1-1-2016

Anonymous identity-based broadcast encryption with revocation for file sharing

Jianchang Lai
University of Wollongong, jl967@uowmail.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Fuchun Guo
University of Wollongong, fuchun@uow.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

Rongmao Chen
University of Wollongong, rc517@uowmail.edu.au

Anonymous identity-based broadcast encryption with revocation for file sharing

Abstract

Traditionally, a ciphertext from an identity-based broadcast encryption can be distributed to a group of receivers whose identities are included in the ciphertext. Once the ciphertext has been created, it is not possible to remove any intended receivers from it without conducting decryption. In this paper, we consider an interesting question: how to remove target designated receivers from a ciphertext generated by an anonymous identity-based broadcast encryption? The solution to this question is found applicable to file sharing with revocation. In this work, we found an affirmative answer to this question. We construct an anonymous identity-based broadcast encryption, which offers the user revocation of ciphertext and the revocation process does not reveal any information of the plaintext and receiver identity. In our proposed scheme, the group of receiver identities are anonymous and only known by the encryptor. We prove that our scheme is semantically secure in the random oracle model.

Keywords

identity, broadcast, encryption, revocation, file, anonymous, sharing

Disciplines

Engineering | Science and Technology Studies

Publication Details

Lai, J., Mu, Y., Guo, F., Susilo, W. & Chen, R. (2016). Anonymous identity-based broadcast encryption with revocation for file sharing. Lecture Notes in Computer Science, 9723 223-239.

Anonymous Identity-Based Broadcast Encryption with Revocation for File Sharing

Jianchang Lai, Yi Mu, Fuchun Guo, Willy Susilo, and Rongmao Chen

Centre for Computer and Information Security Research,
School of Computing and Information Technology
University of Wollongong, Wollongong, Australia
{jl967,ymu,fuchun,wsusilo,rc517}@uow.edu.au

Abstract. Traditionally, a ciphertext from an identity-based broadcast encryption can be distributed to a group of receivers whose identities are included in the ciphertext. Once the ciphertext has been created, it is not possible to remove any intended receivers from it without conducting decryption. In this paper, we consider an interesting question: *how to remove target designated receivers from a ciphertext generated by an anonymous identity-based broadcast encryption?* The solution to this question is found applicable to file sharing with revocation. In this work, we found an affirmative answer to this question. We construct an anonymous identity-based broadcast encryption, which offers the user revocation of ciphertext and the revocation process does not reveal any information of the plaintext and receiver identity. In our proposed scheme, the group of receiver identities are anonymous and only known by the encryptor. We prove that our scheme is semantically secure in the random oracle model.

Keywords: Identity-Based Encryption, Revocation, Anonymity

1 Introduction

In a broadcast encryption system, a file can be encrypted for a group of receivers such that any receiver in the group can decrypt the ciphertext using its respective private key. The users outside the group learn nothing about the encrypted file even if they collude. Broadcast encryption is a useful way for data sharing, where receivers can obtain the broadcast (or shared) data with their private keys. However, directly applying a broadcast encryption for data sharing in database systems or cloud computing might suffer from some drawbacks. For example, it cannot preserve the receiver privacy, since all receiver identities must be attached with the ciphertext. Therefore, if applying an identity-based broadcast encryption scheme to file sharing, an anonymous broadcast encryption would be more desirable.

We consider an application scenario using an anonymous identity-based broadcast encryption, where the file sharing system for a company is supplied by a cloud service. Without losing generality, let's assume that the system involves a cloud server, file owner, and a group of users. The file owner first encrypts a file for a selected group S, and then stores the encrypted file in the cloud for sharing. When some users R leave the company, the server must revoke them from accessing all files. If the revoked users are in S, they cannot decrypt the ciphertext after the server conducts revocation. Mostly important, it requires the cloud server to be able to revoke users from a ciphertext without knowing the encrypted file and the identities of receivers.

A trivial solution to the scenario is to adopt the "decrypt then re-encrypt" approach. It requires the server to have the ability to decrypt the ciphertext. When some identities should be revoked, the server first decrypts the ciphertext and removes them from the original authorized user set. It then re-encrypts the file using the new authorized user set. However, in this trivial solution, the cloud server is able to learn the content and the identity of authorized users who can access the file. Alternatively, the cloud server without decryption right can encrypt the ciphertext by using the broadcast encryption scheme (e.g. [21]) where anyone can decrypt the ciphertext except the revoked users. This method guarantees that the cloud server cannot get any useful information about the content and the authorized users' identities from the original ciphertext. The limitation is that this method could cause a collusion attack. For example, let ID_i be the identity of User i; if $ID_1 \notin S \cup R$, $ID_2 \in S \cap R$, ID_1 can use its private key to help ID_2 recover the original ciphertext, then ID_2 uses its private key to decrypt the original ciphertext.

Our Contributions. We notice that there is *no* ideal trivial solution to the aforementioned problem. In this work, we provide a solution to the stated problem earlier and show how to revoke users' identities from the ciphertext without the knowledge of the plaintext and the knowledge of the receivers. We propose a new cryptographic notion called *anonymous identity-based broadcast encryption with revocation* (AIBBER) to realize this. Our novel solution allows the cloud server to revoke users' identities without decryption and achieves full anonymity where only the sender knows the receivers' identities. We present two security models to meet the requirements of the proposed notion and show that our construction is secure under the attacks in the proposed model. In our setting, both the system public key and user private key are constant. The computation in revocation phase is small, more precisely $O(t)$, where t is the number of revoked identities.

1.1 Related Work

Anonymous Broadcast Encryption. Since Fiat and Naor [15] formally introduced broadcast encryption, subsequent works [9, 10, 3, 8, 16, 25, 6] have proposed broadcast encryption systems with different properties. They mainly focused on reducing public key sizes, private key sizes, ciphertext sizes and computational costs for encryption and decryption. The notion of identity-based broadcast encryption was introduced by Sakai and Furukawa [26], and Delerablée's work [8] achieves constant size ciphertext and private keys. In these schemes, the receiver identities must be attached with the ciphertext, which exposes the privacy of the receivers.

The first work addressing the anonymity in broadcast encryption appeared in [1]. The authors presented the notion of private broadcast encryption to protect the identities of the receivers and gave a generic construction from any key indistinguishable CCA scheme, which achieves receiver anonymity and CCA security. The security in [1] depends on a strongly secure one-time signature. Boneh, Sahai and Waters [4] extended this notion to construct private linear broadcast encryption and proposed a fully collusion resistant tracing traitors scheme with sublinear size ciphertexts and constant size private keys. However, the receivers cannot be arbitrary sets of users. Subsequently, many anonymous ID-based broadcast encryption schemes were proposed [18, 22, 14, 28, 12].

Libert, Paterson and Quaglia [22] examined the security of the number-theoretic construction in [1] and suggested the proof techniques without the random oracle. The authors proposed an anonymous broadcast encryption scheme that achieves adaptive security without random oracles. The ciphertext in their schemes are linear of the number of receivers and the security depends on a one-time signature. Later, Fazio and Perera [14] formalized the notion of outsider-anonymous broadcast encryption, which lies between the complete lack of protection that characterizes traditional broadcast encryption scheme [15] and the full anonymity in [1]. Their constructions achieve sublinear ciphertext length but fail to obtain anonymity among the receiver.

The work of Kiayias and Samari [19] aimed to study the lower bounds for the ciphertext size of private broadcast encryption. They showed that an atomic private broadcast encryption scheme with fully anonymous must have a ciphertext size of $\Omega(n \cdot k)$, where n is the number of broadcast set and k is the security parameter. Recently, Fazio, Nicolosi and Perera [13] studied the broadcast steganography and introduced a new construction called outsider-anonymous broadcast encryption with pseudorandom ciphertexts, which achieves sublinear ciphertext size and is secure without random oracles.

Revocation. The revocation schemes in the literature only guarantee the revoked users cannot decrypt the ciphertext. While the revocation in our paper focuses on how to revoke the identities from a group of users S. Only the users who are in S but not in the revocation set can retrieve the plaintext. Revocation system is a variant of the broadcast encryption system, where it takes a set of revoked users as input to the encryption function. Several elegant revocation constructions [23, 24, 11, 17, 5, 21, 20] have been proposed. Naor, Naor and Lotspiech [23] presented a technique called subset-cover framework, and based on this framework they proposed the first stateless tree-based revocation scheme which was secure against a collision of any number of users. Boneh and Waters [5] introduced a primitive called augmented broadcast encryption which was claimed to be sufficient for constructing trace and revoke schemes. The authors proposed a revocation scheme with sublinear size ciphertexts and private keys. The scheme was proved to be secure against adaptive adversaries.

Lewko, Sahai and Waters [21] proposed a revocation system with very small private keys using the "two equation" technique. The primary challenge is to achieve full collusion resilience. Anyone can decrypt the ciphertext and get the broadcast message except the revoked users even if they collude. In Lewko et al.'s scheme, the ciphertext size is $O(t)$ and the size of the public key is constant, where t is the number of revoked users. Recently, to narrow the scope of decrypter in [21], a single revocation encryption (SRE) scheme was presented by Lee et al. [20], which allows a sender to broadcast a message to a group of selected users and one group user is revoked. Any group member can decrypt the ciphertext except the revoked user. The authors then proposed a public key trace and revoke scheme by combining the layered subset difference scheme and their SRE scheme.

Broadcast Proxy Re-Encryption. The concept of proxy re-encryption (PRE) was introduced by Blaze, Bleumer and Strauss [2], which provides a flexible and secure way to share data. PRE allows an honest-but-curious proxy to turn a ciphertext intended for a receiver into another ciphertext intended for another receiver. While, the proxy cannot learn any useful information about the plaintext during the transformation. Chu et al. [7] extended this notion to construct the proxy broadcast re-encryption (PBRE). Compared with PRE, PBRE allows the proxy to transform a ciphertext intended for a receiver set to another ciphertext intended for another receiver set. Recently, motivated by the cloud email system, Xu et al. [27] presented a conditional identity-based broadcast proxy re-encryption scheme with constant ciphertext based on [8]. In both the PRE and PBRE system, the data owner has to delegate a re-encryption key to the proxy and the proxy knows the new receivers' identities.

Organization. The rest of the paper is organized as follows. In Section 2, we give some preliminaries including complexity assumption, the formal definition of anonymous identity-based broadcast encryption with revocation and the corresponding security models. The concrete construction is presented in Section 3. In Section 4, we show the security proofs of our scheme. Finally, we conclude the paper in Section 5.

2 Preliminaries

2.1 Complexity Assumption

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups of the same prime order p. A bilinear map is a map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ which satisfies the following properties:

1. Bilinear: For all $P, Q \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p^*$, we have $e\left(aP, bQ\right) = e(P, Q)^{ab}$.
2. Non-degeneracy: There exists $P, Q \in \mathbb{G}$ such that $e\left(P, Q\right) \neq 1$.
3. Computability: It is efficient to compute $e\left(P, Q\right)$ for all $P, Q \in \mathbb{G}$.

A bilinear group $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ is composed of objects as described above.

Bilinear Diffie-Hellman Problem (BDH). Let $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ be a bilinear group with a generator $P \in \mathbb{G}$. The BDH problem in $(\mathbb{G}, \mathbb{G}_T, e)$ is as follows: Given a tuple (P, aP, bP, cP) for some unknown $a, b, c \in \mathbb{Z}_p^*$ as input, output $e(P, P)^{abc} \in \mathbb{G}_T$. An algorithm $\mathcal{A}$ has advantage ε in solving BDH in $(\mathbb{G}, \mathbb{G}_T, e)$ if

$$\Pr\left[\mathcal{A}\left(P, aP, bP, cP\right) = e(P, P)^{abc}\right] \geq \varepsilon,$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_p^*$ and $P \in \mathbb{G}$.

Definition 1. *We say that the BDH assumption holds in $\mathbb{G}$ if no PPT adversary has advantage at least ε in solving the BDH problem in $\mathbb{G}$.*

2.2 Anonymous ID-Based Broadcast Encryption with Revocation

The AIBBER system is derived from Identity-Based Broadcast Encryption (IBBE) [8] with more functions. Formally, an AIBBER scheme consists of the algorithms $\mathcal{AIBBER} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Revoke}, \mathsf{Decrypt})$ defined as follows.

Setup (1^λ)**:** Taking a security parameter 1^λ as input, it outputs a master public key mpk and a master secrete key msk. The mpk is publicly known while the msk is kept secretly.

KeyGen $(\mathsf{mpk}, \mathsf{msk}, ID)$**:** Taking the master key pair $(\mathsf{msk}, \mathsf{mpk})$ and a user identity ID as input, it outputs a private key d_{ID} for ID.

Encrypt (mpk, M, S)**:** Taking the master public key mpk, a message M and a set of identities $S = (ID_1, ID_2, ..., ID_n)$ as input, it outputs a ciphertext CT.

Revoke (mpk, R, CT)**:** Taking the master public key mpk, a ciphertext CT and a revocation identity set $R = (ID_1, ID_2, \cdots, ID_t)$ as input, it outputs a new ciphertext CT' with R.

Decrypt $(\mathsf{mpk}, CT', ID, d_{ID})$**:** Taking the master public key mpk, a ciphertext CT', an identity ID and the private key d_{ID} as input. It outputs the message M if $ID \in S$ and $ID \notin R$.

Correctness. Note that if $t = 0$, the AIBBER scheme is AIBBE scheme. Thus, it requires that for any $ID \in S$ and $ID \notin R$, if $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, $d_{ID} \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, ID)$, $CT \leftarrow \mathsf{Encrypt}(\mathsf{mpk}, M, S)$, $CT' \leftarrow \mathsf{Revoke}(\mathsf{mpk}, R, CT)$, we have $\mathsf{Decrypt}(CT, ID, d_{ID}) = M$ and $\mathsf{Decrypt}(CT', ID, d_{ID}) = M$.

2.3 Security Models

The security of AIBBER scheme requires that without a valid private key, both the encrypted message and the intended receivers are unknown to the adversary. Let CT be the original ciphertext for receivers S, R be the revoke users and CT' be the ciphertext after revocation. The security requires:

1. The message in the ciphertext CT cannot be distinguished without a valid private key associated with an identity $ID \in S$. The message in CT' cannot be distinguished without a valid private key associated with an identity $ID' \in S$ and $ID' \notin R$.
2. The identity set in the ciphertext CT cannot be distinguished without a valid private key associated with an identity $ID \in S$. The identity set in CT' cannot be distinguished without a valid private key associated with an identity $ID' \in S$ and $ID' \notin R$.

We define the IND-ID-CPA security and ANON-ID-CPA security for the AIBBER system in a similar way as anonymous IBBE system.

IND-ID-CPA Security. IND-ID-CPA security in AIBBER allows the adversary to issue the private key query to obtain the private key associated with any identity ID of her choice. The adversary is challenged on an identity set S^*, two messages M_0, M_1 of its choice and a revocation identity set R^*. Adversary's goal is to distinguish whether the challenge ciphertext is encrypted under M_0 or M_1 for S^* with some restrictions. We say that adversary breaks the scheme if it guesses the message correctly. Specifically, the notion of IND-ID-CPA is defined under the following game between the challenger $\mathcal{C}$ and the PPT adversary $\mathcal{A}$.

Setup: $\mathcal{C}$ runs the **Setup** algorithm to generate the master public key mpk and master secret key msk. Then it sends the mpk to $\mathcal{A}$ and keeps the msk secretly.

Phase 1: $\mathcal{A}$ issues private key queries. Upon receiving a private key query for ID_i. $\mathcal{C}$ runs the **KeyGen** algorithm to generate the private key d_{ID_i} and sends the result back to $\mathcal{A}$.

Challenge: When $\mathcal{A}$ decides that **Phase 1** is over, it outputs two distinct messages M_0, M_1 from the same message space, a challenge identity set $S^* = (ID_1, ID_2, \cdots, ID_n)$ and a revocation identity set $R^* = (ID'_1, ID'_2, \cdots, ID'_t)$ with the restriction that $\mathcal{A}$ has not queried the private key on ID_i in **Phase 1**, where $ID_i \in S^*$ and $ID_i \notin R^*$. $\mathcal{C}$ randomly picks a bit $b \in \{0,1\}$ and generates the challenge ciphertext CT^* as follows:

$$CT = \mathsf{Encrypt}(\mathsf{mpk}, M_b, S^*), \quad CT' = \mathsf{Revoke}(\mathsf{mpk}, M_b, CT).$$

If $R^* \neq \emptyset$, set $CT^* = CT'$ as the challenge ciphertext, otherwise set $CT^* = CT$ as the challenge ciphertext, then send CT^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues more private key queries as in **Phase 1**, but it cannot query the private key on ID_i where $ID_i \in S^*$ and $ID_i \notin R^*$.

Guess: Finally, $\mathcal{A}$ outputs its guess $b' \in \{0,1\}$ and wins the game if $b' = b$.

We refer to such an adversary $\mathcal{A}$ as an IND-ID-CPA adversary and define adversary $\mathcal{A}$'s advantage in attacking the scheme as $\mathsf{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathcal{AIBER}}(\mathcal{A}) = |\Pr[b = b'] - 1/2|$. The probability is over the random bits used by the challenger and the adversary.

Definition 2. *We say that an AIBBER scheme is IND-ID-CPA secure if there is no IND-ID-CPA adversary $\mathcal{A}$ has a non-negligible advantage in this game.*

ANON-ID-CPA Security. ANON-ID-CPA security in AIBBER allows the adversary to issue the private key query to obtain the private key of any identity ID of its choice. Similarly, the adversary is challenged on a message M^*, two identity sets S_0, S_1 and a revocation identity set R^* of its choice. Adversary's goal is to distinguish whether the challenge ciphertext is generated under S_0 or S_1 with some restrictions. We say that adversary breaks the scheme if it guesses the identity set correctly. Specifically, the notion of ANON-ID-CPA is defined under the following game between the challenger $\mathcal{C}$ and the PPT adversary $\mathcal{A}$.

Setup: $\mathcal{C}$ runs the **Setup** algorithm to generate the master public key mpk and master secret key msk. Then it sends the mpk to $\mathcal{A}$ and keeps the msk secretly.

Phase 1: $\mathcal{A}$ issues private key queries. Upon receiving a private key query for ID_i. $\mathcal{C}$ runs the **KeyGen** algorithm to generate the private key d_{ID_i} and sends the result back to $\mathcal{A}$.

Challenge: When $\mathcal{A}$ decides that **Phase 1** is over, it outputs a message M^*, two distinct identity sets $S_0 = (ID_{0,1}, ID_{0,2}, ..., ID_{0,n})$, $S_1 = (ID_{1,1}, ID_{1,2}, ..., ID_{1,n})$ and a revocation set $R^* = (ID'_1, ID'_2, \cdots, ID'_t)$. We require that $\mathcal{A}$ has not issued the private key queries on ID_i in **Phase 1**, where $ID_i \in (S_0 \cup S_1) \backslash (S_0 \cap S_1)$. $\mathcal{C}$ randomly picks a bit $b \in \{0, 1\}$ and generates the challenge ciphertext CT^* as follows:

$$CT = \mathsf{Encrypt}(\mathsf{mpk}, M^*, S_b), \quad CT' = \mathsf{Revoke}(\mathsf{mpk}, M^*, CT).$$

If $R^* \neq \emptyset$, set $CT^* = CT'$ as the challenge ciphertext, otherwise set $CT^* = CT$ as the challenge ciphertext, then send CT^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues more private key queries as in **Phase 1**, but it cannot query the private key on any ID_i, where $ID_i \in (S_0 \cup S_1) \backslash (S_0 \cap S_1)$.

Guess: Finally, $\mathcal{A}$ outputs its guess $b' \in \{0, 1\}$ and wins the game if $b' = b$.

We refer to such an adversary $\mathcal{A}$ as an ANON-ID-CPA adversary and define adversary $\mathcal{A}$'s advantage in attacking the scheme as $\mathsf{Adv}^{\mathsf{ANON\text{-}ID\text{-}CPA}}_{\mathcal{AIBER}}(\mathcal{A}) = |\Pr[b = b'] - 1/2|$. The probability is over the random bits used by the challenger and the adversary.

Definition 3. *We say that an AIBBER scheme is ANON-ID-CPA secure if there is for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{ANON\text{-}ID\text{-}CPA}}_{\mathcal{AIBER}}(\mathcal{A})$ is negligible.*

3 The Proposed Scheme

3.1 Construction

Setup: Given a security parameter 1^λ, the setup algorithm randomly chooses a bilinear group $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ with a generator $P \in \mathbb{G}$, $s \in \mathbb{Z}_p^*$ and computes $P_{pub} = sP$. It then picks four cryptographic hash functions $H : \{0, 1\}^* \to \mathbb{Z}_p^*$, $H_1 : \{0, 1\}^* \to \mathbb{G}$, $H_2 : \mathbb{G}_T \times \{0, 1\}^* \to \mathbb{G}$, $H_3 : \mathbb{G}_T \times \{0, 1\}^* \to \mathbb{G}$. The master public key and master secret key are

$$mpk = \{\mathbb{BG}, P, P_{pub}, H, H_1, H_2, H_3\}, \quad msk = s.$$

KeyGen: Given the master key pair (mpk, msk) and an identity $ID \in \{0, 1\}^*$, this algorithm outputs the private key

$$d_{ID} = sH_1(ID).$$

Encrypt: Given the master public key mpk, a set of identity $S = (ID_1, ID_2, \ldots, ID_n)$ and a message $M \in \mathbb{G}$, this algorithm randomly chooses $r_1, r_2 \in \mathbb{Z}_p^*$ and $v \in \mathbb{G}$. For $i = 1, 2, \cdots, n$, it computes $x_i = H(ID_i)$,

$$f_i(x) = \prod_{j=1, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} = \sum_{j=0}^{n-1} a_{i,j} x^j \mod p,$$

$$A_i = H_2\Big(e\big(H_1(ID_i), P_{pub}\big)^{r_1}, ID_i\Big), \quad B_i = v + H_3\Big(e\big(H_1(ID_i), P_{pub}\big)^{r_2}, ID_i\Big).$$

We have $f_i(x_i) = 1$ and $f_i(x_j) = 0$ for $i \neq j$. Then it creates the ciphertext CT as $C_0 = v + M, C_1 = r_1 P, C_2 = r_2 P$, together with, for each $i = 1, 2, \cdots, n$:

$$Q_i = \sum_{j=1}^{n} a_{j,i-1} A_j, \quad U_i = \sum_{j=1}^{n} a_{j,i-1} B_j.$$

Revoke: Given a ciphertext $CT = (C_0, C_1, C_2, Q_i, U_i, i \in [1, n])$, the master public key mpk and a revocation identity set R, where $|R| = t$. It requires $t < n$. If $R = \emptyset$, this algorithm sets $CT' = CT$. Otherwise, it randomly chooses $u \in \mathbb{G}$ and computes $C_0' = u + C_0$, $x_i = H(ID_i)$ for $ID_i \in R$,

$$g\left(x\right) = \prod_{i=1}^{t} \left(x - x_i\right) = \sum_{i=0}^{t} b_i x^i \mod p.$$

Then it sets $b_i = 0$ for $i = t+1, t+2, \cdots, n-1$ and for each $i = 1, 2, \cdots, n$ computes

$$Q_i' = Q_i + b_{i-1} u.$$

Then it sets $CT' = (R, C_0', C_1, C_2, Q_i', U_i, i \in [1, n])$.

Decrypt: Given a ciphertext $CT' = (R, C_0', C_1, C_2, Q_i', U_i, i \in [1, n])$, an identity ID_i, a private key d_{ID_i} and the master public key mpk, this algorithm computes $x_i = H(ID_i)$ and

$$U = U_1 + x_i U_2 + x_i^2 U_3 + \cdots + x_i^{n-1} U_n, \quad Q = Q_1' + x_i Q_2' + x_i^2 Q_3' + \cdots + x_i^{n-1} Q_n'.$$

Then it computes $x_j = H(ID_j)$ for each $ID_j \in R$ to reconstruct $g(x)$ as:

$$g\left(x\right) = \prod_{j=1}^{t} \left(x - x_j\right) = \sum_{j=0}^{t} b_j x^j \mod p.$$

Finally, it uses the private key d_{ID_i} to compute

$$v' = U - H_3\big(e(C_2, d_{ID_i}), ID_i\big), \quad u' = g(x_i)^{-1} \left(Q - H_2\big(e(C_1, d_{ID_i}), ID_i\big)\right).$$

and recovers the message $M = C_0' - u' - v'$. If the identity $ID_i \in S$ and $ID_i \notin R$, we have $u' = u$, $v' = v$, then it obtains the correct M after decryption.

Note: For simplicity, we omit the modulo operation and assume that the coefficients of all polynomials are from $\mathbb{Z}_p^*$ in the rest of paper.

3.2 Discussion and Correctness

One may think that after revocation, the revocation set may be updated multiple times. Our scheme allows the server to update the revocation set. For each update, the server uses the original ciphertext and the new revocation set to perform the **Revoke** algorithm. Thus, the server needs to store the original ciphertext CT in our scheme. In our setting, there is no requirement of $R \subset S$. The revocation set R can be arbitrary users.

From our setting, only the users in S can decryption the ciphertext CT. After revocation, the revoked users cannot decrypt the ciphertext CT'. We note that if $ID \in R$, $g(H(ID)) = 0$ and $g(H(ID))u = 1_{\mathbb{G}}$. The user with identity ID cannot retrieve one of the decryption keys u, even all users in R conclude. To obtain the decryption keys u and v, the user must belong to S and not belong to R. Thus our scheme ensures that even if all the revoked users collude, they still cannot access the file and learn the identities of receivers.

Next we show that our construction meets the requirements of correctness as we claimed in the Section 2.3. If $x_i = H(ID_i)$ is computed correctly, for any $ID_i \in S$ and $ID_i \notin R$, we have $g(x_i) \neq 0$ and

$$\begin{aligned}
Q &= Q'_1 + x_i Q'_2 + x_i^2 Q'_3 + \cdots + x_i^{n-1} Q'_n \\
&= \big(Q_1 + x_i Q_2 + x_i^2 Q_3 + \cdots + x_i^{n-1} Q_n\big) + \big(b_0 + b_1 x_i + b_2 x_i^2 + \cdots + b_{n-1} x_i^{n-1}\big)\, u \\
&= (a_{1,0}A_1 + a_{2,0}A_2 + \cdots + a_{n,0}A_n) \\
&\quad + x_i\,(a_{1,1}A_1 + a_{2,1}A_2 + \cdots + a_{n,1}A_n) + \cdots \\
&\quad + x_i^{n-1}\,(a_{1,n-1}A_1 + a_{2,n-1}A_2 + \cdots + a_{n,n-1}A_n) + g(x_i)u \\
&= \big(a_{1,0} + a_{1,1}x_i + a_{1,2}x_i^2 + \cdots + a_{1,n-1}x_i^{n-1}\big)\, A_1 \\
&\quad + \big(a_{2,0} + a_{2,1}x_i + a_{2,2}x_i^2 + \cdots + a_{2,n-1}x_i^{n-1}\big)\, A_2 + \cdots \\
&\quad + \big(a_{n,0} + a_{n,1}x_i + a_{n,2}x_i^2 + \cdots + a_{n,n-1}x_i^{n-1}\big)\, A_n + g(x_i)u \\
&= f_1(x_i)A_1 + f_2(x_i)A_2 + \cdots + f_n(x_i)A_n + g(x_i)u \\
&= A_i + g(x_i)u \\
u' &= g(x_i)^{-1} \cdot \big(Q - H_2\big(e(C_1, d_{ID_i}), ID_i\big)\big) \\
&= g(x_i)^{-1} \cdot \big(A_i + g(x_i)u - H_2\big(e(C_1, d_{ID_i}), ID_i\big)\big) \\
&= g(x_i)^{-1} \cdot \Big(H_2\Big(e\big(H_1(ID_i), P_{pub}\big)^{r_1}, ID_i\Big) - H_2\Big(e\big(r_1P, sH_1(ID_i)\big), ID_i\Big) + g(x_i)u\Big) \\
&= g(x_i)^{-1} \cdot (g(x_i)u) \\
&= u.
\end{aligned}$$

The user ID_i uses its private key d_{ID_i} to remove A_i from Q_i via above computation. As $g(x_i) \neq 0$, the user can obtain u.

$$\begin{aligned}
U &= U_1 + x_i U_2 + x_i^2 U_3 + \cdots + x_i^{n-1} U_n \\
&= (a_{1,0}B_1 + a_{2,0}B_2 + \cdots + a_{n,0}B_n) \\
&\quad + x_i(a_{1,1}B_1 + a_{2,1}B_2 + \cdots + a_{n,1}B_n) + \cdots \\
&\quad + x_i^{n-1}(a_{1,n-1}B_1 + a_{2,n-1}B_2 + \cdots + a_{n,n-1}B_n) \\
&= \big(a_{1,0} + a_{1,1}x_i + a_{1,2}x_i^2 + \cdots + a_{1,n-1}x_i^{n-1}\big)\, B_1 \\
&\quad + \big(a_{2,0} + a_{2,1}x_i + a_{2,2}x_i^2 + \cdots + a_{2,n-1}x_i^{n-1}\big)\, B_2 + \cdots \\
&\quad + \big(a_{n,0} + a_{n,1}x_i + a_{n,2}x_i^2 + \cdots + a_{n,n-1}x_i^{n-1}\big)\, B_n \\
&= f_1(x_i)B_1 + f_2(x_i)B_2 + \cdots + f_n(x_i)B_n \\
&= B_i \\
v' &= U - H_3\big(e(C_2, d_{ID_i}), ID_i\big) \\
&= B_i - H_3\Big(e\big(r_2P, sH_1(ID_i)\big), ID_i\Big) \\
&= v + H_3\Big(e\big(P, H_1(ID_i)\big)^{sr_2}, ID_i\Big) - H_3\Big(e\big(H_1(ID_i), P_{pub}\big)^{r_2}, ID_i\Big) \\
&= v.
\end{aligned}$$

After recovering u and v, we get the message as $C'_0 - u' - v' = M + v + u - u - v = M$.

4 Security Analysis

Theorem 1. *Suppose the hash functions H_1, H_2, H_3 are random oracles. If the BDH problem is hard, the proposed scheme is IND-ID-CPA secure. Specifically, suppose* *there is an IND-ID-CPA adversary $\mathcal{A}$ that has advantage ϵ against our proposed scheme. $\mathcal{A}$ makes at most q_E private key queries and q_{H_1}, q_{H_2}, q_{H_3} queries to the functions H_1, H_2 and H_3 respectively. Then there is an algorithm $\mathcal{S}$ to solve the BDH problem with advantage $\epsilon' \geq \frac{\epsilon}{n \cdot e \cdot q_E \cdot (q_{H_2} + q_{H_3})}$, where n is the number of the broadcast identities.*

Proof. Suppose there exists an adversary $\mathcal{A}$ who can break our scheme with advantage ϵ. We build a simulator $\mathcal{S}$ that can solve the BDH problem with advantage ϵ' by running $\mathcal{A}$. Let (P, aP, bP, cP) be a random instance of BDH problem taken as input by $\mathcal{S}$ and its goal is to compute $e(P,P)^{abc}$. In order to use $\mathcal{A}$ to solve the problem, $\mathcal{S}$ needs to simulate a challenger and respond all the queries for $\mathcal{A}$. For simplicity, we assume that the H_2 and H_3 query is after the H_1 query for the same identity. $\mathcal{S}$ works by interacting with $\mathcal{A}$ in an IND-ID-CPA game as follows:

Setup: $\mathcal{S}$ sets $P_{pub} = aP$ and creates $mpk = (p, P, P_{pub}, e, H)$.

H_1**-queries:** $\mathcal{A}$ makes H_1 queries. $\mathcal{S}$ responds to a query on ID_i as follow. $\mathcal{S}$ maintains a list L_1 of a tuple (ID_i, c_i, r_i, h_i). This list is initially empty. $\mathcal{S}$ first checks the L_1. If the query ID_i already appears on the L_1 in a tuple (ID_i, c_i, r_i, h_i), it returns the corresponding h_i as the value of $H_1(ID_i)$. Otherwise, do the following:

1. Select $c_i \in_R \{0,1\}$ with $Pr[c_i = 0] = \delta$ for some δ (determine later).
2. Pick $r_i \in_R \mathbb{Z}_p^*$, if $c_i = 0$, compute $h_i = r_i bP$. If $c_i = 1$, compute $h_i = r_i P$.
3. Add the tuple (ID_i, c_i, r_i, h_i) to the L_1 and respond with h_i to $\mathcal{A}$.

H_2**-queries:** $\mathcal{A}$ makes H_2 queries. $\mathcal{S}$ responds to a query on (X_i, ID_i) as follow. $\mathcal{S}$ maintains a list L_2 of a tuple (X_i, ID_i, λ_i). This list is initially empty. $\mathcal{S}$ first checks the L_2. If the query (X_i, ID_i) already appears on the L_2 in a tuple (X_i, ID_i, λ_i), it returns the corresponding λ_i as the value of $H_2(X_i, ID_i)$. Otherwise, $\mathcal{S}$ randomly picks a $\lambda_i \in \mathbb{G}$ as the value of $H_2(X_i, ID_i)$, then adds the tuple (X_i, ID_i, λ_i) to the L_2 and responds to $\mathcal{A}$ with λ_i.

H_3**-queries:** $\mathcal{A}$ makes H_3 queries. $\mathcal{S}$ responds to a query on (Y_i, ID_i) as follow. $\mathcal{S}$ maintains a list L_3 of a tuple (Y_i, ID_i, γ_i). This list is initially empty. $\mathcal{S}$ first checks the L_3. If the query (Y_i, ID_i) already appears on the L_3 in a tuple (Y_i, ID_i, γ_i), it returns the corresponding γ_i as the value of $H_3(Y_i, ID_i)$. Otherwise, $\mathcal{S}$ randomly picks a $\gamma_i \in \mathbb{G}$ as the value of $H_3(Y_i, ID_i)$, then adds the tuple (Y_i, ID_i, γ_i) to the L_3 and responds to $\mathcal{A}$ with γ_i.

Phase 1: $\mathcal{A}$ issues the private key queries on ID_i for several times as needed. For each time, $\mathcal{S}$ first runs the H_1 query to get the corresponding c_i and r_i. If $c_i = 0$, $\mathcal{S}$ aborts. If $c_i = 1$, $\mathcal{S}$ computes $d_{ID_i} = sH_1(ID_i) = ar_iP = r_iP_{pub}$.

Challenge: When $\mathcal{A}$ decides **Phase 1** is over, it outputs two distinct messages M_0, M_1, a challenge identity set $S^* = (ID_1, ID_2, \cdots, ID_n)$ and a revocation identity set $R^* = (ID'_1, ID'_2, \cdots, ID'_t)$ under the restriction that $\mathcal{A}$ has not queried the private key on ID_i in **Phase 1**, where $ID_i \in S^*$ and $ID_i \notin R^*$. $\mathcal{S}$ randomly picks a random bit $b \in \{0,1\}$ and does the follows:

Case 1: $R^* = \emptyset$. In this case, $\mathcal{S}$ randomly picks $r^* \in \mathbb{Z}_p^*$, $C_0^* \in \mathbb{G}$, for each $ID_i \in S^*$, $i = 1, 2 \cdots, n$, randomly chooses A_i, $B_i^* \in \mathbb{G}$ and computes $x_i^* = H(ID_i)$,

$$f_i(x) = \prod_{j=1, j \neq i}^{n} \frac{x - x_j^*}{x_i^* - x_j^*} = \sum_{j=0}^{n-1} a_{i,j} x^j,$$

Then $\mathcal{S}$ generates the challenge ciphertext CT^* as $C_0, C_1^* = r^*cP, C_2^* = cP$, together with, for each $i = 1, 2, \cdots, n$:

$$Q_i^* = \sum_{j=1}^{n} a_{j,i-1} A_j^*, \quad U_i^* = \sum_{j=1}^{n} a_{j,i-1} B_j^*.$$

Case 2: $R^* \neq \emptyset$. In this case, $\mathcal{S}$ does the follows:

1. Pick $r^* \in_R \mathbb{Z}_p^*$, $v^*, u^* \in_R \mathbb{G}$, compute $C_0'^* = v^* + u^* + M_b$, $C_1^* = r^*cP$, $C_2^* = cP$.
2. For each $(ID_i \in S^*) \wedge (ID_i \notin R^*)$, $\mathcal{S}$ randomly chooses A_i, $B_i^* \in \mathbb{G}$. For each $(ID_i \in S^*) \wedge (ID_i \in R^*)$, $\mathcal{S}$ gets r_i from the L_1 (If ID_i is not in the L_1, do H_1 queries to get r_i). Then it computes $X_i = e(aP, cP)^{r^* r_i}$ and checks whether the tuple (X_i, ID_i) in the L_2. If yes, it obtains the corresponding λ_i and sets $A_i^* = \lambda_i$. Otherwise, it randomly choose $A_i^* \in \mathbb{G}$ and adds the new tuple (X_i, ID_i, A_i^*) to the L_2. Then $\mathcal{S}$ computes $Y_i = e(aP, cP)^{r_i}$ and checks whether the tuple (Y_i, ID_i) in the L_3. If yes, it obtains the corresponding γ_i and sets $w_i^* = \gamma_i$. Otherwise, it randomly chooses $w_i^* \in \mathbb{G}$ and adds the new tuple (Y_i, ID_i, w_i^*) to the L_3, and computes $B_i^* = w_i^* + v^*$.
3. For each $ID_i \in S^*$, $i = 1, 2 \cdots, n$, compute $x_i^* = H(ID_i)$,

$$f_i(x) = \prod_{j=1, j\neq i}^{n} \frac{x - x_j^*}{x_i^* - x_j^*} = \sum_{j=0}^{n-1} a_{i,j} x^j,$$

$$Q_i^* = \sum_{j=1}^{n} a_{j,i-1} A_j^*, \quad U_i^* = \sum_{j=1}^{n} a_{j,i-1} B_j^*.$$

4. Compute $x_i'^* = H(ID_i)$ for $ID_i \in R^*$ and

$$g\left(x\right) = \prod_{i=1}^{t} (x - x_i'^*) = \sum_{i=0}^{t} b_i x^i.$$

   Then set $b_i = 0$ for $i = t+1, t+2, \cdots, n-1$. For $1 \leq i \leq n$, compute

$$Q_i'^* = Q_i^* + b_{i-1} u^*,$$

   and set $CT^* = (R^*, C_0'^*, C_1^*, C_2^*, Q_i'^*, U_i^*, i \in [1, n])$.

Phase 2: $\mathcal{A}$ issues private key queries as needed, but it cannot query the private key on ID_i, where $ID_i \in S^*$ and $ID_i \notin R^*$. $\mathcal{S}$ responds as in **Phase 1**.

Guess: Finally, $\mathcal{A}$ outputs its guess $b' \in \{0, 1\}$.

Probability Analysis. Note that in the case $R^* = \emptyset$, we can view v^* as the encryption key to encrypt the challenge message. Let $W = (e(H_1(ID_i), P_{pub})^c, ID_i)$ where $ID_i \in S^*$. In the real scheme, $B_i^* = v^* + H_3(W)$, thus we also can regard $H_3(W)$ as the encryption key to encrypt v^*. Before querying the H_3 value of W, the result of $H_3(W)$ is unknown and random. From the view of adversary, v^* is encrypted with a random number key independent of W. Therefore, B_i^* is a one-time pad. In other words, the challenge ciphertext is a one-time pad. According to the assumption($\mathcal{A}$ can break our scheme with advantage ϵ), the adversary will query H_3 on W. In this case, simulator decides the corresponding hard problem's solution is in the L_3 and can solve it with probability $\frac{\delta}{n}$.

When $R^* \neq \emptyset$, we can view v^* and u^* as the encryption key to encrypt the challenge message. However, in this case, the adversary can retrieve v^* by querying the private key of $(ID_i \in S^*) \wedge (ID_i \in R^*)$. That is, the message encryption key is only u^*. Let $\varOmega = \left(e(H_1(ID_i), P_{pub})^{r^*c}, ID_i\right)$, where $(ID_i \in S^*) \wedge (ID_i \notin R^*)$. Similarly, in real scheme $Q^* = A_i^* + g(x_i^*)u^* = H_2(\varOmega) + g(x_i^*)u^*$, we can regard $\varOmega$ as the encryption key to encrypt u^*. Before querying the H_2 value of $\varOmega$, the result of $H_2(\varOmega)$ is unknown and random. From the view of adversary, u^* is encrypted with a random number key independent of $\varOmega$. Therefore, Q^* is a one-time pad, that is, the challenge ciphertext is a one-time pad. According to the assumption($\mathcal{A}$ can break our scheme with advantage ϵ), the adversary will query H_2 on $\varOmega$. In this case, simulator can decides the solution of the corresponding hard problem is in the L_3 and solve it with probability $\frac{\delta}{n-l}$ where $l = |S^* \cap R^*|$. Here, we define the query which can solve the hard problem as *useful query*.

If *useful query* happens, it means $c_j = 0$, $H_1(ID_j) = r_j bP$ and $d_{ID_j} = r_j abP$. From the decryption algorithm, we have $e(C_1^*, d_{ID_j}) = e(P,P)^{r^* r_j abc}$ and $e(C_2^*, d_{ID_j})$ $= e(P,P)^{r_j abc}$. Here $\mathcal{S}$ ign ores the guess of $\mathcal{A}$ and picks a random tuple from the L_2 or L_3. It first obtains the corresponding r_j from the L_1. If $\mathcal{S}$ picks the tuple (X_j, ID_j, λ_j) from the L_2, it computes $X_j^{(r^* r_j)^{-1}}$ as the solution to the given instance of BDH problem. If $\mathcal{S}$ picks the tuple (Y_j, ID_j, γ_j) from the L_3, it computes $X_j^{r^{-1}_j}$ as the solution to the given instance of BDH problem.

The above completes the description of simulation algorithm $\mathcal{S}$. To complete the security proof, it remains to show that $\mathcal{S}$ correctly outputs $e\left(P,P\right)^{abc}$ with advantage at least ϵ'. According to our above analysis, we first define the following events:

E_1: Simulation dose not abort in private key query.
E_2: At least one of the H_1 values of challenge identities contains hard problem.
E_3: Adversary chooses an identity where $c_i = 0$ to distinguish challenge message.
E_4: Simulator correctly chooses the solution from the L_2 or L_3 list.

The simulator can successfully solve the hard problem if and only if all events happen simultaneously. Next, we analyze the probability of all events. From the private key query, we know when each $c_i = 1$, simulation will not abort, thus

$$\Pr[E_1] = \Pr[c_i = 1, i = 1, 2, \cdots, q_E] = (1-\delta)^{q_E}.$$

All c_i are chosen by simulator where $c_i = 0$ with probability δ, $c_i = 1$ with probability $1-\delta$. When $c_i = 0$, the value of H_1 contains the hard problem, thus $\Pr[E_2] = \delta$. Since all c_i are chosen by simulator and they are secretly to adversary, adversary does not know which identity's c_i is equal to 0 or 1. That is, from adversary's point of view, it does not know the probabilities of $c_i = 0$ and $c_i = 1$. Therefore, under event E_2, we have

$$\begin{aligned}\Pr[E_3] &= \Pr[E_3|c_i=0]\Pr[c_i=0] + \Pr[E_3|c_i=1]\Pr[c_i=1] \\ &= \tfrac{1}{n-l}\Pr[c_i=0] + \tfrac{1}{n-l}\Pr[c_i=1] \\ &= \tfrac{1}{n-l} \geq \tfrac{1}{n}.\end{aligned}$$

Note that the identity $ID_i \in S^* \cap R^*$ allows to query the corresponding private key. In our setting, these identities cannot be used to distinguish the challenge messages. Since $|S^* \cap R^*| = l$, the potential useful identity is $n - l$. Thus we have above result $\Pr[E_3] = \frac{1}{n-l} \geq \frac{1}{n}$.

Finally, from the simulator's point of view, if adversary can guess the correct b' and with the conditions that E_1, E_2, E_3 happen, it only knows that the solution of the hard problem is in the L_2 or L_3, but it dose not know which one is, thus $\Pr[E_4] \geq \frac{1}{q_{H_2}+q_{H_3}}$. It is clear that these four events are independent, therefore, we have

$$\begin{aligned}\epsilon' &\geq \Pr[E_1 \wedge E_2 \wedge E_3 \wedge E_4] \cdot \epsilon \\ &= \Pr[E_1] \cdot \Pr[E_2] \cdot \Pr[E_3] \cdot \Pr[E_4] \cdot \epsilon \\ &\geq (1-\delta)^{q_E} \cdot \delta \cdot \tfrac{1}{n} \cdot \tfrac{1}{q_{H_2}+q_{H_3}} \cdot \epsilon \\ &= (1-\delta)^{q_E} \cdot \delta \cdot \tfrac{\epsilon}{n(q_{H_2}+q_{H_3})}.\end{aligned}$$

The function $(1-\delta)^{q_E} \cdot \delta$ is maximized at $\delta = \frac{1}{q_E+1}$, we have

$$(1-\delta)^{q_E} \cdot \delta = \frac{1}{q_E+1} \cdot \left(1 - \frac{1}{q_E+1}\right)^{q_E} = \frac{1}{q_E} \cdot \left(1 - \frac{1}{q_E+1}\right)^{q_E+1}.$$

For a large q_E, $\left(1 - \frac{1}{q_E+1}\right)^{q_E+1} \approx \frac{1}{e}$, thus we have

$$\epsilon' \geq (1-\delta)^{q_E} \cdot \delta \cdot \frac{\epsilon}{n(q_{H_2}+q_{H_3})} \approx \frac{\epsilon}{n \cdot e \cdot q_E \cdot (q_{H_2}+q_{H_3})}.$$

This completes the proof. □

Discussion. When $R^* = \emptyset$, the challenge message is encrypted by v^*. If the adversary can distinguish the message, the simulator can decide it must have queried the H_3 value with the input embedding the hard problem, but simulator does not know which input embeds the hard problem. In this case, $\Pr[E_4] = \frac{1}{q_{H_3}} \geq \frac{1}{q_{H_2}+q_{H_3}}$. When $R^* \neq \emptyset$, even the inputs of H_3 contain the hard problem, the adversary can retrieve v^* by the identity $ID_i \in S^*$ and $ID_i \in R^*$. Thus the *useful queries* are from H_2 and $\Pr[E_4] = \frac{1}{q_{H_2}} \geq \frac{1}{q_{H_2}+q_{H_3}}$.

Theorem 2. *Suppose the hash functions H_1, H_2, H_3 are random oracles. The proposed scheme is ANON-ID-CPA secure under the BDH assumption. Specifically, suppose there is an ANON-ID-CPA adversary $\mathcal{A}$ that has advantage ϵ against our proposed scheme. $\mathcal{A}$ makes at most q_E private key queries and q_{H_1}, q_{H_2}, q_{H_3} queries to the functions H_1, H_2 and H_3 respectively. Then there is an algorithm $\mathcal{S}$ to solve the BDH problem with advantage $\epsilon' \geq \frac{\epsilon}{n \cdot e \cdot q_E \cdot (q_{H_2}+q_{H_3})}$, where n is the number of broadcast identities.*

Proof. The proof of **Theorem 2** is similar to the proof of **Theorem** 1. Given a random instance of BDH problem (P, aP, bP, cP), $\mathcal{S}$ works by interacting with $\mathcal{A}$ in an ANON-ID-CPA game. The **Setup**, H_1**-query**, H_2**-query**, H_3**-query** and **Phase 1** query are the same as in **Theorem** 1.

Challenge: When $\mathcal{A}$ decides **Phase 1** is over, it outputs a challenge message M^*, two distinct identity sets $S_0 = (ID_{0,1}, ID_{0,2}, \cdots, ID_{0,n})$, $S_1 = (ID_{1,1}, ID_{1,2}, \cdots, ID_{1,n})$ and a revocation identity set $R^* = (ID'_1, ID'_2, \cdots, ID'_t)$. We require that any identity $ID_i \in (S_0 \cup S_1) \backslash (S_0 \cap S_1)$ has not been queried the private key in **Phase 1**. $\mathcal{S}$ picks a random bit $b \in \{0, 1\}$ and dose the follows:

1. Pick $r^* \in_R \mathbb{Z}_p^*$, $v^* \in \mathbb{G}$, compute $C_0^* = v^* + M$, $C_1^* = r^* cP$, $C_2^* = cP$.
2. For each $ID_i \in S_b \backslash (S_0 \cap S_1)$, randomly choose $A_i^*, B_i^* \in \mathbb{G}$. For each $ID_i \in S_0 \cap S_1$, $\mathcal{S}$ first gets r_i from the L_1 (If ID_i is not in the L_1, do H_1 queries to get r_i). Then it computes $X_i = e(aP, cP)^{r^* r_i}$ and checks whether the tuple (X_i, ID_i) is in the L_2. If yes, it obtains the corresponding λ_i and sets $A_i^* = \lambda_i$. Otherwise, it randomly chooses $A_i^* \in \mathbb{G}$ and adds the new tuple (X_i, ID_i, A_i^*) to the L_2. Then $\mathcal{S}$ computes $Y_i = e(aP, cP)^{r_i}$ and checks whether the tuple (Y_i, ID_i) in the L_3. If yes, it obtains the corresponding γ_i and sets $w_i^* = \gamma_i$. Otherwise, it randomly chooses $w_i^* \in \mathbb{G}$ and adds the new tuple (Y_i, ID_i, w_i^*) to the L_3, and computes $B_i^* = w_i^* + v^*$.
3. For each $i = 1, 2 \cdots, n$, compute $x_i^* = H(ID_i)$,
$$f_i(x) = \prod_{j=1, j \neq i}^{n} \frac{x - x_j^*}{x_i^* - x_j^*} = \sum_{j=0}^{n-1} a_{i,j} x^j,$$
$$Q_i^* = \sum_{j=1}^{n} a_{j,i-1} A_j^*, \quad U_i^* = \sum_{j=1}^{n} a_{j,i-1} B_j^*,$$
and set $CT = (R^*, C_0^*, C_1^*, C_2^*, Q_i^*, U_i^*, i \in [1, n])$.

Case 1: $R^* = \emptyset$. $\mathcal{S}$ sets the challenge ciphertext $CT^* = CT$.

Case 2: $R^* \neq \emptyset$. $\mathcal{S}$ randomly chooses $u^* \in \mathbb{G}$ and computes $C_0'^* = u^* + C_0^*$. For each $ID_i \in R^*$, $\mathcal{S}$ computes $x_i'^* = H(ID_i)$,
$$g\,(x) = \prod_{i=1}^{t} (x - x_i'^*) = \sum_{i=0}^{t} b_i x^i,$$
and sets $b_i = 0$ for $i = t+1, t+2, \cdots, n-1$. Finally, for each $i = 1, 2, \cdots, n$, $\mathcal{S}$ computes
$$Q_i'^* = Q_i^* + b_{i-1} u^*,$$
and sets $CT^* = (R^*, C_0'^*, C_1^*, C_2^*, Q_i'^*, U_i^*, i \in [1, n])$.

Phase 2: $\mathcal{A}$ issues more private key queries, but it cannot query the private key on ID_i, where $ID_i \in (S_0 \cup S_1) \backslash (S_0 \cap S_1)$. $\mathcal{S}$ responds as in **Phase 1**.

Guess: Finally, $\mathcal{A}$ outputs its guess $b' \in \{0, 1\}$.

The probability analysis is almost similar to the one of **Theorem** 1. Due to space constraints, we omit it here.

5 Conclusion

We presented an anonymous identity-based broadcast encryption with revocation scheme for file sharing. The file owner can encrypt a file for sharing with a group of users and stores the encrypted file in the cloud server (or any other third party). The server can revoke target users without knowing the file and the receiver identities. Our scheme ensures that even if all the revoked users collude, they still cannot access the file and learn the identities of receivers. The cloud server also learns nothing about the file and the receiver identities. Finally, we proved that the proposed scheme is IND-ID-CPA secure and ANON-ID-CPA secure under the BDH assumption in the random oracle model.

References

1. Barth, A., Boneh, D., Waters, B.: Privacy in encrypted content distribution using private broadcast encryption. In: Financial Cryptography 2006, Lecture Notes in Computer Science. vol. 4107, pp. 52 – 64 (2006)
2. Blaze, M., Bleumer, G., Strauss, M.: Divertible protocols and atomic proxy cryptography. In: Advances in Cryptology EUROCRYPT'98, Lecture Notes in Computer Science. vol. 1403, pp. 127–144 (1998)
3. Boneh, D., Gentry, C., Waters, B.: Collusion resistant broadcast encryption with short ciphertext and private keys. In: CRYPTO 2005, Lecture Notes in Computer Science. vol. 3621, pp. 258 – 275 (2005)
4. Boneh, D., Sahai, A., Waters, B.: Fully collusion resistant traitor tracing with short ciphertexts and private keys. In: EUROCRYPT 2006, Lecture Notes in Computer Science. vol. 4004, pp. 573 – 592 (2006)
5. Boneh, D., Waters, B.: A fully collusion resistant broadcast, trace, and revoke system. In: CCS '06 Proceedings of the 13th ACM conference on Computer and communications security. pp. 211 – 220 (2006)
6. Boneh, D., Waters, B., Zhandry, M.: Low overhead broadcast encryption from multilinear maps. In: CRYPTO 2014, Lecture Notes in Computer Science. vol. 8616, pp. 206 – 223 (2014)
7. Chu, C.K., Weng, J., Chow, S.S.M., Zhou, J., Deng, R.H.: Conditional proxy broadcast re-encryption. In: 14th Australasian Conference, ACISP 2009, Lecture Notes in Computer Science. vol. 5594, pp. 327–342 (2009)
8. Delerablée, C.: Identity-based broadcast encryption with constant size ciphertexts and private keys. In: ASIACRYPT 2007, Lecture Notes in Computer Science. vol. 4833, pp. 200 – 215 (2007)
9. Delerablée, C., Paillier, P., Pointcheval, D.: Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys. In: Pairing-Based Cryptography - Pairing 2007, Lecture Notes in Computer Science. vol. 4575, pp. 39–59 (2007)
10. Dodis, Y., Fazio, N.: Public key broadcast encryption for stateless receivers. In: Security and Privacy in Digital Rights Management, ACM CCS-9 Workshop. vol. 2696, pp. 61–80 (2002)
11. Dodis, Y., Fazio, N.: Public key trace and revoke scheme secure against adaptive chosen ciphertext attack. In: Public Key Cryptography PKC 2003, Lecture Notes in Computer Science. vol. 2567, pp. 100–115 (2003)
12. Fan, C., Huang, L., Ho, P.: Anonymous multireceiver identity-based encryption. IEEE Trans. Computers 59(9), 1239–1249 (2010)
13. Fazio, N., Nicolosi, A.R., Perera, I.M.: Broadcast steganography. In: CT-RSA 2014, Lecture Notes in Computer Science. vol. 8366, pp. 64 – 84 (2014)

14. Fazio, N., Perera, I.M.: Outsider-anonymous broadcast encryption with sublinear ciphertexts. In: PKC 2012, Lecture Notes in Computer Science. vol. 7293, pp. 225 – 242 (2012)
15. Fiat, A., Naor, M.: Broadcast encryption. In: Advances in Cryptology-CRYPTO 1993, Lecture Notes in Computer Science. vol. 773, pp. 480 – 491 (1994)
16. Gentry, C., Waters, B.: Adaptive security in broadcast encryption systems (with short ciphertexts). In: EUROCRYPT 2005, Lecture Notes in Computer Science. vol. 5479, pp. 171 – 188 (2009)
17. Goodrich, M.T., Sun, J.Z., Tamassia, R.: Efficient tree-based revocation in groups of low-state devices. In: Advances in Cryptology CRYPTO 2004, Lecture Notes in Computer Science. vol. 3152, pp. 511–527 (2004)
18. Hur, J., Park, C., Hwang, S.: Privacy-preserving identity-based broadcast encryption. Information Fusion 13(4), 296–303 (2012)
19. Kiayias, A., Samari, K.: Lower bounds for private broadcast encryption. In: Information Hiding, Lecture Notes in Computer Science. vol. 7692, pp. 176– 190 (2013)
20. Lee, K., Koo, W.K., Lee, D.H., Park, J.H.: Public-key revocation and tracing schemes with subset difference methods revisited. In: Computer Security - ESORICS 2014, Lecture Notes in Computer Science. vol. 8713, pp. 1 – 18 (2014)
21. Lewko, A., Sahai, A., Waters, B.: Revocation systems with very small private keys. In: 2010 IEEE Symposium on Security and Privacy. pp. 273 – 285 (2010)
22. Libert, B., Paterson, K.G., Quaglia, E.A.: Anonymous broadcast encryption: Adaptive security and efficient constructions in the standard model. In: Public Key Cryptography-PKC 2012, Lecture Notes in Computer Science. vol. 7293, pp. 206 – 224 (2012)
23. Naor, D., Naor, M., Lotspiech, J.: Revocation and tracing schemes for stateless receivers. In: Advances in Cryptology CRYPTO 2001, Lecture Notes in Computer Science. vol. 2139, pp. 41 – 62 (2001)
24. Naor, M., Pinkas, B.: Efficient trace and revoke schemes. In: Financial Cryptography, Lecture Notes in Computer Science. vol. 1962, pp. 1 – 20 (2001)
25. Phan, D.H., Pointcheval, D., Shahandashti, S.F., Strefler, M.: Adaptive cca broadcast encryption with constant-size secret and ciphertexts. In: ACISP 2012. vol. 7372, pp. 308 – 321 (2012)
26. Sakai, R., Furukawa, J.: Identity-based broadcast encryption. IACR Cryptology ePrint Archive 2007, 217 (2007)
27. Xu, P., Jiao, T., Wu, Q., Wang, W., Jin, H.: Conditional identity-based broadcast proxy re-encryption and its application to cloud email. IEEE Transactions on Computers 65(1), 66–79 (2016)
28. Zhang, L., Wu, Q., Mu, Y.: Anonymous identity-based broadcast encryption with adaptive security. In: Cyberspace Safety and Security - 5th International Symposium, CSS 2013. Lecture Notes in Computer Science, vol. 8300, pp. 258–271 (2013)