertexts. Otherwise, in the AND-gates on multi-valued attributes (without wildcard) setting, an encryptor does not have to expect a redundancy part, since an access structure is explicitly described. In addition, the sum of master keys $t_{i,j}$ (described as $\sum_{v_{i,j}\in W} t_{i,j}$) is applied to express an access structure W. This form enables the constant ciphertext length. Although a mapping $W \to \sum_{v_{i,j}\in W} t_{i,j}$ is not one-to-one, the condition $\sum_{v_{i,j}\in W} t_{i,j} \neq \sum_{v_{i,j}\in W'} t_{i,j}$, where $W \neq W'$, holds with overwhelming probability. See Section 4.3 for details. A generic construction of an identity-based encryption scheme with wildcards (called WIBE for short) from any HIBE scheme has been proposed [2]. However, a user's secret key size is exponential in the depth of the hierarchy tree. To solve this problem, WIBE schemes also have been constructed based on the Watre's HIBE [31], the Boneh-Boyen HIBE [6], and the Boneh-Boyen-Goh HIBE [8], respectively. The length of the secret key linearly depends on the maximal hierarchy depth. However, these schemes do not enable the constant ciphertext length, since the length of ciphertext also linearly depends on the maximal hierarchy depth. Next, we discuss the difference between a CP-ABE scheme with AND-gates on multi-valued attributes (without wildcard) and a HIBE scheme. In a HIBE scheme, the user's identity in depth ℓ is described using the set of identities from root node to own node such that $I_\ell := ID_1||ID_1||\ldots||ID_\ell$. The user with the secret key of I_ℓ can generate a new secret key of $I_{\ell'}$, where $I_{\ell'} := I_\ell||ID_{\ell'}$. On the other

[1] Note that, in the original paper [7], the verification key is $(g_1, g_2, u, v, z) \in \mathbb{G}_1^4 \times \mathbb{G}_T$. However, g_1, g_2 and $z = e(g_1, g_2)$ are regarded as common public values.

hand, the CP-ABE scheme has to require the collusion resistance, namely, attackers cannot generate a new secret key by combining their secret keys. Therefore, proposing a constant ciphertext length CP-ABE scheme with AND-gates on multi-valued attributes is a challenging problem, because a HIBE scheme cannot be regarded as a CP-ABE scheme, since HIBE does not satisfy the collusion resistance property.

4.2 Our schemes

Protocol 3. ***Our CPA Secure CP-ABE Scheme with Constant Ciphertext Length***

Setup(1^κ)**:** *A trusted authority TA chooses a prime number p, a bilinear group $(\mathbb{G}_1, \mathbb{G}_T)$ with order p, a generator $g \in \mathbb{G}_1$, $h \in \mathbb{G}_1$, $y \in_R \mathbb{Z}_p$ and $t_{i,j} \in_R \mathbb{Z}_p$ $(i \in [1,n], j \in [1,n_i])$. TA computes $Y = e(g,h)^y$, and $T_{i,j} = g^{t_{i,j}}$ $(i \in [1,n], j \in [1,n_i])$. TA outputs $PK = (e, g, h, Y, \{T_{i,j}\}_{i\in[1,n],j\in[1,n_i]})$ and $MK = (y, \{t_{i,j}\}_{i\in[1,n],j\in[1,n_i]})$. Note that $\forall L, L'$ $(L \neq L')$, $\sum_{v_{i,j}\in L} t_{i,j} \neq \sum_{v_{i,j}\in L'} t_{i,j}$ is assumed.*

KeyGen(PK, MK, L)**:** *TA chooses $r \in_R \mathbb{Z}_p$, outputs $SK_L = (h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r, g^r)$, and gives SK_L to a user with L.*

Encrypt(PK, M, W)**:** *An encryptor chooses $s \in_R \mathbb{Z}_p$, and computes $C_1 = M \cdot Y^s$, $C_2 = g^s$ and $C_3 = (\prod_{v_{i,j}\in W} T_{i,j})^s$. The encryptor outputs $C = (W, C_1, C_2, C_3)$.*

Decrypt(PK, C, SK_L)**:** *A decryptor computes what follows:*

$$\begin{aligned}&\frac{C_1 \cdot e(C_3, g^r)}{e(C_2, h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r)}\\ =&\frac{M \cdot e(g,h)^{sy} e(g,g)^{sr\sum_{v_{i,j}\in W} t_{i,j}}}{e(g,h)^{sy} e(g,g)^{sr\sum_{v_{i,j}\in L} t_{i,j}}}\\ =&M\end{aligned}$$

4.3 Construction of secret keys $t_{i,j}$

In our scheme, $\sum_{v_{i,j}\in L} t_{i,j} \neq \sum_{v_{i,j}\in L'} t_{i,j}$ is assumed. If there exist L and L' $(L \neq L')$ such that $\sum_{v_{i,j}\in L} t_{i,j} = \sum_{v_{i,j}\in L'} t_{i,j}$, a user with the attribute list L' can decrypt a ciphertext associated with W, where $L' \not\models W$ and $L \models W$. Note that the assumption holds with overwhelming probability $\frac{p(p-1)\cdots(p-(N-1))}{p^N} > \frac{(p-(N-1))^N}{p^N} = (1 - \frac{N-1}{p})^N > 1 - \frac{N(N-1)}{p} > 1 - \frac{N^2}{p}$, where $N := \prod_{i=1}^n n_i$. Therefore, if each secret key $t_{i,j}$ is chosen at random from $\mathbb{Z}_p$, then our assumption is natural.

4.4 CCA-conversion scheme

Our scheme can be converted into a CCA-secure CP-ABE scheme by using the conversion method proposed in CN07 [15]. For K_v, let $V_v = \{K_{v,1}, K_{v,2}, \ldots, K_{v,m}\}$ be the set of bits of K_v, $u_{v,i}$ be u_i (if $K_{v,i} = 0$) or u_{m+i} (if $K_{v,i} = 1$), and $U_{v,i} = g^{u_{v,i}}$.

Protocol 4. ***Our CCA-Secure CP-ABE Scheme with Constant Ciphertext Length***

Setup(1^κ)**:** *A trusted authority TA chooses a prime number p, a bilinear group $(\mathbb{G}_1, \mathbb{G}_T)$ with order p, a generator $g \in \mathbb{G}_1$, $h \in \mathbb{G}_1$, $y \in_R \mathbb{Z}_p$, $t_{i,j} \in_R \mathbb{Z}_p$ $(i \in [1,n], j \in [1,n_i])$ and $u_i \in_R \mathbb{Z}_p$ $(i = 1, 2, \ldots, 2m)$. TA computes $Y = e(g,h)^y$, $T_{i,j} = g^{t_{i,j}}$ $(i \in [1,n], j \in [1,n_i])$ and $U_i = g^{u_i}$ $(i = 1, 2, \ldots, 2m)$. TA outputs $PK = (e, g, h, Y, \{T_{i,j}\}_{i\in[1,n],j\in[1,n_i]}, \{U_i\}_{i\in[1,2m]})$ and $MK = (y, \{t_{i,j}\}_{i\in[1,n],j\in[1,n_i]}, \{u_i\}_{i\in[1,2m]})$. Note that $\forall L, L'$ $(L \neq L')$ and $\forall V_v, V_{v'}$ $(V_v \neq V_{v'})$, $(\sum_{v_{i,j}\in L} t_{i,j} + \sum_{K_{v,i}\in V_v} u_{v,i}) \neq (\sum_{v_{i,j}\in L'} t_{i,j} + \sum_{K_{v',i}\in V_{v'}} u_{v',i})$ is assumed.*

KeyGen(PK, MK, L)**:** *TA chooses $r \in_R \mathbb{Z}_p$, computes $h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r, g^r$ and $\{G_i^0 = g^{u_i r}, G_i^1 = g^{u_{m+i} r}\}_{i\in[1,m]}$, and gives $SK_L = (h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r, g^r, \{G_i^0, G_i^1\}_{i\in[1,m]})$ to a user with L.*

Encrypt(PK, M, W)**:** *An encryptor runs* SigGenKey *and obtains a signing/verification key pair $\langle K_s, K_v\rangle$. The encryptor chooses $s \in_R \mathbb{Z}_p$, and computes $C_1 = M \cdot Y^s$, $C_2 = g^s$ and $C_3 = \left((\prod_{v_{i,j}\in W} T_{i,j})(\prod_{K_{v,i}\in V_v} U_{v,i})\right)^s$. The encryptor computes a signature $\sigma =$ Sign$(K_s, \langle W, C_1, C_2, C_3\rangle)$. The encryptor outputs $C = (W, \sigma, K_v, C_1, C_2, C_3)$.*

Decrypt(PK, C, SK_L)**:** *A decryptor checks* Verify$(K_v, \sigma, \langle W, C_1, C_2, C_3\rangle)$*. If σ is valid, then the decryptor computes what follows: $\prod_{i=1}^m G_i^{b_v}$ ($b_v = 0$ if $K_{v,i} = 0$, and $b_v = 1$ if $K_{v,i} = 1$), and*

$$\begin{aligned}&\frac{C_1 \cdot e(C_3, g^r)}{e(C_2, h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r \prod_{i=1}^m G_i^{b_v})}\\ =&\frac{C_1 \cdot e(C_3, g^r)}{e(C_2, h^y(g^{((\sum_{v_{i,j}\in L} t_{i,j})+(\sum_{K_{v,i}\in V_v} u_{v,i}))^r})}\\ =&\frac{M \cdot e(g,h)^{sy} e(g,g)^{sr((\sum_{v_{i,j}\in W} t_{i,j})+(\sum_{K_{v,i}\in V_v} u_{v,i}))}}{e(g,h)^{sy} e(g,g)^{sr((\sum_{v_{i,j}\in L} t_{i,j})+(\sum_{K_{v,i}\in V_v} u_{v,i}))}}\\ =&M\end{aligned}$$

4.5 Order of a finite group

In our CCA-conversion scheme, $(\sum_{v_{i,j}\in L} t_{i,j} + \sum_{K_{v,i}\in V_v} u_{v,i}) \neq (\sum_{v_{i,j}\in L'} t_{i,j} + \sum_{K_{v',i}\in V_{v'}} u_{v',i})$ is assumed. This assumption holds with probability $1 - \frac{(2^m \prod_{i=1}^{n} n_i))^2}{p}$. If we use the BB short signature as a SEU signature scheme, then we can evaluate $m = 322$. We cannot use the same size finite groups of the scheme ($|p| = 160$ bits). Let $n = 10$ and $\prod_{i=1}^{n} n_i = 2^{20}$. We believe that this setting is enough in practice. Then, the length of a prime order p can be 728 bits. Then the ciphertext length of our CCA-secure scheme is $2|\mathbb{G}_1| + |\mathbb{G}_T| = 2 \times 729 + 4368 = 5826$ bits, whereas the ciphertext length of the CN07-2 scheme is $161(n+1) + 1020 + 322m + 321 + m = 55276$ bits. This means that our scheme can enable the CCA security with an approximately 4500 bits overhead, whereas the CN07-2 scheme requires an approximately 52500 bits overhead to enable the CCA security. To sum up, the number of additional bits required from CPA-secure CP-ABE to CCA-secure CP-ABE is reduced by 90% with respect to that of previous scheme.

5 Security Analysis

In this section, we prove that our scheme is CPA-secure under the DBDH assumption.

Theorem 1. *Our scheme satisfies the indistinguishability of messages under the DBDH assumption and the chosen message attack.*

Proof. We suppose that the adversary $\mathcal{A}$ wins the selective CPA game for CP-ABE with the advantage ϵ. Then we can construct an algorithm $\mathcal{B}$ that breaks the DBDH assumption with the advantage $\frac{\epsilon}{2}(1 - \frac{N^2}{p})$, where $N := \prod_{i=1}^{n} n_i$ is the number of expressed access structures. The DBDH challenger selects $a, b, c, z \in_R \mathbb{Z}_p$, $\nu \in_R \{0, 1\}$, and g, where $\langle g \rangle = \mathbb{G}_1$. If $\nu = 0$, then $Z = e(g, g)^{abc}$. Otherwise, if $\nu = 1$, then $Z = e(g, g)^z$. The DBDH challenger gives the DBDH instance $(g, g^a, g^b, g^c, Z) \in \mathbb{G}_1^4 \times \mathbb{G}_T$ to $\mathcal{B}$. First, $\mathcal{B}$ is given the challenge access structure W^* from $\mathcal{A}$. Let $W^* = [W_1^*, \ldots, W_n^*]$. $\mathcal{B}$ selects $u \in_R \mathbb{Z}_p^*$, and sets $h = g^u$ and $Y = e(g^a, (g^b)^u) = e(g, h)^{ab}$. Moreover, $\mathcal{B}$ selects $t'_{i,j} \in_R \mathbb{Z}_p$ ($i \in [1, n], j \in [1, n_i]$), and sets $t_{i,j} = t'_{i,j}$ (in the case where $v_{i,j} = W_i^*$) and $t_{i,j} = bt'_{i,j}$ (in the case where $v_{i,j} \neq W_i^*$), and computes public keys $T_{i,j}$ ($i \in [1, n], j \in [1, n_i]$) as follows:

$$T_{i,j} = g^{t_{i,j}} = \begin{cases} g^{t'_{i,j}} & (v_{i,j} = W_i^*) \\ (g^b)^{t'_{i,j}} & (v_{i,j} \neq W_i^*) \end{cases}$$

$\mathcal{B}$ gives $PK = (e, g, h, Y, \{T_{i,j}\}_{i \in [1,n], j \in [1,n_i]})$ to $\mathcal{A}$. For KeyGen query L, there exists $v_{i,\ell}$ such that $v_{i,\ell} = L_i \wedge v_{i,\ell} \neq W_i^*$, since $L \not\models W^*$. Therefore, $\sum_{v_{i,j} \in L} t_{i,j}$ can be represented as $\sum_{v_{i,j} \in L} t_{i,j} = T_1 + bT_2$, where $T_1, T_2 \in \mathbb{Z}_p$. $\mathcal{B}$ can compute T_1 and T_2, since both T_1 and T_2 are represented by the sum of $t'_{i,j}$. $\mathcal{B}$ chooses $\beta \in_R \mathbb{Z}_p$, sets $r := \frac{\beta - ua}{T_2}$, and computes $SK_L = ((g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}}, g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}})$. We show that SK_L is a valid secret key as follows:

$$\begin{aligned}
(g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}} &= g^{uab} \cdot g^{-uab}(g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}} \\
&= g^{uab} \cdot g^{\frac{T_1}{T_2}(\beta - ua)} \cdot g^{b(\beta - ua)} \\
&= g^{uab}(g^{T_1} \cdot g^{bT_2})^{\frac{\beta - ua}{T_2}} \\
&= g^{uab}(g^{T_1 + bT_2})^{\frac{\beta - ua}{T_2}} \\
&= h^y (g^{\sum_{v_{i,j} \in L} t_{i,j}})^r,
\end{aligned}$$

and

$$g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}} = g^{\frac{\beta - ua}{T_2}} = g^r$$

If $T_2 = 0 \bmod p$, then $\mathcal{B}$ aborts. If $T_2 = 0 \bmod p$ holds, then there exists L such that $\sum_{v_{i,j} \in L} t_{i,j} = \sum_{v_{i,j} \in W^*} t_{i,j}$ holds. Therefore, this probability is at most $\frac{N^2}{p}$. See Section 4.3 for details. For the challenge ciphertext, $\mathcal{B}$ chooses $\mu \in_R \{0, 1\}$, computes $C_1^* = M_\mu \cdot Z^u$, $C_2^* = g^c$ and $C_3^* = (g^c)^{\sum_{v_{i,j} \in W^*} t'_{i,j}}$, and sends (C_1^*, C_2^*, C_3^*) to $\mathcal{A}$. Finally, $\mathcal{A}$ outputs $\mu' \in \{0, 1\}$. $\mathcal{B}$ outputs 1 if $\mu' = \mu$, or outputs 0 if $\mu' \neq \mu$. If $Z = e(g, g)^{abc}$, then (C_1^*, C_2^*, C_3^*) is a valid ciphertext associated with W^*. Therefore, $\mathcal{A}$ has the advantage ϵ. Hence, $\Pr[\mathcal{B} \to 1 | Z = e(g, g)^{abc}] = \Pr[\mu' = \mu | Z = e(g, g)^{abc}] = \frac{1}{2} + \epsilon$. Otherwise, if $Z = e(g, g)^z$, $\mathcal{A}$ has no advantage to distinguish a bit μ, since all parts of the challenge ciphertext when $\mu = 0$ and when $\mu = 1$ have the same distributions. Hence, $\Pr[\mathcal{B} \to 0 | Z = e(g, g)^z] = \Pr[\mu' \neq \mu | Z = e(g, g)^z] = \frac{1}{2}$. It follows that $\mathcal{B}$'s advantage in the DBDH game is $\frac{\epsilon}{2}(1 - \frac{N^2}{p})$. □

The CCA-conversion scheme is CCA secure under both the DBDH assumption and a signature scheme is strongly unforgeable. Proof of theorem 2 is given in the Appendix.

Theorem 2. *Our CCA-conversion scheme satisfies the indistinguishability of messages under the DBDH assumption and the chosen ciphertext attack.*

Although a symmetric bilinear map is required in these proofs, our schemes can be proven with an asymmetric bilinear map such as the Weil or Tate pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ over MNT curves [24], where $\mathbb{G}_1$ and $\mathbb{G}_2$ are distinct groups. Then the indistinguishability of messages can be proven under the DBDH assumption over $\mathbb{G}_2$ [3].

6 Comparison

Let PK, MK, SK and Ciphertext be the size of the public key, of the master key, of the secret key, and the ciphertext length excluding the access structure, respectively. Moreover, Enc. and Dec. are the computational times of encryption and decryption, respectively. We use the terms DBDH, DMBDH [27] and D-Linear [25] to refer to the Decision Bilinear Diffie-Hellman assumption, the Decision Modified Bilinear Diffie-Hellman assumption and the Decision Linear assumption, respectively. The notation $|\mathbb{G}|$

Table 1. Size of each value

	PK	MK	SK	Ciphertext
SW05 [27]	$N'\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(N'+1)\|\mathbb{Z}_p\|$	$r_2\|\mathbb{G}_1\|$	$r_1\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
GPSW06 [18]	$N'\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(N'+1)\|\mathbb{Z}_p\|$	$r_2\|\mathbb{G}_1\|$	$r_1\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
CN07 [15]	$(N'+1)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(N'+1)\|\mathbb{Z}_p\|$	$(2n+1)\|\mathbb{G}_1\|$	$(n+1)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
BSW07 [5]	$3\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$\|\mathbb{Z}_p\| + \|\mathbb{G}\|$	$(2n+1)\|\mathbb{G}_1\|$	$(2r_2+1)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
NYO08 [25]	$(2N'+1)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(2N'+1)\|\mathbb{Z}_p\|$	$(3n+1)\|\mathbb{G}_1\|$	$(2N'+1)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
W08 [32]	$2\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$\|\mathbb{G}_1\|$	$(1+n+r_2)\|\mathbb{G}_1\|$	$(1+r_1 n)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
Our CPA scheme	$(2N'+3)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(N'+1)\|\mathbb{Z}_p\|$	$2\|\mathbb{G}_1\|$	$2\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$
Our CCA scheme	$(2N'+2m+3)\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$	$(N'+2m+1)\|\mathbb{Z}_p\|$	$2\|\mathbb{G}_1\| + 2m$	$2\|\mathbb{G}_1\| + \|\mathbb{G}_T\|$

Table 2. Computational time of each algorithm

	Enc.	Dec.
SW05 [27]	$r_1\mathbb{G}_1 + 2\mathbb{G}_T$	$r_1 e + (r_1+1)\mathbb{G}_T$
GPSW06 [18]	$r_1\mathbb{G}_1 + 2\mathbb{G}_T$	$r_1 e + (r_1+1)\mathbb{G}_T$
CN07 [15]	$(n+1)\mathbb{G}_1 + 2\mathbb{G}_T$	$(n+1)e + (n+1)\mathbb{G}_T$
BSW07 [5]	$(2r_1+1)\mathbb{G}_1 + 2\mathbb{G}_T$	$2r_1 e + (2r_1+2)\mathbb{G}_T$
NYO08 [25]	$(2N'+1)\mathbb{G}_1 + 2\mathbb{G}_T$	$(3n+1)e + (3n+1)\mathbb{G}_T$
W08 [32]	$(1+3r_1 n)\mathbb{G}_1 + 2\mathbb{G}_T$	$(1+n+r_1)e + (3r_1-1)\mathbb{G}_1 + 3\mathbb{G}_T$
Our CPA scheme	$(n+1)\mathbb{G}_1 + 2\mathbb{G}_T$	$2e + 2\mathbb{G}_T$
Our CCA scheme	$(n+m+1)\mathbb{G}_1 + 2\mathbb{G}_T$	$2e + (m+1)\mathbb{G}_2 + 2\mathbb{G}_T$

Table 3. Expressiveness of policy

SW05 [27]	Threshold Structure
GPSW06 [18]	Tree-based Structure
CN07-1 [15]	AND-gates on positive and negative attributes with wildcards
BSW07 [5]	Tree-Based Structure
W08 [32]	Linear Structure
NYO08 [25]	AND-gates on multi-valued attributes with wildcards
Our schemes	AND-gates on multi-valued attributes

Table 4. Performance Results for $n = 3$

	Enc. Time	Dec. Time
CN07-1 [15]	0.028sec	0.031sec
NYO08 [25]	0.032sec	0.078sec
Our CPA scheme	0.015sec	0.015sec

is the bit-length of the element which belongs to $\mathbb{G}$. Let the notations $k\mathbb{G}$ and ke (where $k \in \mathbb{Z}_{>0}$) be the k-times calculation over the group $\mathbb{G}$ and pairing, respectively. Let $\mathcal{U} = \{att_1, att_2, \ldots, att_n\}$ be the set of attributes. Let γ_1 ($|\gamma_1| = r_1$) be a set of attributes associated with the ciphertext, and γ_2 ($|\gamma_2| = r_2$) a set of attributes associated with the secret key. Actually, γ_2 is different for each user. Let $N' := \sum_{i=1}^{n} n_i$ be the total number of possible statements of attributes. The computational time over $\mathbb{Z}_p$ is ignored as usual. Note that SW05 [27] and GPSW06 [18] do not consider the multi-valued attributes. They assign each attribute att_i with a leaf node of an attribute tree. To estimate the same level, we show the result in the case of that each multi-valued attribute $v_{i,j}$ is assigned with a leaf node of an attribute tree. Our scheme is efficient in that the ciphertext length and the costs of decryption do not depend on the number of attributes. In particular, the number of pairing computations is constant. No previous schemes provide these properties. An access structure is constructed by AND-gates on multi-valued attributes defined in section 2.2, which is a subset of the access structures in [25]. To the best of our knowledge, our scheme is the first constant ciphertext length CP-ABE with AND-gates on multi-valued attributes.

Our scheme does not provide recipient anonymity when a symmetric bilinear group is applied. Concretely, for an access structure W', an attacker can run the DDH test $e(C_2, \prod_{v_{i,j} \in W'} T_{i,j}) \stackrel{?}{=} e(C_3, g)$. Then, the attacker can determine whether an encryptor used the policy W' or not. When a DDH-hard bilinear group is applied, namely, the eXternal Diffie-Hellman (XDH) assumption holds, we can show that our scheme enables the property of the hidden encryptor-specified policies in the generic bilinear group model [5, 8, 29]. Let $g_2 \in \mathbb{G}_2$ and $g_1 = \psi(g_2) \in \mathbb{G}_1$ be generators, where ψ is an efficiently computable isomorphism $\mathbb{G}_2 \to \mathbb{G}_1$. We say that the XDH assumption holds if the DDH problem is hard in $\mathbb{G}_1$, namely, ψ^{-1} is uncomputable. In the same way as shown in [25], we have only to show that the adversary cannot run the DDH test, even if the adversary is given g_1, g_1^s, and all $T_{i,j} = g_1^{t_{i,j}}$. Under the XDH assumption, where the adversary cannot compute $g_2^{t_{i,j}} \in \mathbb{G}_2$ from $g_1^{t_{i,j}} \in \mathbb{G}_1$, this condition holds.

The CN07-1 scheme [15], the NYO08 scheme [25] and ours are implemented with *the same access structure* $\{v_{1,1}, v_{2,1}, v_{3,1}\}$, by using the Pairing-Based Cryptography (PBC) Library ver. 0.4.18 [1]. The performance results are shown in Table 4. Our experiment was performed by using a PC with an Intel(R) Core(TM)2 Duo CPU P8400 2.26GHz Windows Vista Home Premium Edition Service Pack 1. The execution of our scheme takes a very small amount of time, which is quite feasible for practical implementation. When $n = 3$, our decryption algorithm is approximately twice as fast as that of the CN07-1 scheme, and approximately five times faster than that of the NYO08 scheme.

7 Conclusion

In this paper, we propose a constant ciphertext length CP-ABE with AND-gates on multi-valued attributes. Moreover, the number of pairing computations is also constant. In addition, the number of additional bits required from CPA-secure CP-ABE to CCA-secure CP-ABE is reduced by 90% with respect to that of previous scheme. To the best of our knowledge, this is the first such construction.

REFERENCES

[1] The pairing-based cryptography (pbc) library. `http://crypto.stanford.edu/pbc/`.

[2] Michel Abdalla, Dario Catalano, Alexander W. Dent, John Malone-Lee, Gregory Neven, and Nigel P. Smart. Identity-based encryption gone wild. In *ICALP (2)*, pages 300–311, 2006.

[3] Michel Abdalla, Alexander W. Dent, John Malone-Lee, Gregory Neven, Duong Hieu Phan, and Nigel P. Smart. Identity-based traitor tracing. In *Public Key Cryptography*, pages 361–376, 2007.

[4] Mihir Bellare and Sarah Shoup. Two-tier signatures, strongly unforgeable signatures, and fiat-shamir without random oracles. In *Public Key Cryptography*, pages 201–216, 2007.

[5] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *IEEE Symposium on Security and Privacy*, pages 321–334, 2007.

[6] Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *EUROCRYPT*, pages 223–238, 2004.

[7] Dan Boneh and Xavier Boyen. Short signatures without random oracles and the sdh assumption in bilinear groups. *J. Cryptology*, 21(2):149–177, 2008.

[8] Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical identity based encryption with constant size ciphertext. In *EUROCRYPT*, pages 440–456, 2005.

[9] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003.

[10] Dan Boneh, Emily Shen, and Brent Waters. Strongly unforgeable signatures based on computational diffie-hellman. In *Public Key Cryptography*, pages 229–240, 2006.

[11] Dan Boneh and Brent Waters. Conjunctive, subset, and range queries on encrypted data. In *TCC*, pages 535–554, 2007.

[12] Xavier Boyen and Brent Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *CRYPTO*, pages 290–307, 2006.

[13] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In *EUROCRYPT*, pages 207–222, 2004.

[14] Melissa Chase. Multi-authority attribute based encryption. In *TCC*, pages 515–534, 2007.

[15] Ling Cheung and Calvin C. Newport. Provably secure ciphertext policy abe. In *ACM Conference on Computer and Communications Security*, pages 456–465, 2007.

[16] Keita Emura, Atsuko Miyaji, Akito Nomura, Kazumasa Omote, and Masakazu Soshi. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length. In *ISPEC*, pages 13–23, 2009.

[17] Vipul Goyal, Abhishek Jain, Omkant Pandey, and Amit Sahai. Bounded ciphertext policy attribute based encryption. In *ICALP (2)*, pages 579–591, 2008.

[18] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM Conference on Computer and Communications Security*, pages 89–98, 2006.

[19] Qiong Huang, Duncan S. Wong, and Yiming Zhao. Generic transformation to strongly unforgeable signatures. In *ACNS*, pages 1–17, 2007.

[20] Luan Ibraimi, Qiang Tang, Pieter H. Hartel, and Willem Jonker. Efficient and provable secure ciphertext-policy attribute-based encryption schemes. In *ISPEC*, pages 1–12, 2009.

[21] Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *EUROCRYPT*, pages 146–162, 2008.

[22] Huang Lin, Zhenfu Cao, Xiaohui Liang, and Jun Shao. Secure threshold multi authority attribute based encryption without a central authority. In *INDOCRYPT*, pages 426–436, 2008.

[23] David Lubicz and Thomas Sirvent. Attribute-based broadcast encryption scheme made efficient. In *AFRICACRYPT*, pages 325–342, 2008.

[24] Atsuko Miyaji, Masaki Nakabayashi, and Shunzo Takano. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transactions*, 84-A(5):1234–1243, 2001.

[25] Takashi Nishide, Kazuki Yoneyama, and Kazuo Ohta. Attribute-based encryption with partially hidden encryptor-specified access structures. In *ACNS*, pages 111–129, 2008.

[26] Leonid Reyzin and Natan Reyzin. Better than biba: Short one-time signatures with fast signing and verifying. In *ACISP*, pages 144–153, 2002.

[27] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, pages 457–473, 2005.

[28] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[29] Victor Shoup. Lower bounds for discrete logarithms and related problems. In *EUROCRYPT*, pages 256–266, 1997.

[30] Isamu Teranishi, Takuro Oyama, and Wakaha Ogata. General conversion for obtaining strongly existentially unforgeable signatures. *IEICE Transactions*, 91-A(1):94–106, 2008.

[31] Brent Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT*, pages 114–127, 2005.

[32] Brent Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. Cryptology ePrint Archive, Report 2008/290, 2008.

Appendix

In this Appendix, we give the proof of Theorem 2:

Proof. We suppose that the adversary $\mathcal{A}$ wins the selective CCA game for CP-ABE with the advantage ϵ. Then we can construct an algorithm $\mathcal{B}$ that breaks the DBDH assumption with the advantage $\frac{\epsilon}{2}(1-\frac{N^2}{p})$, where $N := \prod_{i=1}^{n} n_i$ is the number of expressed access structures. The DBDH challenger selects $a,b,c,z \in_R \mathbb{Z}_p$, $\nu \in_R \{0,1\}$, and g, where $\langle g \rangle = \mathbb{G}_1$. If $\nu = 0$, then $Z = e(g,g)^{abc}$. Otherwise, if $\nu = 1$, then $Z = e(g,g)^z$. The DBDH challenger gives the DBDH instance $(g, g^a, g^b, g^c, Z) \in \mathbb{G}_1^4 \times \mathbb{G}_T$ to $\mathcal{B}$. $\mathcal{B}$ runs $\mathsf{SigKeyGen}$, and obtains $\langle K_{s^*}, K_{v^*}\rangle$. First, $\mathcal{B}$ is given the challenge access structure W^* from $\mathcal{A}$. Let $W^* = [W_1^*, \ldots, W_n^*]$. $\mathcal{B}$ selects $u \in_R \mathbb{Z}_p^*$, and sets $h = g^u$ and $Y = e(g^a, (g^b)^u) = e(g,h)^{ab}$. Moreover, $\mathcal{B}$ selects $t'_{i,j} \in_R \mathbb{Z}_p$ $(i \in [1,n], j \in [1,n_i])$ and $u_i \in_R \mathbb{Z}_p$ $(i \in [1,2m])$, and sets $t_{i,j} = t'_{i,j}$ (in the case where $v_{i,j} = W_i^*$) and $t_{i,j} = bt'_{i,j}$ (in the case where $v_{i,j} \neq W_i^*$), and computes public keys U_i $(i \in [1,2m])$ and $T_{i,j}$ $(i \in [1,n], j \in [1,n_i])$ as follows:

$$U_i = g^{u_i}$$

$$T_{i,j} = g^{t_{i,j}} = \begin{cases} g^{t'_{i,j}} & (v_{i,j} = W_i^*) \\ (g^b)^{t'_{i,j}} & (v_{i,j} \neq W_i^*) \end{cases}$$

$\mathcal{B}$ gives $PK = (e, g, h, Y, \{T_{i,j}\}_{i\in[1,n],j\in[1,n_i]}, \{U_i\}_{i\in[1,2m]})$ to $\mathcal{A}$. For KeyGen query L, there exists $v_{i,\ell}$ such that $v_{i,\ell} = L_i \wedge v_{i,\ell} \neq W_i^*$, since $L \not\models W^*$. Therefore, $\sum_{v_{i,j}\in L} t_{i,j}$ can be represented as $\sum_{v_{i,j}\in L} t_{i,j} = T_1 + bT_2$, where $T_1, T_2 \in \mathbb{Z}_p$. $\mathcal{B}$ can compute T_1 and T_2, since both T_1 and T_2 are represented by the sum of $t'_{i,j}$. $\mathcal{B}$ chooses $\beta \in_R \mathbb{Z}_p$, sets $r := \frac{\beta - ua}{T_2}$, and computes $SK_L = ((g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}}, g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}}, \{G_i^0 = (g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}})^{u_i}, G_i^1 = (g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}})^{u_{m+i}}\})$. We show that SK_L is a valid secret key as follows:

$$\begin{aligned}
(g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}} &= g^{uab} \cdot g^{-uab}(g^b)^\beta g^{\frac{T_1}{T_2}\beta}(g^a)^{-\frac{T_1 u}{T_2}} \\
&= g^{uab} \cdot g^{\frac{T_1}{T_2}(\beta - ua)} \cdot g^{b(\beta-ua)} \\
&= g^{uab}(g^{T_1} \cdot g^{bT_2})^{\frac{\beta-ua}{T_2}} \\
&= g^{uab}(g^{T_1+bT_2})^{\frac{\beta-ua}{T_2}} \\
&= h^y(g^{\sum_{v_{i,j}\in L} t_{i,j}})^r,
\end{aligned}$$

and

$$g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}} = g^{\frac{\beta-ua}{T_2}} = g^r,$$

and

$$(g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}})^{u_i} = g^{u_i r}$$
$$(g^{\frac{\beta}{T_2}}(g^a)^{-\frac{u}{T_2}})^{u_{m+i}} = g^{u_{m+i} r}$$

If $T_2 = 0 \bmod p$, then $\mathcal{B}$ aborts. If $T_2 = 0 \bmod p$ holds, then there exists L such that $\sum_{v_{i,j}\in L} t_{i,j} = \sum_{v_{i,j}\in W^*} t_{i,j}$ holds. Therefore, this probability is at most $\frac{N^2}{p}$. See Section 4.3 for details. Note that this probability does not depend on m since all u_i are not included in T_2. For Decryption query $C = (W, \sigma, K_v, C_1, C_2, C_3)$, $\mathcal{B}$ checks σ. If σ is invalid, then $\mathcal{B}$ aborts. If $K_v = K_{v^*}$ (we call this a forge event), then $\mathcal{B}$ gives a random answer to the DBDH challenger. Otherwise, if $K_v \neq K_{v^*}$, then $\mathcal{B}$ computes SK_L, where $L \models W$, using the same procedure as a KeyGen query. By using SK_L, $\mathcal{B}$ decrypts C, obtains M, and returns M to $\mathcal{A}$. For the challenge ciphertext, $\mathcal{B}$ chooses $\mu \in_R \{0,1\}$, computes $C_1^* = M_\mu \cdot Z^u$, $C_2^* = g^c$, $C_3^* = (g^c)^{(\sum_{v_{i,j} \in W^*} t'_{i,j}) + (\sum_{K_{v^*,i} \in V_{v^*}} u_{v^*,i})}$ and $\sigma^* = \mathsf{Sign}(K_{s^*}, \langle W^*, C_1^*, C_2^*, C_3^* \rangle)$, and sends $(\sigma^*, K_{s^*}, C_1^*, C_2^*, C_3^*)$ to $\mathcal{A}$. Finally, $\mathcal{A}$ outputs $\mu' \in \{0,1\}$. $\mathcal{B}$ outputs 1 if $\mu' = \mu$, or outputs 0 if $\mu' \neq \mu$. If $Z = e(g,g)^{abc}$, then (C_1^*, C_2^*, C_3^*) is a valid ciphertext associated with W^*. Therefore, $\mathcal{A}$ has the advantage ϵ. Hence, $\Pr[\mathcal{B} \to 1 | Z = e(g,g)^{abc}] \geq \frac{1}{2} + \epsilon - \Pr[\mathsf{forge} | Z = e(g,g)^{abc}]$. Otherwise, if $Z = e(g,g)^z$, $\mathcal{A}$ has no advantage in distinguishing a bit μ, since all parts of the challenge ciphertext, when $\mu = 0$ and when $\mu = 1$ have the same distributions. Hence, $\Pr[\mathcal{B} \to 0 | Z = e(g,g)^z] \geq \frac{1}{2} - \Pr[\mathsf{forge} | Z = e(g,g)^z]$. It follows that $\mathcal{B}$'s advantage in the DBDH game is $(\frac{\epsilon}{2} - \Pr[\mathsf{forge}])(1 - \frac{N^2}{p})$. Next, we prove that $\Pr[\mathsf{forge}]$ is negligible. We construct an algorithm $\mathcal{B}'$ which can win the SEU game with probability of at least $\Pr[\mathsf{forge}]$. $\mathcal{B}'$ obtains K_{v^*} from the SEU challenger, instead of executing SigKeyGen to obtain $\langle K_{s^*}, K_{v^*} \rangle$. $\mathcal{B}'$ proceeds as $\mathcal{B}$ using the SEU challenger. In the challenge phase of the CCA game, $\mathcal{B}'$ obtains σ^* from the SEU challenger. Therefore, $\mathcal{B}'$ makes at most one signature query. If the event forge occurs, namely $\mathcal{A}$ sends a decryption query $(W, \sigma, K_v, C_1, C_2, C_3)$, where $K_v = K_{v^*}$, then $\mathcal{B}'$ submits a forge signature σ to the SEU challenger and wins. Therefore, $\Pr[\mathsf{forge}]$ is negligible, since we assume that the signature scheme is SEU. $\square$