Tsinghua Science and Technology

Volume 23 | Issue 5 Article 5

2018

Revocable Hierarchical Identity-Based Broadcast Encryption

Dawei Li
the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China.

Jianwei Liu
the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China.

Zongyang Zhang
the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China.

Qianhong Wu
the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China.

Weiran Liu
the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China.

TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 03/10 pp539–549
DOI: 10.26599/TST.2018.9010023
Volume 23, Number 5, October 2018

Revocable Hierarchical Identity-Based Broadcast Encryption

Dawei Li, Jianwei Liu, Zongyang Zhang*, Qianhong Wu*, and Weiran Liu

Abstract: Hierarchical Identity-Based Broadcast Encryption (HIBBE) organizes users into a tree-like structure, and it allows users to delegate their decryption ability to subordinates and enable encryption to any subset of users while only intended users can decrypt. However, current HIBBE schemes do not support efficient revocation of private keys. Here, a new primitive called Revocable Hierarchical Identity-Based Broadcast Encryption (RHIBBE) is formalized that allows revocation of the HIBBE. Ciphertext indistinguishability is defined against the selectively Bounded Revocable Identity-Vector-Set and Chosen-Plaintext Attack (IND-sBRIVS-CPA). An IND-sBRIVS-CPA secure RHIBBE scheme is constructed with efficient revocation on prime-order bilinear groups. The unbounded version of the scheme is also shown to be secure but a little weaker than the former under the decisional n-Weak Bilinear Diffie-Hellman inversion assumption.

Key words: Revocable Hierarchical Identity-Based Broadcast Encryption (RHIBBE); revocation; provable security

1 Introduction

Hierarchical Identity-Based Broadcast Encryption (HIBBE), first proposed by Liu et al.[1] in 2014, combines the function of Hierarchical Identity-Based Encryption (HIBE) and Broadcast Encryption (BE). In a HIBBE system, users are organized into a tree-like structure, and they can delegate their private keys to lower-level users, which reduces the workload of the Private Key Generator (PKG). If senders need to encrypt the same message to a large number of recipients, they do not have to separately encrypt for each recipient. They encrypt the message only once, which reduces computation costs and saves communication bandwidth. For example, consider the scenario where a message is sent by email to Alice, Bob, and so on. One can encrypt this message using their public keys, i.e., Alice@mail.com, . . . , Bob@mail.com, and broadcast the ciphertext. Only the intended recipients can decrypt this message.

In some cases, if a user's private key is leaked or a user is cheating, their private key should be revoked. If there is no revocation mechanism in the HIBBE system, he/she has to change their identity to apply for a new private key. It takes a great deal of effort to convince all the other users of this alteration. Two kinds of mechanisms are commonly used to achieve revocation in HIBE, namely direct and indirect revocations[2]. In direct revocation, senders directly specify the revocation list and have to always confirm the private keys of revoked users are invalid when encrypting, which makes the efficiency of encryption relatively low. Indirect revocation also consists of two kinds of revocation. In the first kind of indirect revocation, a PKG keeps the revocation list and they can transmit private keys to all non-revoked users at intervals. As PKG has to compute new keys frequently for all non-revoked users, so this puts a heavy burden on the PKG. In addition, a secure channel is required to transmit private keys to each user every time. The second kind of indirect revocation was proposed by Boldyreva et al.[3] in a Revocable Identity-

• Dawei Li, Jianwei Liu, Zongyang Zhang, Qianhong Wu, and Weiran Liu are with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China. E-mail: lidawei@buaa.edu.cn; liujianwei@buaa.edu.cn; zongyangzhang84@gmail.com; qianhong.wu@buaa.edu.cn; liuweiran900217@gmail.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2017-09-22; accepted: 2017-09-29

Based Encryption (RIBE). Here, the private key of each user is divided into a secret key and an update key based on the idea of the fuzzy Identity-Based Encryption (IBE)[4], where the identity is contained in a secret key and the time is contained in an update key. With this method, the PKG just publicly publishes the update keys but does not need a secure channel. Moreover, the subset-cover revocation framework plays an important role in reducing the number of executions in updating keys from linearly to logarithmically correlated with the number of users.

Seo and Emura[5,6] developed a RIBE scheme to Revocable Hierarchical Identity-Based Encryption (RHIBE), where the users can delegate secret keys and update keys for their child users in the tree-like structure and share the burden with the PKG. However, HIBBE does not have efficient and secure mechanism for revocation. Inspired by these revocation mechanisms, a HIBBE with a new primitive that can execute revocation much more efficiently has been developed.

1.1 Contributions

A new cryptographic primitive was defined, called Revocable Hierarchical Identity-Based Broadcast Encryption (RHIBBE), to meet the functional requirements in the above scenarios. The main contributions of this work are summarized as follows:

(1) The security notion is formalized with ciphertext indistinguishability against selectively Bounded Revocable Identity-Vector-Set and Chosen-Plaintext Attack (IND-sBRIVS-CPA) for RHIBBE. In this security notion, an adversary should declare an identity vector set that he/she will attack, and they can make private key queries with some restrictions. In addition, the order of the identity vector set that an adversary queries is bounded. They still cannot distinguish which plaintext is encrypted by the selected identity vector set. This security notion already captures many powerful attacks to RHIBBE in reality.

(2) A concrete RHIBBE scheme is constructed on prime-order bilinear groups, which has an efficient performance in revocation and encryption. Inspired by HIBE, the broadcast identity vector set in RHIBBE is taken as the single identity vector in HIBE to execute the encryption with similar principles. When decrypting, the redundant identity in the identity vector set would be cancelled out to make sure the ciphertext can be decrypted by the corresponding decryption keys. In HIBE, the public parameter is related to the total hierarchy in the scheme. However, the method would bring many security problems to RHIBBE. For example, a ciphertext is encrypted by an identity vector set that contains two identity vectors, and then if an adversary exchanges one identity from an identity vector into another at the same hierarchy, it can decrypt the ciphertext successfully. Thus, the public parameter is related to the total number of users in the scheme, which means the elements in the broadcast identity vector set is in order so as to avoid such a trivial attack.

(3) The RHIBBE scheme is proven to be IND-sBRIVS-CPA secure based on the decisional weak Bilinear Diffie-Hellman Inversion (wBDHI) assumption, where the target identity vector set selected to be attacked is bounded. This attack can capture the most realistic attack types. If the broadcast set is required to be unbounded to improve the broadcast performance, the scheme is indistinguishable against selectively Revocable Identity-Vector-Set and static Chosen-Plaintext Attacks (IND-sRIVS-sCPA), which are also secure but a little weaker than the former.

This new revocation strategy is efficient and flexible. The subset-cover revocation framework is used and a secret key is separated into two parts related to identity and time. This will reduce the workload and bandwidth of the PKG and negate the need to issue secret keys to all non-revoked users every time with a highly secure key transmission channel in the normal HIBBE. The PKG shares its burden with higher-level users, who can delegate the secret keys and update keys for the corresponding lower-level users. The number of update keys, which can be broadcasted publicly, is a logarithm with the number of non-revoked users.

1.2 Related work

IBE was first proposed in 1984 by Shamir[7]. However, it was first practically constructed in 2001 based on bilinear groups by Boneh and Franklin[8], which proved to be secure in the random oracle model. Since then, many IBE schemes with different properties have been proposed[9–12]. HIBE was first introduced by Horwitz and Lynn[13], and was first achieved in the random oracle model by Gentry and Silverberg[14]. Then Boneh et al.[15] presented a more efficient HIBE with constant-size ciphertext in a selective security model. Boyen and Waters[16] proposed an anonymous HIBE. Waters[11]

proposed the dual system encryption to realize fully secure HIBE under simple assumptions. Since then, many fully secure IBE and HIBE schemes have been constructed[11, 12, 17, 18].

BE was first proposed in 1993 by Fiat and Naor[19], where a dealer can encrypt messages to a subset of the users and only the assigned users can decrypt to get the messages. Dealers only need to execute the encryption procedure once for each broadcast, which greatly reduces the workload. The fully functional Identity-Based Broad Encryption (IBBE) was constructed by Delerablée[20], which allows the identity to represent a receiver's public key and dealers only need to encrypt messages by using a set of identities as the public key. HIBBE was first proposed by Liu et al.[1] to share the burden of the PKG. However, the scheme is constructed on composite bilinear group. To improve the efficiency of HIBBE, Liu et al.[21] presented a practical chosen-ciphertext secure HIBBE scheme. Since then, several subsequent works focused on improving efficiency or security of HIBBE[22–25].

There are many direct methods to achieve revocable IBE with a third party to help users decrypt[26–30], but the third party should be disallowed to collude with the users or it must hold the shares of all users' private keys, which is difficult to achieve in reality[2].

IBE with an indirect revocation method was first proposed by Boneh and Franklin[8] in a trivial way, where a PKG had to issue new keys every time to each non-revoked user. It required secure channels between the PKG and every non-revoked user. Thus, this kind of revocation puts a heavy burden on the PKG and occupies significant bandwidth to transmit the keys to all non-revoked users securely. Another kind of indirect revocation method for IBE was put forward by Boldyreva et al.[3] that improves the key-update efficiency on the side of the trusted party from linear to logarithmic in the number of users. Based on this, Seo et al.[5, 6, 31] constructed a series of revocation scheme for HIBE. They took advantage of the subset-cover revocation framework to update keys for as few as possible nodes in a binary tree to improve the revocation efficiency. Besides, there have been many revocable HIBE schemes with different properties[32, 33]. To revoke the recipients for HIBBE, Susilo et al.[24] presented the notion of recipient-revocable IBBE scheme.

1.3 Organization

Necessary preliminaries are introduced for RHIBBE in Section 2, including the prime-order bilinear groups, decisional weak bilinear Diffie-Hellman inversion assumption, and the subset-cover revocation framework, which is the basis of this scheme. Then the RHIBBE definition is presented in Section 3, where the correctness and security notion IND-sBRIVS-CPA are described. Based on all of the above, a concrete RHIBBE scheme is constructed in Section 4, which is verified to be correct according to the correctness definition. Finally, in Section 5, the security of the RHIBBE scheme is analyzed and shown that it is IDN-sBRIVS-CPA secure.

2 Preliminaries

2.1 Prime-order bilinear groups

Let p be a large prime number, and $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups of order p. If g is a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear map, then $\mathbb{G}$ and $\mathbb{G}_T$ are bilinear groups if e satisfies all of the properties[34]:

- Bilinearity: $\forall u, v \in \mathbb{G}$ and $\forall a, b \in \mathbb{Z}_p$, $e(u^a, v^b) = e(u, v)^{ab} = e(u^b, v^a)$.
- Non-degeneracy: $e(g, g) \neq 1$.
- Computability: group operation $e(u, v)$ for $u, v \xleftarrow{R} \mathbb{G}$ can be efficiently computed.

2.2 Decisional weak bilinear diffie-hellman inversion assumption

2.2.1 n-wBDHI problem

Let $\mathbb{G}$ and $\mathbb{G}_T$ be the bilinear groups of order q, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, g and h be generators of $\mathbb{G}$, and $\alpha \in \mathbb{Z}_p$. The n-wBDHI problem is as follows. Given the tuple $(g, g', y_1 = g^{\alpha}, \ldots, y_n = g^{\alpha^n})$, an algorithm $\mathcal{A}$ computes $e(g, g')^{(\alpha^{n+1})}$[15]. It is defined that $\mathcal{A}$ has advantage ϵ in solving n-wBDHI problem if

$$\Pr[\mathcal{A}(g, g', y_1, \ldots, y_n) = e(g, g')^{(\alpha^{n+1})}] \geqslant \epsilon,$$

where the probability is over the random choice of $g, h \in \mathbb{G}, \alpha \in \mathbb{Z}_p$ and random bits used in $\mathcal{A}$.

2.2.2 Decisional n-wBDHI problem

The decisional n-wBDHI problem is as follows. Given the tuple $(g, g', y_1 = g^{\alpha}, \ldots, y_n = g^{\alpha^n}, e(g, g')^{(\alpha^{n+1})})$ or $(g, g', y_1 = g^{\alpha}, \ldots, y_n = g^{\alpha^n}, R)$, in which $R \xleftarrow{R} \mathbb{G}_T$, an algorithm $\mathcal{B}$ outputs a bit $b \in \{0, 1\}$. It is defined that $\mathcal{B}$ has advantage ϵ in solving decisional n-wBDHI problem if

$$|\Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, e(g, g')^{(\alpha^{n+1})}) = 0] - \Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, R) = 0]| \geqslant \epsilon,$$

where the probability is over the random choice of g, $h \in \mathbb{G}$, $R \in \mathbb{G}_T$, $\alpha \in \mathbb{Z}_p$, and random bits used in $\mathcal{B}$.

2.2.3 Decisional n-wBDHI assumption

The (t, ϵ, n)-decisional n-wBDHI assumption holds in $\mathbb{G}$ if there is no algorithm that has advantage of at least ϵ in solving the decisional n-wBDHI problem in polynomial time t[21].

2.3 Subset-cover revocation framework

The subset-cover revocation framework was proposed by Naor et al.[35], for which the Complete Subtree (CS) and Subset Difference (SD) are instances used in practice[6]. The revocation in this paper is mainly based on CS method. Upon input of a binary tree, BT, the current time, T, and a revocation list, RL, the algorithm outputs a set of users that has not been revoked until time T. What's more important, this set allows to update keys for the least nodes, which is logarithmic in the number of users.

Let v denote a non-leaf node, and let v_L (v_R) denote the left (right) child of v. In the binary tree BT, each user is assigned to a leaf node, and if it is revoked on time T, it will be added into the revocation list RL. The $\mathsf{KUNode}(BT, RL, T)$ function is defined as follows:

$\mathsf{KUNode}(BT, RL, T)$
  $X, Y \leftarrow \varnothing$
  $\forall (v_i, T_i) \in RL$
    if $T_i \leqslant T$ then add $\mathrm{Path}(v_i)$ to X
  $\forall x \in X$
    if $x_L \notin X$, then add x_L to Y
    if $x_R \notin X$, then add x_R to Y
  if $Y = \varnothing$, then add root to Y
  Return Y

3 Revocable Hierarchical Identity-Based Broadcast Encryption

3.1 RHIBBE

A RHIBBE scheme consists of the following seven polynomial-time algorithms: $\mathcal{SETUP}$, $\mathcal{SK}$, $\mathcal{KU}$, $\mathcal{DK}$, $\mathcal{ENC}$, $\mathcal{DEC}$, and $\mathcal{REV}$.

- $(mpk, msk) \leftarrow \mathcal{SETUP}(1^\lambda, n, \ell)$: The setup algorithm is executed by the PKG to initialize the system. Upon input, a security parameter, λ, is expressed in the unary representation, a maximum number of users $n = O(poly(\lambda))$, and a maximum hierarchical depth $\ell = O(poly(\lambda))$. Next, a master public key, mpk, and a master secret key, msk, are output. The master public key, mpk, contains the initial system state information, st_0, and an empty revocation list, RL. The PKG publishes mpk and keeps msk for itself.
- $sk_{\mathrm{ID}_k} \leftarrow \mathcal{SK}(st_{\mathrm{ID}_{k-1}}, \mathrm{ID}_k)$: The secret key generation algorithm is executed by ID_{k-1} for $k = 1, 2, \ldots, n$ to generate the secret key for its branch ID_k. The state information $st_{\mathrm{ID}_{k-1}}$ of the binary tree kept by ID_{k-1} and the identity ID_k are input and the secret key sk_{ID_k} is output.
- $ku_{\mathrm{ID}_{k-1},T} \leftarrow \mathcal{KU}(dk_{\mathrm{ID}_{k-1},T}, st_{\mathrm{ID}_{k-1}}, RL_{\mathrm{ID}_{k-1}}, T)$: The key update algorithm is executed by ID_{k-1} to generate the update key for its non-revoked branch, where the ID_0 represents the PKG. The current decryption key, $dk_{\mathrm{ID}_{k-1},T}$, which is equal to msk for $k = 1$, state information $st_{\mathrm{ID}_{k-1}}$, revocation list $RL_{\mathrm{ID}_{k-1}}$ and the time T are input, and the update key $ku_{\mathrm{ID}_{k-1},T}$ is output.
- $dk_{\mathrm{ID}_k,T} \leftarrow \mathcal{DK}(sk_{\mathrm{ID}_k}, ku_{\mathrm{ID}_{k-1},T})$: The decryption key generation algorithm is executed by a user with identity ID_k to compute its decryption key. A secret key, sk_{ID_k}, and current update key, $ku_{\mathrm{ID}_{k-1},T}$ are input, and a decryption key, $dk_{\mathrm{ID}_k,T}$, which can be used for decryption and key update is output.
- $C \leftarrow \mathcal{ENC}(M, S, T)$: This algorithm is executed by senders to encrypt message into ciphertext. A message, M, a set of receiver's identity, S, and current time, T, are input, and a ciphertext, C, is output.
- $M \leftarrow \mathcal{DEC}(C, S, dk_{\mathrm{ID}_k,T})$: The decryption algorithm is executed by a user with identity ID_k to decrypt ciphertext into message. A ciphertext, C, a set of receiver's identity, S, and a decryption key, $dk_{\mathrm{ID}_k,T}$, are input. Only if $\mathrm{ID}_k \in S$, a message, M, is output.
- $RL_{\mathrm{ID}_{k-1}} \leftarrow \mathcal{REV}(\mathrm{ID}_k, T, RL_{\mathrm{ID}_{k-1}})$: The revocation algorithm is executed by a user with identity ID_{k-1} to revoke ID_k. A revocation list, $RL_{\mathrm{ID}_{k-1}}$, kept by ID_{k-1}, an identity ID_k that needs to be revoked, and a time, T, are input, and an updated revocation list, $RL_{\mathrm{ID}_{k-1}}$ is the output.

3.2 Correctness

A RHIBBE scheme is said to satisfy the correct condition if the correctness-game in Fig. 1 returns true with overwhelming probability for $k, j = 1, \ldots, \ell$ and $T \xleftarrow{R} \mathbb{Z}_p$.

$(mpk, msk) \leftarrow \mathcal{SETUP}(1^\lambda, n, \ell)$
$sk_{\mathrm{ID}_k} \leftarrow \mathcal{SK}(st_{\mathrm{ID}_{k-1}}, \mathrm{ID}_k)$
$sk_{\mathrm{ID}'_j} \leftarrow \mathcal{SK}(st_{\mathrm{ID}'_{j-1}}, \mathrm{ID}'_j)$
$RL_{\mathrm{ID}'_{j-1}} \leftarrow \mathcal{REV}(\mathrm{ID}'_j, T, RL_{\mathrm{ID}'_{j-1}})$
$ku_{\mathrm{ID}_{k-1},T} \leftarrow \mathcal{KU}(dk_{\mathrm{ID}_{k-1},T}, st_{\mathrm{ID}_{k-1}}, RL_{\mathrm{ID}_{k-1}}, T)$
$ku_{\mathrm{ID}'_{j-1},T} \leftarrow \mathcal{KU}(dk_{\mathrm{ID}'_{j-1},T}, st_{\mathrm{ID}'_{j-1}}, RL_{\mathrm{ID}'_{j-1}}, T)$
$\boldsymbol{C} \leftarrow \mathcal{ENC}(\boldsymbol{M}, S = \{\mathrm{ID}_k, \mathrm{ID}'_j, \ldots\}, T)$
$dk_{\mathrm{ID}_k,T} \leftarrow \mathcal{DK}(sk_{\mathrm{ID}_k}, ku_{\mathrm{ID}_{k-1},T})$
$\boldsymbol{M}' \leftarrow \mathcal{DEC}(\boldsymbol{C}, S, dk_{\mathrm{ID}_k,T})$
Return $true$, if $\boldsymbol{M}' = \boldsymbol{M} \wedge \perp \leftarrow \mathcal{DK}(\mathrm{ID}'_j, ku_{\mathrm{ID}'_{j-1},T})$
Otherwise, return $false$.

Fig. 1 Correctness game of a RHIBBE scheme.

3.3 Security model

We define the security model, indistinguishability against selectively Bounded Revocable-Identity-Vector-Set and Chosen-Plaintext Attack (IND-sBRIVS-CPA) for RHIBBE. Let $\Pi = \{\mathcal{SETUP}, \mathcal{SK}, \mathcal{DK}, \mathcal{KU}, \mathcal{ENC}, \mathcal{DEC}, \mathcal{REV}\}$ be a revocable HIBBE scheme. $\mathcal{A} = \{\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2\}$ is the adversary who can capture realistic attacks on RHIBBE. There are four oracles in this system, including $\mathcal{SK}(\cdot)$, $\mathcal{DK}(\cdot,\cdot)$, $\mathcal{KU}(\cdot,\cdot)$, and $\mathcal{REV}(\cdot,\cdot)$.

- $\mathcal{SK}(\cdot)$: Upon inputting an identity ID_k, this oracle outputs a secret key sk_{ID_k} and its state information st_{ID_k}.
- $\mathcal{DK}(\cdot,\cdot)$: Upon inputting an identity ID_k and a time T, this oracle outputs a decryption key $dk_{\mathrm{ID}_k,T}$.
- $\mathcal{KU}(\cdot,\cdot)$: Upon inputting identity ID_{k-1} and a time T, this oracle outputs an update key $ku_{\mathrm{ID}_{k-1},T}$.
- $\mathcal{REV}(\cdot,\cdot)$: Upon inputting an identity ID_k and a time T, this oracle adds ID_k to a revocation list RL_{ID_k}.

Let $\mathcal{O}$ represent a set of oracles $\{\mathcal{SK}(\cdot), \mathcal{DK}(\cdot,\cdot), \mathcal{KU}(\cdot,\cdot), \mathcal{REV}(\cdot,\cdot)\}$, and let S^* represent the identity vector set that adversary selects to attack. Note that S^* is abridged by the method if an identity and its ancestor are both in the set, then the ancestor is omitted. In reality, however, the identity vectors in the set S^* that an adversary would attack cannot be infinite, because the total number of identities at all levels $\ell = O(poly(\lambda))$ is not greater than $n = O(poly(\lambda))$. Thus the number of identity vectors in S^* is bounded such that $\ell^{|S^*|} = O(poly(\lambda))$ to ensure security, which is huge enough to satisfy the broadcast function to realize the reduction in computation and bandwidth.

The security model is defined through the following experiment $\mathsf{EXP}^{\text{IND-sBRIVS-CPA}}_{\Pi,\mathcal{A}}(1^\lambda, n, \ell)$:

$\mathsf{EXP}^{\text{IND-sBRIVS-CPA}}_{\Pi,\mathcal{A}}(1^\lambda, n, \ell)$
$(S^*, T^*, state_0) \xleftarrow{R} \mathcal{A}_0$
$(mpk, msk) \leftarrow \mathcal{SETUP}(1^\lambda, n, \ell)$
$(M_0, M_1, state_1) \xleftarrow{R} \mathcal{A}_1^{\mathcal{O}}(mpk, state_0)$
$b \xleftarrow{R} \{0, 1\}$
$C \leftarrow \mathcal{ENC}(M_b, S^*, T^*)$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}}(mpk, C, state_1)$
If $b = b'$ then return 1; else return 0.

The following conditions must always hold to avoid trivial attacks:

(1) $|M_0| = |M_1|$.

(2) $\mathcal{A}$ must query to $\mathcal{KU}(\cdot,\cdot)$ and $\mathcal{REV}(\cdot,\cdot)$ in an increasing order of time, which means that adversary $\mathcal{A}$ cannot query to $\mathcal{REV}(\cdot,\cdot)$ for time T_1 earlier and then $\mathcal{KU}(\cdot,\cdot)$ for time T_2, or query to $\mathcal{KU}(\cdot,\cdot)$ for time T_1 earlier and then $\mathcal{REV}(\cdot,\cdot)$ for time T_2, where $T_1 \geqslant T_2$.

(3) $\mathcal{A}$ cannot query decryption key $dk_{\mathrm{ID}^*_{k^*},T}$ for each $\mathrm{ID}^*_{k^*} \in S^*$ or their ancestors.

(4) If $\mathcal{A}$ has query a secret key $sk_{\mathrm{ID}^*_{k^*}}$ for some $\mathrm{ID}^*_{k^*} \in S^*$ or their ancestors on time $T \leqslant T^*$, $\mathcal{A}$ has to revoke $\mathrm{ID}^*_{k^*}$ or its ancestor on time T', where $T \leqslant T' \leqslant T^*$.

The advantage that adversary, $\mathcal{A}$, has in the experiment is defined as $\mathsf{Adv}^{\text{IND-sBRIVS-CPA}}_{\Pi,\mathcal{A}}(1^\lambda) = |\Pr[\mathsf{EXP}^{\text{IND-sBRIVS-CPA}}_{\Pi,\mathcal{A}}(1^\lambda, n, \ell) = 1] - \frac{1}{2}|$

Definition 1 Let $\Pi = \{\mathcal{SETUP}, \mathcal{SK}, \mathcal{DK}, \mathcal{KU}, \mathcal{ENC}, \mathcal{DEC}, \mathcal{REV}\}$ be a RHIBBE scheme with maximum n users and maximum ℓ hierarchies. We say Π is IND-sBRIVS-CPA secure if for any Probabilistin Polynomial Time (PPT) adversary $\mathcal{A}$, the advantage $\mathsf{Adv}^{\text{IND-sBRIVS-CPA}}_{\Pi,\mathcal{A}}(1^\lambda, n, \ell)$ is negligible w.r.t security parameter λ.

4 Concrete Scheme

Several notations are introduced to simplify the description of the scheme. Each user in the tree-like structure would have a unique identity vector denoted as $\mathbf{ID}_i = (\mathrm{ID}_1, \ldots, \mathrm{ID}_i)$ and ID_{i-1} is used to represent its parent node. However, the figures i and $i+1$ are not in the relationship of digital addition, but just represent the inheritance relation. Let $pref(\mathbf{ID}_i) = \{\mathrm{ID}_1, \ldots, \mathrm{ID}_i\}$ denote a set of all ancestors of an identity vector. In addition, all users are in a binary tree whose root is their parent, and they also have their position in the hierarchical tree whose root is the PKG. For the ID_{i-1} that can normally decrypt the message that is sent to its child ID_i, the identities like ID_{i-1} in S are omitted and only the ID_i are maintained. $\overline{S}$ is used to represent the set of ancestors of all elements and themselves in

an identity vector set S. Note that I_S denotes a set of index of all elements in $\overline{S}$, while I_{ID_i} represents the index set of all ancestors of ID_i and itself. $\mathrm{Path}(\mathrm{ID}_i)$ denotes all ancestor nodes and itself of ID_i in a binary tree $BT_{\mathrm{ID}_{i-1}}$. An example is given in Figs. 2 and 3 for explanation. Thus $\mathbf{ID}_9 = (\mathrm{ID}_1, \mathrm{ID}_8, \mathrm{ID}_9)$, and $I_{\mathrm{ID}_9} = \{1, 8, 9\}$. $pref(\mathbf{ID}_9) = \{\mathrm{ID}_1, \mathrm{ID}_8, \mathrm{ID}_9\}$ Suppose a vector set is $S = \{\mathbf{ID}_4, \mathbf{ID}_5, \mathbf{ID}_9\}$, $\overline{S} = \{\mathrm{ID}_1, \mathrm{ID}_2, \mathrm{ID}_4, \mathrm{ID}_5, \mathrm{ID}_8, \mathrm{ID}_9\}$, and $I_S = \{1, 2, 4, 5, 8, 9\}$. In the binary tree BT_{ID_2}, $\mathrm{Path}(\mathrm{ID}_4) = \{\theta_1, \theta_2, \theta_4\}$.

• $(mpk, msk) \leftarrow Setup(1^\lambda, n, \ell)$: It picks a prime order bilinear group generator $\mathcal{G}$, and runs $(\mathbb{G}, p, g) \xleftarrow{R} \mathcal{G}(1^\lambda)$. It picks random $g, h, g_2, u_1, \ldots, u_n, u'$, $h' \xleftarrow{R} \mathbb{G}$ and $\alpha \xleftarrow{R} \mathbb{Z}_p$, then it publishes $mpk = \{n, \ell, g, g_1 = g^\alpha, h, g_2, u_1, \ldots, u_n, u', h', RL\}$ and keeps $msk = g_2^\alpha$ by itself.

• $sk_{\mathrm{ID}_k} \leftarrow \mathcal{SK}(st_{\mathrm{ID}_{k-1}}, \mathrm{ID}_k)$: Each user ID_{k-1} for $k = 1, 2, \ldots, n$ can act as a key generator for its son users which are assigned as leaf nodes in the binary tree, thus the state information $st_{\mathrm{ID}_{k-1}}$ includes the binary tree BT_{k-1} and the msk-shade P_θ for each node θ. When generating secret key sk_{ID_k} for $\mathbf{ID}_k = (\mathrm{ID}_1, \mathrm{ID}_2, \ldots, \mathrm{ID}_k)$, ID_{k-1} runs the algorithm. For each node $\theta \in \mathrm{Path}(\mathrm{ID}_k)$, it picks P_θ from $st_{\mathrm{ID}_{k-1}}$ or picks random $P_\theta \xleftarrow{R} \mathbb{G}$ if P_θ has not been assigned and then the algorithm picks random $r_\theta \xleftarrow{R} \mathbb{Z}_p$ to compute $sk_{\mathrm{ID}_k} = \{P_\theta(h \cdot \prod_{i \in I_{\mathrm{ID}_k}} u_i^{\mathrm{ID}_i})^{r_\theta}, g^{r_\theta}, \{u_i^{r_\theta}\}_{i \in [1,n] \setminus I_{\mathrm{ID}_k}}\}_{\theta \in \mathrm{Path}(\mathrm{ID}_k)}$.

• $ku_{0,T} \leftarrow \mathcal{KU}(msk, st_0, RL_0, T)$: It keeps msk, the state information st_0, the revocation list RL_0 and time T. For each node $\theta \in \mathsf{KUNode}(BT_0, RL_0, T)$, this algorithm recalls P_θ if it has been defined or assigns a random $P_\theta \xleftarrow{R} \mathbb{G}$ if it has not been defined. It picks random $t_\theta \xleftarrow{R} \mathbb{Z}_p$ for each node $\theta \in \mathsf{KUNode}(BT_0, RL_0, T)$, and computes $ku_{0,T} = \{P_\theta^{-1} g_2^\alpha (u'^T h')^{t_\theta}, g^{t_\theta}\}_{\theta \in \mathsf{KUNode}(BT_0, RL_0, T)}$.

• $dk_{\mathrm{ID}_k,T} \leftarrow \mathcal{DK}(sk_{\mathrm{ID}_k}, ku_{\mathrm{ID}_{k-1},T})$: For $\mathbf{ID}_k = (\mathrm{ID}_1, ID_2, \ldots, ID_k) \notin RL_{\mathrm{ID}_{k-1}}$, the secret key $sk_{\mathrm{ID}_k} = \{a_{\theta,0}, a_{\theta,1}, \{b_{\theta,j}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}}\}_{\theta \in \mathrm{Path}(\mathrm{ID}_k)}$ and the update key on time T is $ku_{\mathrm{ID}_{k-1},T} = \{a_{t,0}, a_{t,1}, a_{t,2}, \{b_{t,j}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_{k-1}}}\}_{\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)}$, where $a_{t,1} = b_{t,k} = \ldots = b_{t,n} = 1_{\mathbb{G}}$ for $k = 1$. Compute $(a_0, a_1, a_2, \{b_j\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}}) = (a_{\theta,0} a_{t,0} b_{t,k}^{\mathrm{ID}_k}, a_{\theta,1} a_{t,1}, a_{t,2}, \{b_{t,j} \cdot b_{\theta,j}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}})$, and re-randomize it with the random integer $r', t' \xleftarrow{R} \mathbb{Z}_p$ to get the decryption key $dk_{\mathrm{ID}_k,T} = (a_0(h \cdot \prod_{i \in I_{\mathrm{ID}_k}} u_i^{\mathrm{ID}_i})^{r'} (u'^T h')^{t'}, a_1 g^{r'}, a_2 g^{t'}, \{b_j u_j^{r'}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}})$.

• $ku_{\mathrm{ID}_{k-1},T} \leftarrow \mathcal{KU}(dk_{\mathrm{ID}_{k-1},T}, st_{\mathrm{ID}_{k-1}}, RL_{\mathrm{ID}_{k-1}}, T)$: This algorithm first runs the algorithm KUNode to get the nodes $\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)$, and recalls the P_θ if it has been defined, or picks a random $P_\theta \xleftarrow{R} \mathbb{G}$ otherwise. Then for $dk_{\mathrm{ID}_{k-1},T} = (a_0, a_1, a_2, \{b_j\}_{j \in [1,n] \setminus I_{\mathrm{ID}_{k-1}}})$, re-randomize it with random $r_\theta, t_\theta \xleftarrow{R} \mathbb{Z}_p$ for each $\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)$ to get the re-randomized decryption key $\{a_{\theta,0}, a_{\theta,1}, a_{\theta,2}, \{b_{\theta,j}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_{k-1}}}\}_{\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)} = \{h \cdot \prod_{i \in I_{\mathrm{ID}_{k-1}}} u_i^{\mathrm{ID}_i})^{r_\theta} (u'^T h')^{t_\theta}, a_1 g^{r_\theta}, a_2 g^{t_\theta}, \{b_j u_j^{r_\theta}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_{k-1}}}\}_{\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)}$. The update key is $ku_{\mathrm{ID}_{k-1},T} = \{P_\theta^{-1} a_{\theta,0}, a_{\theta,1}, a_{\theta,2}, \{b_{\theta,j}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_{k-1}}}\}_{\theta \in \mathsf{KUNode}(BT_{k-1}, RL_{k-1}, T)}$.

• $C \leftarrow \mathcal{ENC}(M, S, T)$: Let the receivers identity set be S, and the encryption algorithm picks random $s \xleftarrow{R} \mathbb{Z}_p$ and output the ciphertext $C = (C_0, C_1, C_2, C_3) = (M \cdot e(g_1, g_2)^s, g^s, (h \prod_{i \in I_S} u_i^{\mathrm{ID}_i})^s, (u'^T h')^s)$.

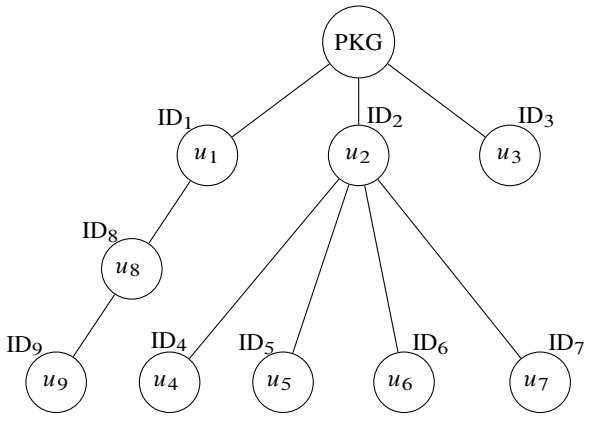


Fig. 2 The structure of a RHIBBE.

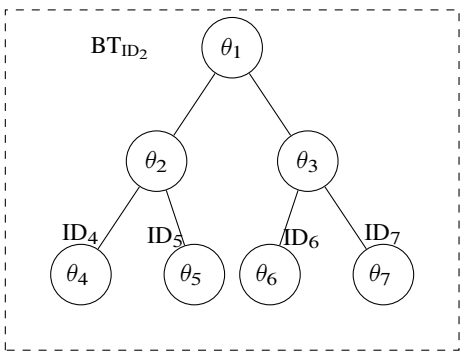


Fig. 3 The binary tree structure of the node ID_2.

• $M \leftarrow \mathcal{DEC}(C, S, dk_{\mathrm{ID}_k,T})$: Given the ciphertext $C = (C_0, C_1, C_2, C_3)$, the receiver $\mathrm{ID}_k \in S$ with decryption key $dk_{\mathrm{ID}_k,T}$ first computes $a_0' = a_0 \cdot \prod_{j \in I_S \setminus I_{\mathrm{ID}_k}} b_j^{\mathrm{ID}_j}$, then it outputs the message $M = C_0 \cdot \frac{e(a_1,C_2)\cdot e(a_2,C_3)}{e(a_0',C_1)}$.

• $RL_{\mathrm{ID}_{k-1}} \leftarrow \mathcal{REV}(\mathrm{ID}_k, T, RL_{\mathrm{ID}_{k-1}})$: This algorithm adds (ID_k, T) to $RL_{\mathrm{ID}_{k-1}}$ to update the revocation list RL.

Correctness. The decryption key would always be in the form of $dk = (g_2^{\alpha}(h \cdot \prod_{i \in I_{\mathrm{ID}_k}} u_i^{\mathrm{ID}_i})^r (u'^T h')^t, g^r, g^t, \{u_i^r\}_{i \in [1,n] \setminus I_{\mathrm{ID}_k}})$ and the ciphertext would always be in the form of $C = (C_0, C_1, C_2, C_3) = (M \cdot e(g_1, g_2)^s, g^s, (h \prod_{i \in I_S} u_i^{\mathrm{ID}_i})^s, (u'^T h')^s)$. Thus the following equations hold:

$$\frac{e(a_1, C_2) \cdot e(a_2, C_3)}{e(a_0', C_1)} = \frac{e(g^r, (h \prod_{i \in I_S} u_i^{\mathrm{ID}_i})^s) \cdot e(g^t, (u'^T h')^s)}{e(g_2^{\alpha}(h \prod_{i \in I_S} u_i^{\mathrm{ID}_i})^r (u'^T h')^t, g^s)} =$$

$$\frac{e(g, (h \prod_{i \in I_S} u_i^{\mathrm{ID}_i}))^{rs} \cdot e(g, (u'^T h'))^{ts}}{e(g_2^{\alpha}, g^s) \cdot e((h \prod_{i \in I_S} u_i^{\mathrm{ID}_i}), g)^{rs} \cdot e((u'^T h'), g)^{st}} =$$

$$\frac{1}{e(g_2^{\alpha}, g^s)} = \frac{1}{e(g_1, g_2)^s}.$$

5 Security Analysis

5.1 IND-sBRIVS-CPA secure

Theorem 1 Let $\mathcal{G}$ be a prime-order bilinear group generator and $\Pi = \{\mathcal{SETUP}, \mathcal{SK}, \mathcal{DK}, \mathcal{KU}, \mathcal{ENC}, \mathcal{DEC}, \mathcal{REV}\}$ be a concrete RHIBBE scheme constructed in Section 4 with maximum n users and maximum ℓ hierarchies. Assume that the Decision n-wBDHI assumption holds on $\mathcal{G}$. Then the proposed RHIBBE scheme Π is IND-sBRIVS-CPA secure where the order of the selected identity vector set is bounded that $\ell^{|S^*|} = O(poly(\lambda))$.

Proof In the following part of this section, we prove the RHIBBE scheme is IND-sBRIVS-CPA secure under the Decision n-wBDHI assumption.

Init. Suppose the challenger tuple for Decision n-wBDHI problem is $(g, g', y_1, \ldots, y_n, R)$. $\mathcal{B}$ and $\mathcal{A}$ should obey all of the rules in security model. Adversary, $\mathcal{A}$, first selects a target identity vector set $S^* = \{\mathbf{ID}_1^*, \ldots, \mathbf{ID}_{|S^*|}^*\}$ and the time T^* that it will attack, where $\ell^{|S^*|} = O(poly(\lambda))$. This gives S^*, T^*, together with an initial system state, $state_0$, to the simulator, $\mathcal{B}$. Let $\overline{S}^* = \{\mathrm{ID}_1^*, \ldots, \mathrm{ID}_N^*\}$ be the set of ancestors of all elements and themselves in $\mathcal{S}^*$, where there must be $N \leqslant n$, and $I^* = \{i : \mathrm{ID}_i^* \in \overline{S}^*\}$.

Setup. Simulator $\mathcal{B}$ picks random $\gamma_0, \gamma_1, \ldots, \gamma_n, \delta, c, d \xleftarrow{R} \mathbb{Z}_p$, and sets $g_1 = y_1 = g^{\alpha}$, $g_2 = y_n g^{\gamma_0} = g^{\gamma_0 + \alpha^n}$, and $h = g^{\delta} \cdot \prod_{i \in I^*} y_{n-i+1}^{\mathrm{ID}_i^*}$. Besides, it sets $u' = g_2^c$ and $h' = u'^{-T^*} g^d$. For $i \in I_{S^*}$, set $u_i = g^{\gamma_i} \cdot y_{\ell-i+1}^{-1}$. Finally, $\mathcal{B}$ derives an initial revocation list RL from $state_0$, and outputs the public parameter $mpk = (g, g_1, g_2, h, u_1, \ldots, u_n, u', h', RL)$. The master key $msk = g_2^{\alpha} = (y_n g^{\gamma_0})^{\alpha} = y_{n+1} y_1^{\gamma_0}$, which can not be computed by $\mathcal{B}$ because the y_{n+1} part is not known to $\mathcal{B}$.

Phase 1. Adversary $\mathcal{A}$ makes $\mathcal{SK}(\cdot)$, $\mathcal{DK}(\cdot,\cdot)$, $\mathcal{KU}(\cdot,\cdot)$, $\mathcal{REV}(\cdot,\cdot)$ queries to simulator $\mathcal{B}$ by the rule that is the same as it in security model. $\mathcal{B}$ responds to $\mathcal{A}$ by the following methods.

$\mathcal{DK}(\cdot,\cdot)$ query: Adversary $\mathcal{A}$ can not query for dk_{ID^*,T^*} in time T^* by the rule, in which $\mathrm{ID}^* \in \overline{S}^*$.

(1) If the queried identity ID_k and its ancestors are not in $\overline{S}^*$, $\mathcal{B}$ picks random $r, t \xleftarrow{R} \mathbb{Z}_p$ and computes the decryption key $dk_{\mathrm{ID}_k,T}$. $\mathcal{B}$ sorts the $\overline{S^*}$ by the identity index, which can be denoted as $\{\ell_1^*, \ldots, \ell_m^*\}$ and extends the set by adding $n-m$ zeroes to the right. $I_{\ell_k^*}$ denotes the set $\{\ell_1^*, \ldots, \ell_k^*\}$. For $dk_{\mathrm{ID}_k,T} = (a_0, a_1, a_2, \{b_j\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}}) = (g_2^{\alpha} \cdot (h \cdot \prod_{j \in I_{\mathrm{ID}_k}} u_j^{\mathrm{ID}_j})^{r'} \cdot (u'^T h')^t, g^{r'}, g^t, \{u_j^{r'}\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}})$, where $r' = \frac{\alpha^k}{\mathrm{ID}_k - \ell_k^*} + r$. All parts of the $dk_{\mathrm{ID}_k,T}$ can be computed by $\mathcal{B}$ using the known parameters. For example, $a_0 = y_{n+1} \cdot y_1^{\gamma_0} \cdot (g^{\delta} \cdot \prod_{j \in I_{\mathrm{ID}_k}} g^{\mathrm{ID}_j \gamma_j} \cdot \prod_{j \in I_{\mathrm{ID}_{k-1}}} y_{n-j+1}^{\ell_j^* - \mathrm{ID}_j} \cdot y_{n-k+1}^{\ell_k^* - \mathrm{ID}_k} \cdot \prod_{j \in [1,n] \setminus I_{\mathrm{ID}_k}} y_{n-j+1}^{\ell_j^*})^{r'} \cdot (u'^T h')^t = y_1^{\gamma_0} \cdot (g^{\delta} \cdot \prod_{j \in I_{\mathrm{ID}_k}} g^{\mathrm{ID}_j \gamma_j} \cdot \prod_{j \in I_{\mathrm{ID}_{k-1}}} y_{n-j+1}^{\ell_j^* - \mathrm{ID}_j} \cdot \prod_{j \in [1,n] \setminus I_{\mathrm{ID}_k}} y_{n-j+1}^{\ell_j^*})^{r'} \cdot y_{n-k+1}^{r(\ell_k^* - \mathrm{ID}_k)} \cdot (u'^T h')^t$.

(2) If the queried time $T \neq T^*$, $\mathcal{B}$ picks random $r, t \xleftarrow{R} \mathbb{Z}_p$ outputs the decryption key $dk_{\mathrm{ID}_k,T} = ((h \cdot \prod_{j \in I_{\mathrm{ID}_k}} u_j^{\mathrm{ID}_i})^r \cdot g_1^{\frac{-d}{c(T-T^*)}} \cdot (u'^{T-T^*} g^d)^t, g^r, g_1^{\frac{-1}{c(T-T^*)}} \cdot g^t, \{u_j^r\}_{j \in [1,n] \setminus I_{\mathrm{ID}_k}})$, in which we can regard $t = t' + \frac{-\alpha}{c(T-T^*)}$ such that the $dk_{\mathrm{ID}_k,T}$ distributes as the real decryption key.

(3) If the query meets both above conditions, either of them can be applied.

$\mathcal{REV}(\cdot,\cdot)$ query: $\mathcal{B}$ just runs the normal revocation

algorithm.

The adversary can be divided into $L+1$ types including the type-* adversary and the type-i adversaries for $i \in [1, L]$. The adversary is defined as type-* if all of the $\mathcal{SK}$ queried identities $\mathrm{ID} \notin \overline{S^*}$. The type-i adversary means that $\{\mathrm{ID}^*_{i,1}, \ldots, \mathrm{ID}^*_{i,|S^*|}\}$ are the oldest ancestor identities $\mathcal{A}$ queries to $\mathcal{SK}$ for each element in S^* before time T^*, where $\mathrm{ID}^*_{i,j}$ is an identity element in $\mathbf{ID}^*_{i,j} \in S^*$. If there are some identities in $\{\mathrm{ID}^*_{i,1}, \ldots, \mathrm{ID}^*_{i,|S^*|}\}$ that have inheritance relation such that $\mathrm{ID}^*_{i,j} \in pref(\mathrm{ID}^*_{i,j'})$, just keep the oldest one. The $|S^*|$ identity vectors are treated separately and just use ID^*_i to denote the oldest ancestor identities $\mathcal{A}$ queries each element in S^*. Thus, the number of adversary type would be $L+1 \leqslant \ell^{|S^*|}+1 = O(poly(\lambda))+1$.

If $\mathcal{B}$'s guess is not correct, it will output a random bit. If $\mathcal{B}$'s guess is correct that the adversary is type-i, $\mathcal{B}$ can generate the secret key and update key for the children of $\mathrm{ID}^* \in pref(\mathbf{ID}^*_j)$ whose level is not greater than i by the following method. While for the others $\mathcal{B}$ can generate the keys with the normal algorithm.

$\mathcal{SK}(\cdot)$ query: Suppose that $\mathcal{A}$ queries for the identity ID_j which is a child of ID^*_{j-1}, where $j \in [1, i]$.

(1) If $j<i$, $\mathcal{B}$ generates the secret key by computing $\{(P_\theta a_0, a_1, \{b_k\}_{k\in[1,n]\setminus I_{\mathrm{ID}_j}})\}_{\theta\in\mathrm{Path}(\mathrm{ID}_j)} = \{P_\theta(g_2^\alpha \cdot (h \cdot \prod_{k\in I_{\mathrm{ID}_j}} u_k^{\mathrm{ID}_k})^{r_\theta}, g^{r_\theta}, \{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_j}})\}_{\theta\in\mathrm{Path}(\mathrm{ID}_j)}$, and re-randomizing it, in which all elements can be computed by the known parameters just as it shown in $\mathcal{DK}(\cdot,\cdot)$ query, and P_θ can be recalled from the storage if it has been stored before, otherwise it should be randomly picked by $\mathcal{B}$.

(2) If $j = i$ and $\mathrm{ID}_i \neq \mathrm{ID}^*_i$, $\mathcal{B}$ computes $\{P_\theta(g_2^\alpha \cdot (h \cdot \prod_{k\in I_{\mathrm{ID}_j}} u_k^{\mathrm{ID}_k})^{r_\theta}, g^{r_\theta}, \{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_j}})\}$ for $\theta \in \mathrm{Path}(\mathrm{ID}_j)\setminus\mathrm{Path}(\mathrm{ID}^*_i)$ and computes $\{P_\theta(h\cdot\prod_{k\in I_{\mathrm{ID}_j}} u_k^{\mathrm{ID}_k})^{r_\theta}, g^{r_\theta}, \{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_j}})\}$ for $\theta \in \mathrm{Path}(\mathrm{ID}_j)\cap\mathrm{Path}(\mathrm{ID}^*_i)$, and re-randomizes it after all.

(3) If $j = i$ and $\mathrm{ID}_i = \mathrm{ID}^*_i$, $\mathcal{B}$ computes $\{P_\theta(h\cdot\prod_{k\in I_{\mathrm{ID}_j}} u_k^{\mathrm{ID}_k})^{r_\theta}, g^{r_\theta}, \{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_j}})\}_{\theta\in\mathrm{Path}(\mathrm{ID}^*_j)}$ and returns it to $\mathcal{A}$.

$\mathcal{KU}(\cdot,\cdot)$ query: Suppose that $\mathcal{A}$ queries for (ID^*_{j-1}, T), in which ID^*_{j-1} has not been revoked at time T, and $j \in [1, i]$.

(1) If $j < i$, $\mathcal{B}$ computes the update key $ku_{\mathrm{ID}^*_{j-1},T} = \{(P_\theta^{-1}(u'^T h')^{t_\theta}, g^{t_\theta})\}_\theta$ for $j = 1$ and computes $(P_\theta^{-1}\cdot(h\cdot\prod_{j\in I_{\mathrm{ID}_{j-1}}} u_j^{\mathrm{ID}_j})^{r_\theta}\cdot(u'^T h')^{t_\theta}, g^{r_\theta}, g^{t_\theta}, \{u_j^{r_\theta}\}_{j\in[1,n]\setminus I_{\mathrm{ID}_{j-1}}})$ for $1 < j < i$.

(2) If $j = i$ and $T \neq T^*$, for $j = 1$ $\mathcal{B}$ computes $ku_{\mathrm{ID}^*_{j-1},T} = \{(P_\theta^{-1} g_1^{\frac{-d}{c(T-T^*)}}\cdot(u'^{T-T^*} g^d)^{t_\theta}, g_1^{\frac{-1}{c(T-T^*)}}\cdot g^{t_\theta})\}_{\theta\in K(0)}\cup\{(P_\theta^{-1}(u'^T h')^{t_\theta}, g^{t_\theta})\}_{\theta\in K'(0)}$. For $j > 1$, $\mathcal{B}$ first queries for $\mathcal{DK}(\cdot,\cdot)$ oracle to generate the decryption of ID_{j-1} on time T, $dk_{\mathrm{ID}_{j-1},T} = (a_0, a_1, a_2, \{b_k\}_{k\in[1,n]\setminus I_{\mathrm{ID}_{j-1}}})$, and then it computes the update key

$$
\begin{aligned}
&ku_{\mathrm{ID}^*_{j-1},T} = \\
&\quad\{(P_\theta^{-1}a_0, a_1, a_2, \{b_k\}_{k\in[1,n]\setminus I_{\mathrm{ID}_{j-1}}})\}_{\theta\in K(i-1)}\cup \\
&\quad\{(P_\theta^{-1}(h\cdot\prod_{k\in I_{\mathrm{ID}_{j-1}}} u_k^{\mathrm{ID}_k})^{r_\theta}(u'^T h')^{t_\theta}, g^{r_\theta}, g^{t_\theta}, \\
&\quad\{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_{j-1}}})\}_{\theta\in K'(i-1)},
\end{aligned}
$$

where $K(i)=\mathsf{KUNode}(BT_{\mathrm{ID}^*_i}, RL_{\mathrm{ID}^*_i}, T)\cap\mathrm{Path}(\mathrm{ID}^*_j)$, and the other part of it $K'(i) = \mathsf{KUNode}(BT_{\mathrm{ID}^*_i}, RL_{\mathrm{ID}^*_i}, T)\setminus\mathrm{Path}(\mathrm{ID}^*_j)$.

(3) If $j = i$ and $T = T^*$, $\mathcal{B}$ computes

$$
\begin{aligned}
&\{(P_\theta^{-1}(h\cdot\prod_{k\in I_{\mathrm{ID}_{j-1}}} u_k^{\mathrm{ID}_k})^{r_\theta}(u'^T h')^{t_\theta}, g^{r_\theta}, g^{t_\theta}, \\
&\{u_k^{r_\theta}\}_{k\in[1,n]\setminus I_{\mathrm{ID}_{j-1}}})\}_{\theta\in\mathsf{KUNode}(BT_{\mathrm{ID}^*_{i-1}}, RL_{\mathrm{ID}^*_{i-1}}, T)}.
\end{aligned}
$$

If $\mathcal{B}$'s guess is right that the adversary is type-*, it will generate the decryption and update keys normally, except that for the children of $pref(\mathbf{ID}^*_{i-1})$. In that case, $\mathcal{B}$ will generate their keys like the type-i adversary does.

Challenge. Adversary $\mathcal{A}$ outputs two messages M_0 and M_1 with equal length, and the state information $state_1$. $\mathcal{B}$ randomly picks $b \xleftarrow{R} \{0,1\}$, and computes the ciphertext

$$
\begin{aligned}
C = (C_0, C_1, C_2, C_3) = (&M\cdot R\cdot e(y_1, g'^{\gamma_0}), g', \\
&(g'^{\delta+\sum_{i\in I^*}\mathrm{ID}^*_i\cdot\gamma_i}, u'^{T-T^*} g'^d),
\end{aligned}
$$

where R and g' are delegated from the Decision n-wBDHI challenge tuple. Because g' is an element in $\mathbb{G}$, we can say $g' = g^s$ in which s is not known. Thus the ciphertext C is correct distributed as it in $(C_0, C_1, C_2, C_3) \leftarrow \mathcal{ENC}(M_b, S^*, T^*)$.

If $R = e(g, g')^{(\alpha^{n+1})}$, C is the valid ciphertext encrypted from M_b, while if $R \xleftarrow{R} \mathbb{G}_T$, the C_0 part of the ciphertext is randomly picked. After all, $\mathcal{B}$ returns C to $\mathcal{A}$.

Phase 2. Adversary $\mathcal{A}$ queries just like it does in Phase 1.

Guess. Finally, $\mathcal{A}$ outputs the bit $b' \in \{0, 1\}$.

Analysis. If $R = e(g, g')^{(\alpha^{n+1})}$, $\mathcal{A}$ can attack the RHIBBE scheme successfully, thus the advantage that $\mathcal{A}$ wins the game is $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{RHIBBE}} = |\Pr[b = b'] - \frac{1}{2}| \geqslant \epsilon$. If $R \xleftarrow{R} \mathbb{G}_T$, the advantage that $\mathcal{A}$ succeeds is $Pr[b = b'] = \frac{1}{2}$. The simulator $\mathcal{B}$ would guess out the adversary's type with the probability at least $\frac{1}{L+1}$, where $L+1 \leqslant \ell^{|S^*|}+1 = O(poly(\lambda))+1$. Then $\mathcal{B}$ can break Decision wBDHI assumption with the advantage $\mathsf{Adv}_{\mathcal{B}} = |\Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, e(g, h')^{\alpha^{n+1}}) = 0] - \Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, R) = 0]| \geqslant \frac{\epsilon}{L+1}$.

5.2 IND-sRIVS-sCPA secure

Theorem 2 Let $\mathcal{G}$ be a prime-order bilinear group generator and $\Pi = \{\mathcal{SETUP}, \mathcal{SK}, \mathcal{DK}, \mathcal{KU}, \mathcal{ENC}, \mathcal{DEC}, \mathcal{REV}\}$ be a concrete RHIBBE scheme constructed in Section 4 with maximum n users and maximum ℓ hierarchies. Assume that the Decision n-wBDHI assumption holds on $\mathcal{G}$. Then the proposed RHIBBE scheme Π is IND-sRIVS-sCPA secure.

Proof The security proof of Theorem 2 is the same as it of Theorem 1, except the following aspects:

(1) In the **Init** stage, the order of the selected identity vector set is not bounded.

(2) At the beginning of **Phase 1** and **Phase 2** stages, adversary $\mathcal{A}$ should make $\mathcal{SK}(\cdot)$ and $\mathcal{KU}(\cdot, \cdot)$ queries in a static way, which means $\mathcal{A}$ outputs a list $\mathcal{L}$ that contains all elements that it would query in the game, and gives it to $\mathcal{B}$.

(3) To reply the $\mathcal{SK}(\cdot)$ and $\mathcal{KU}(\cdot, \cdot)$ queries in **Phase 1** and **Phase 2** stages, $\mathcal{B}$ needs not guess the adversary's type, because $\mathcal{B}$ can judge the adversary's type by analyzing the queried list $\mathcal{L}$.

Analysis. If $R = e(g, g')^{(\alpha^{n+1})}$, $\mathcal{A}$ can attack the RHIBBE scheme successfully, thus the advantage that $\mathcal{A}$ wins the game is $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{RHIBBE}} = |\Pr[b = b'] - \frac{1}{2}| \geqslant \epsilon$. If $R \xleftarrow{R} \mathbb{G}_T$, the advantage that $\mathcal{A}$ succeeds is $\Pr[b = b'] = \frac{1}{2}$. Then $\mathcal{B}$ can break Decision wBDHI assumption with the advantage $\mathsf{Adv}_{\mathcal{B}} = |\Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, e(g, h')^{\alpha^{n+1}}) = 0] - \Pr[\mathcal{B}(g, g', y_1, \ldots, y_n, R) = 0]| \geqslant \epsilon$.

6 Conclusion

A new cryptographic primitive was formalized and is referred to as a revocable hierarchical identity-based broadcast encryption. This method allows an identity in a broadcast encryption scheme to be revoked efficiently. Messages can be broadcast to a set of receivers with the encryption being executed only once, which makes the encryption and transmission much more manageable. Then, the IND-sBRIVS-CPA security is defined for this primitive. A concrete scheme of RHIBBE is proposed on the prime-order bilinear groups that have efficient performance for revocation. Finally the RHIBBE scheme was shown to be IND-sBRIVS-CPA secure under the decisional n-wBDHI assumption. The RHIBBE scheme was shown to be IND-sRIVS-sCPA secure when the restriction on broadcast set is removed to improve performance. This scheme provides a new method to achieve efficient revocation for hierarchical identity-based broadcast encryption and can be deployed in practice.

Acknowledgment

This work was supported by the National Key Research and Development Program of China (No. 2017YFB0802502), the National Natural Science Foundation of China (Nos. 61672083, 61370190, 61532021, 61472429, 61402029, 61702028, and 61571024), the National Cryptography Development Fund (No. MMJJ20170106), the Planning Fund Project of Ministry of Education (No. 12YJAZH136), the Beijing Natural Science Foundation (No. 4132056), and the Fund of the State Key Laboratory of Information Security, the Institute of Information Engineering, and the Chinese Academy of Sciences (No. 2017-MS-02).

References

[1] W. Liu, J. Liu, Q. Wu, and B. Qin, Hierarchical identity-based broadcast encryption, in *Information Security and Privacy—19th Australasian Conference*, Wollongong, Australia, 2014, pp. 242–257.

[2] H. Cui, R. H. Deng, Y. Li, and B. Qin, Server-aided revocable attribute-based encryption, in *21st European Symposium on Research in Computer Security*, Heraklion, Greece, 2016, pp. 570–587.

[3] A. Boldyreva, V. Goyal, and V. Kumar, Identity-based encryption with efficient revocation, in *Proceedings of the 2008 ACM Conference on Computer and Communications Security*, Alexandria, VA, USA, pp. 417–426.

[4] A. Sahai and B. Waters, Fuzzy identity-based encryption, in *24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Aarhus, Denmark, 2005, pp. 457–473.

[5] J. H. Seo and K. Emura, Efficient delegation of key generation and revocation functionalities in identity-based

encryption, in *the Cryptographers' Track at the RSA Conference 2013*, San Francisco, CA, USA, 2013, pp. 343–358.

[6] J. H. Seo and K. Emura, Revocable hierarchical identity-based encryption via history-free approach, *Theor. Comput. Sci.*, vol. 615, pp. 45–60, 2016.

[7] A. Shamir, Identity-based cryptosystems and signature schemes, in *Proceedings of CRYPTO'84*, Santa Barbara, CA, USA, 1984, pp. 47–53.

[8] D. Boneh and M. K. Franklin, Identity-based encryption from the weil pairing, in *Annual International Cryptology Conference*, Heidelberg, Germany, 2011, pp. 213–229.

[9] D. Boneh and X. Boyen, Efficient selective-id secure identity-based encryption without random oracles, in *International Conference on the Theory and Applications of Cryptographic Techniques*, Interlaken, Switzerland, 2004, pp. 223–238.

[10] C. Gentry, Practical identity-based encryption without random oracles, in *25th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Petersburg, Russia, 2006, pp. 445–464.

[11] B. Waters, Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions, in *29th Annual International Cryptology Conference*, Santa Barbara, CA, USA, 2009, pp. 619–636.

[12] A. B. Lewko and B. Waters, Unbounded HIBE and attribute-based encryption, in *30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Tallinn, Estonia, 2011, pp. 547–567.

[13] J. Horwitz and B. Lynn, Toward hierarchical identity-based encryption, in *International Conference on the Theory and Applications of Cryptographic Techniques*, Amsterdam, Netherlands, 2002, pp. 466–481.

[14] C. Gentry and A. Silverberg, Hierarchical id-based cryptography, in *8th International Conference on the Theory and Application of Cryptology and Information Security*, Queenstown, New Zealand, 2002, pp. 548–566.

[15] D. Boneh, X. Boyen, and E. Goh, Hierarchical identity based encryption with constant size ciphertext, in *24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Aarhus, Denmark, 2005, pp. 440–456.

[16] X. Boyen and B. Waters, Anonymous hierarchical identity-based encryption (without random oracles), in *26th Annual International Cryptology Conference*, Santa Barbara, CA, USA, 2006, pp. 290–307.

[17] K. Lee, J. H. Park, and D. H. Lee, Anonymous HIBE with short ciphertexts: Full security in prime order groups, *Des. Codes Cryptography*, vol. 74, no. 2, pp. 395–425, 2015.

[18] C. Gentry and S. Halevi, Hierarchical identity based encryption with polynomially many levels, in *6th Theory of Cryptography Conference*, San Francisco, CA, USA, 2009, pp. 437–456.

[19] A. Fiat and M. Naor, Broadcast encryption, in *13th Annual International Cryptology Conference*, Santa Barbara, CA, USA, 1993, pp. 480–491.

[20] C. Delerablée, Identity-based broadcast encryption with constant size ciphertexts and private keys, in *13th International Conference on the Theory and Application of Cryptology and Information Security*, Kerchirg, Malaysia, 2007, pp. 200–215.

[21] W. Liu, J. Liu, Q. Wu, B. Qin, and Y. Li, Practical chosen-ciphertext secure hierarchical identity-based broadcast encryption, *Int. J. Inf. Sec.*, vol. 15, no. 1, pp. 35–50, 2016.

[22] M. H. Ameri, J. Mohajeri, and M. Salmasizadeh, Efficient and provable secure anonymous Hierarchical Identity-Based Broadcast Encryption (HIBBE) scheme without random oracle, *IACR Cryptology ePrint Archive*, vol. 2016, p. 780, 2016.

[23] K. He, J. Weng, M. H. Au, Y. Mao, and R. H. Deng, Generic anonymous identity-based broadcast encryption with chosen-ciphertext security, in *Information Security and Privacy-21st Australasian Conference*, Melbourne, Australia, 2016, pp. 207–222.

[24] W. Susilo, R. Chen, F. Guo, G. Yang, Y. Mu, and Y. Chow, Recipient revocable identity-based broadcast encryption: How to revoke some recipients in IBBE without knowledge of the plaintext, in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, Xi'an, China, 2016, pp. 201–210.

[25] P. Xu, J. Li, W. Wang, and H. Jin, Anonymous identity-based broadcast encryption with constant decryption complexity and strong security, in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, Xi'an, China, 2016, pp. 223–233.

[26] D. Boneh, X. Ding, G. Tsudik, and C. Wong, A method for fast revocation of public key certificates and security capabilities, in *10th USENIX Security Symposium*, Washington, DC, USA, 2001.

[27] J. Baek and Y. Zheng, Identity-based threshold decryption, in *7th International Workshop on Theory and Practice in Public Key Cryptography*, Singapore, 2004, pp. 262–276.

[28] B. Libert and J. Quisquater, Efficient revocation and threshold pairing based cryptosystems, in *Proceedings of the Twenty-Second ACM Symposium on Principles of Distributed Computing*, Boston, MA, USA, 2003, pp. 163–171.

[29] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, Identity-based encryption with outsourced revocation in cloud computing, *IEEE Trans. Computers*, vol. 64, no. 2, pp. 425–437, 2015.

[30] B. Qin, R. H. Deng, Y. Li, and S. Liu, Server-aided revocable identity-based encryption, in *20th European Symposium on Research in Computer Security*, Vienna, Austria, 2015, pp. 286–304.

[31] K. Emura, J. H. Seo, and T. Youn, Semi-generic transformation of revocable hierarchical identity-based encryption and its DBDH instantiation, *IEICE Transactions*, vol. 99-A, no. 1, pp. 83–91, 2016.

[32] G. Ryu, K. Lee, S. Park, and D. H. Lee, Unbounded hierarchical identity-based encryption with efficient revocation, in *Information Security Applications-16th International Workshop*, Jeju Island, Korea, 2015, pp. 122–133.

[33] S. Park, D. H. Lee, and K. Lee, Revocable hierarchical identity-based encryption from multilinear maps, arXiv: 1610.07948, 2016.

[34] D. Li, J. Liu, and W. Liu, Secure and anonymous data transmission system for cluster organised space information network, in *IEEE International Conference on Smart Cloud*, New York, NY, USA, 2016, pp. 228–233.

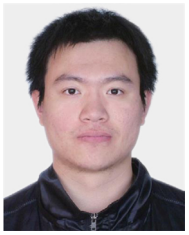
[35] D. Naor, M. Naor, and J. Lotspiech, Revocation and tracing schemes for stateless receivers, in *21st Annual International Cryptology Conference*, Santo Barbara, CA, USA, 2001, pp. 41–62.

Dawei Li received the BS degree from Beihang University, Beijing, China, in 2015. He is currently working toward the PhD degree in electronic and information engineering at Beihang University, Beijing, China. His research interests include applied cryptography and blockchain.

Jianwei Liu received the BS and MS degrees from Shandong University, China, in 1985 and 1988, respectively. He received the PhD degree in communication and electronic system from Xidian University, China, in 1998. He is now a professor of electronic and information engineering at Beihang University, Beijing, China. His current research interests include wireless communication network, cryptography, and information security.

Zongyang Zhang received the BS degree from Hohai University, Jiangsu, China, in 2005. He received the MS and PhD degrees from Shanghai Jiao Tong University in 2008 and 2012, respectively. He is now a lecturer of electronic and information engineering at Beihang University, Beijing, China. His current research interests include cryptography, information, and network security and blockchain.

Qianhong Wu received the PhD degree in cryptography from Xidian University, China, in 2004. Since then, he has been with Wollongong University (Australia) as an associate research fellow, with Wuhan University (China) as an associate professor, with Universitat Rovira i Virgili (Catalonia) as a research director, and now with Beihang University (China) as a professor. He is a member of the International Association for Cryptologic Research (IACR), Association for Computing Machinery (ACM), and Institute of Electrical and Electronics Engineers (IEEE). His current research interests include cryptography, data security and privacy, and information theory.



Weiran Liu received the BS degree from Beihang University, Beijing, China, in 2012. He is currently working toward the PhD degree in electronic and information engineering at Beihang University, Beijing, China. His research interests include applied cryptography and cloud security.