Hindawi
Security and Communication Networks
Volume 2017, Article ID 1404279, 16 pages
https://doi.org/10.1155/2017/1404279

Research Article

Adaptive Security of Broadcast Encryption, Revisited

Bingxin Zhu, Puwen Wei, and Mingqiang Wang

Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China

Correspondence should be addressed to Puwen Wei; pwei@sdu.edu.cn and Mingqiang Wang; wangmingqiang@sdu.edu.cn

Received 21 February 2017; Accepted 27 April 2017; Published 3 July 2017

Academic Editor: Paolo D'Arco

We provide a strong security notion for broadcast encryption, called adaptive security in the multichallenge setting (MA-security), where the adversary can adaptively have access to the key generation oracle and the encryption oracle many times (multichallenge). The adversary specially can query for the challenge ciphertexts on different target user sets adaptively, which generalizes the attacks against broadcast encryptions in the real world setting. Our general result shows that the reduction of the adaptive secure broadcast encryption will lose a factor of q in the MA setting, where q is the maximum number of encryption queries. In order to construct tighter MA-secure broadcast encryptions, we investigate Gentry and Water's transformation and show that their transformation can preserve MA-security at the price of reduction loss on the advantage of the underlying symmetric key encryption. Furthermore, we remove the q-type assumption in Gentry and Water's semistatically secure broadcast encryption by using Hofheinz-Koch-Striecks techniques. The resulting scheme instantiated in a composite order group is MA-secure with constant-size ciphertext header.

1. Introduction

Broadcast encryptions (BE), introduced by Fiat and Naor [1], allow a sender to broadcast encrypted messages in such a way that only a specified group of users can decrypt the messages. Such schemes are useful in many applications, for example, pay-TV systems, internet multicasting of video and music, DVD content protection, file system access control, and wireless sensor networks [2]. One basic security requirement for broadcast encryption is the fully collusion resistance, which means that even a coalition of all users outside of target user set S learns nothing about the target plaintext. Naor et al. [3] proposed a fully collusion secure broadcast encryption scheme with the private key overhead $O(\log^2(n))$, where n is the total number of users. Subsequent works [4, 5] reduced the private key size to $O(\log n)$. However, the ciphertexts size of collusion resistant schemes, for example, [3–6], usually grows linearly with either the number of receivers or the number of revoked users. Boneh et al. [7] constructed a fully collusion secure broadcast encryption systems with low ciphertext overhead and short secret keys. But the security of their scheme was proven in a static model, where the adversary needs to choose the target user set before seeing the system parameter. To capture more powerful attacks, Gentry and Waters [8] provided a stronger security model, called adaptive security, where the adversary can compromise users' keys and choose the target user set adaptively. They showed a generic method to construct adaptively secure broadcast encryption scheme by transforming semistatically secure broadcast encryption scheme, while the underlying semistatically secure scheme in [8] is based on a q-type assumption, which is considered to be too strong. By introducing the dual system, Waters [9] presented a broadcast encryption scheme with ciphertext overhead of constant size, and the resulting scheme can be proven adaptively secure under static assumption (non-q-type assumption). Then, Boneh, Waters, and Zhandry [10] made use of multilinear maps to construct a broadcast encryption where ciphertext overhead, private key size, and public key size are all poly-logarithmic in n. Other works [11–15] focus on the improvements of broadcast encryptions with special functionalities, for example, identity-based BE, anonymous BE, and traitor-tracing BE. Recently, Wee [16] presented the first broadcast encryption scheme with constant-size ciphertext overhead, constant-size user secret keys, and linear-size public parameters under static assumptions, while the resulting scheme is proven secure under static security model.

It is worth noting that although adaptive security defined in [8] seems strong enough to capture the security of broadcast encryptions, attacks in the real world are more complex,

for example, the adversary may adaptively get multiple challenge ciphertexts instead of only one. Such attacks are described in the so-called multiuser, multichallenge setting. Bellare et al. [17] initiated the study of the formal security in the multiuser setting, which shows that one-user, one-ciphertext security implies security in the multiuser, multichallenge setting. But the reduction loss of the proof is $n_u \cdot n_c$, where n_u and n_c denote the number of users and the number of challenge ciphertexts per user, respectively. However, large reduction loss usually implies large cryptographic parameters, which leads to low efficiency in practice. Recent breakthrough was made by Hofheinz and Jager [18], which provided the first IND-CCA secure PKE in the multiuser/multichallenge setting and the security tightly relates to the decision linear assumption. Here, tight security means that the security loss is a constant. Hofheinz, Koch, and Striecks [19] extended Chen and Wee's proof technique [20] to the multiuser/multichallenge setting and provided an almost tightly secure identity-based encryption (IBE) in the same setting, where the security loss only relies on the security parameter instead of the number of queries or instances of the scheme. Hence, an extension of broadcast encryptions in the multiuser/multichallenge setting is natural. However, the problem of constructing tightly secure broadcast encryptions in the multiuser/multichallenge setting is more subtle.

Our Contribution. We define a stronger notion for broadcast encryption, called the adaptive security in the multichallenge setting (MA-security), where the adversary can not only adaptively have access to the key generation oracle and the encryption oracle many times (multichallenge) but also adaptively query for the challenge ciphertexts on different target user sets instead of only one target set as in previous security model. Since each target user set is actually the combination of different users chosen by the adversary adaptively, it is more challenging for the reduction algorithm to prepare the parameters of broadcast encryptions than that of ordinary PKE or IBE.

Our general result shows that the reduction of the adaptive secure broadcast encryption will lose a factor of q in the MA setting, where q is the maximum number of encryption queries. To achieve tighter MA-security, we investigate the following two methods. The first method is from Gentry and Waters transformation [8] mentioned above. By exploring the random self-reducibility of BDHE assumption, we show that their transformation still holds in terms of MA-security, but at the cost of reduction loss q on the advantage of underlying symmetric key encryption. We emphasize that the resulting broadcast encryption scheme's security depends on both the BDHE assumption and the security of the symmetric key encryption. The reduction loss on the underlying symmetric key encryption is q, while the reduction on BDHE is tight due to the random self-reducibility of BDHE assumption, which is not implied by the general result of [17]. To remove the BDHE assumption, our second method applies the Hofheinz-Koch-Striecks techniques [19] to Gentry-Waters' semistatic secure broadcast encryption. The resulting scheme is essentially the Hofheinz-Koch-Striecks IBE scheme instantiated in a composite order group, while the user's decryption key of broadcast encryption is expressed in a different way from that of [19]. Both methods can turn Gentry-Waters' semistatically secure broadcast encryption into a MA-secure one with constant-size ciphertext header.

Note that the public key size of both schemes is linear with the number of users. An interesting problem is how to reduce the public key size of a MA-secure broadcast encryption under standard assumptions while preserving constant ciphertext header size.

2. Preliminaries

Notations. Let $[1, n] := \{1, \ldots, n\}$, where $n \in \mathbb{N}$. For a finite set $\mathcal{S}$, we denote by $x \xleftarrow{R} \mathcal{S}$ the fact that x is picked uniformly at random from $\mathcal{S}$. S can be denoted as a binary string; that is, $S = s_1 \cdots s_n$, where $s_i \in \{0, 1\}$ for $i \in [1, n]$. We write vectors in bold font; for example, $\mathbf{K} = (K_0, \ldots, K_{2n})$ for a vector of length $2n + 1$. $\mathsf{SD}(X; Y)$ denotes the statistical distance of X and Y, where X and Y are random variables. We say X and Y are ε-close if $\mathsf{SD}(X; Y) \leq \varepsilon$.

2.1. Bilinear Map. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two groups of prime order p, and let g be a generator of $\mathbb{G}$. $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear map with the following properties.

(1) Bilinearity: for all $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.

(2) Nondegeneracy: $e(g, g) \neq 1$.

(3) Computability: there exists an efficient algorithm to compute $e(u, v)$, for any $u, v \in \mathbb{G}$.

2.2. Assumptions

Decisional BDHE Problem [8]. Let $(\mathbb{G}, \mathbb{G}_T, e, p)$ be the description of the group parameter which is the output of group generator $\mathcal{G}(\lambda)$, where λ is the security parameter. Choose $b \xleftarrow{R} \{0, 1\}$ and given $2n + 2$ elements

$$\left(g^s, g, g^a, g^{a^2}, \ldots, g^{a^n}, g^{a^{n+2}}, \ldots, g^{a^{2n}}\right) \in \mathbb{G}^{2n+1}, \quad Z \in \mathbb{G}_T, \tag{1}$$

where $a, s \xleftarrow{R} \mathbb{Z}_p^*$, $Z \leftarrow e(g^s, g^{a^{n+1}})$ if $b = 0$ and $Z \xleftarrow{R} \mathbb{G}_T$ if $b = 1$. The problem is to guess b.

The decisional BDHE assumption states that for any PPT adversary $\mathcal{A}$ which takes as inputs the description of $(\mathbb{G}, \mathbb{G}_T, e)$ and the above elements and outputs b^*, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}}^{\mathrm{BDHE}}(\lambda) := \left|\Pr\left[b = b^*\right] - \frac{1}{2}\right| \tag{2}$$

is negligible in λ.

2.3. Broadcast Encryption Systems. A broadcast encryption system consists of four randomized algorithms described below.

Table 1: MA experiment.

$\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda)$
$(\mathrm{PK}, \mathrm{SK}) \longleftarrow \mathsf{Setup}(n, \ell)$;
$b \xleftarrow{R} \{0,1\}$;
$b' \longleftarrow \mathscr{A}^{\mathcal{O}_{\mathrm{KeyGen}}(\cdot,\mathrm{SK}),\mathcal{O}_{\mathrm{Enc}}(\cdot,\mathrm{PK})}$;
If $b' = b$, return 1; otherwise, 0.

Table 2: MS experiment.

$\mathsf{Exp}^{\mathrm{MS}}_{\mathscr{A},\mathsf{BE}}(\lambda)$
$S^* \longleftarrow \mathscr{A}$;
$(\mathrm{PK}, \mathrm{SK}) \longleftarrow \mathsf{Setup}(n, \ell)$;
$b \xleftarrow{R} \{0,1\}$;
$b' \longleftarrow \mathscr{A}^{\mathcal{O}_{\mathrm{KeyGen}}(\cdot,\mathrm{SK}),\mathcal{O}_{\mathrm{Enc}}(\cdot,\mathrm{PK})}$;
If $b' = b$, return 1; otherwise, 0.

Setup(n, ℓ). Take as input the number of users n and the maximal size $\ell \leq n$ of a broadcast recipient group and output a public/secret key pair (PK, SK). (The security parameter λ is taken as parts of the input implicitly.)

KeyGen(i, SK). Take as input a user index $i \in [1, n]$ and the secret key SK and output a private key d_i.

Enc(S, PK). Take a user set $S \subseteq [1, n]$ and the public key PK as input. It outputs a pair (Hdr, K), where Hdr is the header and $K \in \mathscr{K}$ is the message encryption key from a key space $\mathscr{K}$.

Dec(S, i, d_i, Hdr, PK). Take as input a user set $S \subseteq [1, n]$, a user index $i \in [1, n]$, and the corresponding private key d_i for user i, a header Hdr, and the public key PK. If $i \in S$, then the algorithm outputs the message encryption key $K \in \mathscr{K}$.

3. Adaptive Security in the Multichallenge Setting (MA-Security)

In this section, we define the adaptive security of broadcast encryption in the multichallenge setting. Let $\mathsf{BE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a broadcast encryption scheme. The experiment for BE is described in Table 1.

During the experiment, the adversary $\mathscr{A}$ takes λ and the description of BE including PK as inputs and $\mathscr{A}$ can have access to the following two kinds of oracles.

(i) $\mathcal{O}_{\mathrm{KeyGen}}(\cdot, \mathrm{SK})$ is the secret key generation oracle which takes a user index i as input and outputs $\mathsf{KeyGen}(i, \mathrm{SK})$. Note that $\mathscr{A}$ cannot make i as the key generation query if $i \in S_j$, where S_j has been queried to the encryption oracle. Suppose the adversary can make q_{key} key generation queries at most.

(ii) $\mathcal{O}_{\mathrm{Enc}}(\cdot, \mathrm{PK})$ is the encryption oracle which takes S_j as input and outputs the challenge ciphertext $C_j = (Hdr^*_j, K_{b,j})$, where $(Hdr^*_j, K_{0,j}) \leftarrow \mathsf{Enc}(S_j, \mathrm{PK})$, $K_{1,j} \xleftarrow{R} \mathscr{K}$. The restriction on encryption query is that S_j can not include any user index i which has been queried to $\mathcal{O}_{\mathrm{KeyGen}}$. Suppose that the adversary $\mathscr{A}$ can only query encryption oracle $\mathcal{O}_{\mathrm{Enc}}(\cdot, \mathrm{PK})$ at most q_{Enc} times.

A broadcast encryption scheme BE is adaptively secure in the multichallenge setting (MA-secure) if, for any PPT adversary $\mathscr{A}$, the advantage

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) := \left| \Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1\right] - \frac{1}{2} \right| \tag{3}$$

is negligible in λ.

Remark 1. The main difference between our MA-security and the adaptive security defined in [8] is the encryption queries. In MA-security experiment, the adversary can not only adaptively have access to the encryption oracle many times but also query for the challenge ciphertexts on different target user sets, while the adversary can make only one encryption query for one target user set in adaptive security experiment $\mathsf{Exp}^{\mathrm{A}}_{\mathscr{A},\mathsf{BE}}$ [8], where the related advantage of $\mathsf{Exp}^{\mathrm{A}}_{\mathscr{A},\mathsf{BE}}$ is denoted as $\mathsf{Adv}^{\mathrm{A}}_{\mathscr{A},\mathsf{BE}}(\lambda) := |\Pr[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1] - 1/2|$.

To investigate Gentry and Waters transformation in the multichallenge setting, we also need to extend semistatic security defined in [8] to the multichallenge setting, which is called semistatic security in the multichallenge setting (MS-security). The MS-security is defined in a similar way as that of MA-security, where the adversary also takes λ and the description of BE including PK as inputs and can have access to $\mathcal{O}_{KeyGen}(\cdot, \mathrm{SK})$ and $\mathcal{O}_{\mathrm{Enc}}(\cdot, \mathrm{PK})$ as defined in MA-security. But additional restrictions in MS-security are that $\mathscr{A}$ has to choose a target user set S^* at the beginning of the experiment and encryption queries S_j are such that $S_j \subseteq S^*$. Details of MS experiment are shown in Table 2

A broadcast encryption scheme BE is semistatically secure in the multichallenge setting (MS-secure) if, for any PPT adversary $\mathscr{A}$, the advantage

$$\mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A},\mathsf{BE}} := \left| \Pr\left[\mathsf{Exp}^{\mathrm{MS}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1\right] - \frac{1}{2} \right| \tag{4}$$

is negligible in λ.

4. MA-Secure Broadcast Encryption

First we give a general result on the reduction loss of an adaptive secure broadcast encryption in the MA setting. Then, to derive a tighter reduction, we show how to extend Gentry-Waters transformation to the multichallenge setting and construct a concrete MA-secure broadcast encryption based on BDHE assumption.

4.1. General Construction

Theorem 2. *For any PPT adversary $\mathscr{A}$ which can make at most $q_{key} = q_{key}(\lambda)$ key generation queries and $q_{Enc} = q_{Enc}(\lambda)$ encryption queries with running time t', there exists an algorithm $\mathscr{B}$ with about the same running time as $\mathscr{A}$, such that*

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) \leq q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda). \tag{5}$$

Proof. The proof proceeds via the following games.

(i) Game_0: Game_0 is the real MA experiment except the following differences. When the adversary adaptively makes encryption query for set S_j, the challenger responds with $\mathsf{Enc}(S_j, \mathrm{PK}) = (Hdr_j, K_{0,j})$, where $j \in [1, q_{\mathrm{Enc}}]$.

(ii) Game_1: Game_1 is identical to Game_0 except that the challenger replies the encryption queries with $(Hdr_j, K_{1,j})$ for $j \in [1, q_{\mathrm{Enc}}]$, where $(Hdr_j, K_{0,j}) \leftarrow \mathsf{Enc}(S_j, \mathrm{PK}), K_{1,j} \xleftarrow{R} \mathscr{K}^*$ and $\mathscr{K}^*$ denotes the key space.

Now we construct a series of subgames $0, \iota$ for $\iota = 0, \ldots, q_{\mathrm{Enc}}$ to prove the indistinguishability between Game_0 and Game_1.

(i) $\mathsf{Game}_{0,1}$. $\mathsf{Game}_{0,1}$ is the same as Game_0 except that the challenger chooses $K_{1,1} \xleftarrow{R} \mathscr{K}^*$ to construct challenge $(Hdr_1, K_{1,1})$ for the first encryption query S_1.

(ii) $\mathsf{Game}_{0,\iota}$. $\mathsf{Game}_{0,\iota}$ is the same as $\mathsf{Game}_{0,\iota-1}$ except that the challenger chooses $K_{1,\iota} \xleftarrow{R} \mathscr{K}^*$ to construct challenge $(Hdr_\iota, K_{1,\iota})$ for the ιth encryption query S_ι, where $\iota \in [1, q_{\mathrm{Enc}}]$.

Let $\mathsf{Game}_{0,\iota} = 1$ denote the event that the adversary outputs 1 in $\mathsf{Game}_{0,\iota}$. Note that $\mathsf{Game}_{0,0}$ and $\mathsf{Game}_{0,q_{\mathrm{Enc}}}$ are identical to Game_0 and Game_1, respectively. Thus,

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) &:= \left|\Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1\right] - \frac{1}{2}\right| = \left|\frac{1}{2}\right. \\
&\cdot \Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 0 \mid b = 0\right] + \frac{1}{2} \\
&\left.\cdot \Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right] - \frac{1}{2}\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right] - \left(1\right.\right. \\
&\left.\left. - \Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 0 \mid b = 0\right]\right)\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right]\right. \\
&\left. - \Pr\left[\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) = 1 \mid b = 0\right]\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Game}_1 = 1\right] - \Pr\left[\mathsf{Game}_0 = 1\right]\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Game}_{0,q_{\mathrm{Enc}}} = 1\right] - \Pr\left[\mathsf{Game}_{0,0} = 1\right]\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Game}_{0,q_{\mathrm{Enc}}} = 1\right] - \Pr\left[\mathsf{Game}_{0,q_{\mathrm{Enc}}-1} = 1\right]\right. \\
&+ \Pr\left[\mathsf{Game}_{0,q_{\mathrm{Enc}}-1} = 1\right] - \Pr\left[\mathsf{Game}_{0,q_{\mathrm{Enc}}-2} = 1\right] \\
&\left. + \cdots + \Pr\left[\mathsf{Game}_{0,1} = 1\right] - \Pr\left[\mathsf{Game}_{0,0} = 1\right]\right| \le \frac{1}{2} \\
&\cdot \sum_{\iota=1}^{q_{\mathrm{Enc}}} \left|\Pr\left[\mathsf{Game}_{0,\iota} = 1\right] - \Pr\left[\mathsf{Game}_{0,\iota-1} = 1\right]\right|.
\end{aligned} \tag{6}$$

Next, we show that $|\Pr[\mathsf{Game}_{0,\iota} = 1] - \Pr[\mathsf{Game}_{0,\iota-1} = 1]|$ is negligible, for $\iota \in [1, q_{\mathrm{Enc}}]$. That is, if there exists a PPT adversary $\mathscr{A}_1$ which can distinguish the adjacent games for some ι, we can construct a PPT algorithm $\mathscr{B}$ which can break the adaptive security of the underlying scheme.

Claim ($\mathsf{Game}_{0,\iota-1}$ *to* $\mathsf{Game}_{0,\iota}$). For any PPT adversary $\mathscr{A}_1$ which can make at most $q_{\mathrm{key}} = q_{\mathrm{key}}(\lambda)$ key generation queries and $q_{\mathrm{Enc}} = q_{\mathrm{Enc}}(\lambda)$ encryption queries with running time t', there exists algorithm $\mathscr{B}$ with about the same running time as $\mathscr{A}_1$, such that

$$\begin{aligned}
&\frac{1}{2} \cdot \left|\Pr\left[\mathsf{Game}_{0,\iota} = 1\right] - \Pr\left[\mathsf{Game}_{0,\iota-1} = 1\right]\right| \\
&\quad = \mathsf{Adv}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda).
\end{aligned} \tag{7}$$

Proof. $\mathscr{B}$ simulates the experiment as follows.

(i) The challenger runs $\mathsf{Setup}(n, \ell)$ and sends PK to $\mathscr{B}$ which will send PK to $\mathscr{A}_1$.

(ii) $\mathscr{A}_1$ adaptively makes key generation queries for user index i.

(iii) $\mathscr{B}$ sends user index i to the challenger which runs $\mathsf{KeyGen}(i, \mathrm{SK})$ and sends back the secret key d_i for user i. Then $\mathscr{B}$ sends d_i to $\mathscr{A}_1$.

(iv) $\mathscr{A}_1$ adaptively makes encryption queries for S_j, where j denotes the jth query.

(a) If $j \in [1, \iota - 1]$, $\mathscr{B}$ runs $\mathsf{Enc}(S_j, \mathrm{PK}) = (Hdr_j, K_{0,j})$ and chooses $K_{1,j} \xleftarrow{R} \mathscr{K}^*$ and sends $(Hdr_j, K_{1,j})$ to $\mathscr{A}_1$.

(b) If $j \in [\iota + 1, q_{\mathrm{Enc}}]$, $\mathscr{B}$ runs $\mathsf{Enc}(S_j, \mathrm{PK}) = (Hdr_j, K_{0,j})$ and sends $(Hdr_j, K_{0,j})$ to $\mathscr{A}_1$.

(c) If $j = \iota$, $\mathscr{B}$ sends S_ι to the challenger which then chooses $b \xleftarrow{R} \{0, 1\}$ and sends back $C^*_\iota = (Hdr_\iota, K_{b,\iota})$ where $K_{1,\iota} \xleftarrow{R} \mathscr{K}^*, (Hdr_\iota, K_{0,\iota}) \leftarrow \mathsf{Enc}(S_\iota, \mathrm{PK})$. Next $\mathscr{B}$ sends C^*_ι to $\mathscr{A}_1$.

(v) $\mathscr{A}_1$ outputs b'. If $b' = b$, $\mathscr{B}$ outputs 1, otherwise, 0.

Observe that if $b = 1$, $\mathscr{A}_1$'s view is identical to that of $\mathsf{Game}_{0,\iota}$. Otherwise $\mathscr{A}_1$'s view is identical to that of $\mathsf{Game}_{0,\iota-1}$. Thus

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}} &:= \left|\Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 1\right] - \frac{1}{2}\right| = \left|\frac{1}{2}\right. \\
&\cdot \Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right] + \frac{1}{2} \\
&\left.\cdot \Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 0 \mid b = 0\right] - \frac{1}{2}\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right]\right. \\
&\left. - \left(1 - \Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 0 \mid b = 0\right]\right)\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 1 \mid b = 1\right]\right. \\
&\left. - \Pr\left[\mathsf{Exp}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda) = 1 \mid b = 0\right]\right| = \frac{1}{2} \\
&\cdot \left|\Pr\left[\mathsf{Game}_{0,\iota} = 1\right] - \Pr\left[\mathsf{Game}_{0,\iota-1} = 1\right]\right|.
\end{aligned} \tag{8}$$

Hence we have

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) \le q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{A}}_{\mathscr{B},\mathsf{BE}}(\lambda), \tag{9}$$

which completes the proof of Theorem 2. □

4.2. MS-Secure Broadcast Encryption Based on BDHE Assumption. To reduce the reduction loss, we investigate Gentry-Waters broadcast encryption [8] in the MA setting. First we briefly recall the semistatically secure broadcast encryption scheme in [8]. Let $\mathscr{G}(\lambda, n)$ be a PPT algorithm which takes as input the security parameter λ and the number of users n and generates the description of group parameter $(\mathbb{G}, \mathbb{G}_T, e, p)$, where $\mathbb{G}, \mathbb{G}_T$ denotes the group of prime order $p = p(\lambda, n)$ and e is the bilinear map.

Setup(n). $(\mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathscr{G}(\lambda, n)$, $g, h_1, \ldots, h_n \xleftarrow{R} \mathbb{G}^{n+1}$, where $g, h_1, \ldots, h_n$ are generators of $\mathbb{G}$ and $\alpha \xleftarrow{R} \mathbb{Z}_p$. Set

$$\mathrm{PK} = \left(\mathbb{G}, \mathbb{G}_T, e; g, e(g,g)^{\alpha}, h_1, \ldots, h_n\right), \tag{10}$$

$$\mathrm{SK} = g^{\alpha}. \tag{11}$$

Output (PK, SK).

KeyGen(i, SK). Choose $r_i \xleftarrow{R} \mathbb{Z}_p$ and output user i's private key

$$\begin{aligned} d_i &= (d_{i,0}, d_{i,1}, \ldots, d_{i,i-1}, d_{i,i}, d_{i,i+1}, \ldots, d_{i,n}) \\ &= \left(g^{-r_i}, h_1^{r_i}, \ldots, h_{i-1}^{r_i}, g^{\alpha} h_i^{r_i}, h_{i+1}^{r_i}, \ldots, h_n^{r_i}\right). \end{aligned} \tag{12}$$

Enc(S, PK). Choose $t \xleftarrow{R} \mathbb{Z}_p$ and compute $C_1 = g^t, C_2 = (\prod_{j \in S} h_j)^t$. Set

$$Hdr = \left(g^t, \left(\prod_{j \in S} h_j\right)^t\right), \tag{13}$$

$$K = e(g,g)^{\alpha \cdot t}. \tag{14}$$

Output (Hdr, K).

Dec(S, i, d_i, Hdr, PK). If $i \in S$, parse d_i as $(d_{i,0}, \ldots, d_{i,n})$ and Hdr as (C_1, C_2) and output

$$K = e\left(d_{i,i} \cdot \prod_{j \in S \setminus \{i\}} d_{i,j}, C_1\right) \cdot e(d_{i,0}, C_2). \tag{15}$$

Theorem 3. *For any PPT adversary $\mathscr{A}$ which can make at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries with running time t', there exists an algorithm $\mathscr{B}$ with about the same running time as $\mathscr{A}$, such that*

$$\mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A},\mathsf{BE}}(\lambda) = \mathsf{Adv}^{\mathrm{BDHE}}_{\mathscr{G},\mathscr{B}}(\lambda). \tag{16}$$

The proof is similar to that of [8] except that we have to deal with multiple challenges in the simulation. Furthermore, to derive a tighter reduction, we need the following lemma which makes use of the random self-reducibility of BDHE.

Lemma 4. *There exists an efficient algorithm that takes as input $(g^s, g, g^a, g^{a^2}, \ldots, g^{a^n}, g^{a^{n+2}}, \ldots, g^{a^{2n}}, Z)$ for $Z = e(g^{a^n}, g^c)$ and generates many tuples of the form*

$$\left(g^{s_j}, g, g^a, g^{a^2}, \ldots, g^{a^n}, g^{a^{n+2}}, \ldots, g^{a^{2n}}, Z_j\right), \tag{17}$$

where $Z_j = e(g^{a^n}, g^{c_j})$ and $s_j, c_j \in \mathbb{Z}_p$.

Proof. Compute $g^{s_j} = (g^s)^{v_j} \cdot g^{u_j}$ and $Z_j = Z^{v_j} \cdot e(g^{a^n}, g^a)^{u_j}$, where $u_j, v_j \xleftarrow{R} \mathbb{Z}_p$. Let $e = c - as \bmod p$. We implicitly set

$$\begin{aligned} s_j &= s \cdot v_j + u_j \bmod p, \\ c_j &= a s_j + e v_j \bmod p. \end{aligned} \tag{18}$$

Hence, we have

$$Z_j = \begin{cases} e\left(g^{a^n}, g^{a s_j}\right) & \text{if } c = as \bmod p \\ e\left(g^{a^n}, g^{a s_j + e v_j}\right) & \text{if } c \ne as \bmod p. \end{cases} \tag{19}$$

If $e = 0$, namely, $c = as$, then $c_j = as_j$. If $e \ne 0$, namely, $c \ne as$, then $c_j = as_j + ev_j$. Since ev_j are uniformly distributed, we have c_j uniformly distributed over $\mathbb{Z}_p$. □

Next, more details of the concrete proof of Theorem 3 can be found in Appendix A.

4.3. Transforming MS-Security to MA-Security. In this section, we show that Gentry-Waters transformation still holds in the multichallenge setting, but at the cost of reduction loss q_{Enc} in the advantage of underlying symmetric encryption scheme. First, we briefly recall Gentry-Waters transformation [8]. Let $\mathsf{BE}_{\mathrm{MS}} = (\mathsf{Setup}_{\mathrm{MS}}, \mathsf{KeyGen}_{\mathrm{MS}}, \mathsf{Enc}_{\mathrm{MS}}, \mathsf{Dec}_{\mathrm{MS}})$ be a MS-secure broadcast system and $\Pi_{\mathrm{sym}} = (\mathrm{SymSetup}, \mathrm{SymEnc}, \mathrm{SymDec})$ be a symmetric encryption scheme with key space $\mathscr{K}'$.

Setup(n). Run $\mathsf{Setup}_{\mathrm{MS}}(2n) \to (PK', SK')$. Let $s \xleftarrow{R} \{0,1\}^n$ and s_i denotes ith bit of s. Let $\mathrm{PK} = PK'$ and $\mathrm{SK} = (SK', s)$. Output (PK, SK).

KeyGen(i, SK). Run $\mathsf{KeyGen}_{\mathrm{MS}}(2i - s_i, SK') \to d_i'$. Set $d_i = (d_i', s_i)$. Output private key d_i.

Table 3: IND-CPA experiment.

$\mathsf{Exp}^{\mathrm{CPA}}_{\mathscr{A},\Pi_{\mathrm{sym}}}(\lambda)$
$b \xleftarrow{R} \{0,1\}$;
$k \longleftarrow \mathrm{SymSetup}\left(1^{\lambda}\right)$;
$b' \longleftarrow \mathscr{A}^{\mathscr{O}'_{\mathrm{Enc}}(k,\cdot)}$;
If $b' = b$, return 1; otherwise, 0.

$Enc(S, PK)$. Generate random $|S|$ bits: $t \leftarrow \{t_i \xleftarrow{R} \{0,1\} : i \in S\}$ and $K \xleftarrow{R} \mathscr{K}$. Set

$$\begin{aligned} S_0 &\longleftarrow \{2i - t_i : i \in S\}, \\ \mathsf{Enc}_{\mathrm{MS}}\left(S_0, PK'\right) &\longrightarrow \left(Hdr_0, \kappa_0\right), \\ \mathrm{SymEnc}\left(\kappa_0, K\right) &\longrightarrow C_0; \\ S_1 &\longleftarrow \{2i - (1 - t_i) : i \in S\}, \\ \mathsf{Enc}_{\mathrm{MS}}\left(S_1, PK'\right) &\longrightarrow \left(Hdr_1, \kappa_1\right), \\ \mathrm{SymEnc}\left(\kappa_1, K\right) &\longrightarrow C_1; \\ Hdr &\longleftarrow \left(Hdr_0, Hdr_1, C_0, C_1, t\right). \end{aligned} \tag{20}$$

Output (Hdr, K).

$Dec(S, i, d_i, Hdr, PK)$. Parse Hdr as $(Hdr_0, Hdr_1, C_0, C_1, t)$ and d_i as (d'_i, s_i). Set S_0 and S_1 as above. Run

$$\begin{aligned} \kappa_{s_i \oplus t_i} &\longleftarrow \mathsf{Dec}_{\mathrm{MS}}\left(S_{s_i \oplus t_i}, i, d'_i, Hdr_{s_i \oplus t_i}, PK'\right), \\ K &\longleftarrow \mathrm{SymDec}\left(\kappa_{s_i \oplus t_i}, C_{s_i \oplus t_i}\right). \end{aligned} \tag{21}$$

Output K.

Theorem 5. *For any PPT adversary $\mathscr{A}$ which can make at most $q_{key} = q_{key}(\lambda)$ key generation queries and $q_{Enc} = q_{Enc}(\lambda)$ encryption queries with running time t', there exist algorithms $\mathscr{A}_1, \mathscr{A}_2, \mathscr{A}_3$, and $\mathscr{A}_4$, each with about the same running time as $\mathscr{A}$, such that*

$$\begin{aligned} \mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE},n}(\lambda) \le\ & \mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A}_1,\mathsf{BE},2n}(\lambda) + \mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A}_2,\mathsf{BE},2n}(\lambda) \\ & + q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathscr{A}_3,\Pi_{\mathrm{sym}}}(\lambda) + q_{\mathrm{Enc}} \\ & \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathscr{A}_4,\Pi_{\mathrm{sym}}}(\lambda). \end{aligned} \tag{22}$$

Notice that $\mathsf{Adv}^{\mathrm{CPA}}_{\mathscr{B},\Pi_{\mathrm{sym}}}(\lambda)$ denotes the advantage of $\Pi_{\mathrm{sym}} = (\mathrm{SymSetup}, \mathrm{SymEnc}, \mathrm{SymDec})$, which is defined by the following one-time symmetric key IND-CPA experiment described in Table 3.

During the experiment, $\mathscr{A}$ takes the security parameter and the description of Π_{sym} as input and can make only one encryption query to encryption oracle $\mathscr{O}'_{\mathrm{Enc}}(k,\cdot)$. More precisely, A chooses a pair of plaintexts (m_0, m_1) of the same length as the query and $\mathscr{O}'_{\mathrm{Enc}}(k,\cdot)$ returns $\mathrm{SymEnc}(k, m_b)$ as the challenge ciphertext.

We say the symmetric key encryption scheme Π_{sym} is one-time CPA-secure if, for any PPT adversary $\mathscr{A}$, the advantage

$$\mathsf{Adv}^{\mathrm{CPA}}_{\mathscr{A},\Pi_{\mathrm{sym}}} := \left|\Pr\left[\mathsf{Exp}^{\mathrm{CPA}}_{\mathscr{A},\Pi_{\mathrm{sym}}}(\lambda) = 1\right] - \frac{1}{2}\right| \tag{23}$$

is negligible in λ, where the probability is taken over the random coins used in the experiment, as well as the random coins used by $\mathscr{A}$.

Proof of Sketch. The main idea of the proof is similar to that of [8] except that we need to deal with multiple challenges, which incurs a reduction loss in the advantage of symmetric key encryption scheme. More precisely, we need to prove the indistinguishability of the following games.

(i) Game_0 is identical to $\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda)$.

(ii) Game_1 is the same as Game_0 except that for each encryption query the challenger chooses $\kappa_{0,j} \xleftarrow{R} \mathscr{K}'$ to construct $C_{0,j} = \mathrm{SymEnc}(\kappa_{0,j}, K_{0,j})$, where $K_{0,j} = K_{1,j}$.

(iii) Game_2 is the same as Game_1 except that for each encryption query the challenger chooses $\kappa_{1,j} \xleftarrow{R} \mathscr{K}'$ to construct $C_{1,j} = \mathrm{SymEnc}(\kappa_{1,j}, K_{1,j})$, where $K_{0,j} = K_{1,j}$.

(iv) Game_3 is the same as Game_2 except that the challenger chooses $K_{0,j} \xleftarrow{R} \mathscr{K}$ to construct $C_{0,j} = \mathrm{SymEnc}(\kappa_{0,j}, K_{0,j})$.

(v) Game_4 is the same as Game_3 except that the challenger chooses $K_{1,j} \xleftarrow{R} \mathscr{K}$ to construct $C_{1,j} = \mathrm{SymEnc}(\kappa_{1,j}, K_{1,j})$.

The indistinguishability among Game_0, Game_1, and Game_2 relies on the MS-security of BE. By using hybrid arguments, we show the indistinguishability between Game_2 and Game_3 (Game_3 and Game_4), which relies on the one-time CPA security of the underlying symmetric key encryption. It is easy to check that the adversary has no advantage in Game_4. More details are shown in Appendix B.

5. Remove q-Type Assumption

In this section, we show how to remove the q-type assumption of the MS-secure Gentry-Waters scheme in Section 4 by using Hofheinz-Koch-Striecks techniques [19], where the original Gentry-Waters scheme is lifted to composite order groups.

Let $\mathscr{G}(\lambda, 4)$ be a composite-order group generator which generates group parameters $(G, G_T, N, e, g, g_1, g_2, g_3, g_4)$, where $e : G \times G \to G_T$ is a nondegenerate bilinear map and G, G_T are cyclic groups of order N and N is the product of different primes p_1, p_2, p_3, and p_4, and let g be the generator of group G and g_i, for $i \in [1,4]$, be the

random generators of subgroups $G_{p_1}, G_{p_2}, G_{p_3}, G_{p_4}$ of orders p_1, p_2, p_3, p_4, respectively.

Let $\mathcal{UH}$ be a family of universal hash functions $\mathsf{H} : G_T \rightarrow \{0,1\}^\tau$ with the property that for any nontrivial subgroup $G_T' \subseteq G_T$ and for $\mathsf{H} \leftarrow \mathcal{UH}, X \leftarrow G_T'$ and $U \leftarrow \{0,1\}^\tau$, we have $\mathsf{SD}((\mathsf{H}, \mathsf{H}(X)), (\mathsf{H}, U)) = \mathbf{O}(2^{-\tau})$. In addition, the resulting scheme relies on the following assumptions [19].

Dual System Assumption 1 (DS1). For any PPT adversary $\mathcal{A}$, the advantage function

$$\begin{aligned}&\mathsf{Adv}_{\mathcal{A}}^{\mathrm{DS1}}(\lambda)\\ &\quad := \left|\Pr\left[\mathcal{A}(D, T_0) = 1\right] - \Pr\left[\mathcal{A}(D, T_1) = 1\right]\right|\end{aligned} \tag{24}$$

is negligible in λ, where

$$\begin{aligned}&\left(G, G_T, N, e, g, (g_i)_i\right) \longleftarrow \mathcal{G}(\lambda, 4);\\ &\qquad D := \left(G, G_T, N, e, g, g_1, g_3, g_4\right);\\ &\qquad T_0 \xleftarrow{R} G_{p_1},\\ &\qquad T_1 \xleftarrow{R} G_{p_1 p_2}.\end{aligned} \tag{25}$$

Dual System Assumption 2 (DS2). For any PPT adversary $\mathcal{A}$, the advantage function

$$\begin{aligned}&\mathsf{Adv}_{\mathcal{A}}^{\mathrm{DS2}}(\lambda)\\ &\quad := \left|\Pr\left[\mathcal{A}(D, T_0) = 1\right] - \Pr\left[\mathcal{A}(D, T_1) = 1\right]\right|\end{aligned} \tag{26}$$

is negligible in λ, where

$$\begin{aligned}&\left(G, G_T, N, e, g, (g_i)_i\right) \longleftarrow \mathcal{G}(\lambda, 4);\\ &D := \left(G, G_T, N, e, g, g_1, g_4, g_{\{1,2\}}, g_{\{2,3\}}\right);\\ &g_{\{1,2\}} \xleftarrow{R} G_{p_1 p_2},\\ &g_{\{2,3\}} \xleftarrow{R} G_{p_2 p_3};\\ &T_0 \xleftarrow{R} G_{p_1 p_2},\\ &T_1 \xleftarrow{R} G_{p_1 p_3}.\end{aligned} \tag{27}$$

Dual System Assumption 3 (DS3). For any PPT adversary $\mathcal{A}$, the advantage function

$$\begin{aligned}&\mathsf{Adv}_{\mathcal{A}}^{\mathrm{DS3}}(\lambda)\\ &\quad := \left|\Pr\left[\mathcal{A}\left(D, \widehat{T}_0, \widetilde{T}_0\right) = 1\right] - \Pr\left[\mathcal{A}\left(D, \widehat{T}_1, \widetilde{T}_1\right) = 1\right]\right|\end{aligned} \tag{28}$$

is negligible in λ, where

$$\begin{aligned}&\left(G, G_T, N, e, g, (g_i)_i\right) \longleftarrow \mathcal{G}(\lambda, 4);\\ &D := \left(G, G_T, N, e, g, (g_i)_i, g_2^x \widehat{X}_4, g_2^y \widehat{Y}_4, g_3^x \widetilde{X}_4, g_3^y \widetilde{Y}_4\right);\\ &\widehat{X}_4, \widetilde{X}_4, \widehat{Y}_4, \widetilde{Y}_4 \xleftarrow{R} G_{p_4},\\ &x, y \longleftarrow \mathbb{Z}_N^*, \gamma' \longleftarrow \mathbb{Z}_N^*;\\ &\widehat{T}_0 = g_2^{xy},\\ &\widehat{T}_1 = g_2^{xy+\gamma'};\\ &\widetilde{T}_0 = g_3^{xy},\\ &\widetilde{T}_1 = g_3^{xy+\gamma'}.\end{aligned} \tag{29}$$

Dual System Bilinear DDH Assumption (DS-BDDH). For any PPT adversary $\mathcal{A}$, the advantage function

$$\begin{aligned}&\mathsf{Adv}_{\mathcal{A}}^{\text{DS-BDDH}}(\lambda)\\ &\quad := \left|\Pr\left[\mathcal{A}(D, T_0) = 1\right] - \Pr\left[\mathcal{A}(D, T_1) = 1\right]\right|\end{aligned} \tag{30}$$

is negligible in λ, where

$$\begin{aligned}&\left(G, G_T, N, e, g, (g_i)_i\right) \longleftarrow \mathcal{G}(\lambda, 4);\\ &D := \left(G, G_T, N, e, g, (g_i)_i, g_1^a, g_2^a, g_2^b, g_{\{2,4\}}, g_{\{2,4\}}^b, g_{\{2,4\}}^c\right);\\ &T_0 = e(g_2, g_2)^{abc},\\ &T_1 = e(g_2, g_2)^z;\\ &\qquad g_{\{2,4\}} \xleftarrow{R} G_{p_2 p_4},\ a, b, c, z \longleftarrow \mathbb{Z}_N^*.\end{aligned} \tag{31}$$

5.1. Construction

Setup(n, n). Generate $\alpha, \omega_1, \ldots, \omega_{2n} \xleftarrow{R} \mathbb{Z}_N$ and compute $(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}) \in G_{p_1}^{2n+1}$. Set $h \xleftarrow{R} G_N$, generate $(R_{4,1}, \ldots, R_{4,2n}) \xleftarrow{R} G_{p_4}^{2n}$, and compute $(h_1, \ldots, h_{2n}) = (h^{\omega_1} \cdot R_{4,1}, \ldots, h^{\omega_{2n}} \cdot R_{4,2n})$. Set

$$\begin{aligned}&\mathsf{mpk}\\ &\quad = \left(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}, g_4, h, e(g_1, h)^\alpha, h_1, \ldots, h_{2n}\right),\end{aligned} \tag{32}$$

$$\mathsf{msk} = h^\alpha. \tag{33}$$

Output $(\mathsf{mpk}, \mathsf{msk})$.

KeyGen(u, msk). Take an index $u \in [1, n]$ and the master key msk as input. Set $r_u \xleftarrow{R} \mathbb{Z}_N$, generate $(R_{4,1}', \ldots, R_{4,2n}') \xleftarrow{R} G_{p_4}^{2n}$, and compute $K_0 = h^{r_u}$, $K_1 = h_1^{r_u} \cdot R_{4,1}', \ldots, K_{2n} = h_{2n}^{r_u} \cdot R_{4,2n}'$

and output a user secret key

$$sk_u = \left(K_0, K_1, \ldots, \mathsf{msk} \cdot K_{2u-1}, K_{2u+1}, \ldots, K_{2n}\right) = \left(h^{r_u}, h_1^{r_u} \cdot R'_{4,1}, \ldots, \mathsf{msk} \cdot h_{2u-1}^{r_u} \cdot R'_{4,2u-1}, h_{2u+1}^{r_u} \cdot R'_{4,2u+1}, \ldots, h_{2n}^{r_u} \cdot R'_{4,2n}\right). \quad (34)$$

Note that K_{2u} is not used in sk_u.

$Enc(S, \mathsf{mpk})$. Take a set $S \subseteq [1, n]$ as well as a master public key mpk as input. We denote S as a binary string; that is, $S = s_1 \cdots s_n$, where $s_u \in \{0, 1\}$ and $u \in [1, n]$. That is, $s_u = 1$ if user u is in S. Otherwise, $s_u = 0$. Generate $t \xleftarrow{R} \mathbb{Z}_N$ and output

$$Hdr = (C_0, C_1) = \left(g_1^t, \left(\prod_{j=1}^{n} g_1^{\omega_{2j-s_j}}\right)^t\right), \quad (35)$$

$$K = \mathsf{H}\left(e\left(g_1, h\right)^{\alpha t}\right). \quad (36)$$

$Dec(S, u, sk_u, Hdr, \mathsf{mpk})$. If $u \in S$, parse sk_u as $(K_0, K_1, \ldots, \mathsf{msk} \cdot K_{2u-1}, K_{2u+1}, \ldots, K_{2n})$ and Hdr as (C_0, C_1) and output

$$K = \mathsf{H}\left(\frac{e\left(C_0, \mathsf{msk} \cdot \prod_{j=1}^{n} K_{2j-s_j}\right)}{e\left(C_1, K_0\right)}\right). \quad (37)$$

Correctness.

$$\begin{aligned} &\frac{e\left(C_0, \mathsf{msk} \cdot \prod_{j=1}^{n} K_{2j-s_j}\right)}{e\left(C_1, K_0\right)} \\ &= \frac{e\left(g_1^t, \mathsf{msk} \cdot \prod_{j=1}^{n} h_{2j-s_j}^{r_u} \cdot R'_{4,2j-s_j}\right)}{e\left(\prod_{j=1}^{n} g_1^{\omega_{2j-s_j}}, h\right)^{t \cdot r_u}} \\ &= \frac{e\left(g_1^t, h^{\alpha}\right) \cdot e\left(g_1, \prod_{j=1}^{n} h_{2j-s_j}\right)^{t \cdot r_u}}{e\left(\prod_{j=1}^{n} g_1^{\omega_{2j-s_j}}, h\right)^{t \cdot r_u}} \\ &= \frac{e\left(g_1, h\right)^{\alpha t} \cdot e\left(g_1, \prod_{j=1}^{n} h^{\omega_{2j-s_j}}\right)^{t \cdot r_u}}{e\left(\prod_{j=1}^{n} g_1^{\omega_{2j-s_j}}, h\right)^{t \cdot r_u}} = e\left(g_1, h\right)^{\alpha t}. \end{aligned} \quad (38)$$

5.2. Security Proof

Theorem 6. *For any PPT adversary $\mathcal{A}$ which can make at most $q_{key} = q_{key}(\lambda)$ key generation queries and $q_{Enc} = q_{Enc}(\lambda)$ encryption queries with running time t', there exist algorithm $\mathcal{B}_1$ on DS1, $\mathcal{B}_2$ on DS2, $\mathcal{B}_3$ on DS3, and $\mathcal{B}_4$ on DS-BDDH with running time $t' + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, respectively, for some constant $c \in \mathbb{N}$, such that*

$$\mathsf{Adv}_{\mathcal{A},\mathsf{BE}}^{\mathrm{MA}}(\lambda) \le \mathsf{Adv}_{\mathcal{B}_1}^{\mathrm{DS1}}(\lambda) + 2n \cdot \mathsf{Adv}_{\mathcal{B}_2}^{\mathrm{DS2}}(\lambda) + n \cdot \mathsf{Adv}_{\mathcal{B}_3}^{\mathrm{DS3}}(\lambda) + \mathsf{Adv}_{\mathcal{B}_4}^{\text{DS-BDDH}} + q_{\mathrm{Enc}} \cdot \mathbf{O}\left(2^{-\tau}\right). \quad (39)$$

The proof follows that of [19] and proceeds via a series of games described in Appendix C, where the user set S is considered as a special kind of identity id.

The main difference between games is presented in Table 4. Random function families, auxiliary secret key generation, auxiliary encryption function, semifunctional user secret keys, pseudo-normal ciphertexts, and semifunctional ciphertexts are defined as follows. More details can be found in Table 4.

(i) Random Function Families. In our scheme each user index $u \in [1, n]$ is interpreted as n-bit binary string $u = 0 \cdots 010 \cdots 0$, where only u-th bit is 1. Both user index u and user set S are denoted as identity id. Let $id|_i := id_1 \cdots id_i$ be i-bit prefix of id and denote $\mathrm{ID}|_i := \{0, 1\}^i$. For $i \in [0, n]$, define two random functions as follows.

(a) $\widehat{\mathsf{RF}}_i(id|_i) : \mathrm{ID}|_i \to G_{p_2} G_{p_4}; id|_i \mapsto (g_2 g_4)^{\widehat{\gamma}_i(id|_i)}$;

$\widehat{\gamma}_i(id|_i) : \mathrm{ID}|_i \to \mathbb{Z}_{p_2 p_4}^*; id|_i \mapsto \widehat{\gamma}_i$, where $\widehat{\gamma}_i \xleftarrow{R} \mathbb{Z}_{p_2 p_4}^*$.

(b) $\widetilde{\mathsf{RF}}_i(id|_i) : \mathrm{ID}|_i \to G_{p_3} G_{p_4}; id|_i \mapsto (g_3 g_4)^{\widetilde{\gamma}_i(id|_i)}$;

$\widetilde{\gamma}_i(id|_i) : \mathrm{ID}|_i \to \mathbb{Z}_{p_3 p_4}^*; id|_i \mapsto \widetilde{\gamma}_i$, where $\widetilde{\gamma}_i \xleftarrow{R} \mathbb{Z}_{p_3 p_4}^*$.

(ii) Auxiliary Secret Key Generation

$$\overline{\mathsf{KeyGen}}\left(u, \mathsf{msk}, \mathbf{K}\right) = \left(K_0, K_1, \ldots, \mathsf{msk} \cdot K_{2u-1}, K_{2u+1}, \ldots, K_{2n}\right), \quad (40)$$

where $\mathbf{K} = (K_0, K_1, K_2, \ldots, K_{2n}) \in G_N^{2n+1}$.

(iii) Auxiliary Encryption Function

$$\overline{\mathsf{Enc}}\left(S^*, \mathsf{mpk}, \mathbf{g}_1, \mathsf{msk}\right) = (Hdr; K) = \left(g_1^t, \left(\prod_{j=1}^{n} g_1^{\omega_{2j-s_j}}\right)^t; \mathsf{H}\left(e\left(g_1, \mathsf{msk}\right)^t\right)\right), \quad (41)$$

where $\mathbf{g}_1 = (g_1^t, g_1^{\omega_1 t}, \ldots, g_1^{\omega_{2n} t})$.

(iv) Semi-Functional Type-i User Secret Keys

$$\overline{\mathsf{KeyGen}}\left(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id|_i\right) \cdot \widetilde{\mathsf{RF}}_i\left(id|_i\right), \mathbf{K}\right) = \left(K_0, K_1, \ldots, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id|_i\right) \cdot \widetilde{\mathsf{RF}}_i\left(id|_i\right) \cdot K_{2u-1}, K_{2u+1}, \ldots, K_{2n}\right), \quad (42)$$

where user $u = id_1 \cdots id_n$ is denoted as id.

Table 4: Sequence of games, where a dash(−) means the same as in the previous game. Note that user set S^* and user identity u are interpreted as id^* and id, respectively.

Game	Challenge ciphertexts for identity S^*	User secret keys for identity $u \in [1, n]$
0	$\mathsf{Enc}(S^*, \mathsf{mpk})$	$\mathsf{KeyGen}(u, \mathsf{msk})$
1	$\overline{\mathsf{Enc}}(S^*, \mathsf{mpk}, \mathbf{g}_{\{1,2\}}, \mathsf{msk})$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk}, \mathbf{K})$
2, i, 0	$\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i-1}(id^*\vert_{i-1}))$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i-1}(id\vert_{i-1}) \cdot \widetilde{\mathsf{RF}}_{i-1}(id\vert_{i-1}), \mathbf{K})$
2, i, 1	if $id_i^* = 0$: $\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i-1}(id^*\vert_{i-1}))$ if $id_i^* = 1$: $\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,3\}}, \mathsf{msk} \cdot \widetilde{\mathsf{RF}}_{i-1}(id^*\vert_{i-1}))$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i-1}(id\vert_{i-1}) \cdot \widetilde{\mathsf{RF}}_{i-1}(id\vert_{i-1}), \mathbf{K})$
2, i, 2	if $id_i^* = 0$: $\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i}(id^*\vert_{i}))$ if $id_i^* = 1$: $\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,3\}}, \mathsf{msk} \cdot \widetilde{\mathsf{RF}}_{i}(id^*\vert_{i}))$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i}(id\vert_{i}) \cdot \widetilde{\mathsf{RF}}_{i}(id\vert_{i}), \mathbf{K})$
3	$\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{n}(id^*))$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{n}(id) \cdot \widetilde{\mathsf{RF}}_{n}(id), \mathbf{K})$
4	$\overline{\mathsf{Enc}}(-, -, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{n}(id^*)), K_b \xleftarrow{R} \{0,1\}^\tau$	$\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{n}(id) \cdot \widetilde{\mathsf{RF}}_{n}(id), \mathbf{K})$

(v) Pseudo-Normal Ciphertexts

$$\overline{\mathsf{Enc}}\left(S^*, \mathsf{mpk}, \mathbf{g}_{\{1,2\}}, \mathsf{msk}\right) = (Hdr; K) = \left(g_{\{1,2\}}^t, \left(\prod_{j=1}^{n} \left(g_{\{1,2\}}\right)^{\omega_{2j-s_j}} \right)^t ; \mathsf{H}\left(e\left(g_{\{1,2\}}, \mathsf{msk}\right)^t\right) \right), \tag{43}$$

where $\mathbf{g}_{\{1,2\}} = (g_{\{1,2\}}^t, g_{\{1,2\}}^{\omega_1 t}, \ldots, g_{\{1,2\}}^{\omega_{2n} t})$.

(vi) Semi-Functional Type-$(\wedge, i)$ Ciphertexts

$$\begin{aligned} &\overline{\mathsf{Enc}}\left(S^*, \mathsf{mpk}, \mathbf{g}_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id\vert_i\right) \cdot \widetilde{\mathsf{RF}}_i\left(id\vert_i\right)\right) \\ &= (Hdr; K) = \left(g_{\{1,2\}}^t, \left(\prod_{j=1}^{n} g_{\{1,2\}}^{\omega_{2j-s_j}} \right)^t ; \right. \\ &\left. \mathsf{H}\left(e\left(g_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id\vert_i\right)\right) \cdot \widetilde{\mathsf{RF}}_i\left(id\vert_i\right)^t\right) \right) \\ &= \left(g_{\{1,2\}}^t, \left(\prod_{j=1}^{n} g_{\{1,2\}}^{\omega_{2j-s_j}} \right)^t ; \right. \\ &\left. \mathsf{H}\left(e\left(g_{\{1,2\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id\vert_i\right)\right)^t\right) \right), \end{aligned} \tag{44}$$

where $\mathbf{g}_{\{1,2\}} = (g_{\{1,2\}}^t, g_{\{1,2\}}^{\omega_1 t}, \ldots, g_{\{1,2\}}^{\omega_{2n} t})$ and id denotes $S^* = s_1 \cdots s_n$.

(vii) Semi-Functional Type-$(\sim, i)$ Ciphertexts

$$\begin{aligned} &\overline{\mathsf{Enc}}\left(S^*, \mathsf{mpk}, \mathbf{g}_{\{1,3\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id\vert_i\right) \cdot \widetilde{\mathsf{RF}}_i\left(id\vert_i\right)\right) \\ &= (Hdr; K) = \left(g_{\{1,3\}}^t, \left(\prod_{j=1}^{n} g_{\{1,3\}}^{\omega_{2j-s_j}} \right)^t ; \right. \\ &\left. \mathsf{H}\left(e\left(g_{\{1,3\}}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id\vert_i\right) \cdot \widetilde{\mathsf{RF}}_i\left(id\vert_i\right)\right)^t\right) \right) \\ &= \left(g_{\{1,3\}}^t, \left(\prod_{j=1}^{n} g_{\{1,3\}}^{\omega_{2j-s_j}} \right)^t ; \right. \\ &\left. \mathsf{H}\left(e\left(g_{\{1,3\}}, \mathsf{msk} \cdot \widetilde{\mathsf{RF}}_i\left(id\vert_i\right)\right)^t\right) \right), \end{aligned} \tag{45}$$

where $\mathbf{g}_{\{1,3\}} = (g_{\{1,3\}}^t, g_{\{1,3\}}^{\omega_1 t}, \ldots, g_{\{1,3\}}^{\omega_{2n} t})$ and id denotes $S^* = s_1 \cdots s_n$.

Appendix

A. Proof of Theorem 3

Proof. If there exists a PPT adversary $\mathscr{A}$ which can break the MS-security of the broadcast encryption, then we show how to construct a PPT algorithm $\mathscr{B}$ to break BDHE assumption.

Upon receiving the BDHE problem instance, which consists of $(g^s, g, g^a, g^{a^2}, \ldots, g^{a^n}, g^{a^{n+2}}, \ldots, g^{a^{2n}})$ and Z, $\mathscr{B}$ simulates the experiment $\mathsf{Exp}_{\mathscr{A},\mathsf{BE}}^{\mathrm{MS}}(\lambda)$ for $\mathscr{A}$ as follows.

(i) $\mathscr{A}$ chooses a set $S^* \subseteq [1, n]$.

(ii) Setup. $\mathscr{B}$ generates $y_0, \ldots, y_n \overset{R}{\leftarrow} \mathbb{Z}_p$ and sets

$$h_i = g^{y_i} \quad \text{if } i \in S^*,$$
$$h_i = g^{y_i + a^i} \quad \text{if } i \notin S^*. \tag{A.1}$$

Since $g^{a^{n+1}}$ is unknown, we implicitly set $\alpha = y_0 \cdot a^{n+1}$ and $e(g,g)^\alpha$ can be computed as $e(g^a, g^{a^n})^{y_0}$. Now the public key is

$$\mathrm{PK} = \left(\mathbb{G}, \mathbb{G}_T, e; g, e(g,g)^\alpha, h_1, \ldots, h_n\right). \tag{A.2}$$

$\mathscr{B}$ sends PK to $\mathscr{A}$.

(iii) Key generation queries: for $i \in [1,n] \setminus S^*$, $\mathscr{B}$ chooses $z_i \overset{R}{\leftarrow} \mathbb{Z}_p$ and computes $r_i = z_i - y_0 \cdot a^{n+1-i}$ and outputs

$$d_i = (d_{i,0}, d_{i,1}, \ldots, d_{i,i-1}, d_{i,i}, d_{i,i+1}, \ldots, d_{i,n})$$
$$= \left(g^{-r_i}, h_1^{r_i}, h_2^{r_i}, \ldots, h_{i-1}^{r_i}, g^\alpha h_i^{r_i}, h_{i+1}^{r_i}, \ldots, h_n^{r_i}\right), \tag{A.3}$$

where $d_{i,i} = g^\alpha h_i^{r_i} = g^{y_0 \cdot a^{n+1} + (y_i + a^i)(z_i - y_0 \cdot a^{n+1-i})} = g^{y_i r_i + a^i \cdot z_i}$.

(iv) Encryption queries: $\mathscr{A}$ adaptively makes subset $S_j \subseteq S^*$ as encryption query. $\mathscr{B}$ runs the algorithm of Lemma 4 to generate (g^{s_j}, Z_j) for $j \in [1, q_{\mathrm{Enc}}]$ and set $Hdr_j^* = (C_{1,j}, C_{2,j}) = (g^{s_j}, (\prod_{i \in S_j} h_i)^{s_j}) = (g^{s_j}, (g^{s_j})^{\sum_{i \in S_j} y_i})$, $K_j = Z_j^{y_0}$. Return (Hdr_j^*, K_j) as the answer to S_j.

Eventually, $\mathscr{A}$ outputs a bit b' which is also the output of $\mathscr{B}$. It is easy to check that $\mathscr{B}$ perfectly simulates $\mathsf{Exp}^{\mathrm{MS}}_{\mathscr{A},\mathsf{BE}}(\lambda)$. Therefore, $\mathscr{B}$'s advantage in deciding the BDHE instance is precisely $\mathscr{A}$'s advantage against the MS-security of the broadcast encryption scheme, which completes the proof. □

B. Proof of Theorem 5

Proof. Let $\mathsf{Game}_i = 1$ denote the event that the adversary outputs 1 in Game_i.

Game_0. This is the real game which is identical to experiment $\mathsf{Exp}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}$. Thus,

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE},n}(\lambda) = \left|\Pr\left[\mathsf{Game}_0 = 1\right] - \frac{1}{2}\right|. \tag{B.1}$$

(Game_1 *to* Game_0). Game_1 is identical to Game_0 except that the challenge ciphertext $C_{0,j}$ for $j \in [1, q_{\mathrm{Enc}}]$ is computed as follows: $\kappa_{0,j} \overset{R}{\leftarrow} \mathscr{K}'$ and $C_{0,j} \leftarrow \mathrm{SymEnc}(\kappa_{0,j}, K_j)$.

For any PPT adversary $\mathscr{A}$ which can distinguish Game_1 from Game_0, there exists an algorithm $\mathscr{A}_1$ which can break the MS-security of $\mathsf{BE}_{\mathrm{MS}}$ scheme. Suppose q_{Enc} is the maximal number of encryption queries that adversary can make. $\mathscr{A}_1$ acts as follows:

(i) Choose $b^* \overset{R}{\leftarrow} \{0,1\}$.

(ii) Choose $s \overset{R}{\leftarrow} \{0,1\}^n$ and generate $S^* \leftarrow \{2i - (1 - s_i) : i \in [1,n]\}$, where s_i denotes ith bit of s. $\mathscr{A}_1$ sends S^* to the challenger. Notice that $S^* \subseteq [1, 2n]$.

(iii) The challenger runs $\mathsf{Setup}_{\mathrm{MS}}$ to obtain $(\mathrm{PK}', \mathrm{SK}')$ and sends PK' to $\mathscr{A}_1$. Then $\mathscr{B}_1$ sets $\mathrm{PK} \leftarrow \mathrm{PK}'$ and sends PK to $\mathscr{A}$.

(iv) $\mathscr{A}$ adaptively makes key generation queries for $i' \in [1,n]$. Then $\mathscr{A}_1$ sends $2i' - s_{i'}$ to the challenger of the MS experiment.

(v) The challenger runs $\mathsf{KeyGen}_{\mathrm{MS}}(2i' - s_{i'}, \mathrm{SK}')$ to obtain $d'_{i'}$ and sends $d'_{i'}$ to $\mathscr{A}_1$, which returns $d_{i'} \leftarrow (d'_{i'}, s_{i'})$ to $\mathscr{A}$.

(vi) $\mathscr{A}$ adaptively makes encryption queries S_j. Then $\mathscr{A}_1$ sets $T_j = \{t_i = 1 - s_i : i \in S_j\}$, $S_{0,j} = \{2i - t_i : i \in S_j\}$, $S_{1,j} = \{2i - (1 - t_i) : i \in S_j\}$ and sends $S_{0,j}$ for $j \in [1, q_{\mathrm{Enc}}]$ to the challenger. Notice that $S_{0,j} \subseteq S^*$.

(vii) The challenger sends back $(Hdr_{0,j}, \kappa_{0,j}^{(b)})$, where $(Hdr_{0,j}, \kappa_{0,j}^{(0)}) \leftarrow \mathsf{Enc}_{\mathrm{MS}}(S_{0,j}, \mathrm{PK}')$ and $\kappa_{0,j}^{(1)} \overset{R}{\leftarrow} \mathscr{K}'$. Note that $b \leftarrow \{0,1\}$ has been chosen by the challenger at the beginning of the experiment.

(viii) $\mathscr{A}_1$ sets $(Hdr_{1,j}, \kappa_{1,j}) \leftarrow \mathsf{Enc}_{\mathrm{MS}}(S_{1,j}, \mathrm{PK}')$ and gener ates $K_{0,j}, K_{1,j} \overset{R}{\leftarrow} \mathscr{K}$. Then it sets $C_{0,j} \leftarrow \mathrm{SymEnc}(\kappa_{0,j}^{(b)}, K_{0,j})$, $C_{1,j} \leftarrow \mathrm{SymEnc}(\kappa_{1,j}, K_{0,j})$, and $Hdr_j^* \leftarrow (Hdr_{0,j}, C_{0,j}, Hdr_{1,j}, C_{1,j}, T_j)$ and sends $(Hdr_j^*, K_{b^*,j})$ to adversary $\mathscr{A}$.

(ix) $\mathscr{A}$ outputs a bit b'. If $b^* = b'$, $\mathscr{A}_1$ outputs 0, otherwise, 1.

Notice that if $b = 0$, $\mathscr{A}$'s view is identical to that of Game_0. Otherwise, $\mathscr{A}$'s view is identical to that of Game_1. Hence, we have

$$\left|\Pr\left[\mathsf{Game}_0 = 1\right] - \Pr\left[\mathsf{Game}_1 = 1\right]\right| \le \mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A}_1,\mathsf{BE},2n}(\lambda). \tag{B.2}$$

(Game_2 *to* Game_1). Game_2 is identical to Game_1 except that for each encryption query the challenger chooses $\kappa_{1,j} \overset{R}{\leftarrow} \mathscr{K}'$ to construct $C_{1,j} = \mathrm{SymEnc}(\kappa_{1,j}, K_{0,j})$. The proof of the indistinguishability between Game_2 and Game_1 is similar to that of Game_1 and Game_0. So we have

$$\left|\Pr\left[\mathsf{Game}_1 = 1\right] - \Pr\left[\mathsf{Game}_2 = 1\right]\right| \le \mathsf{Adv}^{\mathrm{MS}}_{\mathscr{A}_2,\mathsf{BE},2n}(\lambda). \tag{B.3}$$

(Game_3 *to* Game_2). Game_3 is identical to Game_2 except that for each encryption query the challenger chooses $K_{0,j} \overset{R}{\leftarrow} \mathscr{K}$ to construct $C_{0,j} \overset{R}{\leftarrow} \mathrm{SymEnc}(\kappa_{0,j}, K_{0,j})$ for $j \in [1, q_{\mathrm{Enc}}]$.

We construct a series of subgame $2,\iota$ for $\iota = 1, \ldots, q_{\mathrm{Enc}}$, to prove the indistinguishability between Game 2 and Game 3.

(i) $\mathsf{Game}_{2,1}$. $\mathsf{Game}_{2,1}$ is identical to Game_2 except that the challenger chooses $K_{0,1} \xleftarrow{R} \mathcal{K}$ to construct $C_{0,1} \xleftarrow{R} \mathrm{SymEnc}(\kappa_{0,1}, K_{0,1})$.

(ii) $\mathsf{Game}_{2,\iota}$. $\mathsf{Game}_{2,\iota}$ is identical to $\mathsf{Game}_{2,\iota-1}$ except that the challenger chooses $K_{0,\iota} \xleftarrow{R} \mathcal{K}$ to construct $C_{0,\iota} \xleftarrow{R} \mathrm{SymEnc}(\kappa_{0,\iota}, K_{0,\iota})$.

Note that Game_3 is identical to $\mathsf{Game}_{2,q_{\mathrm{Enc}}}$.

Claim ($\mathsf{Game}_{2,\iota-1}$ *to* $\mathsf{Game}_{2,\iota}$). For any PPT adversary $\mathcal{M}$ which can make at most $q_{\mathrm{key}} = q_{\mathrm{key}}(\lambda)$ key generation queries and $q_{\mathrm{Enc}} = q_{\mathrm{Enc}}(\lambda)$ encryption queries with running time $\tilde{t}$, there exists an algorithm $\mathcal{A}_3$ with running about the same time as $\mathcal{M}$, such that

$$\left|\Pr\left[\mathsf{Game}_{2,\iota-1} = 1\right] - \Pr\left[\mathsf{Game}_{2,\iota} = 1\right]\right| \leq \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_3,\Pi_{\mathrm{sym}}}(\lambda). \tag{B.4}$$

Proof. $\mathcal{A}_3$ chooses $b^* \xleftarrow{R} \{0,1\}$ and simulates the experiment as follows.

(i) $\mathcal{A}_3$ runs $\mathsf{Setup}(n, \ell)$ and $\mathsf{KeyGen}(i', \mathrm{SK})$ as in Game_2.

(ii) $\mathcal{M}$ adaptively makes encryption queries for S_j, where j denotes jth query. If $j \in [1, \iota - 1]$, $\mathcal{A}_3$ returns the answer as in $\mathsf{Game}_{2,\iota-1}$. If $j \in [\iota + 1, q_{\mathrm{Enc}}]$, $\mathcal{A}_3$ returns the answer as in Game_2. If $j = \iota$, $\mathcal{A}_3$ chooses $K_{0,\iota}, K_{1,\iota} \xleftarrow{R} \mathcal{K}$ and sends $(K_{0,\iota}, K_{1,\iota})$ to the challenger. The challenger chooses $b \xleftarrow{R} \{0,1\}$ and returns $C_{0,\iota} = \mathrm{SymEnc}(\kappa, K_{b,\iota})$. Then $\mathcal{A}_3$ chooses $\kappa_{1,\iota} \xleftarrow{R} \mathcal{K}'$, computes $C_{1,\iota} = \mathrm{SymEnc}(\kappa_{1,\iota}, K_{1,\iota})$, and returns $(C_{0,\iota}, C_{1,\iota})$.

(iii) $\mathcal{M}$ outputs b'. If $b^* = b'$, $\mathcal{A}_3$ outputs 0, otherwise, 1.

Observe that if $b = 1$, $\mathcal{M}$'s view is identical to that of $\mathsf{Game}_{2,\iota-1}$. Otherwise, $\mathcal{M}$'s view is identical to that of $\mathsf{Game}_{2,\iota}$. Thus

$$\left|\Pr\left[\mathsf{Game}_{2,\iota-1} = 1\right] - \Pr\left[\mathsf{Game}_{2,\iota} = 1\right]\right| \leq \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_3,\Pi_{\mathrm{sym}}}(\lambda), \tag{B.5}$$

which concludes the proof of the Claim.

Due to the Claim, we have

$$\left|\Pr\left[\mathsf{Game}_2 = 1\right] - \Pr\left[\mathsf{Game}_3 = 1\right]\right| \leq q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_3,\Pi_{\mathrm{sym}}}(\lambda). \tag{B.6}$$

(Game_4 *to* Game_3). Game_4 is identical to Game_3 except that the challenger sets $K_{1,j} \xleftarrow{R} \mathcal{K}$ to construct $C_{1,j} = \mathrm{SymEnc}(\kappa_{1,j}, K_{1,j})$ for $j \in [1, q_{\mathrm{Enc}}]$. The proof of the indistinguishability between Game_4 and Game_3 is similar to that of Game_3 and Game_2. So we have

$$\left|\Pr\left[\mathsf{Game}_3 = 1\right] - \Pr\left[\mathsf{Game}_4 = 1\right]\right| \leq q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_4,\Pi_{\mathrm{sym}}}(\lambda). \tag{B.7}$$

In Game_4, all $\kappa_{0,j}, \kappa_{1,j}, K_{0,j}, K_{1,j}$ for $j \in [1, q_{\mathrm{Enc}}]$ are chosen at random and Hdr^*_j is independent of $K_{b,j}$. Hence, the adversary has no advantage. That is,

$$\left|\Pr\left[\mathsf{Game}_4 = 1\right] - \frac{1}{2}\right| = 0. \tag{B.8}$$

Hence, we have

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathcal{A},\mathsf{BE},n}(\lambda) \leq \mathsf{Adv}^{\mathrm{MS}}_{\mathcal{A}_1,\mathsf{BE},2n}(\lambda) + \mathsf{Adv}^{\mathrm{MS}}_{\mathcal{A}_2,\mathsf{BE},2n}(\lambda) + q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_3,\Pi_{\mathrm{sym}}}(\lambda) + q_{\mathrm{Enc}} \cdot \mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}_4,\Pi_{\mathrm{sym}}}(\lambda). \tag{B.9}$$

□

C. Proof of Theorem 6

Game Sequence

(i) Game_0 is the real experiment $\mathsf{Exp}^{\mathrm{MA}}_{\mathcal{A},\mathsf{BE}}$.

(ii) Game_1 is the same as Game_0 except that all the challenge ciphertexts are pseudo-normal.

(iii) $\mathsf{Game}_{2,i,0}$ is the same as Game_1 except that all user secret keys are semifunctional of type-$(i-1)$, while the challenge ciphertexts are semifunctional of type-$(\wedge, i-1)$ for $i \in [1, n]$.

(iv) $\mathsf{Game}_{2,i,1}$ is the same as $\mathsf{Game}_{2,i,0}$ except that if ith bit of a challenge identity id^* is 0 (i.e., $id^*_i = 0$), then the corresponding challenge ciphertexts are semifunctional of type-$(\wedge, i-1)$. Otherwise, the corresponding challenge ciphertexts are semifunctional of type-$(\sim, i-1)$.

(v) $\mathsf{Game}_{2,i,2}$ is the same as $\mathsf{Game}_{2,i,1}$ except that if ith bit of a challenge identity id^* is 0 (i.e., $id^*_i = 0$), then the corresponding challenge ciphertexts are semifunctional of type-$(\wedge, i)$. Otherwise, the corresponding challenge ciphertexts are semifunctional of type-$(\sim, i)$.

(vi) Game_3 is the same as $\mathsf{Game}_{2,n,0}$ except that all the challenge ciphertexts and user secret keys are semifunctional of type-$(\wedge, n)$ and semifunctional of type-n, respectively.

(vii) Game_4 is the same as Game_3 except that the challenge keys K_b output by oracle $\mathcal{O}_{\mathrm{Enc}}(\cdot, \mathsf{mpk})$ are uniform bitstrings over $\{0,1\}^\tau$.

Lemma C.1 (Game_0 to Game_1). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\tilde{t}$, there exists an*

algorithm $\mathcal{B}_1$ on DS1 with running time $t_1 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$, such that

$$\left|\Pr\left[\mathsf{Game}_0 = 1\right] - \Pr\left[\mathsf{Game}_1 = 1\right]\right| \leq \mathsf{Adv}^{DS1}_{\mathcal{B}_1}(\lambda). \quad \text{(C.1)}$$

Proof. $\mathcal{B}_1$ receives the instance $(G, G_T, N, e, g, g_1, g_3, g_4, T)$ from the challenger, where $T \overset{R}{\leftarrow} G_{p_1}$ or $T \overset{R}{\leftarrow} G_{p_1p_2}$, and simulates the experiment for $\mathcal{A}$ as follows.

Setup. Choose a bit $b \leftarrow \{0,1\}$. Pick $\alpha, \omega_1, \ldots, \omega_{2n} \overset{R}{\leftarrow} \mathbb{Z}_N$ and compute $(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}) \in G_{p_1}^{2n+1}$. Let $h = g \in G_N$, $r_\mu \overset{R}{\leftarrow} \mathbb{Z}_N^*$, $R_{4,\mu} = g_4^{r_\mu}$ for $\mu \in [1, 2n]$. Compute $h_\mu = h^{\omega_\mu} \cdot R_{4,\mu}$ for $\mu \in [1, 2n]$. Set

$$\begin{aligned}\mathsf{mpk} &= \left(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}, g_4, h, e\left(g_1, h\right)^{\alpha}, h_1, \ldots, h_{2n}\right), \quad \text{(C.2)}\\ \mathsf{msk} &= h^{\alpha}.\end{aligned}$$

Key Generation Queries. To answer key generation queries $u \in [1, n]$, set $r'_\mu \overset{R}{\leftarrow} \mathbb{Z}_N^*$ and compute $R'_{4,\mu} = g_4^{r'_\mu}$ for $\mu \in [1, 2n]$ and generate $\mathbf{K} = (K_0, K_1, K_2, \ldots, K_{2n}) = (h^t, h_1^t \cdot R'_{4,1}, \ldots, h_{2n}^t \cdot R'_{4,2n}) \in G_N^{2n+1}$, where $t \overset{R}{\leftarrow} \mathbb{Z}_N^*$, and return $\overline{\mathsf{KeyGen}}(u, \mathsf{msk}, \mathbf{K})$.

Encryption Queries. $\mathcal{A}$ can adaptively make encryption queries S_l at most $q_{\mathrm{Enc}} = q_{\mathrm{Enc}}(\lambda)$ times. $\mathcal{B}_1$ sets

$$\begin{aligned}\mathsf{Enc}\left(S_l, \mathsf{mpk}\right) &= \left(T^{t_l}, \prod_{j=1}^{n} T^{t_l \cdot \omega_{2j-s_j}}, \mathsf{H}\left(e\left(T^{t_l}, \mathsf{msk}\right)\right)\right) \quad \text{(C.3)}\\ &= \left(Hdr_l^*, K_{l,0}^*\right),\end{aligned}$$

where $t_l \overset{R}{\leftarrow} \mathbb{Z}_N^*, l \in [1, q_{\mathrm{Enc}}]$ and returns $(Hdr_l^*, K_{l,b}^*)$, where $K_{l,1}^* \leftarrow \{0,1\}^\tau$.

Finally, $\mathcal{A}$ outputs a guess b'. $\mathcal{B}_1$ outputs 1 if $b' = b$, otherwise, 0.

The distribution of the master public key and user secret keys that $\mathcal{A}$ requests are identical to those in Game_0 as well as in Game_1. If $T \overset{R}{\leftarrow} G_{p_1}$, the distribution of challenge ciphertexts is identical to that of Game_0, while if $T \overset{R}{\leftarrow} G_{p_1p_2}$, the distribution of challenge ciphertexts is identical to that of Game_1. Therefore, we have (C.1). □

Lemma C.2 (Game_1 to $\mathsf{Game}_{2,1,0}$). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, we have*

$$\Pr\left[\mathsf{Game}_1 = 1\right] = \Pr\left[\mathsf{Game}_{2,1,0} = 1\right]. \quad \text{(C.4)}$$

Proof. As shown in Table 4, $\mathsf{msk} \leftarrow G_N$ in Game_1, while $\mathsf{msk}' = \mathsf{msk} \cdot \widehat{\mathsf{RF}}_0(\varepsilon) \cdot \widetilde{\mathsf{RF}}_0(\varepsilon)$ in $\mathsf{Game}_{2,1,0}$ is also uniform in G_N, where $id|_0$ denotes the empty string ε. Since the distribution of msk and $\mathsf{msk} \cdot \widehat{\mathsf{RF}}_0(\varepsilon) \cdot \widetilde{\mathsf{RF}}_0(\varepsilon)$ is identical, (C.4) holds. □

Lemma C.3 ($\mathsf{Game}_{2,i,0}$ to $\mathsf{Game}_{2,i,1}$). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, there exists an algorithm $\mathcal{B}_2$ on DS2 with running time $t_2 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$, such that*

$$\begin{aligned}&\left|\Pr\left[\mathsf{Game}_{2,i,0} = 1\right] - \Pr\left[\mathsf{Game}_{2,i,1} = 1\right]\right|\\ &\quad \leq \mathsf{Adv}^{DS2}_{\mathcal{B}_2}(\lambda).\end{aligned} \quad \text{(C.5)}$$

Proof. $\mathcal{B}_2$ receives the instance $(G, G_T, N, e, g, g_1, g_4, g_{\{1,2\}}, g_{\{2,3\}}, T)$ from the challenger, where $T \overset{R}{\leftarrow} G_{p_1p_2}$ or $T \overset{R}{\leftarrow} G_{p_1p_3}$ and chooses $b \overset{R}{\leftarrow} \{0,1\}$. Then $\mathcal{B}_2$ can use the parameter to generate $\mathsf{msk}, \mathsf{mpk}$ as in the proof of Lemma C.1 and define a truly random function $\mathsf{RF} : \{0,1\}^{i-1} \to G_{p_2p_3p_4}$.

Key Generation Queries. Next, it can answer the secret key queries for $u \in [1, n]$ as $\overline{\mathsf{KeyGen}}(u, \mathsf{msk} \cdot \mathsf{RF}(id|_{i-1}), \mathbf{K})$, where $\mathsf{RF}(id|_{i-1}) : id|_{i-1} \to (g_{\{2,3\}}g_4)^{\gamma_{i-1}(id|_{i-1})}$, $\gamma_{i-1}(id|_{i-1}) : id|_{i-1} \mapsto \gamma_{i-1}$, and $\gamma_{i-1} \overset{R}{\leftarrow} \mathbb{Z}_N^*$.

Encryption Queries. Upon receiving encryption queries S_l^*, $\mathcal{B}_2$ chooses $t_l \overset{R}{\leftarrow} \mathbb{Z}_N^*$, $l \in [1, q_{\mathrm{Enc}}]$ and returns

$$\begin{cases}\overline{\mathsf{Enc}}\left(S_l^*, \mathsf{mpk}, \mathbf{g}_{\{1,2\}}^{t_l}, \mathsf{msk} \cdot \mathsf{RF}\left(id_l^*|_{i-1}\right)\right) & \text{if } \left(id_l^*\right)_i = 0,\\ \overline{\mathsf{Enc}}\left(S_l^*, \mathsf{mpk}, T^{t_l}, \mathsf{msk} \cdot \mathsf{RF}\left(id_l^*|_{i-1}\right)\right) & \text{if } \left(id_l^*\right)_i = 1.\end{cases} \quad \text{(C.6)}$$

Finally, $\mathcal{A}$ outputs a guess bit b'. $\mathcal{B}_2$ outputs 1 if $b' = b$; otherwise 0.

Note that $\mathbf{g}_{\{1,2\}}^{t_l}$ is distributed uniformly over $G_{p_1p_2}$. If $T \overset{R}{\leftarrow} G_{p_1p_2}$, the challenge ciphertexts are distributed identically as in $\mathsf{Game}_{2,i,0}$. Otherwise, the distribution is the same as in $\mathsf{Game}_{2,i,1}$. Hence (C.5) holds. □

Lemma C.4 ($\mathsf{Game}_{2,i,1}$ to $\mathsf{Game}_{2,i,2}$). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, there exists an algorithm $\mathcal{B}_3$ on DS3 with running time $t_3 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$, such that*

$$\begin{aligned}&\left|\Pr\left[\mathsf{Game}_{2,i,1} = 1\right] - \Pr\left[\mathsf{Game}_{2,i,2} = 1\right]\right|\\ &\quad \leq \mathsf{Adv}^{DS3}_{\mathcal{B}_3}(\lambda).\end{aligned} \quad \text{(C.7)}$$

Proof. $\mathcal{B}_3$ gets the instance (D, T), where $T = (\widehat{T}, \widetilde{T})$ is either (g_2^{xy}, g_3^{xy}) or $(g_2^{xy+\gamma'}, g_3^{xy+\gamma'})$ for

$$\begin{aligned}D = \Big(&G, G_T, N, e, g, g_1, g_2, g_3, g_4, g_2^x\widehat{X}_4, g_2^y\widehat{Y}_4, g_3^x\widetilde{X}_4,\\ &g_3^y\widetilde{Y}_4\Big),\end{aligned} \quad \text{(C.8)}$$

$$\widehat{X}_4, \widehat{Y}_4, \widetilde{X}_4, \widetilde{Y}_4 \overset{R}{\leftarrow} G_{p_4}, x, y, \gamma' \longleftarrow \mathbb{Z}_N^*. \quad \text{(C.9)}$$

Setup. $\mathcal{B}_3$ chooses $b \leftarrow \{0,1\}$ and $(\alpha, \omega_1, \ldots, \omega_{2n}) \in \mathbb{Z}_N^{2n+1}$, computes $(g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}) \in G_{p_1}^{2n}$, and generates $r, \hat{r}, \tilde{r} \overset{R}{\leftarrow} \mathbb{Z}_N^*$ and sets

$$\begin{aligned} h &= (g_1 g_2 g_3 g_4)^r \in G_N, \\ \widehat{h} &= (g_2 g_4)^{\hat{r}} \in G_{p_2 p_4}, \\ \widetilde{h} &= (g_3 g_4)^{\tilde{r}} \in G_{p_3 p_4}. \end{aligned} \tag{C.10}$$

Next $\mathcal{B}_3$ generates $r_\mu \overset{R}{\leftarrow} \mathbb{Z}_N^*$ for $\mu \in [1, 2n]$ and $(R_{4,1}, \ldots, R_{4,2n}) = (g_4^{r_1}, \ldots, g_4^{r_{2n}}) \in G_{p_4}^{2n}$ and sets

$$\begin{aligned} &(h_1, \ldots, h_{2i-1}, h_{2i}, \ldots, h_{2n}) = \Big(h^{\omega_1} \cdot R_{4,1}, \ldots, h^{\omega_{2i-1}} \\ &\quad \cdot \left(g_2^y \widehat{Y}_4\right)^r \cdot R_{4,2i-1}, h^{\omega_{2i}} \cdot \left(g_3^y \widetilde{Y}_4\right)^r \cdot R_{4,2i}, \ldots, h^{\omega_{2n}} \\ &\quad \cdot R_{4,2n}\Big), \\ &\mathsf{mpk} = \left(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}, g_4, h, e(g_1, h)^{\alpha}, h_1, \ldots, h_{2n}\right). \end{aligned} \tag{C.11}$$

Key Generation Queries. During the experiment $\mathcal{B}_3$ can answer key generation queries for identity id_ℓ for $\ell \in [1, q_{\text{key}}]$ as follows.

By running the algorithm in Lemma 6 of [20] (or algorithm in [19]) which takes as input $(1^{q_{\text{key}}}, (g_2, g_4, g_2^x \widehat{X}_4, g_2^y \widehat{Y}_4), \widehat{T})$ and $(1^{q_{\text{key}}}, (g_3, g_4, g_3^x \widetilde{X}_4, g_3^y \widetilde{Y}_4), \widetilde{T})$, $\mathcal{B}_3$ can generate the following tuples:

$$\begin{aligned} &\left(g_2^{\hat{r}_\ell} \widehat{X}_{4,\ell}, \widehat{T}_\ell\right)_{\ell=1}^{q_{\text{key}}}, \\ &\left(g_3^{\tilde{r}_\ell} \widetilde{X}_{4,\ell}, \widetilde{T}_\ell\right)_{\ell=1}^{q_{\text{key}}} \end{aligned} \tag{C.12}$$

respectively, where

$$\begin{aligned} \widehat{T}_\ell &= \begin{cases} g_2^{\hat{r}_\ell y} \cdot \widehat{Y}_{4,\ell}, & \text{if } \widehat{T} = g_2^{xy} \\ g_2^{\hat{r}_\ell y} \cdot \widehat{Y}_{4,\ell} \cdot g_2^{\hat{\gamma}'_\ell}, & \text{if } \widehat{T} = g_2^{xy+\gamma'}, \end{cases} \\ \widetilde{T}_\ell &= \begin{cases} g_3^{\tilde{r}_\ell y} \cdot \widetilde{Y}_{4,\ell}, & \text{if } \widetilde{T} = g_3^{xy} \\ g_3^{\tilde{r}_\ell y} \cdot \widetilde{Y}_{4,\ell} \cdot g_3^{\tilde{\gamma}'_\ell}, & \text{if } \widetilde{T} = g_3^{xy+\gamma'}. \end{cases} \end{aligned} \tag{C.13}$$

Then $\mathcal{B}_3$ generates $r'_\ell \leftarrow \mathbb{Z}_N^*$, $(g_4^{t_{\ell,1}}, \ldots, g_4^{t_{\ell,2n}}) = (R_{\ell,1}, \ldots, R_{\ell,2n}) \in G_{p_4}^{2n}$, where $(t_{\ell,1}, \ldots, t_{\ell,2n}) \overset{R}{\leftarrow} (\mathbb{Z}_N^*)^{2n}$, and sets

$$\begin{aligned} K_0 &= h^{r'_\ell} \cdot g_2^{\hat{r}_\ell} \widehat{X}_{4,\ell} \cdot g_3^{\tilde{r}_\ell} \widetilde{X}_{4,\ell}, \\ K_1 &= {K_0}^{\omega_1} \cdot R_{\ell,1}, \\ &\vdots \\ K_{2j-1} &= {K_0}^{\omega_{2j-1}} \cdot \left(g_2^y \widehat{Y}_4\right)^{r'_\ell \cdot r} \cdot \widehat{T}_\ell \cdot R_{\ell,2j-1}, \\ K_{2j} &= {K_0}^{\omega_{2j}} \cdot \left(g_3^y \widetilde{Y}_4\right)^{r'_\ell \cdot r} \cdot \widetilde{T}_\ell \cdot R_{\ell,2j}, \\ &\vdots \\ K_{2n} &= {K_0}^{\omega_{2n}} \cdot R_{\ell,2n}. \end{aligned} \tag{C.14}$$

Thus $\mathbf{K}_\ell = (K_0, K_1, \ldots, K_{2j-1}, K_{2j}, \ldots, K_{2n})$. For identity id_ℓ and $\ell \in [1, q_{\text{key}}]$, $\mathcal{B}_3$ defines the random functions below:

$$\begin{aligned} &\widehat{\mathsf{RF}}_i\left(id_\ell|_i\right) = \widehat{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right), \\ &\widetilde{\mathsf{RF}}_i\left(id_\ell|_i\right) = \widetilde{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right) \cdot \left(\widetilde{h}\right)^{\tilde{\gamma}_\ell} \\ &\qquad \text{if } \left(id_\ell\right)_i = 0, \\ &\widetilde{\mathsf{RF}}_i\left(id_\ell|_i\right) = \widetilde{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right), \\ &\widehat{\mathsf{RF}}_i\left(id_\ell|_i\right) = \widehat{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right) \cdot \left(\widehat{h}\right)^{\hat{\gamma}_\ell} \\ &\qquad \text{if } \left(id_\ell\right)_i = 1, \end{aligned} \tag{C.15}$$

where $\tilde{\gamma}_\ell, \hat{\gamma}_\ell \leftarrow \mathbb{Z}_N^*$. Next $\mathcal{B}_3$ answers ℓ-th secret key generation query for identity id_ℓ with prefix $id_\ell|_i$ that is not a prefix of an already queried identity as

$$\begin{aligned} &\overline{\mathsf{KeyGen}}\left(u_\ell, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id_\ell|_i\right) \cdot \widetilde{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right), \mathbf{K}_\ell\right) \\ &\qquad \text{if } \left(id_\ell\right)_i = 0, \\ &\overline{\mathsf{KeyGen}}\left(u_\ell, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_{i-1}\left(id_\ell|_{i-1}\right) \cdot \widetilde{\mathsf{RF}}_i\left(id_\ell|_i\right), \mathbf{K}_\ell\right) \\ &\qquad \text{if } \left(id_\ell\right)_i = 1. \end{aligned} \tag{C.16}$$

For an identity prefix $id_\ell|_i$ that is a prefix of an already queried identity, we rerandomize the element of $\mathbf{K}_\ell$.

Encryption Queries. Upon receiving encryption queries S_l^*, $\mathcal{B}_3$ chooses $t_l \overset{R}{\leftarrow} \mathbb{Z}_N^*$ and returns

$$\begin{aligned} &\overline{\mathsf{Enc}}\left(S_l^*, \mathsf{mpk}, (\mathbf{g}_1\mathbf{g}_2)^{t_l}, \mathsf{msk} \cdot \widehat{\mathsf{RF}}_i\left(id_l^*|_i\right)\right) \\ &\qquad \text{if } \left(id_l^*\right)_i = 0, \\ &\overline{\mathsf{Enc}}\left(S_l^*, \mathsf{mpk}, (\mathbf{g}_1\mathbf{g}_3)^{t_l}, \mathsf{msk} \cdot \widetilde{\mathsf{RF}}_i\left(id_l^*|_i\right)\right) \\ &\qquad \text{if } \left(id_l^*\right)_i = 1, \end{aligned} \tag{C.17}$$

where $\mathbf{g}_1\mathbf{g}_2 = ((g_1 g_2)^{t^*}, (g_1 g_2)^{\omega_1 t^*}, \ldots, (g_1 g_2)^{\omega_{2n} t^*})$, $\mathbf{g}_1\mathbf{g}_3 = ((g_1 g_3)^{\hat{t}}, (g_1 g_3)^{\omega_1 \hat{t}}, \ldots, (g_1 g_3)^{\omega_{2n} \hat{t}})$, $t^*, \hat{t} \overset{R}{\leftarrow} \mathbb{Z}_N^*$.

Finally, $\mathcal{A}$ outputs a guess bit b'. $\mathcal{B}_3$ outputs 1 if $b' = b$, otherwise 0. If $T = (g_2^{xy}, g_3^{xy})$ (i.e., $T_\ell = (g_2^{\hat{r}_\ell y} \cdot \widehat{Y}_{4,\ell}, g_3^{\tilde{r}_\ell y} \cdot \widetilde{Y}_{4,\ell})$), then the secret keys are distributed identically as in $\mathsf{Game}_{2,i,1}$. If $T = (g_2^{xy+\gamma'}, g_3^{xy+\gamma'})$ (i.e., $T_\ell = (g_2^{\hat{r}_\ell y} \cdot \widehat{Y}_{4,\ell} \cdot g_2^{\hat{r}'_\ell}, g_3^{\tilde{r}_\ell y} \cdot \widetilde{Y}_{4,\ell} \cdot g_3^{\tilde{r}'_\ell})$), we have

$$\begin{aligned} &\left(\widehat{h}, g_2^{\hat{r}_\ell} \cdot \widehat{Y}_{4,\ell}\right), \\ &\left(\widehat{h}, \left(\widehat{h}\right)^{\hat{r}_\ell} \cdot \widehat{Y}_{4,\ell}\right) \\ &\left(\widetilde{h}, g_3^{\tilde{r}_\ell} \cdot \widetilde{Y}_{4,\ell}\right), \\ &\left(\widetilde{h}, \left(\widetilde{h}\right)^{\tilde{r}_\ell} \cdot \widetilde{Y}_{4,\ell}\right) \end{aligned} \tag{C.18}$$

identically distributed, respectively. Therefore, in this case, the distribution is the same as in $\mathsf{Game}_{2,i,2}$. Besides, the distribution of the challenge ciphertexts is identical in these two games. Hence, (C.7) holds. □

Lemma C.5 ($\mathsf{Game}_{2,i-1,2}$ to $\mathsf{Game}_{2,i,0}$). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, there exists an algorithm $\mathcal{B}_2$ on DS2, with running time $t_2 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$, such that*

$$\left|\Pr\left[\mathsf{Game}_{2,i-1,2} = 1\right] - \Pr\left[\mathsf{Game}_{2,i,0} = 1\right]\right| \leq \mathsf{Adv}^{DS2}_{\mathcal{B}_2}(\lambda). \tag{C.19}$$

Lemma C.6 ($\mathsf{Game}_{2,n,2}$ to Game_3). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, there exists an algorithm $\mathcal{B}_2$ on DS2, with running time $t_2 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$, such that*

$$\left|\Pr\left[\mathsf{Game}_{2,n,2} = 1\right] - \Pr\left[\mathsf{Game}_3 = 1\right]\right| \leq \mathsf{Adv}^{DS2}_{\mathcal{B}_2}(\lambda). \tag{C.20}$$

In $\mathsf{Game}_{2,i,0}$ all the challenge ciphertexts are semifunctional of type-$(\wedge, i-1)$, while in $\mathsf{Game}_{2,i-1,2}$ if ith bit of challenge identity id is 0 (i.e., $id_i = 0$), the challenge ciphertexts are totally identical to those in $\mathsf{Game}_{2,i,0}$. Otherwise, the challenge ciphertexts are semifunctional of type-$(\sim, i-1)$. Actually, the proof of ($\mathsf{Game}_{2,i-1,2}$ to $\mathsf{Game}_{2,i,0}$) and ($\mathsf{Game}_{2,n,2}$ to Game_3) is similar to that in Lemma C.3 and thus omitted.

Lemma C.7 (Game_3 to Game_4). *For any PPT adversary $\mathcal{A}$ with at most $q_{key} = q_{key}(\lambda)$ key generation queries, $q_{Enc} = q_{Enc}(\lambda)$ encryption queries, and running time $\widetilde{t}$, there exists an algorithm $\mathcal{B}_4$ on DS-BDDH with running time $t_4 \approx \widetilde{t} + \mathbf{O}(n\lambda^c(q_{key} + q_{Enc}))$, for some constant $c \in \mathbb{N}$ such that*

$$\left|\Pr\left[\mathsf{Game}_3 = 1\right] - \Pr\left[\mathsf{Game}_4 = 1\right]\right| \leq \mathsf{Adv}^{DS\text{-}BDDH}_{\mathcal{B}_4}(\lambda) + q_{Enc} \cdot \mathbf{O}\left(2^{-\tau}\right). \tag{C.21}$$

Proof. $\mathcal{B}_4$ is provided with the instance (D, T) where T is either $e(g_2, g_2)^{abc}$ or $e(g_2, g_2)^z$, for

$$D = \left(G, G_T, \mathrm{N}, e, g, (g_i), g_1^a, g_2^a, g_2^b, g_{\{2,4\}}, g_{\{2,4\}}^b, g_{\{2,4\}}^c\right), \tag{C.22}$$

where $a, b, c, z \leftarrow \mathbb{Z}_N^*$.

Setup. $\mathcal{B}_4$ chooses $(\alpha, \omega_1, \ldots, \omega_{2n}) \xleftarrow{R} \mathbb{Z}_N^{2n+1}$ and computes $(g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}) \in G_{p_1}^{2n}$. Set $h = g \in G_N$, $(r_1, r_2, \ldots, r_{2n}) \xleftarrow{R} (\mathbb{Z}_N^*)^{2n}$ and compute $(R_{4,1}, \ldots, R_{4,2n}) = (g_4^{r_1}, \ldots, g_4^{r_{2n}}) \in G_{p_4}^{2n}$. Thus

$$\begin{aligned}
&(h_1, \ldots, h_{2n}) = \left(h^{\omega_1} \cdot R_{4,1}, \ldots, h^{\omega_{2n}} \cdot R_{4,2n}\right);\\
&\mathsf{mpk} = \left(g_1, g_1^{\omega_1}, \ldots, g_1^{\omega_{2n}}, h, e\left(g_1, h\right)^{\alpha}, h_1, \ldots, h_{2n}\right),\\
&\mathsf{msk} = h^{\alpha}.
\end{aligned} \tag{C.23}$$

Key Generation Queries. $\mathcal{B}_4$ defines a truly random function $\mathsf{RF}' : \{0,1\}^n \to G_{p_2 p_4}$. Next it can answer the secret key generation queries for identity $u \in [1, n]$ as

$$\overline{\mathsf{KeyGen}}\left(u, \mathsf{msk} \cdot \mathsf{RF}'(id), \mathbf{K}\right), \tag{C.24}$$

where $\mathsf{RF}' : id \mapsto (g_{\{2,4\}})^{\gamma'(id)}$, $\gamma'(id) : id \mapsto \gamma'$ for $\gamma' \xleftarrow{R} \mathbb{Z}_N^*$, and $\mathbf{K} = (h^{\hat{t}}, h_1^{\hat{t}} \cdot R'_{4,1}, \ldots, h_{2n}^{\hat{t}} \cdot R'_{4,2n})$ for $\hat{t} \xleftarrow{R} \mathbb{Z}_N^*$, $(R'_{4,1}, \ldots, R'_{4,2n}) = (g_4^{r'_1}, \ldots, g_4^{r'_{2n}})$, where $(r'_1, \ldots, r'_{2n}) \xleftarrow{R} (\mathbb{Z}_N^*)^{2n}$.

Encryption Queries. Upon receiving encryption queries S_l^* for some $l \in [1, q_{\mathrm{Enc}}]$, $\mathcal{B}_4$ sets

$$\begin{aligned}
&\mathbf{g} = \left(g_1^s, g_1^{s\omega_1}, \ldots, g_1^{s\omega_{2n}}\right),\\
&\mathbf{g}^a = \left(g_1^{sa}, g_1^{s\omega_1 a}, \ldots, g_1^{s\omega_{2n} a}\right);\\
&\widehat{\mathbf{g}} = \left(g_2^{\hat{s}}, g_2^{\hat{s}\omega_2}, \ldots, g_2^{\hat{s}\omega_{2n}}\right),\\
&\widehat{\mathbf{g}}^a = \left(g_2^{\hat{s}a}, g_2^{\hat{s}\omega_2 a}, \ldots, g_2^{\hat{s}\omega_{2n} a}\right);\\
&\left(g_2^{\hat{s}}\right)^b, g_{\{2,4\}}, g_{\{2,4\}}^b, g_{\{2,4\}}^c,
\end{aligned} \tag{C.25}$$

where $s, \hat{s} \xleftarrow{R} \mathbb{Z}_N^*$. Then $\mathcal{B}_4$ runs the algorithm $\mathsf{Rerand}_{abc}(N, \mathbf{g}, \mathbf{g}^a, \widehat{\mathbf{g}}, \widehat{\mathbf{g}}^a, (g_2^{\hat{s}})^b, g_{\{2,4\}}, g_{\{2,4\}}^b, g_{\{2,4\}}^c, T)$ in [19] and outputs $(\mathbf{g}^{a_l}, \widehat{\mathbf{g}}^{a_l}, (g_2^{\hat{s}})^{b_l}, g_{\{2,4\}}^{b_l}, g_{\{2,4\}}^{c_l}, T_{abc}^l)$, where

$$\begin{aligned}
\mathbf{g}^{a_l} &= \left(g_1^{sa_l}, g_1^{s\omega_1 a_l}, \ldots, g_1^{s\omega_{2n} a_l}\right) = \Big(\left(g_1^{sa}\right)^{r_1}\\
&\quad \cdot \left(g_1^{s}\right)^{t_1}, \left(g_1^{s\omega_1 a}\right)^{r_1} \cdot \left(g_1^{s\omega_1}\right)^{t_1}, \ldots, \left(g_1^{s\omega_{2n} a}\right)^{r_1}\\
&\quad \cdot \left(g_1^{s\omega_{2n}}\right)^{t_1}\Big),\\
\widehat{\mathbf{g}}^{a_l} &= \left(\widehat{g}_2^{\hat{s}a_l}, \widehat{g}_2^{\hat{s}\omega_1 a_l}, \ldots, \widehat{g}_2^{\hat{s}\omega_{2n} a_l}\right) = \Big(\left(\widehat{g}_2^{\hat{s}a}\right)^{r_1}\\
&\quad \cdot \left(\widehat{g}_2^{\hat{s}}\right)^{t_1}, \left(\widehat{g}_2^{\hat{s}\omega_1 a}\right)^{r_1} \cdot \left(\widehat{g}_2^{\hat{s}\omega_1}\right)^{t_1}, \ldots, \left(\widehat{g}_2^{\hat{s}\omega_{2n} a}\right)^{r_1}\\
&\quad \cdot \left(\widehat{g}_2^{\hat{s}\omega_{2n}}\right)^{t_1}\Big),\\
\left(g_2^{\hat{s}}\right)^{b_l} &= \left(g_2^{\hat{s}b}\right)^{r_2} \cdot \left(g_2^{\hat{s}}\right)^{t_2} = \left(g_2^{\hat{s}}\right)^{br_2 + t_2},\\
g_{\{2,4\}}^{b_l} &= \left(g_{\{2,4\}}^{b}\right)^{r_2} \cdot g_{\{2,4\}}^{t_2} = g_{\{2,4\}}^{br_2 + t_2},\\
g_{\{2,4\}}^{c_l} &= \left(g_{\{2,4\}}^{c}\right)^{r_3} \cdot g_{\{2,4\}}^{t_3} = g_{\{2,4\}}^{cr_3 + t_3},\\
T_{abc}^l &= e\left(g_2^{\hat{s}}, g_{\{2,4\}}\right)^{z_{abc}},
\end{aligned} \tag{C.26}$$

where $r_1, r_2, r_3, t_1, t_2, t_3 \xleftarrow{R} \mathbb{Z}_N^*$, $z_a = zr_1 + bct_1$; $z_{ab} = z_a r_2 + a_l c t_2$, $z_{abc} = z_{ab} r_3 + a_l b_l t_3$. It is easy to check that a_l, b_l, c_l

are uniformly distributed in $\mathbb{Z}_N$. If $z = abc$, then $z_a = a_l bc$, $z_{ab} = a_l b_l c, z_{abc} = a_l b_l c_l$ and $T^l_{abc} = T^{a_l b_l c_l}$. For the case $z \neq abc$, since $a, b, c, z, r_1, t_1, r_2, t_2, r_3, t_3 \leftarrow \mathbb{Z}^*_N$ and a_l, b_l, c_l are uniformly distributed over $\mathbb{Z}_N$, we have z_a, z_{ab}, z_{abc} all uniformly distributed in $\mathbb{Z}_N$ and $T^l_{abc} = e(\widehat{g}_2^{\hat{s}}, g_{\{2,4\}})^{z_{abc}}$.

(i) If the challenge identity was not queried before, then $\mathcal{B}_4$ computes

$$\left(\mathbf{g}^{a_l}, \widehat{\mathbf{g}}^{a_l}, \left(g_2^{\hat{s}}\right)^{b_l}, g_{\{2,4\}}^{b_l}, g_{\{2,4\}}^{c_l}, T^l_{abc}\right) \longleftarrow \mathsf{Rerand}_{abc}\left(N, \mathbf{g}, \mathbf{g}^a, \widehat{\mathbf{g}}, \widehat{\mathbf{g}}^a, \left(g_2^{\hat{s}}\right)^b, g_{\{2,4\}}, g^b_{\{2,4\}}, g^c_{\{2,4\}}, T\right) \tag{C.27}$$

and returns

$$\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \mathsf{msk}\right) \cdot T^l_{abc}\right)\right). \tag{C.28}$$

(ii) If the challenge identity was queried before, then $\mathcal{B}_4$ uses the algorithm Rerand_a in [19] to compute

$$\left(\mathbf{g}^{a'_l}, \widehat{\mathbf{g}}^{a'_l}, \left(g_2^{\hat{s}}\right)^{b_l}, g_{\{2,4\}}^{b_l}, g_{\{2,4\}}^{c_l}, T'_l\right) \longleftarrow \mathsf{Rerand}_a\left(N, \mathbf{g}, \mathbf{g}^{a_l}, \widehat{\mathbf{g}}, \widehat{\mathbf{g}}^{a_l}, \left(g_2^{\hat{s}}\right)^{b_l}, g_{\{2,4\}}, g_{\{2,4\}}^{b_l}, g_{\{2,4\}}^{c_l}, T^l_{abc}\right) \tag{C.29}$$

and returns

$$\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a'_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a'_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a'_l}, \mathsf{msk}\right) \cdot T'_l\right)\right). \tag{C.30}$$

The distribution of mpk and the requested user secret keys are identical to the real scheme.

If $T = e(g_2, g_2)^{abc} = e((g_1 g_2)^a, g_{\{2,4\}})^{bc}$ and the challenge identity was not queried before,

$$\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \mathsf{msk}\right) \cdot T^l_{abc}\right) = \left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \mathsf{msk} \cdot \left(g_{\{2,4\}}\right)^{b_l c_l}\right)\right)\right). \tag{C.31}$$

Note that the exponents a, b, c, and z are required to be uniformly distributed in $\mathbb{Z}^*_N$, but when we reuse the outputs of Rerand_a and Rerand_b, a_l, b_l are uniformly distributed in $\mathbb{Z}_N$. Since the uniform distribution in $\mathbb{Z}_N$ is statistically indistinguishable from the uniform distribution in $\mathbb{Z}^*_N$, we have that the distribution of challenge ciphertexts are $\mathbf{O}(2^{-\tau})$-close to that of Game_3. Note that we implicitly set $\widehat{\gamma}_l = b_l c_l$. For the challenge identity queried before, we can just rerandomize the previously used query value a_l.

If $T = e(g_2, g_2)^z$, then

$$\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \mathsf{msk}\right) \cdot T^l_{abc}\right)\right) = \left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \left(\prod_{j=1}^{n}\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{\omega_{2j-s_j}}\right)^{a_l}, \mathsf{H}\left(e\left(\left(g_1^s \widehat{g}_2^{\hat{s}}\right)^{a_l}, \mathsf{msk} \cdot \left(g_{\{2,4\}}\right)^{z_{abc} \cdot a_l^{-1}}\right)\right)\right). \tag{C.32}$$

Since a_l, z_{abc} are uniformly distributed over $\mathbb{Z}_N$. So $e((g_1^s \widehat{g}_2^{\hat{s}})^{a_l}, \mathsf{msk} \cdot (g_{\{2,4\}})^{z_{abc} \cdot a_l^{-1}})$ is statically close to the uniform distribution of subgroup of G_T. Due to the property of universal hash function we have $\mathsf{SD}((\mathsf{H}, \mathsf{H}(e((g_1^s \widehat{g}_2^{\hat{s}})^{a_l}, \mathsf{msk} \cdot (g_{\{2,4\}})^{z_{abc} \cdot a_l^{-1}}))); (\mathsf{H}, U)) = \mathbf{O}(2^{-\tau})$ for $U \leftarrow \{0,1\}^\tau$. Thus the challenge ciphertexts are distributed $\mathbf{O}(2^{-\tau})$-close to that of Game_4. Hence (C.21) holds.

Finally, we have

$$\mathsf{Adv}^{\mathrm{MA}}_{\mathscr{A},\mathsf{BE}}(\lambda) \leq \mathsf{Adv}^{\mathrm{DS1}}_{\mathscr{B}_1}(\lambda) + 2n \cdot \mathsf{Adv}^{\mathrm{DS2}}_{\mathscr{B}_2}(\lambda) + n \cdot \mathsf{Adv}^{\mathrm{DS3}}_{\mathscr{B}_3}(\lambda) + \mathsf{Adv}^{\text{DS-BDDH}}_{\mathscr{B}_4} + q_{\mathrm{Enc}} \cdot \mathbf{O}\left(2^{-\tau}\right), \tag{C.33}$$

which completes the proof of Theorem 6. □

Disclosure

Parts of this paper are presented at Inscrypt 2016.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

All authors were funded by National 973 Grant 2013CB834205, NSFC Grant 61672019, and The Fundamental Research Funds of Shandong University Grant 2016JC029. Puwen Wei was also funded by NSFC Grant 61502276.

References

[1] A. Fiat and M. Naor, "Broadcast encryption," in *Advances in Cryptology—(CRYPTO '93)*, D. R. Stinson, Ed., vol. 773 of *Lecture Notes in Computer Science*, pp. 480–491, Springer, Berlin, Germany, 1993.

[2] I. Kim and S. O. Hwang, "An optimal identity-based broadcast encryption scheme for wireless sensor networks," *IEICE Transactions on Communications*, vol. E96-B, no. 3, pp. 891–895, 2013.

[3] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," in *Proceedings of the Advances in Cryptology—(CRYPTO '01), 21st Annual International Cryptology Conference*, pp. 41–62, Springer, Berlin, Germany.

[4] D. Halevy and A. Shamir, "The LSD broadcast encryption scheme," in *Advances in Cryptology—(CRYPTO '02)*, M. Yung, Ed., vol. 2442 of *Lecture Notes in Computer Science*, pp. 47–60, Springer, Berlin, Germany, 2002.

[5] M. T. Goodrich, J. Z. Sun, and R. Tamassia, "Efficient tree-based revocation in groups of low-state devices," in *Advances in Cryptology—(CRYPTO '04)*, M. K. Franklin, Ed., vol. 3152 of *Lecture Notes in Computer Science*, pp. 511–527, Springer, Berlin, Germany, 2004.

[6] Y. Dodis and N. Fazio, "Public key broadcast encryption for stateless receivers," in *Security and Privacy in Digital Rights Management, ACM CCS-9 Workshop*, J. Feigenbaum, Ed., vol. 2696 of *Lecture Notes in Computer Science*, pp. 61–80, Springer, Berlin, Germany, 2003.

[7] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," in *Advances in Cryptology—(CRYPTO '05)*, V. Shoup, Ed., vol. 3621 of *Lecture Notes in Computer Science*, pp. 258–275, Springer, Berlin, Germany, 2005.

[8] C. Gentry and B. Waters, "Adaptive security in broadcast encryption systems (with short ciphertexts)," in *Advances in Cryptology—(EUROCRYPT '09)*, A. Joux, Ed., vol. 5479 of *Lecture Notes in Computer Science*, pp. 171–188, Springer, Berlin, Germany, 2009.

[9] B. Waters, "Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions," in *Advances in Cryptology—(CRYPTO '2009)*, S. Halevi, Ed., vol. 5677 of *Lecture Notes in Computer Science*, pp. 619–636, Springer, Berlin, Germany, 2009.

[10] D. Boneh, B. Waters, and M. Zhandry, "Low overhead broadcast encryption from multilinear maps," in *Advances in Cryptology—(CRYPTO '14)*, J. A. Garay and R. Gennaro, Eds., vol. 8616 of *Lecture Notes in Computer Science*, pp. 206–223, Springer, Berlin, Germany, 2014.

[11] D. Boneh and B. Waters, "A fully collusion resistant broadcast, trace, and revoke system," in *Proceedings of CCS 2006: 13th ACM Conference on Computer and Communications Security*, pp. 211–220, Alexandria, Virginia, USA, November 2006.

[12] J. H. Han, J. H. Park, and D. H. Lee, "Transmission-efficient broadcast encryption scheme with personalized messages," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E96-A, no. 4, pp. 796–806, 2013.

[13] M. Zhang, B. Yang, Z. Chen, and T. Takagi, "Efficient and adaptively secure broadcast encryption systems," *Security and Communication Networks*, vol. 6, no. 8, pp. 1044–1052, 2013.

[14] J. Kim, W. Susilo, M. H. Au, and J. Seberry, "Adaptively secure identity-based broadcast encryption with a constant-sized ciphertext," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 679–693, 2015.

[15] S. H. Islam, M. K. Khan, and A. M. Al-Khouri, "Anonymous and provably secure certificateless multireceiver encryption without bilinear pairing," *Security and Communication Networks*, vol. 8, no. 13, pp. 2214–2231, 2015.

[16] H. Wee, "Déjà Q: Encoreé un petit IBE," in *Theory of Cryptography—13th International Conference, TCC 2016-A*, E. Kushilevitz and T. Malkin, Eds., vol. 9563 of *Lecture Notes in Computer Science*, pp. 237–258, Springer, Berlin, Germany, 2016.

[17] M. Bellare, A. Boldyreva, and S. Micali, "Public-key encryption in a multi-user setting: security proofs and improvements," in *Advances in Cryptology—(EUROCRYPT '2000)*, B. Preneel, Ed., vol. 1807 of *Lecture Notes in Computer Science*, pp. 259–274, Springer, Berlin, Germany, 2000.

[18] D. Hofheinz and T. Jager, "Tightly secure signatures and public-key encryption," *Designs, Codes and Cryptography. An International Journal*, vol. 80, no. 1, pp. 29–61, 2016.

[19] D. Hofheinz, J. Koch, and C. Striecks, "Identity-based encryption with (almost) tight security in the multi-instance, multi-ciphertext setting," in *Public-key cryptography—(PKC '15)*, J. Katz, Ed., vol. 9020 of *Lecture Notes in Computer Science*, pp. 799–822, Springer, Berlin, Germany, 2015.

[20] J. Chen and H. Wee, "Fully, (almost) tightly secure IBE and dual system groups," in *Advances in Cryptology—(CRYPTO '13)*, R. Canetti and J. A. Garay, Eds., vol. 8043 of *Lecture Notes in Computer Science*, pp. 435–460, Springer, Berlin, Germany, 2013.