Identity-Based Broadcast Encryption

Abstract. Broadcast encryption schemes enable senders to efficiently broadcast ciphertexts to a large set of receivers in a way that only non-revoked receivers can decrypt them. Identity-based encryption schemes are public key encryption schemes that can use arbitrary strings as public keys. We propose the first public key broadcast encryption scheme that can use any string as a public key of each receiver. That is, identity-based broadcast encryption scheme. Our scheme has many desirable properties. The scheme is fully collusion resistant, and the size of ciphertexts and that of private key are small constants. The size of public key is proportional to only the maximum number of receiver sets to each of which the ciphertext is sent. Note that its size remains to be so although the number of potential receivers is super-polynomial size. Besides these properties, the achieving the first practical identity-based broadcast encryption scheme itself is the most interesting point of this paper. The security of our scheme is proved in the generic bilinear group model.

Keywords: broadcast encryption, ID based encryption, bilinear map, revocation

1 Introduction

Broadcast encryption schemes are cryptosystems that enable senders to efficiently broadcast ciphertexts to a large set of receivers such that only this chosen receivers can decrypt them. Their notion was first introduced by Berkovits in [4] and was given formal analysis by Fiat and Naor in [14]. Many schemes, such as [14, 1, 21, 20, 16, 13, 3, 15, 2, 9], have been proposed since then, and their main purpose is to decrease private key size, ciphertext size, public key size, and computational costs for encryption and decryption.

One notable broadcast encryption scheme is that proposed by Boneh et al. in [9]. This is a public key broadcast encryption scheme that features many desirable properties. Particularly notable is the fact that both the receiver's private key size and ciphertexts size are small constants when broadcasting to any set of receivers. However, the length of its public key is proportional to the number of potential receivers.

Identity-based encryption schemes are public key cryptosystems that can use any string as a public key of each receiver. If the public key broadcast encryption is identity-based, senders are able to send ciphertexts to any set of receivers who had never engaged any setup procedure with the system. This implies that its public key size does not depend on the number of potential receivers. None of previous schemes has achieved such a property.

In this paper we propose the first identity-based broadcast encryption scheme that has many desirable property. It is fully collusion resistant, its ciphertexts excluding the information to specifies receiver set is short constant length, each private keys is short constant length, and the length of its public key is proportional to only the maximum number of receiver sets. Although the required computational costs are proportional to the square of the number of receivers when the set of receivers is newly determined, the computational costs that is required for each sender and each receiver are small constants as long as messages are encrypted for the same set of receivers as previous one. When some receivers are added or removed from the set of receivers, the computational costs required for each sender and each receiver are proportional to the product of the number of receivers in this set and that of added and removed receivers.

The security of our scheme is proved in the generic bilinear group model under existence of random oracles. The random oracle model and generic group model are much weaker security model than the standard model. In fact, there are protocols whose security can be proved in the random oracle model but are not secure at all when the random oracle is instantiated. However, these weak protocols are only those who have special and artificial structure. No natural scheme with a security proof in the generic group model are vulnerable. At the end of paper, we mention that our scheme does not in fact require random oracles. And we discuss how the security of our scheme can be proved outside of the generic bilinear group model.

Our scheme is the most advantageous over the previous scheme, such as Boneh's scheme in [9], when the number of potential receivers is huge but the maximum size of its receiver set is rather small and receivers set does not change drastically on an average day. Example of such a case is when all Internet users are potential receivers and some of them form a group based on their identities and communicate each other. Beside such a direct application, our scheme is a new primitive and has a potential to be employed by many application as a building block.

Our paper is organized as follow. Section 2 defines the algorithms and security requirements of identity-based broadcast encryption schemes. Section 3 introduce the generic bilinear group model and a related theorem that our proof of security depends on. Section 4 proposes our scheme, the first identity-based broadcast encryption scheme. Section 5 proves that security of our scheme in the generic bilinear group model. Section 6 analyzes the performance of our scheme. Section 7 discusses how much the security of our scheme depend on the random oracles model and the generic bilinear group model, and then it also discusses about the technique to make our scheme secure under chosen ciphertext attacks. Section 8 concludes our paper.

2 Model of Identity-Based Broadcast Encryption

2.1 Algorithms

Three types of players participate in identity-based broadcast encryption schemes: a manager $\mathcal{M}$, sender $\mathcal{E}$, and receivers. We let n be the maximum size of receiver sets to each of which ciphertext is sent.

An identity-based broadcast encryption scheme consists of four algorithms: Setup, KeyExt, Enc, and Dec.

Setup: A probabilistic setup algorithm for a manager $\mathcal{M}$ that, given a security parameter 1^k, size of identity string ℓ, and the maximum size n of receiver sets, outputs a public key *pkey* and a master key *mkey*. Here, *pkey* includes n, ℓ, k, and descriptions of shared key space $\mathcal{KS}$ and a ciphertext space $\mathcal{HS}$. *pkey* is published and *mkey* is given to only $\mathcal{M}$.

$$(\mathit{pkey}, \mathit{mkey}) \leftarrow \mathsf{Setup}(k, \ell, n)$$

We note that n determines only the maximum size of receiver sets to each of which ciphertexts are sent. While ℓ determines the number of potential receivers which is superpolynomial of ℓ if the scheme is identity-based.

KeyExt: A probabilistic key extraction algorithm for $\mathcal{M}$ that, given *mkey*, *pkey*, and an identity string $id \in \{0,1\}^\ell$ of receiver, outputs a private key skey_{id} for receiver of identity id

$$\mathit{skey}_{id} \leftarrow \mathsf{KeyExt}(\mathit{pkey}, \mathit{mkey}, id)$$

Enc: A probabilistic encryption algorithm for senders that, given a receiver set $\mathcal{S}$ and a public key *pkey*, outputs a shared key $\mathit{key} \in \mathcal{KS}$ and a header $\mathit{hdr} \in \mathcal{HS}$. Here, each element in $\mathcal{S}$ is in $\{0,1\}^\ell$ and $|\mathcal{S}| \leq n$.

$$(\mathit{key}, \mathit{hdr}) \leftarrow \mathsf{Enc}(\mathcal{S}, \mathit{pkey}).$$

Dec: A probabilistic decryption algorithm for receivers that, given a receiver set $\mathcal{S}$, a header *hdr* for a receiver set $\mathcal{S}$, $id \in \mathcal{S}$, skey_{id}, and the public key *pkey*, outputs *key*.

$$\mathit{key} \leftarrow \mathsf{Dec}(\mathcal{S}, id, \mathit{skey}_{id}, \mathit{hdr}, \mathit{pkey}).$$

2.2 Security

We introduce security requirements of identity-based broadcast encryption schemes.

Definition 1. *Consider the following* key distinguishing game *between a challenger $\mathcal{C}$ and a probabilistic polynomial-time adversary $\mathcal{A}$.*

1. *$\mathcal{C}$, that is given maximum number of receivers $n \in \mathbb{N}$, a security parameter k, a size of identity string ℓ, and a random tape, runs* Setup *to output pkey and mkey. pkey is given to $\mathcal{A}$ while mkey is kept secret.*
2. *$\mathcal{A}$ may adaptively sends the key extraction queries described bellow. During this phase, $\mathcal{A}$ may also send a challenge query described below for once and only once.*

   Key extraction query: *$\mathcal{A}$ sends $id \in \{0,1\}^\ell$ to $\mathcal{C}$. Then, $\mathcal{C}$ runs $\mathit{skey}_{id} \leftarrow \mathsf{KeyExt}(\mathit{pkey}, \mathit{mkey}, id)$ and returns skey_{id} to $\mathcal{A}$.*

Challenge query: *$\mathcal{A}$ sends $\mathcal{S}^*$ such that each element in $\mathcal{S}^*$ is in $\{0,1\}^\ell$ and $|\mathcal{S}^*| \leq n$. Then, $\mathcal{C}$ runs $(key_0, hdr^*) \leftarrow \mathsf{Enc}(\mathcal{S}^*, pkey)$ and randomly chooses $key_1 \in \mathcal{KS}$. Then, $\mathcal{C}$ randomly chooses $b \in \{0,1\}$ and returns (key_b, hdr^*)*

3. *After the above phase is over, $\mathcal{A}$ outputs $b' \in \{0,1\}$.*
4. *At the end of the game, $\mathcal{C}$ outputs $\perp$ if $\mathcal{A}$ asked key extraction query with respect to $id \in \mathcal{S}^*$, else output 1 if $b = b'$ and 0 if $b \neq b'$.*

Let $Adv_{\mathcal{A}}(k, \ell, n, b)$ be the probability that $\mathcal{C}$ outputs $b \in \{0,1\}$ after the game with $\mathcal{A}$, where the probability is taken over random tapes of $\mathcal{C}$ and $\mathcal{A}$.

We say an identity-based broadcast encryption scheme is key indistinguishable under adaptive key extraction attacks *if the*

$$|Adv_{\mathcal{A}}(k, \ell, n, 0) - Adv_{\mathcal{A}}(k, \ell, n, 1)|$$

is negligible in k for any probabilistic polynomial-time $\mathcal{A}$ in the above game.

3 Preliminaries

Our scheme leverages bilinear groups that has the following properties.

Definition 2. Bilinear groups

1. *$\mathcal{G}$ and $\mathcal{G}_T$ are cyclic groups of prime order p.*
2. *$e : \mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}_T$ is an efficient map such that:*
   - *(i) Bilinear: for all $u, v \in \mathcal{G}$, and $\alpha, \beta \in \mathbb{Z}/p\mathbb{Z}$, we have $e(u^\alpha, v^\beta) = e(u, v)^{\alpha\beta}$.*
   - *(ii) Non-degenerate: if $g \in \mathcal{G}$ is a generator of $\mathcal{G}$, then $e(g, g)$ is a generator of $\mathcal{G}_T$.*

We prove the security of our scheme in the generic bilinear group model. We here briefly review this model. The generic group model introduced in [22] is extended to the generic bilinear groups model of prime order in [5].

Definition 3. The generic bilinear group model: *Let us consider the case the bilinear groups of prime order p are $\mathcal{G}$ and $\mathcal{G}_T$ and g is a generator of $\mathcal{G}$. In this model, elements of $\mathcal{G}$ and $\mathcal{G}_T$ appear to be encoded as unique random strings, so that no property other than equality can be directly tested by the adversary. There exist three oracles in this model. Those are, oracles that perform group operations in each $\mathcal{G}$ and $\mathcal{G}_T$ and an oracle that performs paring e. The opaque encoding of an element in $\mathcal{G}$ is modeled as an injective function $\psi : \mathbb{Z}/p\mathbb{Z} \rightarrow \Sigma \subset \{0,1\}^*$, which maps all $\alpha \in \mathbb{Z}/p\mathbb{Z}$ to the string representation $\psi(g^\alpha)$ of $g^\alpha \in \mathcal{G}$. We similarly define $\psi_T : \mathbb{Z}/p\mathbb{Z} \rightarrow \Sigma_T$ for $\mathcal{G}_T$. The attacker communicates with the oracles using the (ψ, ψ_T)-representations of the group elements only.*

We define a “General Bilinear Decision Problem” in the generic bilinear group, which is a straight forward generalization of “General Diffie-Hellman Exponent problem” of [7] in the generic bilinear group model [5].

Definition 4. General Bilinear Decision Problem: *Let p be a prime and let s and m be two positive integer constants. Let $\mathcal{G}$ and $\mathcal{G}_T$ be order p cyclic groups with an efficient bilinear mapping $e : \mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}_T$ and g is a generator of $\mathcal{G}$ and $g_T = e(g, g)$. Let $P = (p_1, \ldots, p_s), Q = (q_1, \ldots, q_s), P' = (p'_1, \ldots, p'_s), Q' = (q'_1, \ldots, q'_s) \in \mathbb{F}_p[X_1, \ldots, X_m]^s$ and where $p_1 = q_1 = p'_1 = q'_1 = 1$. Let $P(x_1, \ldots, x_n)$ denote $(p_1(x_1, \ldots, x_m), \ldots, p_s(x_1, \ldots, x_m))$ and $[P(x_1, \ldots, x_m)]g = ([p_1(x_1, \ldots, x_m)]g, \ldots, [p_s(x_1, \ldots, x_m)]g)$. We use similar notation for Q, P', Q'.*

We say an algorithm $\mathcal{B}$ has an advantage ϵ solving general bilinear decision problem with respect to (P, Q) and (P', Q') *if*

$$\begin{aligned} &| \Pr[\mathcal{B}([P(x_1, \ldots, x_m)]g, [Q(x_1, \ldots, x_m)]g_T = 1] \\ &- \Pr[\mathcal{B}([P'(x_1, \ldots, x_m)]g, [Q'(x_1, \ldots, x_m)g_T) = 1]| > \epsilon \end{aligned}$$

where the probability is taken over random choice of $x_0, \ldots, x_m \in_R \mathbb{Z}/p\mathbb{Z}$ and random tapes of $\mathcal{B}$.

Definition 5. Dependent and Independent Polynomials: *We say (P, Q) and (P', Q') are* dependent *if there exists tuple of $s^2 + s$ constants $\{a_{ij}\}_{i=1,\ldots,s,j=1,\ldots,s}$, $\{b_i\}_{i=1,\ldots,s}$ such that either*

$$0 \equiv \sum_{i,j=1}^{s} a_{ij} p_i p_j + \sum_{i=1}^{s} b_i q_i \wedge 0 \not\equiv \sum_{i,j=1}^{s} a_{ij} p'_i p'_j + \sum_{i=1}^{s} b_i q'_i$$

or

$$0 \not\equiv \sum_{i,j=1}^{s} a_{ij} p_i p_j + \sum_{i=1}^{s} b_i q_i \wedge 0 \equiv \sum_{i,j=1}^{s} a_{ij} p'_i p'_j + \sum_{i=1}^{s} b_i q'_i$$

holds. We let $(P, Q) \not\sim (P', Q')$ denote this.

We say that a general bilinear decision problem with respect to (P, Q) and (P', Q') is independent *if (P, Q) and (P', Q') are not dependent. We let $(P, Q) \sim (P', Q')$ denote this.*

The general Diffie-Hellman Exponent problem in [7] is a special case of the above problem when each of $\{p_i = p'_i\}_{i=1,\ldots,s}$ is a polynomial of $x_1, \ldots, x_{m-1}$, $Q = (1, f(x_1, \ldots, x_{m-1}))$, and $Q' = (1, x_m)$.

Theorem 1. *Let $d_P, d_{P'}, d_Q$, and $d_{Q'}$ be, respectively, the maximum degree of polynomials in P, P', Q, and Q' and let $d = \max(2d_P, 2d_{P'}, d_Q, d_{Q'})$. In the generic bilinear group model, no algorithm $\mathcal{A}$ that makes a total of at most q_g queries to the oracles computing group operations in $\mathcal{G}$, $\mathcal{G}_T$, and $e : \mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}_T$ has an advantage ϵ in solving any of general bilinear decision problem with respect to (P, Q) and (P', Q') which are independent. Where,*

$$\epsilon = \frac{(q_g + 2s)^2 d}{p}.$$

The proof of the theorem is given in Appendix A.

4 Proposed Scheme

We now propose our identity-based broadcast encryption scheme. Let $\mathcal{H} : \{0,1\}^\ell \rightarrow \mathbb{Z}/p\mathbb{Z}$ be a cryptographic hash function. Our scheme is as in the following.

Setup: Given a $n, \ell, k \in \mathbb{N}$, Setup first chooses $(\mathcal{G}, \mathcal{G}_T, e)$ of size polynomial of k and let $\mathit{param} = (n, \ell, p, \mathcal{G}, \mathcal{G}_T, e)$. Then, it randomly chooses

$$\begin{aligned} &P, Q \in_R \mathcal{G} \\ &\omega, \sigma \in_R \mathbb{Z}/p\mathbb{Z}. \end{aligned}$$

Next, it generates

$$\begin{aligned} \tilde{P} &= [\omega]P \\ \tilde{Q} &= [\omega]Q \\ (P_i)_{i=1,\ldots,n} &= ([\sigma^i]P) \\ (Q_i)_{i=1,\ldots,n} &= ([\sigma^i]Q) \\ y &= e(\tilde{P}, Q). \end{aligned}$$

Finally, it outputs

$$\begin{aligned} \mathit{pkey} &= \left(\mathit{param}, y, \begin{pmatrix} & & P_1, & \ldots, & P_n \\ \tilde{Q}, & Q, & Q_1, & \ldots, & Q_n \end{pmatrix}\right) \\ \mathit{mkey} &= (\sigma, \tilde{P}) \end{aligned}$$

KeyExt: Given $id \in \{0,1\}^\ell$, pkey, and mkey, KeyExt generates

$$\begin{aligned} I_{id} &= \mathcal{H}(id) \\ \mathit{skey}_{id} = K_{id} &= \left[\frac{1}{\sigma + I_{id}}\right]\tilde{P} \end{aligned}$$

and then outputs skey_{id}

Enc: Given a set of identities $\mathcal{S}$ such that $|\mathcal{S}| \leq n$, Enc randomly chooses $\rho \in \mathbb{Z}/p\mathbb{Z}$ and generates

$$\begin{aligned} &\mathit{key} = K = y^\rho \\ &\mathit{hdr} = (H_1, H_2, \mathcal{S}) = ([\rho \prod_{id' \in \mathcal{S}} (\sigma + I_{id'})]Q, [\rho]\tilde{Q}, \mathcal{S}) \end{aligned}$$

where $I_{id'} = \mathcal{H}(id')$. Finally, it outputs $(\mathit{key}, \mathit{hdr})$.

Dec: Given $\mathit{pkey}, \mathit{hdr} = (H_1, H_2, \mathcal{S})$, and skey_{id} such that $id \in \mathcal{S}$, it generates

$$\begin{aligned} &v = e(K_{id}, H_1) \cdot e([\prod_{id' \in \mathcal{S}, id' \neq id} I_{id'}]P - \prod_{id' \in \mathcal{S}, id' \neq id} (\sigma + I_{id'})]P, H_2) \\ &K = v^{\prod_{id' \in \mathcal{S}, id' \neq id} I_{id'}{}^{-1}} \end{aligned}$$

Note that $[\rho\prod_{id'\in\mathcal{S}}(\sigma + I_{id'})]Q = \sum_{i=0}^{|\mathcal{S}|}([f_i][\sigma^i]Q) = [f_0]Q + \sum_{i=1}^{|\mathcal{S}|}([f_i]Q_i)$ holds for some $\{f_i \in \mathbb{Z}/p\mathbb{Z}\}_{i=0,\dots,|\mathcal{S}|}$. Here, $\{f_i \in \mathbb{Z}/p\mathbb{Z}\}_{i=0,\dots,|\mathcal{S}|}$ can be computed from ρ and $\{I_{id'}\}_{id'\in\mathcal{S}}$ with less than $|\mathcal{S}|^2$ multiplications and additions in $\mathbb{Z}/p\mathbb{Z}$ and H_1 can be computed from these $\{f_i\}_{i=0,\dots,|\mathcal{S}|}$ and $Q, \{Q_i\}_{i=1,\dots,|\mathcal{S}|}$ with $|\mathcal{S}|$ scalar multiplications and additions in $\mathcal{G}$.

We also note that $[\prod_{id'\in\mathcal{S},id'\neq id} I_{id'}]P - \prod_{id'\in\mathcal{S},id'\neq id}(\sigma+I_{id'})]P) = \sum_{i=1}^{|\mathcal{S}|-1}([f'_i]P_i)$ holds for some $\{f'_i \in \mathbb{Z}/p\mathbb{Z}\}_{i=1,\dots,|\mathcal{S}|-1}$. Here, we do not need P, which is not included in public key. Decryption also requires less than $|\mathcal{S}|^2$ multiplications and additions in $\mathbb{Z}/p\mathbb{Z}$ and $|\mathcal{S}|$ scalar multiplications and additions in $\mathcal{G}$.

5 Security of the Proposed Scheme

Theorem 2. *The proposed scheme is key-indistinguishable under adaptive key extraction attacks in the generic bilinear group model with random oracle.*

Proof. We first list elements that the adversary obtains from the game in the generic bilinear group model. We assume hash function $\mathcal{H}$ is now considered as a random oracle and suppose that the adversary asks at most q random oracle queries. If $\mathcal{A}$ asks a key extraction query with respect to a string $id \in \{0,1\}^\ell$ and it has never asked random oracle query with respect the query id, $\mathcal{C}$ asks the random oracle a query for this string. We count this random oracle query as the one asked by $\mathcal{A}$. We assume the response to i-th query of the random oracle is υ_i and let $\mathcal{U}$ be the set of integers $\{1,\dots,q\}$. We let $\prod_j$ and $\prod_{j\notin\mathcal{S}}$ denote $\prod_{j\in\mathcal{U}}$ and $\prod_{j\notin\mathcal{S}\wedge j\in\mathcal{U}}$ respectively. Similar notations are defined naturally.

Let $\alpha, \beta, \sigma, \omega, \rho'$ be variables and let us see what $\mathcal{A}$ obtains in the key distinguishing game in the generic bilinear group model.

Setup: Suppose G is a generator of $\mathcal{G}$. Let

$$P = [\alpha\prod_j(\sigma+\upsilon_j)]G \;\;,\;\; Q = [\beta\prod_j(\sigma+\upsilon_j)]G$$

$$\tilde{P} = [\omega\alpha\prod_j(\sigma+\upsilon_j)]G \;\;,\;\; \tilde{Q} = [\omega\beta\prod_j(\sigma+\upsilon_j)]G$$

$$P_i = ([\sigma^i\alpha\prod_j(\sigma+\upsilon_j)]G) \;\;,\;\; Q_i = ([\sigma^i\beta\prod_j(\sigma+\upsilon_j)]G)$$

$$y = e(G,G)^{\omega\alpha\beta\prod_j(\sigma+\upsilon_j)^2}.$$

Then, the data that $\mathcal{A}$ obtains from public key are ψ and ψ_T representation of the following polynomials. They are, polynomials

$$\alpha\sigma\prod_j(\sigma+\upsilon_j), \dots, \alpha\sigma^n\prod_j(\sigma+\upsilon_j)$$

$$\omega\beta\prod_j(\sigma+\upsilon_j), \beta\prod_j(\sigma+\upsilon_j), \beta\sigma\prod_j(\sigma+\upsilon_j), \dots, \alpha\sigma^n\prod_j(\sigma+\upsilon_j)$$

which are included in P, P' and a polynomial

$$\alpha\beta\prod_{j}(\sigma+\upsilon_j)^2$$

which is included in Q, Q'.

KeyExt: Since, $skey_{id} = K_{id} = \left[\frac{1}{\sigma+\mathcal{H}(id)}\right]\tilde{P}$, $\mathcal{A}$ obtains ψ representation of the polynomial

$$\omega\alpha\prod_{j\neq i}^{n}(\sigma+\upsilon_j)$$

when it asked a key extraction query with respect id. Here we assume that random oracle with respect to this string id is i-th query. Hence, this polynomial is included in P and P'.

Challenge: Valid encryptions with respect to $\mathcal{S}^*$ are of the form (key^*, hdr^*) such that

$$\begin{aligned} key^* &= K = y^{\rho'} \\ hdr^* &= (H_1, H_2, \mathcal{S}^*) = ([\rho' \prod_{id'\in\mathcal{S}^*}(\sigma + I_{id'})]Q, [\rho']\tilde{Q}, \mathcal{S}^*), \end{aligned}$$

Here, $I_{id} = \upsilon_i$ such that i-th random oracle query is with respect to this id. Hence, letting $\rho = \rho' \prod_{j\in\mathcal{S}^*}(\sigma+\upsilon_j)$, polynomials of which ψ, ψ_T representations $\mathcal{A}$ obtains are as in the following.

The distribution of challenge is that of valid ciphertext when $b = 0$, but that of key^* in it is independently distributed when $b = 1$. Therefore, polynomials

$$\rho\beta\prod_{j}(\sigma+\upsilon_j)\ ,\ \rho\omega\beta\prod_{j\notin\mathcal{S}^*}(\sigma+\upsilon_j)$$

are included in both P and P' and

$$\rho\omega\alpha\beta\prod_{j\notin\mathcal{S}^*}(\sigma+\upsilon_j)\prod_{j}(\sigma+\upsilon_j)$$

is included in only Q but not in Q'. Another random variable τ is included in Q' instead.

We assume no key extraction query with respect to id such that $id \in \mathcal{S}^*$ is asked. Here id is assumed to be i-th query to the random oracle.

Since $\mathcal{A}$ obtains ψ and ψ_T representations of (P, Q) when $b = 0$ and those of (P', Q'), what we are required to prove is the independence of (P, Q) and (P', Q').

We assume they are dependent and find a contradiction. Since τ is independent variable, we may assume that the following equation holds with respect to (P, Q) because of the dependence:

$$\sum_{i=1}^{s} c_i q_i \equiv \sum_{i,j=1}^{s} a_{ij} p_i p_j .$$

The equation should hold for each degree of polynomials. Let us see the terms that contains $\rho\omega\alpha\beta$. Then, there should be a set of coefficients c and $\{a_i\}_{i=1,\ldots,n}$ in$\mathbb{Z}/p\mathbb{Z}$ such that

$$c\rho\omega\alpha\beta \prod_{j\notin\mathcal{S}^*}(\sigma+\upsilon_j)\prod_j(\sigma+\upsilon_j) = \sum_{i\notin\mathcal{S}^*} a_i(\rho\beta\prod_j(\sigma+\upsilon_j)\cdot\omega\alpha\prod_{j\neq i}(\sigma+\upsilon_j))$$

holds for non zero c. Hence,

$$c\prod_{j\notin\mathcal{S}^*}(\sigma+\upsilon_j)\prod_j(\sigma+\upsilon_j) = \sum_{i\notin\mathcal{S}^*} a_i(\prod_j(\sigma+\upsilon_j)\cdot\prod_{j\neq i}(\sigma+\upsilon_j))$$

$$\Leftrightarrow$$

$$c\prod_j(\sigma+\upsilon_j) = \sum_{i\notin\mathcal{S}^*} a_i(\prod_{j\in\mathcal{S}^*}(\sigma+\upsilon_j)\cdot\prod_{j\neq i}(\sigma+\upsilon_j))$$

$$\Leftrightarrow$$

$$c\prod_{j\notin\mathcal{S}^*}(\sigma+\upsilon_j) = \sum_{i\notin\mathcal{S}^*} a_i(\prod_{j\in\mathcal{S}^*}(\sigma+\upsilon_j)\prod_{j\notin\mathcal{S}^*,j\neq i}(\sigma+\upsilon_j))$$

should holds.

Inserting each υ_k such that $k\notin\mathcal{S}^*$ into σ,

$$\begin{aligned}0 &= \sum_{i\notin\mathcal{S}^*} a_{2i}\prod_{j\in\mathcal{S}^*}(\upsilon_k+\upsilon_j)\prod_{j\notin\mathcal{S}^*,j\neq i}(\upsilon_k+\upsilon_j)\\ &= a_k\prod_{j\in\mathcal{S}^*}(\upsilon_k+\upsilon_j)\prod_{j\notin\mathcal{S}^*,j\neq k}(\upsilon_k+\upsilon_j)\end{aligned}$$

Hence, for all $k\notin\mathcal{S}^*$, $a_k = 0$ holds. This implies $c = 0$ and contradicts to the assumption of dependence. Hence, the independence is proved.

Now, counting the degrees and the number of polynomials, t-time adversary wins the key distinguishing game with the probability at most,

$$\epsilon \leq \frac{2(t+2n+6)(n+q+1)}{p}.$$

where we assumed $t > q$ and that t is a polynomial of k. Since p which is chosen to be superpolynomial of k, ϵ is negligible in k. Therefore, the theorem is proved.

6 Performance Analysis

Now we analyze the performance of our scheme. Many efficiency benchmarks for broadcast encryption schemes exist. They are, length of ciphertext, length of private key, length of public key, computational cost for encryption and decryption etc. Those values vary according to the size of receiver set, the number of potential users, and how much its receiver set has changed. We compare these efficiency with the scheme of Boneh et al. in [9].

Table 1. Comparison of schemes

	Our scheme	[10]
private key length	$\mathcal{O}(1)$	$\mathcal{O}(1)$
ciphertext length (excluding $\mathcal{S}$)	$\mathcal{O}(1)$	$\mathcal{O}(1)$
public key length	$\mathcal{O}(n)$	$\mathcal{O}(N = 2^\ell)$
cost for encryption (for the same receiver set)	$\mathcal{O}(1)$	$\mathcal{O}(1)$
cost for decryption (for the same receiver set)	$\mathcal{O}(1)$	$\mathcal{O}(1)$
# of multiplication in $\mathbb{Z}/p\mathbb{Z}$ or # of additions in $\mathcal{G}$	$\mathcal{O}(t^2)$	$\mathcal{O}(t)$
# of scalar multiplication in $\mathcal{G}$	$\mathcal{O}(t)$	–

In Table 1, n is the maximum size of receiver set $\mathcal{S}$. $N = 2^\ell$ is the maximum number of users, i.e., potential receivers. t is the number of receivers who is in either of previous receiver set or current receiver set but not in the other. As we can see in this table, our scheme has most of nice features of Boneh's scheme. Although our scheme requires more computational power when receiver sets are changed, its computational cost remains as small as the scheme is realistic in use.

The main problem of our scheme is that the size of $\mathcal{S}$ is proportional to the size of receiver set since its each element is of size ℓ. Hence, our scheme can not be in advantageous position when its receiver sets vary very often.

Our scheme is the most advantageous over the previous scheme, such as Boneh's scheme in [9], when the number of potential receivers is huge but the maximum size of its receiver set is rather small and receivers set does not change drastically on an average day. Example of such a case is when all Internet users are potential receivers and some of them form a group based on their identities and communicate each other.

7 Fully Secure Scheme, Removing Random Oracle and Generic Group Assumptions

From our scheme, we can construct a scheme that has key indistinguishability under chosen ciphertext attacks. This is possible if we use the technique of [12]. With the technique of [23] and random oracle, our scheme can also acquire key indistinguishability under chosen ciphertext attacks.

Although the security of our scheme is proved in the random oracle model. It was not essential. We used random oracle so that we are able to determine $\upsilon_i = \mathcal{H}(id)$ before the game starts. However, If we consider rational polynomial rather than polynomial for representing the values in $\mathcal{G}$ and $\mathcal{G}_T$ in the generic bilinear group, only essential property is that no υ_i and υ_j collude as long as i-th and j-th queries are different. Hence, if we use rational polynomial and collision resistant hash function, our scheme can be proved only in generic bilinear group model.

Proving the security outside of the generic bilinear group model is possible if we assume hardness of some problem in which adversaries are allowed to call oracles, such as one that returns $[\frac{1}{\sigma+\upsilon}]g$ when the adversary sent υ. Such an approach is adopted in many works such as LRSW assumption in [17, 11]. However, we could not see any essential improvement as long as underlying assumptions can be proved only in generic group model. Hence, we did not choose this kind of approach. We believe a proof based on mild assumptions in which adversaries are not allowed to call oracles, such as q-strong Diffie-Hellman assumptions has a significance, even if these assumptions are proved in generic group model. Constructing identity-based broadcast encryption whose security can be proved in the standard model (or only with mild assumptions) is still an open problem.

8 Conclusion

We have proposed the first truly identity-based broadcast encryption scheme which has most of desirable properties that the previous nice scheme has. The scheme is the most advantageous when the number of potential receivers is huge but the maximum size of its receiver set is rather small and receive sets do not change so often. Our scheme can also be an essential building block without no alternative for other protocols.

References

1. Jun Anzai, Natsume Matsuzaki, Tsutomu Matsumoto: A Quick Group Key Distribution Scheme with "Entity Revocation". ASIACRYPT 1999: 333-347
2. Nuttapong Attrapadung, Hideki Imai: Graph-Decomposition-Based Frameworks for Subset-Cover Broadcast Encryption and Efficient Instantiations. ASIACRYPT 2005: 100-120
3. Nuttapong Attrapadung, Kazukuni Kobara, Hideki Imai: Sequential Key Derivation Patterns for Broadcast Encryption and Key Predistribution Schemes. ASIACRYPT 2003: 374-391
4. Shimshon Berkovits: How To Broadcast A Secret. EUROCRYPT 1991: 535-541
5. Dan Boneh, Xavier Boyen: Short Signatures Without Random Oracles. EUROCRYPT 2004: 56-73
6. Dan Boneh, Xavier Boyen: Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. EUROCRYPT 2004: 223-238
7. Dan Boneh, Xavier Boyen, Eu-Jin Goh: Hierarchical Identity Based Encryption with Constant Size Ciphertext. EUROCRYPT 2005: 440-456
8. Dan Boneh, Matthew K. Franklin: An Efficient Public Key Traitor Tracing Scheme. CRYPTO 1999: 338-353
9. Dan Boneh, Craig Gentry, Brent Waters: Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys. CRYPTO 2005: 258-275
10. Dan Boneh, Brent Waters: A fully collusion resistant broadcast, trace, and revoke system. ACM Conference on Computer and Communications Security 2006: 211-220

11. Jan Camenisch, Anna Lysyanskaya: Signature Schemes and Anonymous Credentials from Bilinear Maps. CRYPTO 2004: 56-72
12. Ran Canetti, Shai Halevi, Jonathan Katz: Chosen-Ciphertext Security from Identity-Based Encryption. EUROCRYPT 2004: 207-222
13. Yevgeniy Dodis, Nelly Fazio: Public Key Trace and Revoke Scheme Secure against Adaptive Chosen Ciphertext Attack. Public Key Cryptography 2003: 100-115
14. Amos Fiat, Moni Naor: Broadcast Encryption. CRYPTO 1993: 480-491
15. Michael T. Goodrich, Jonathan Z. Sun, Roberto Tamassia: Efficient Tree-Based Revocation in Groups of Low-State Devices. CRYPTO 2004: 511-527
16. Dani Halevy, Adi Shamir: The LSD Broadcast Encryption Scheme. CRYPTO 2002: 47-60
17. Anna Lysyanskaya, Ronald L. Rivest, Amit Sahai, Stefan Wolf: Pseudonym Systems. Selected Areas in Cryptography 1999: 184-199
18. S. Mitsunari, R. Sakai, and M. Kasahara: A new Traitor tracing. IEICE Trans. Fundamentals, E85-A(2), pp.481-484, Feb. 2002
19. Moni Naor and Moti Yung: "Public-key Cryptosystems Provably Secure against Chosen Ciphertext Attacks." STOC 1990: 427-437
20. Dalit Naor, Moni Naor, Jeffery Lotspiech: Revocation and Tracing Schemes for Stateless Receivers. CRYPTO 2001: 41-62
21. Moni Naor, Benny Pinkas: Efficient Trace and Revoke Schemes. Financial Cryptography 2000: 1-20
22. Victor Shoup: Lower Bounds for Discrete Logarithms and Related Problems. EUROCRYPT 1997: 256-266
23. Victor Shoup and Rosario Gennaro, Securing Threshold Cryptosystems against Chosen Ciphertext Attack. EUROCRYPT 1998, pp.1-16
24. Douglas R. Stinson, Ruizhong Wei: Combinatorial Properties and Constructions of Traceability Schemes and Frameproof Codes. SIAM J. Discrete Math. 11(1): 41-53 (1998)
25. Douglas R. Stinson, Ruizhong Wei: Key Preassigned Traceability Schemes for Broadcast Encryption. Selected Areas in Cryptography 1998: 144-156

A Proof of Theorem 1

Proof. The proof is similar to that of Theorem A.2 in [7].

Consider an algorithm $\mathcal{B}$ that plays the following game with $\mathcal{A}$. Algorithm $\mathcal{B}$ maintains two lists of pairs,

$$L_P = \{(p_i, \psi_{P,i}) : i = 1, \ldots, \tau_P\}, L_Q = \{(q_i, \psi_{Q,i}) : i = 1, \ldots, \tau_Q\}$$

under the invariant that at step τ in the game, $\tau_P + \tau_Q = \tau + 2s$. Here, $p_i \in \mathbb{F}_p[X_1, \ldots, X_m]$ and $q_i \in \mathbb{F}_p[X_1, \ldots, X_m]$ are multi-variate polynomials. The $\psi_{P,i}$ and $\psi_{Q,i}$ are strings in $\{0,1\}^k$. The lists are initialized at step $\tau = 0$ by initializing $\tau_P = \tau_Q = s$. $\mathcal{B}$ chooses $b \in \{0,1\}$ at the beginning of the game. We set $p_1, \ldots, p_s$ in L_P and $q_1, \ldots, q_s$ in L_Q to be, respectively, the polynomials in P and Q if $b = 0$. We set them to be polynomials in P' and Q' if $b = 1$. Algorithm $\mathcal{B}$ completes the preparation of the lists L_P and L_Q by setting the ψ-strings associated with distinct polynomials to random strings in $\{0,1\}^k$.

We can assume that $\mathcal{A}$ makes oracle queries only on strings obtained from $\mathcal{B}$, since $\mathcal{B}$ can make the strings in $\mathcal{G}$ and $\mathcal{G}_T$ arbitrarily hard to guess by increasing k.

We note that $\mathcal{B}$ can easily determine the index i of any given string $\psi_{P,i} \in L_P$ and $\psi_{Q,i} \in L_Q$. $\mathcal{B}$ starts the game by providing $\mathcal{A}$ with the value of p and a tuple of strings

$$\{\psi_{P,i}\}_{i=1,\ldots,s}, \{\psi_{Q,i}\}_{i=1,\ldots,s}$$

meant to encode some tuple in $\mathcal{G}^s \times \mathcal{G}_T^s$. Algorithm $\mathcal{B}$ responds to $\mathcal{A}$'s oracle queries as follows.

Group operation in $\mathcal{G}, \mathcal{G}_T$. A query in $\mathcal{G}$ consists of two operands $\psi_{P,i}$ and $\psi_{P,j}$ with $1 \leq i, j \leq \tau_P$ and a selection bit indicating whether to multiply or divide the group elements. To answer, let $\tau_P' \leftarrow \tau_P + 1$. Perform the polynomial addition or subtraction $p_{\tau_P'} = p_i \pm p_j$ depending on whether multiplication or division is requested. If the result $p_{\tau_P'} = p_l$ for some $l \leq \tau_P$, then set $\psi_{P\tau_P'} = p_{Pl}$; otherwise, set $\psi_{P\tau_P'}$ to a new random string in $\{0,1\}^k \setminus \{\psi_{P1}, \ldots, \psi_{P\tau_P}\}$. Insert the pair $(p_{\tau_P'}, \psi_{P\tau_P'})$ into the list L_P and update the counter $\tau_P \leftarrow \tau_P'$. Algorithm $\mathcal{B}$ replies to $\mathcal{A}$ with the string $\psi_{P\tau_P}$. $\mathcal{G}_T$ queries are handled analogously, this time by working with the list L_Q and the counter τ_Q.

Bilinear pairing. A query of this type consists of two operands $\psi_{P,i}$ and $\psi_{P,j}$ with $1 \leq i, j \leq \tau_P$. To answer, let $\tau_Q' \leftarrow \tau_Q + 1$. Perform the polynomial multiplication $q_{\tau_Q'} = p_i \cdot p_j$. If the result $q_{\tau_Q'} = q_l$ for some $l \leq \tau_Q$, then set $\psi_{Q,\tau_Q'} = p_{Ql}$; otherwise, set $\psi_{Q,\tau_Q'}$ to a new random string in $\{0,1\}^k \setminus \{\psi_{Q,1}, \ldots, \psi_{Q,\tau_Q}\}$. Insert the pair $(q_{\tau_Q'}, \psi_{Q\tau_Q'})$ into the list L_Q and update the counter $\tau_Q \leftarrow \tau_Q'$. Algorithm $\mathcal{B}$ replies to $\mathcal{A}$ with the string ψ_{Q,τ_Q}.

After at most q_g queries, $\mathcal{A}$ terminates and returns a guess $b' \in \{0,1\}$. At this point $\mathcal{B}$ chooses random $\{x_1, \ldots, x_m\}$ For $i = 1, \ldots, m$, we set $X_i = x_i$. It follows that the simulation provided by $\mathcal{B}$ is perfect unless the chosen random values for the variables $X_1, \ldots, X_m$ result in an equality relation between intermediate values that is not an equality of polynomials. In other words, the simulation is perfect unless for some i, j one of the following holds:

1. $p_i(x_1, \ldots, x_m) - p_j(x_1, \ldots, x_m) = 0$, yet the polynomials p_i and p_j are not equal.
2. $q_i(x_1, \ldots, x_m) - q_j(x_1, \ldots, x_m) = 0$, yet the polynomials q_i and q_j are not equal.

Let fail be the event that one of these two conditions holds. When event fail occurs, then $\mathcal{B}$'s responses to $\mathcal{A}$'s queries deviate from the real oracle's responses when the input tuple is derived from the vector $(x_1, \ldots, x_m) \in \mathbb{F}_p^m$.

We first bound the probability that event fail occurs. We need to bound the probability that for some i, j we get $(p_i - p_j)(x_1, \ldots, x_m) = 0$ even though $p_i - p_j \neq 0$ or that $(q_i - q_j)(x_1, \ldots, x_m) = 0$ even though $q_i \; q_j \neq 0$. By construction, the maximum total degree of these polynomials is at most $d = \max(2d_P, 2d_{P'}, d_Q, d_{Q'})$. Therefore, for a given i, j the probability that a random assignment to $X_1, \ldots, X_n$ is a root of $q_i - q_j$ is at most d/p. The same holds for $p_i - p_j$. Since there are no more than $2(\frac{q+2s}{2})$ such pairs (p_i, p_j) and (q_i, q_j) in total, we have that

$$Pr[\mathsf{fail}] \leq (\frac{q+2s}{2})\frac{2d}{p} \leq (q+2s)^2 d/p.$$

If event fail does not occur, then $\mathcal{B}$'s simulation is perfect.

Since (P, Q) and (P', Q') are independent, the distributions $\mathcal{A}$ is given are exactly the same unless fail happens. Therefore, the theorem is proved.