Adaptive Security in Broadcast Encryption Systems (with Short Ciphertexts)

Craig Gentry[1,⋆] and Brent Waters[2,⋆⋆]

[1] Stanford University and IBM
cgentry@cs.stanford.edu
[2] University of Texas at Austin
bwaters@cs.utexas.edu

Abstract. We present new techniques for achieving *adaptive* security in broadcast encryption systems. Previous work on fully collusion resistant broadcast encryption systems with very short ciphertexts was limited to considering only static security.

First, we present a new definition of security that we call *semi-static* security and show a generic "two-key" transformation from semi-statically secure systems to adaptively secure systems that have comparable-size ciphertexts. Using bilinear maps, we then construct broadcast encryption systems that are semi-statically secure in the standard model and have constant-size ciphertexts. Our semi-static constructions work when the number of indices or identifiers in the system is polynomial in the security parameter.

For identity-based broadcast encryption, where the number of potential indices or identifiers may be exponential, we present the first adaptively secure system with sublinear ciphertexts. We prove security in the standard model.

1 Introduction

Broadcast encryption systems [17] allow a sender, who wants to send a message to a dynamically chosen subset $S \subseteq [1, n]$ of users, to construct a ciphertext such that only users in S can decrypt; the sender can then safely transmit this ciphertext over a broadcast channel to all users. It is preferable if the system is *public key* (anybody can encrypt), permits *stateless receivers* (users do not need to update their private keys), and is *fully collusion resistant* (even if all users outside of S collude, they cannot decrypt). Typically in this paper, when we speak of a broadcast encryption system, we will assume that it has these properties. The main challenge in building efficient broadcast systems is to encrypt messages with *short* ciphertexts.

Traditionally, broadcast encryption systems have relied on combinatorial techniques. Such systems include a collusion bound t, where using larger values of

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-3-642-01001-9_35

⋆ Supported by IBM Fellowship and Herbert Kunzel Stanford Graduate Fellowship.
⋆⋆ Supported by NSF CNS-0716199; and the U.S. Department of Homeland Security under Grant Award Number 2006-CS-001-000001.

A. Joux (Ed.): EUROCRYPT 2009, LNCS 5479, pp. 171–188, 2009.

t impacts system performance. If an adversary compromises more than t keys, the system would no longer guarantee security even for encryptions solely to uncompromised users. Among systems that are fully collusion resistant, the ciphertext typically grows linearly with either the number of privileged receivers (in the broadcast subset) or the number of revoked users [22,15,20,19,24]. Recently, Boneh, Gentry, and Waters [8] broke through this barrier. They presented new methods for achieving *fully collusion resistant* systems with short (i.e., $\mathcal{O}(\lambda)$, where λ is the security parameter) ciphertexts by applying computational techniques using groups with bilinear maps. However, they used a *static* model of security in which an adversary declares the target set S^* of his challenge ciphertext *before* even seeing the system parameters.

Unfortunately, the weaker static model of security does not capture the power of several types of attackers. Attackers might choose which keys to attempt to compromise and ciphertexts to attack based on the system parameters or the structure of previously compromised keys. To capture general attackers we must use an *adaptive* definition of security.

Adaptive Security. We would like to achieve a system that is provably fully collusion resistant under adaptive attacks. Arguably, this is the “right” model for security in broadcast encryption systems.

Achieving this goal, however, seems challenging. In a security reduction, intuitively, we would expect that a simulation must know all the private keys requested by the attacker, but not know any of the private keys for S^*, the set encrypted to in the challenge ciphertext. Once the public parameters are published, the simulator is essentially bound to what keys it knows. Therefore, in the adaptive setting it might appear that the best we can do in a reduction is to simply guess what keys the adversary might request. Unfortunately, for a system with n users a reduction might guess correctly only a negligible (in n) fraction of the time.

One approach for achieving adaptive security is to apply a hybrid argument. Instead of doing a reduction in one step, one can break the reduction into $n+1$ hybrid experiments $H_0, \ldots, H_n$, such that hybrid games H_i and H_{i+1} are indistinguishable to the adversary. In this reduction in Hybrid H_i the challenge ciphertext is to set $S^* \setminus [1, i]$, where S^* is the challenge specified by the adversary. Since each reduction in the hybrid games lops off only one user at a time, the reduction needs only to guess whether user $i+1$ will be in S^* when distinguishing between H_i and H_{i+1}, thus avoiding an exponential drop-off.

The key leverage that this solution needs is the ability to reduce the target set *anonymously.* This can be done with $\mathcal{O}(\lambda \cdot |S|)$ size ciphertexts. Recently, Boneh and Waters [10] achieved $\mathcal{O}(\lambda \cdot \sqrt{n})$ size ciphertexts. They combine the BGW broadcast techniques with the private linear broadcast techniques of Boneh, Sahai, and Waters [9] (that were originally designed for building traitor tracing techniques). Unfortunately, the $\sqrt{n}$ factor seems to be inherent in this approach with groups that have bilinear (as opposed to say trilinear) maps.

Our Methods. First, we introduce a new general technique for proving systems adaptively secure. The first component of our methodology is the introduction of the *semi-static* model of security. In the semi-static model of security an attacker must first commit to a set $\tilde{S}$ before setup, but then can later attack any set S^* that is a subset of $\tilde{S}$. This gives the attacker more flexibility than the static model, in which it had to exactly commit to the set it attacks.

At first glance the semi-static model might appear as simply a minor variant of the static model. However, we will also show a generic transformation from semi-static security to adaptive security. Suppose a ciphertext in the semi-static scheme was of size C for a set S of users; then in our transformation the ciphertexts will be of size $2 \cdot C$ plus $|S|$ bits. At the heart of our transformation is a two-key technique where two keys are assigned to each user, but the user is given only one of them. We note that our techniques are partially inspired from those used by Katz and Wang [21] to achieve tight security for IBE systems in the random oracle model.

Using this transformation we might simply hope to prove the BGW system to be semi-statically secure. Unfortunately, the BGW proof of security requires an "exact cancellation" and there is not an obvious way to prove BGW to be semi-statically secure. Instead, we provide two new constructions with constant-size ciphertexts, and prove semi-static security in the standard model. The first construction is a variant of the BGW that still has short ciphertexts, but that requires longer-size private keys. Like the BGW encryption system, we prove our security under the decisional Bilinear Diffie-Hellman Exponent (BDHE) assumption.

Our first construction has two principal limitations. First, it has long private keys. Second, our semi-static transformation works only when $n = \text{poly}(\lambda)$, since the time complexities of the security reductions are at least linear in n. For identity-based broadcast encryption (IBBE), where n may be exponential in λ, we use a different approach.

To solve these problems we use techniques from the Gentry IBE system [18]. We begin by building an "initial" identity-based broadcast encryption system with core component of size $\mathcal{O}(\lambda)$ plus an additional "tag" of size $\mathcal{O}(\lambda \cdot |S|)$. The tag represents a random polynomial in $\mathbb{Z}_p$. The public key is of size $\mathcal{O}(\ell \cdot \lambda)$ for when we can broadcast to at most ℓ users.

While a system with ciphertexts of size $\mathcal{O}(\lambda \cdot |S|)$ is not immediately useful, we can build on this in several ways.

- First, we show that for standard (non-identity-based) broadcast systems we can omit the tag and achieve $\mathcal{O}(\lambda)$ size ciphertexts and private keys while retaining semi-static security.
- Second, we show how in the random oracle model the tag can be generated from a short $O(\lambda)$ size seed and get adaptively secure ID-based broadcast encryption with $O(\lambda)$ size ciphertexts.
- Finally, in the standard model we show how to achieve ID-based encryption with $\mathcal{O}(\lambda \cdot \sqrt{|S|})$ size ciphertexts. In this approach we essentially perform $\sqrt{|S|}$ encryptions to $\sqrt{|S|}$ of the recipients, but share one tag polynomial across all these encryptions.

We prove the security of this base scheme and its derivatives under a new *non-interactive* assumption.

1.1 Related Work

Dodis and Fazio [16] showed how to build an adaptively secure revocation system building upon the techniques of Cramer and Shoup [12] and Naor and Pinkas [23]. In their system the ciphertext size is $\mathcal{O}(\lambda \cdot |R|)$, where R is the set of *revoked* users.

Delerablée, Paillier, and Pointcheval [14] describe a system that is somewhat incomparable to ours and the others discussed here; it allows the adversary to wait until just before each dynamic join operation to declare whether it is joining as an honest or corrupt party (the challenge broadcast is for the honest parties), but then each join operation triggers a change to the public key.

The concept of identity-based broadcast encryption (IBBE) was proposed in [13] (and independently in [27]). This concept is related to identity-based encryption [25], in which the maximal size of a broadcast group is $\ell = 1$. It is also related to multi receiver ID-based KEM (mID-KEM), introduced in [26] and further developed in [4,5,11,2]. We also note that Panjwani [1] considered adaptive corruptions, but in the context of *stateful* protocols such as Logical Key Hierarchy.

2 Adaptive Security in Broadcast Encryption

We present background material on broadcast encryption systems. Then we show our main transformation; we describe how to build adaptive securely broadcast encryption systems from those that are secure against a "semi-static" adversary.

2.1 Broadcast Encryption Systems

We begin by formally defining the notion of security for a public-key broadcast encryption system. For simplicity we define broadcast encryption as a key encapsulation mechanism. In addition, we make our definition general enough to capture identity-based encryption systems.

A broadcast encryption system is made up of four randomized algorithms:

Setup(n, ℓ) Takes as input the number of receivers n and the maximal size $\ell \leq n$ of a broadcast recipient group. It outputs a public/secret key pair $\langle PK, SK \rangle$. (We leave another input, the input security parameter λ, implicit.)

KeyGen(i, SK) Takes as input an index $i \in \{1, \ldots, n\}$ and the secret key SK. It outputs a private key d_i.

Enc(S, PK) Takes as input a subset $S \subseteq \{1, \ldots, n\}$ and a public key PK. If $|S| \leq \ell$, it outputs a pair $\langle \mathrm{Hdr}, K \rangle$ where Hdr is called the header and $K \in \mathcal{K}$ is a message encryption key.

Let $\mathcal{E}_{sym}$ be a symmetric encryption scheme with key-space $\mathcal{K}$, and algorithms *SymEnc* and *SymDec*. Let M be a message to be broadcast to the set S, and let $C_M \overset{R}{\leftarrow} \mathit{SymEnc}(K, M)$ be the encryption of M under the symmetric key K. The broadcast to users in S consists of $\langle S, \mathrm{Hdr}, C_M \rangle$.

Dec$(S, i, d_i, \mathrm{Hdr}, PK)$ Takes as input a subset $S \subseteq \{1, \ldots, n\}$, an index $i \in \{1, \ldots, n\}$, a private key d_i for i, a header Hdr, and the public key PK. If $|S| \leq \ell$ and $i \in S$, then the algorithm outputs the message encryption key $K \in \mathcal{K}$. The key K can then be used to decrypt C_M to obtain M.

As usual, we require that the system be correct, namely, that for all $S \subseteq \{1, \ldots, n\}^{\leq \ell}$ and all $i \in S$, if $\langle PK, SK \rangle \overset{R}{\leftarrow} \mathit{Setup}(n, \ell)$, $d_i \overset{R}{\leftarrow} \mathit{KeyGen}(i, SK)$, and $\langle \mathrm{Hdr}, K \rangle \overset{R}{\leftarrow} \mathit{Enc}(S, PK)$, then
$\mathit{Dec}(S, i, d_i, \mathrm{Hdr}, PK) = K$.

Our goal is to illustrate the issues for adaptive security. For simplicity, we define security against chosen plaintext attacks. However, our definitions can readily be extended to reflect chosen-ciphertext attacks.

2.2 Security Definitions

Arguably, the "correct" definition for security in broadcast encryption systems is that of adaptive security. In an adaptively secure system, the adversary is allowed to see PK and then ask for several private keys before choosing the set of indices that it wishes to attack.

Adaptive security in broadcast encryption is defined using the following game between an attack algorithm $\mathcal{A}$ and a challenger. Both the challenger and $\mathcal{A}$ are given n and ℓ as input.

Setup. The challenger runs $\mathit{Setup}(n, \ell)$ to obtain a public key PK, which it gives to the adversary.

Key Query Phase. Algorithm $\mathcal{A}$ adaptively issues private key queries for indices $i \in \{1, \ldots, n\}$.

Challenge. The adversary then specifies a challenge set S^*, such that for all private keys i queried we have that $i \notin S^*$. The challenger sets $\langle \mathrm{Hdr}^*, K_0 \rangle \overset{R}{\leftarrow} \mathit{Enc}(S^*, PK)$ and $K_1 \overset{R}{\leftarrow} \mathcal{K}$. It sets $b \overset{R}{\leftarrow} \{0, 1\}$ and gives (Hdr^*, K_b) to algorithm $\mathcal{A}$.

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

We define $\mathcal{A}$'s advantage in attacking the broadcast encryption system BE with parameters (n, ℓ) and security parameter λ as

$$\mathsf{AdvBr}_{\mathcal{A},\mathsf{BE},n,\ell}(\lambda) = \left| \Pr[b = b'] - \frac{1}{2} \right|$$

We may omit the system name when it can be understood from the context.

Definition 1. *We say that a broadcast encryption system* BE *is* adaptively *secure if for all poly-time algorithms* $\mathcal{A}$ *we have that* $\mathsf{AdvBr}_{\mathcal{A},\mathsf{BE},n,\ell}(\lambda) = \mathrm{negl}(\lambda)$.

In addition to the adaptive game for broadcast security, we consider two other weaker security notions. The first is *static* security, where the adversary must commit to the set S^* of identities that it will attack in an "Init" phase before the setup algorithm is run. This is the security definition that is used by recent broadcast encryption systems [8].

We also propose a new security definition called *semi-static* security. In this game the adversary must commit to a set $\tilde{S}$ of indices at the Init phase. The adversary cannot query a private key for any $i \in \tilde{S}$, and it must choose a target group S^* for the challenge ciphertext that is a subset of $\tilde{S}$. A semi-static adversary is weaker than an adaptive adversary, but it is stronger than a static adversary, in that its choice of *which* subset of $\tilde{S}$ to attack can be adaptive.

2.3 Transforming Semi-static Security to Adaptive Security

At first the benefits of achieving semi-static security versus just static security might appear incremental. Indeed, in both games, the adversary is forced to restrict its queries before it even sees the public key.

Despite this apparent shortcoming, we will show that the semi-static security definition is a very useful tool for achieving adaptive security. We will show how to transform any semi-static broadcast encryption scheme to one secure under adaptive attacks with a modest increase in overhead.

Our main idea is to apply a simulation for a two-key technique. In such a system each user will be associated with two potential private keys; however, the authority will give it only one of the two. An encryptor (that does not know which private key the receiver possesses) will need to encrypt the ciphertext twice, once for each key. This technique was used by Katz and Wang [21] to create tightly secure signature and identity-based encryption systems in the random oracle model.

The main benefit is that a simulator will have the private keys for every identity. In the Katz-Wang constructions this enabled tight security reductions. In the context of broadcast encryption, the impact will be much stronger, since trying to guess S^* would otherwise result in an exponential loss of security in the reduction. We now show how to apply the two-key idea to broadcast encryption.

Suppose we are given a semi-static secure broadcast system BE_{SS} with algorithms $\mathit{Setup}_{\mathrm{SS}}$, $\mathit{KeyGen}_{\mathrm{SS}}$, $\mathit{Enc}_{\mathrm{SS}}$, $\mathit{Dec}_{\mathrm{SS}}$. Then we can build our adaptively secure broadcast system BE_A as follows.

Setup(n, ℓ)**:** Run $\langle PK', SK' \rangle \xleftarrow{R} \mathit{Setup}_{\mathrm{SS}}(2n, \ell)$. Set $s \xleftarrow{R} \{0,1\}^n$. Set $PK \leftarrow PK'$ and $SK \leftarrow (SK', s)$. Output $\langle PK, SK \rangle$.

KeyGen(i, SK)**:** Run $d_i' \xleftarrow{R} \mathit{KeyGen}_{\mathrm{SS}}(2i - s_i, SK')$. Set $d_i \leftarrow \langle d_i', s_i \rangle$. Output d_i.

Enc(S, PK)**:** Generate a random set of $|S|$ bits: $t \leftarrow \{t_i \xleftarrow{R} \{0,1\} : i \in S\}$. Generate $K \xleftarrow{R} \mathcal{K}$. Set

$$S_0 \leftarrow \{2i - t_i : i \in S\}\ , \qquad \langle \mathrm{Hdr}_0, \kappa_0 \rangle \xleftarrow{R} \mathit{Enc}_{\mathrm{SS}}(S_0, PK')$$
$$S_1 \leftarrow \{2i - (1 - t_i) : i \in S\}\ , \quad \langle \mathrm{Hdr}_1, \kappa_1 \rangle \xleftarrow{R} \mathit{Enc}_{\mathrm{SS}}(S_1, PK')$$

Set $C_0 \xleftarrow{R} \mathit{SymEnc}(\kappa_0, K), C_1 \xleftarrow{R} \mathit{SymEnc}(\kappa_1, K)$, $\mathrm{Hdr} \leftarrow \langle \mathrm{Hdr}_0, C_0, \mathrm{Hdr}_1, C_1, t \rangle$. Output $\langle \mathrm{Hdr}, K \rangle$.

Dec$(S, i, d_i, \mathrm{Hdr}, PK)$**:** Parse d_i as $\langle d_i', s_i \rangle$ and Hdr as $\langle \mathrm{Hdr}_0, C_0, \mathrm{Hdr}_1, C_1, t \rangle$. Set S_0 and S_1 as above. Run

$$\kappa_{s_i \oplus t_i} \leftarrow \mathit{Dec}_{\mathrm{SS}}(S_{s_i \oplus t_i}, i, d_i', \mathrm{Hdr}_{s_i \oplus t_i}, PK')$$

Run $K \leftarrow \mathit{SymDec}(\kappa_{s_i \oplus t_i}, C_{s_i \oplus t_i})$. Output K.

Note that, aside from the string t, the BE_A ciphertext is only about twice as long as a BE_{SS} ciphertext. Suppose that we have a semi-static broadcast encryption system in which ciphertexts are "constant-size" – i.e., $\mathcal{O}(\lambda)$ for security parameter λ. Then, our transformation gives an adaptively secure broadcast encryption system with ciphertexts that are $\mathcal{O}(\lambda + |S|)$, versus $\mathcal{O}(\lambda \cdot |S|)$. In particular, the ciphertext size in BE_A increases by only one bit per additional recipient.

Later, we will describe a semi-static broadcast encryption system in which Hdr contains only two group elements of, say, 200 bits apiece – a total of 400 bits. As an example, suppose we apply the transformation above to this scheme to encrypt to 1000 users, and use AES for the symmetric system. In this case, the Hdr size of the induced adaptively-secure broadcast encryption system is $2 \cdot 400 + 2 \cdot 128 + 1000 = 2056$ bits, versus say $400 \cdot 1000 = 400000$ bits.

It is easy to see that, assuming BE_A is adaptively secure, we can get adaptively secure broadcast encryption system with truly constant-size $\mathcal{O}(\lambda)$ ciphertexts in the random oracle model as follows. Put a hash function $H : \{0,1\}^{\mathcal{O}(\lambda)} \times \{1, \ldots, n\} \to \{0,1\}$ in the public key. The sender encrypts as before, except that it generates t by setting $u \xleftarrow{R} \{0,1\}^{\mathcal{O}(\lambda)}$ and $t_i \leftarrow H(u, i)$; it replaces t by u in the ciphertext. The recipient decrypts as before, except that it recovers t from u using H.

Alternatively, without random oracles, we get an adaptively secure broadcast encryption system with $\mathcal{O}(\sqrt{\lambda \cdot |S|})$ size ciphertexts from a semi-static system with $\mathcal{O}(\lambda)$ size ciphertexts by partitioning the $|S|$ users into $\sqrt{|S|/\lambda}$ groups of $\sqrt{\lambda \cdot |S|}$ users, and then re-using the same $\sqrt{\lambda \cdot |S|}$-bit string t for every group. Asymptotically, this beats the adaptively-secure system of [10], but often the system above with $\mathcal{O}(\lambda + |S|)$ size ciphertexts will still be preferable in practice. Security follows from the security of the underlying semi-static system by a hybrid argument (omitted).

We now show that BE_A is secure if BE_{SS} is secure.

Theorem 1. *Let $\mathcal{A}$ be an adaptive adversary against BE_A. Then, there exist algorithms $\mathcal{B}_1$, $\mathcal{B}_2$, $\mathcal{B}_3$, and $\mathcal{B}_4$, each running in about the same time as $\mathcal{A}$, such that*

$$\begin{aligned}\mathsf{AdvBr}_{\mathcal{A},\mathsf{BE}_A,n,\ell}(\lambda) \leq\ & \mathsf{AdvBrSS}_{\mathcal{B}_1,\mathsf{BE}_{SS},2n,\ell}(\lambda) \ + \ \mathsf{AdvBrSS}_{\mathcal{B}_2,\mathsf{BE}_{SS},2n,\ell}(\lambda) \\ & + \mathsf{AdvSym}_{\mathcal{B}_3,\mathcal{E}_{sym}}(\lambda) \ + \ \mathsf{AdvSym}_{\mathcal{B}_4,\mathcal{E}_{sym}}(\lambda)\end{aligned}$$

Proof. We present the proof as a sequence of games. Let W_i denote the event that $\mathcal{A}$ wins game i.

Game 0. The first game is identical to the adaptive security game given above. Thus,

$$\left|\Pr[W_0] - \frac{1}{2}\right| = \mathsf{AdvBr}_{\mathcal{A},\mathsf{BE}_A,n,\ell}(\lambda) \tag{1}$$

Game 1. Game 1 is identical to Game 0, except that the challenger generates C_0 in the challenge ciphertext as follows: set $\kappa_0^\dagger \xleftarrow{R} \mathcal{K}$ and then $C_0 \xleftarrow{R} \mathit{SymEnc}(\kappa_0^\dagger, K_0)$.

We claim that there exists an algorithm $\mathcal{B}_1$, whose running time is about the same as $\mathcal{A}$, such that

$$|\Pr[W_1] - \Pr[W_0]| = \mathsf{AdvBrSS}_{\mathcal{B}_1,\mathsf{BE}_{SS},2n,\ell}(\lambda) \tag{2}$$

To break BE_{SS}, $\mathcal{B}_1$ sets $s \xleftarrow{R} \{0,1\}^n$ and $\tilde{S} \leftarrow \{2i - (1 - s_i) : i \in \{1, \ldots, n\}\}$. It sends $\tilde{S}$ to the challenger, which sends back PK'. $\mathcal{B}$ sets $PK \leftarrow PK'$ and forwards PK to $\mathcal{A}$.

When $\mathcal{A}$ queries the BE_A private key for $i \in \{1, \ldots, n\}$, $\mathcal{B}$ queries the challenger for the BE_{SS} private key for $2i - s_i$. The challenger sends back d_i'; $\mathcal{B}$ sends (d_i', s_i) to $\mathcal{A}$.

$\mathcal{A}$ requests a challenge ciphertext on some $S^* \subseteq \{1, \ldots, n\}$. $\mathcal{B}$ sets $t \leftarrow \{t_i \leftarrow 1 - s_i : i \in S^*\}$. It sets $S_0 \leftarrow \{2i - t_i : i \in S^*\}$ and $S_1 \leftarrow \{2i - (1 - t_i) : i \in S^*\}$, and queries the challenger for a challenge ciphertext on S_0. The challenger sends back $(\mathrm{Hdr}_0, \kappa_0^{(b)})$, where b denotes the bit flipped by the challenger. $\mathcal{B}$ sets $(\mathrm{Hdr}_1, \kappa_1) \xleftarrow{R} \mathit{Enc}(S_1, PK')$. It generates $K_0, K_1 \xleftarrow{R} \mathcal{K}$, $b^\dagger \xleftarrow{R} \{0,1\}$, $C_0 \xleftarrow{R} \mathit{SymEnc}(\kappa_0^{(b)}, K_0)$ and $C_1 \xleftarrow{R} \mathit{SymEnc}(\kappa_1, K_0)$. It sets $\mathrm{Hdr} \leftarrow \langle \mathrm{Hdr}_0, C_0, \mathrm{Hdr}_1, C_1, t\rangle$. It sends $(\mathrm{Hdr}, K_{b^\dagger})$ to $\mathcal{A}$.

Eventually, $\mathcal{A}$ outputs a bit b'. If $b' = b^\dagger$, $\mathcal{B}$ sends 0 to the challenger; else, it sends 1.

If $b = 0$, $\mathcal{A}$'s view is as in Game 0. The private keys sent by $\mathcal{B}$ are appropriately distributed. The string t appears to be uniformly random, since $\mathcal{A}$'s private key queries reveal only the values of s_i for $i \notin S^*$. Also, $\kappa_0^{(0)}$ is generated correctly, and so the dependent values are as well. If $b = 1$, $\mathcal{A}$'s view is as in Game 1. The claim follows.

Game 2. Game 2 is identical to Game 1, except that the challenger sets $\kappa_1 \xleftarrow{R} \mathcal{K}$ when constructing the challenge ciphertext. By an analysis similar to above, we conclude that there exists an algorithm $\mathcal{B}_2$, which runs in about the same time as $\mathcal{A}$, for which

$$|\Pr[W_2] - \Pr[W_1]| = \mathsf{AdvBrSS}_{\mathcal{B}_2,\mathsf{BE}_{SS},2n,\ell}(\lambda) \tag{3}$$

Game 3. Game 3 is identical to Game 2, except that the challenger sets $K_0^\dagger \xleftarrow{R} \mathcal{K}$ and $C_0 \xleftarrow{R} \mathit{SymEnc}(\kappa_0^\dagger, K_0^\dagger)$. We claim that there exists an algorithm $\mathcal{B}_3$, which runs in about the same time as $\mathcal{A}$, for which

$$|\Pr[W_3] - \Pr[W_2]| = \mathsf{AdvSym}_{\mathcal{B}_3,\mathcal{E}_{sym}}(\lambda) \tag{4}$$

This follows, since it is straightforward to construct $\mathcal{B}_3$ as an algorithm that attacks the semantic security of $\mathcal{E}_{sym}$.

Game 4. Game 4 is identical to Game 3, except that the challenger sets $K_1^\dagger \xleftarrow{R} \mathcal{K}$ and $C_1 \xleftarrow{R} SymEnc(\kappa_1^\dagger, K_1^\dagger)$. As above, we obtain

$$|\Pr[W_4] - \Pr[W_3]| = \mathsf{AdvSym}_{\mathcal{B}_4,\mathcal{E}_{sym}}(\lambda) \qquad (5)$$

Finally, the theorem follows if the following claim is true:

$$\left|\Pr[W_4] - \frac{1}{2}\right| = 0 \qquad (6)$$

This claim follows since, in Game 4, Hdr is independent of K_b, and hence b.

3 BE Construction with Small Ciphertexts

Now that we have our transformation of semi-static security to adaptive security, we would like to leverage it to create new adaptively secure broadcast encryption systems. One obvious candidate to examine is the Boneh-Gentry-Waters [8] broadcast encryption system. Unfortunately, it was proven only to be statically secure and there does not appear to be an obvious way to make the proof semi-static.[1]

To prove semi-static security we will need to use a variant of the BGW system. We first describe our construction. Then we describe the decisional-BDHE assumption (the same one used by BGW). Then we prove our system to be semi-statically secure under this assumption.

3.1 Our Construction

Let $GroupGen(\lambda, n)$ be an algorithm that, on input security parameter λ, generates groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order $p = p(\lambda, n) > n$ with bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.

Setup(n, n)**:** Run $\langle \mathbb{G}, \mathbb{G}_T, e\rangle \xleftarrow{R} GroupGen(\lambda, n)$. Set $\alpha \xleftarrow{R} \mathbb{Z}_p$ and $g, h_1, \ldots, h_n \xleftarrow{R} \mathbb{G}^{n+1}$. Set PK to include a description of $\langle \mathbb{G}, \mathbb{G}_T, e\rangle$, as well as

$$g \ , \quad e(g,g)^\alpha \ , \quad h_1 \ , \ \ldots, \quad h_n .$$

The secret key is $SK \leftarrow g^\alpha$. Output $\langle PK, SK\rangle$

KeyGen(i, SK)**:** Set $r_i \xleftarrow{R} \mathbb{Z}_p$ and output

$$d_i \leftarrow \langle d_{i,0}, \ldots, d_{i,n}\rangle \text{ where } d_{i,0} \leftarrow g^{-r_i} \ , \quad d_{i,i} \leftarrow g^\alpha h_i^{r_i} \ , \quad \forall_{j \neq i}\, d_{i,j} \leftarrow h_j^{r_i}$$

[1] The BGW reduction depends upon an *exact* cancellation between a value embedded by the simulator in the parameters and a function of the target set S^*.

Enc(S, PK)**:** Set $t \xleftarrow{R} \mathbb{Z}_p$ and

$$\text{Hdr} \leftarrow \langle C_1, C_2 \rangle \quad \text{where} \quad C_1 \leftarrow g^t \ , \quad C_2 \leftarrow (\prod_{j \in S} h_j)^t$$

Set $K \leftarrow e(g,g)^{\alpha \cdot t}$. Output $\langle \text{Hdr}, K \rangle$.

Dec$(S, i, d_i, \text{Hdr}, PK)$**:** If $i \in S$, parse d_i as $\langle d_{i,0}, \dots, d_{i,n} \rangle$ and Hdr as $\langle C_1, C_2 \rangle$ and output

$$K \leftarrow e(d_{i,i} \cdot \prod_{j \in S \setminus \{i\}} d_{i,j}, C_1) \cdot e(d_{i,0}, C_2)$$

Correctness: We check that decryption recovers the correct value of K.

$$\begin{aligned} e(d_{i,i} \cdot \prod_{j \in S \setminus \{i\}} d_{i,j}, C_1) \cdot e(d_{i,0}, C_2) = \ & e(g^\alpha \cdot (\prod_{j \in S} h_j)^{r_i}, g^t) \cdot e(g^{-r_i}, (\prod_{j \in S} h_j)^t) \\ = \ & e(g,g)^{\alpha \cdot t} \end{aligned}$$

as required.

3.2 The BDHE Assumption

We base the security of the above system on the decision BDHE assumption, used in [8]. The decision BDHE problem is as follows.

Definition 2 (Decision BDHE problem (for m)). *Let $\mathbb{G}$ and $\mathbb{G}_T$ be groups of order p with bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, and let g be a generator for $\mathbb{G}$. Set $a, s \xleftarrow{R} \mathbb{Z}_p^*$ and $b \xleftarrow{R} \{0,1\}$. If $b = 0$, set $Z \leftarrow e(g,g)^{a^{m+1} \cdot s}$; else, set $Z \xleftarrow{R} \mathbb{G}_T$. The problem instance consists of g^s, Z, and the set*

$$\{g^{\alpha^i} : i \in [0, m] \cup [m+2, 2m]\}$$

The problem is to guess b.

We define $\mathsf{AdvBDHE}_{\mathcal{A},m}(\lambda)$ in the expected way. We have the following theorem.

Theorem 2. *Let $\mathcal{A}$ be a semi-static adversary against the above system. Then, there is an algorithm $\mathcal{B}$, which runs in about the same time as $\mathcal{A}$, such that*

$$\mathsf{AdvBrSS}_{\mathcal{A},n,n}(\lambda) = \mathsf{AdvBDHE}_{\mathcal{B},n}(\lambda)$$

We provide the proof in Appendix A.

3.3 Semi-static BE with Small Ciphertexts and Private Keys

In the semi-static system described in Section 3, the public key and private keys are of size $\mathcal{O}(\lambda \cdot n)$. However, we have an alternative construction that has a public key of size $\mathcal{O}(\lambda \cdot \ell)$ and constant-sized private keys (i.e., $\mathcal{O}(\lambda)$). This construction is a special case of the identity-based broadcast encryption system that we provide in Section 4.1. We provide more details in Section 4.3.

4 Identity-Based BE with Small Ciphertexts and Private Keys

The essential property of an *identity-based* broadcast encryption (IBBE) system is that it remains efficient when n is exponential in the security parameter λ. Adaptive security is even more challenging in this setting. In particular, our semi-static constructions do not give adaptively secure IBBE, since the time complexities of the reduction algorithms are at least linear in n.

Here we first describe an initial IBBE system with adaptive security, where the ciphertext size is constant aside from a random "tag" that has length $\mathcal{O}(\lambda \cdot |S|)$. This long tag is needed by the simulator to handle the fact that the adversary chooses the target set S^* adaptively. The public key has size $\mathcal{O}(\lambda \cdot \ell)$, and private keys are constant size (i.e., $\mathcal{O}(\lambda)$). This system is an extension of Gentry's IBE system [18].

At first, a system with such a long tag appears to be pointless. However, there are several ways to address this apparent problem. First, for polynomial-size n, we show that the system is semi-statically secure if we replace the random tag with a *constant* tag; the ciphertext size then becomes constant. Second, we make the straightforward observation that, in the random oracle model, we obtain an adaptively secure IBBE system with constant-size ciphertexts if we generate the tag from the random oracle. Finally, we construct an adaptively secure IBBE system (in the standard model) that, for a recipient group of size $k \leq \ell$, has $\mathcal{O}(\lambda \cdot \sqrt{k})$-size ciphertexts, a $\mathcal{O}(\lambda \cdot \sqrt{\ell})$-size public key, and still constant-size private keys, by reusing the same $\mathcal{O}(\lambda \cdot \sqrt{k})$-size tag in $\mathcal{O}(\sqrt{k})$ separate sub ciphertexts from the initial system. As far as we know, this is the first IBBE system with sub-linear ciphertexts secure against adaptive adversaries.

4.1 An Initial IBBE Construction

Let $GroupGen(\lambda, n, \ell)$ be an algorithm that outputs suitable bilinear group parameters $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, where $\mathbb{G}$ is of order $p \geq n + \ell$.

Setup(n, ℓ)**:** Run $\langle \mathbb{G}, \mathbb{G}_T, e \rangle \overset{R}{\leftarrow} GroupGen(\lambda, n, \ell)$. Set $g_1, g_2 \overset{R}{\leftarrow} \mathbb{G}$. Set $\alpha, \beta, \gamma \overset{R}{\leftarrow} \mathbb{Z}_p$. Set $\hat{g}_1 \leftarrow g_1^{\beta}$ and $\hat{g}_2 \leftarrow g_2^{\beta}$. PK contains a description of $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, the parameters n and ℓ, along with g_1^{γ}, $g_1^{\gamma \cdot \alpha}$ and the set

$$\{g_1^{\alpha^j}, \hat{g}_1^{\alpha^j}, g_2^{\alpha^k}, \hat{g}_2^{\alpha^k} : j \in [0, \ell], k \in [0, \ell - 2]\}$$

Generate a random key κ for a PRF $\Psi : [1, n] \to \mathbb{Z}_p$. The private key is $SK \leftarrow (\alpha, \gamma, \kappa)$.

KeyGen(i, SK)**:** Similar to Gentry's IBE system, set $r_i \leftarrow \Psi_\kappa(i)$ and output the private key

$$d_i \leftarrow \langle r_i, h_i \rangle \ , \quad \text{where} \quad h_i \leftarrow g_2^{\frac{\gamma - r_i}{\alpha - i}}$$

Enc(S, PK)**:** Run $\tau \overset{R}{\leftarrow} TagGen(S,PK)$. Output $\langle \text{Hdr}, K \rangle \overset{R}{\leftarrow} TagEncrypt(\tau,S,PK)$.

TagGen(S, PK)**:** Let $k = |S|$. Set $F(x) \in \mathbb{Z}_p[x]$ to be a random $(\ell-1)$-degree polynomial such that $F(n+j) = 1$ for $j \in [k+1, \ell]$. Output $\tau \leftarrow F(x)$. Note that τ can be expressed by k values in $\mathbb{Z}_p$ – e.g., $\{F(i) : i \in S\}$; $F(x)$ can be interpolated from these values and $\{F(n+j) = 1 : j \in [k+1, \ell]\}$.

TagEncrypt(τ, S, PK)**:** Parse τ as $F(x)$ and S as $\{i_1, \dots, i_k\}$. Set $i_j \leftarrow n + j$ for $j \in [k+1, \ell]$. Set $P(x) = \prod_{j=1}^{\ell}(x - i_j)$. Set $t \xleftarrow{R} \mathbb{Z}_p$ and set $K \leftarrow e(g_1, \hat{g}_2)^{\gamma \cdot \alpha^{\ell-1} \cdot t}$. Next, set

$$\text{Hdr} \leftarrow \langle C_1, \dots, C_4 \rangle \leftarrow \langle \hat{g}_1^{P(\alpha)\cdot t} \ , \quad g_1^{\gamma \cdot t} \ , \quad g_1^{F(\alpha)\cdot t} \ , \quad e(g_1, \hat{g}_2)^{\alpha^{\ell-1}\cdot F(\alpha) \cdot t} \rangle \ .$$

Output $\langle \tau, \text{Hdr}, K \rangle$.

Dec$(S, i, d_i, \tau, \text{Hdr}, PK)$**:** Suppose $i \in S = \{i_1, \dots, i_k\}$. Parse d_i as $\langle r_i, h_i \rangle$, τ as $F(x)$, and Hdr as $\langle C_1, \dots, C_4 \rangle$. Define $P(x)$ as above. Let

$$P_i(x) = x^{\ell-1} - \frac{P(x)}{(x-i)} \ , \quad F_i(x) = \frac{F(x) - F(i)}{x - i} \ , \quad \text{and} \quad e_i = -\frac{r_i}{F(i)} \ .$$

Set

$$K \leftarrow e(C_1, h_i \cdot g_2^{e_i \cdot F_i(\alpha)}) \cdot e(C_2 \cdot C_3^{e_i}, \hat{g}_2^{P_i(\alpha)})/C_4^{e_i} \qquad (7)$$

Note that the recipient can compute $g_2^{F_i(\alpha)}$ and $\hat{g}_2^{P_i(\alpha)}$ from PK, since $F_i(x)$ and $P_i(x)$ are polynomials of degree $\ell - 2$.

Correctness: We verify that decryption recovers the message. First, we note that $K = K_1 \cdot K_2$, where we gather the terms containing a γ in K_1, and the other terms in K_2. (Recall $h_i = g_2^{\gamma/(\alpha-i)} \cdot g_2^{-r_i/(\alpha-i)}$.)

$$\begin{aligned} K_1 &= e(C_1, g_2^{\gamma})^{1/(\alpha-i)} \cdot e(C_2, \hat{g}_2^{P_i(\alpha)}) \\ K_2 &= e(C_1, g_2^{-r_i/(\alpha-i)+e_i\cdot F_i(\alpha)}) \cdot e(C_3, \hat{g}_2^{P_i(\alpha)})^{e_i}/C_4^{e_i} \end{aligned}$$

We have that

$$K_1^{1/t} = e(g_1, \hat{g}_2)^{\gamma(P(\alpha)/(\alpha-i)+P_i(\alpha))} = e(g_1, \hat{g}_2)^{\gamma \cdot \alpha^{\ell-1}}$$

We also have that

$$\begin{aligned} K_2^{1/t} &= e(g_1, \hat{g}_2)^{-r_i \cdot P(\alpha)/(\alpha-i) + e_i \cdot P(\alpha)\cdot F_i(\alpha) + e_i \cdot P_i(\alpha) \cdot F(\alpha) - e_i \cdot \alpha^{\ell-1} \cdot F(\alpha)} \\ &= e(g_1, \hat{g}_2)^{e_i \cdot P(\alpha) \cdot F(\alpha)/(\alpha-i) + e_i \cdot P_i(\alpha) \cdot F(\alpha) - e_i \cdot \alpha^{\ell-1} \cdot F(\alpha)} \\ &= e(g_1, \hat{g}_2)^{e_i \cdot F(\alpha)\left(P(\alpha)/(\alpha-i)+P_i(\alpha)-\alpha^{\ell-1}\right)} \\ &= e(g_1, \hat{g}_2)^0 \quad = \quad 1 \end{aligned}$$

as required.

4.2 Security of the Initial IBBE Construction

Below, we define a class of assumptions that is narrower than the general bilinear DH exponent "uber-assumption" defined by Boneh et al. [6], but broad enough to cover some frequently used assumptions. One reason that we think carving out this class of assumptions is useful is that it is much easier to glance at an assumption in this class and verify that it at least superficially makes sense than it is for some of the wilder assumptions within general BDHE.

Definition 3 (The Decision BDHE Sum Problem for (S, m)). *Fix $S \subset \mathbb{Z}$ and $m \in \mathbb{Z} \setminus (S + S)$. Let $\mathbb{G}$ and $\mathbb{G}_T$ be groups of order p with bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, and let g be a generator for $\mathbb{G}$. Set $\alpha \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and $b \overset{R}{\leftarrow} \{0, 1\}$. If $b = 0$, set $Z \leftarrow e(g, g)^{\alpha^m}$; otherwise, set $Z \overset{R}{\leftarrow} \mathbb{G}_T$. Output*

$$\{g^{\alpha^i} : i \in S\} \quad \text{and} \quad Z$$

The problem is to guess b.

In the decision n-BDHI problem, $S = [0, n]$ and $m = -1$. One can reduce the Decision BDHE Sum problem for $S = [0, n] \cup [n + 2, 2n] \cup [3n]$ and $m = 4n + 1$ to the decision BDHE problem for n – i.e., s in the BDHE problem is replaced by α^{3n}.

Although we do not use it in this paper, we mention an obvious (possibly easier) variant of the problem:

Definition 4 (The Decision BDHE Sum Problem for (S, m) (variant)). *As above, except Z is replaced in the instance by random $(z_1, z_2) \in \mathbb{G}^2$ satisfying $e(z_1, z_2) = Z$.*

A recent paper [3] builds the first adaptively secure hierarchical identity based encryption (HIBE) system that allows a polynomial number of levels by building on our IBBE system and using this variant of the Decision BDHE Sum problem.

We base the security of our system on the Decision BDHE Sum problem for $m = 4d + 4\ell - 1$ and

$$\begin{aligned} S = {} & [0, \ell - 2] \cup [d + \ell, 2d + \ell - 1] \cup [2d + 2\ell, 2d + 3\ell - 1] \\ & \cup [3d + 3\ell, 4d + 3\ell] \cup [4d + 4\ell, 5d + 4\ell + 1] \end{aligned}$$

where $d = q + 2\ell$, q and ℓ non-negative. We define $\mathsf{AdvBDHES}_{\mathcal{A},q,\ell}(\lambda)$ in the expected way, using these particular values of S and m.

We have the following theorem.

Theorem 3. *Let $\mathcal{A}$ be an adaptive adversary against the above initial IBBE system that makes at most q queries. Then, there exist algorithms $\mathcal{B}_1$ and $\mathcal{B}_2$ such that*

$$\mathsf{AdvBr}_{\mathcal{A},n,\ell}(\lambda) \leq \mathsf{AdvPRF}_{\mathcal{B}_1,\Psi}(\lambda) \; + \; \mathsf{AdvBDHES}_{\mathcal{B}_2,q,\ell}(\lambda) \; + \; (\ell + 2)/p \qquad (8)$$

where $\mathcal{B}_1$ runs in about the same time as $\mathcal{A}$, and $\mathcal{B}_2$ runs in time $t(\mathcal{A}) + \mathcal{O}((q + \ell)^2 \cdot \lambda^3)$, assuming exponentiations take time $\mathcal{O}(\lambda^3)$.

We provide the proof in Appendix B.

4.3 Variants of the IBBE Construction

Semi-Static BE with Constant-Size Ciphertexts and Private Keys. When $n = \text{poly}(\lambda)$, we obtain a semi-statically secure variant of the above system with constant-size ciphertexts by making the following simple change.

TagGen(S, PK)**:** Output $\tau \leftarrow F(x) \leftarrow 1$.

Since τ is always 1, we do not need to include it in the ciphertext. Also, some terms in PK become unnecessary – in particular, $\{g_2^{\alpha^i} : i \in [0, \ell - 2]\}$.

We have the following theorem. Let $q = n$.

Theorem 4. *Let $\mathcal{A}$ be a semi-static adversary against the above system. Then, there exist algorithms $\mathcal{B}_1$ and $\mathcal{B}_2$ such that*

$$\mathsf{AdvBr}_{\mathcal{A},n,\ell}(\lambda) \leq \mathsf{AdvPRF}_{\mathcal{B}_1,\Psi}(\lambda) \ + \ \mathsf{AdvBDHES}_{\mathcal{B}_2,q,\ell}(\lambda) \ + \ (\ell+2)/p \quad (9)$$

where $\mathcal{B}_1$ runs in about the same time as $\mathcal{A}$ and $\mathcal{B}_2$ runs in time $t(\mathcal{A}) + \mathcal{O}((q+\ell)^2 \cdot \lambda^3)$, assuming exponentiations take time $\mathcal{O}(\lambda^3)$.

We prove this simultaneously with Theorem 3 in Appendix B.

Adaptively Secure IBBE with Constant-Size Ciphertexts in the ROM. In the random oracle model, the obvious way to modify the initial IBBE system to obtain constant-size ciphertexts is to generate τ using a hash function $H : \{0,1\}^{\mathcal{O}(\lambda)} \times [1, n] \rightarrow \mathbb{Z}_p$. In particular, we make the following modification.

TagGen(S, PK)**:** Output $\tau \leftarrow \{0,1\}^{\mathcal{O}(\lambda)}$.

In *TagEncrypt* and *Dec*, $F(x)$ is set to be the $(\ell - 1)$-degree polynomial that interpolates $F(i) = H(\tau, i)$ for $i \in S$ and $F(i) = 1$ for $i \in [n+j]$ with $j \in [k+1, \ell]$. The ciphertext size is constant, since the size of τ is constant (i.e., $\mathcal{O}(\lambda)$). We omit the easy tight reduction from an adversary that breaks the initial system to an adversary that breaks this system.

Adaptively Secure IBBE with Sublinear-Size Ciphertexts. Let $\ell = \ell_1 \cdot \ell_2$. Below, we describe a system that builds on the initial IBBE system and allows one to encrypt to a set S with $|S| = k_1 \cdot k_2$, $k_1 \leq \ell_1$, $k_2 \leq \ell_2$.

Setup$_{\boldsymbol{SL}}(n, \ell)$**:** Run $(PK', SK') \leftarrow$ *Setup*(n, ℓ_2). Set $PK \leftarrow (PK', \ell_1)$ and $SK \leftarrow SK'$. Output $\langle PK, SK \rangle$.

KeyGen$_{\boldsymbol{SL}}(i, SK)$**:** Run $d_i \xleftarrow{R}$ *KeyGen*(i, SK'). Output d_i.

Encrypt$_{\boldsymbol{SL}}(S, PK)$**:** Partition S into $k_1 \leq \ell_1$ sets $\langle S_1, \ldots, S_{k_1} \rangle$ of size $k_2 \leq \ell_2$. Run $\tau \xleftarrow{R}$ *TagGen*(S_1, PK'). Generate $K \xleftarrow{R} \mathcal{K}$. For $j \in [1, k_1]$, set

$$\langle \text{Hdr}_j, \kappa_j \rangle \xleftarrow{R} \mathit{TagEncrypt}(\tau, S_j, PK') \ , \quad c_j \leftarrow \mathit{SymEnc}(\kappa_j, K)$$

Set $\text{Hdr} \leftarrow \langle \text{Hdr}_1, c_1, \ldots, \text{Hdr}_{k_1}, c_{k_1} \rangle$. Output $\langle \tau, \text{Hdr}, K \rangle$.

Decrypt$_{SL}$ $(S, i, d_i, \tau, \mathrm{Hdr}, PK)$**:** Parse Hdr as $\langle \mathrm{Hdr}_1, c_1, \ldots, \mathrm{Hdr}_{k_1}, c_{k_1} \rangle$ and S as $\langle S_1, \ldots, S_{k_1} \rangle$. Suppose $i \in S_j$. Run

$$\kappa_j \leftarrow Dec(S_j, i, d_i, \tau, \mathrm{Hdr}_j, PK') \quad \text{and} \quad K \leftarrow SymDec(\kappa_j, c_j)$$

Output K.

We have the following theorem.

Theorem 5. *Let $\mathcal{A}$ be an adaptive adversary against this system that makes at most q queries. Then, there exist algorithms $\mathcal{B}_1$ and $\mathcal{B}_2$, the former being an adversary against the initial IBBE system that makes at most q queries, each algorithm running in about the same time as $\mathcal{A}$, such that*

$$\mathsf{AdvBr}_{\mathcal{A},n,\ell}(\lambda) \leq \ell_1 \cdot \Big(\mathsf{AdvBr}_{\mathcal{B}_1,n,\ell_2}(\lambda) \ + \ \mathsf{AdvSym}_{\mathcal{B}_2,\mathcal{E}_{sym}}(\lambda)\Big) \tag{10}$$

As before $\mathcal{E}_{sym}$ is a symmetric encryption scheme. We omit the proof, since it is a simple hybrid argument similar to the proof of Theorem 1.

It is easy to handle the case where $|S|$ cannot be expressed as a product $k_1 \cdot k_2$ with $k_1, k_2 = \mathcal{O}(\sqrt{|S|})$. Let S' consist of the first $k_1 \cdot k_2$ identities in S, where $k_1 = k_2 = \lfloor \sqrt{|S|} \rfloor$. Encrypt to S' using the above system, and to $S \setminus S'$ using any reasonable system – e.g., the initial system. The overall size of the ciphertext is still $\mathcal{O}(\lambda \cdot \sqrt{|S|})$. One can prove the security of this double encryption by a sequence of games similar to the proof of Theorem 1.

References

1. Panjwani, S.: Tackling adaptive corruptions in multicast encryption protocols. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 21–40. Springer, Heidelberg (2007)
2. Abdalla, M., Kiltz, E., Neven, G.: Generalized Key Delegation for Hierarchical Identity-Based Encryption. In: Biskup, J., López, J. (eds.) ESORICS 2007. LNCS, vol. 4734, pp. 139–154. Springer, Heidelberg (2007)
3. Anonymous. Hierarchical Identity Based Encryption with Polynomially Many Levels (Manuscript, 2008)
4. Baek, J., Safavi-Naini, R., Susilo, W.: Efficient Multi-receiver Identity-Based Encryption and Its Application to Broadcast Encryption. In: Vaudenay, S. (ed.) PKC 2005. LNCS, vol. 3386, pp. 380–397. Springer, Heidelberg (2005)
5. Barbosa, M., Farshim, P.: Efficient Identity-Based Key Encapsulation to Multiple Parties. In: Smart, N.P. (ed.) Cryptography and Coding 2005. LNCS, vol. 3796, pp. 428–441. Springer, Heidelberg (2005)
6. Boneh, D., Boyen, X., Goh, E.-J.: Hierarchical Identity Based Encryption with Constant Size Ciphertext. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 440–456. Springer, Heidelberg (2005)
7. Boneh, D., Gentry, C., Hamburg, M.: Space Efficient Identify Based Encryption without Pairings. In: FOCS 2007 (2007)
8. Boneh, D., Gentry, C., Waters, B.: Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 258–275. Springer, Heidelberg (2005)

9. Boneh, D., Sahai, A., Waters, B.: Fully Collusion Resistant Traitor Tracing with Short Ciphertexts and Private Keys. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 573–592. Springer, Heidelberg (2006)
10. Boneh, D., Waters, B.: A Fully Collusion Resistant Broadcast, Trace, and Revoke System. In: CCS 2006 (2006)
11. Chatterjee, S., Sarkar, P.: Multi-receiver Identity-Based Key Encapsulation with Shortened Ciphertext. In: Barua, R., Lange, T. (eds.) INDOCRYPT 2006. LNCS, vol. 4329, pp. 394–408. Springer, Heidelberg (2006)
12. Cramer, R., Shoup, V.: Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (2002)
13. Delerablée, C.: Identity-Based Broadcast Encryption with Constant Size Ciphertexts and Private Keys. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 200–215. Springer, Heidelberg (2007)
14. Delerablée, C., Paillier, P., Pointcheval, D.: Fully Collusion Secure Dynamic Broadcast Encryption with Constant-Size Ciphertexts or Decryption Keys. In: Takagi, T., Okamoto, T., Okamoto, E., Okamoto, T. (eds.) Pairing 2007. LNCS, vol. 4575, pp. 39–59. Springer, Heidelberg (2007)
15. Dodis, Y., Fazio, N.: Public Key Broadcast Encryption for Stateless Receivers. In: Feigenbaum, J. (ed.) DRM 2002. LNCS, vol. 2696, pp. 61–80. Springer, Heidelberg (2003)
16. Dodis, Y., Fazio, N.: Public Key Trace and Revoke Scheme Secure against Adaptive Chosen Ciphertext Attack. In: Desmedt, Y.G. (ed.) PKC 2003. LNCS, vol. 2567, pp. 100–115. Springer, Heidelberg (2002)
17. Fiat, A., Naor, M.: Broadcast Encryption. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 480–491. Springer, Heidelberg (1994)
18. Gentry, C.: Practical Identity-Based Encryption Without Random Oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer, Heidelberg (2006)
19. Goodrich, M.T., Sun, J.Z., Tamassia, R.: Efficient Tree-Based Revocation in Groups of Low-State Devices. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 511–527. Springer, Heidelberg (2004)
20. Halevy, D., Shamir, A.: The LSD Broadcast Encryption Scheme. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 47–60. Springer, Heidelberg (2002)
21. Katz, J., Wang, N.: Efficiency Improvements for Signature Schemes with Tight Security Reductions. In: CCS 2003 (2003)
22. Naor, D., Naor, M., Lotspiech, J.: Revocation and Tracing Schemes for Stateless Receivers. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 41–62. Springer, Heidelberg (2001)
23. Naor, M., Pinkas, B.: Efficient Trace and Revoke Schemes. In: Frankel, Y. (ed.) FC 2000. LNCS, vol. 1962, pp. 1–20. Springer, Heidelberg (2001)
24. Sharmila Deva Selvi, S., Sree Vivek, S., Gopalakrishnan, R., Karuturi, N.N., Pandu Rangan, C.: Provably Secure ID-Based Broadcast Signcryption (IBBSC) Scheme. Eprint 2008/225
25. Shamir, A.: Identity-Based Cryptosystems and Signature Schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
26. Smart, N.P.: Efficient Key Encapsulation to Multiple Parties. In: Blundo, C., Cimato, S. (eds.) SCN 2004. LNCS, vol. 3352, pp. 208–219. Springer, Heidelberg (2005)
27. Sakai, R., Furukawa, J.: Identity-Based Broadcast Encryption. Eprint 2007/217

A Proof of Theorem 2

$\mathcal{B}$ receives the problem instance, which includes g^s, Z, and the set

$$\{g^{a^i} : i \in [0,n] \cup [n+2, 2n]\}$$

Init $\mathcal{A}$ commits to a set $\tilde{S} \subseteq [1,n]$.

Setup $\mathcal{B}$ generates $y_0, \ldots, y_n \xleftarrow{R} \mathbb{Z}_p$. It sets

$$\begin{array}{lllll} h_i & \leftarrow & g^{y_i} & \text{for} \;\; i & \in \;\; \tilde{S} \\ h_i & \leftarrow & g^{y_i + a^i} & \text{for} \;\; i & \in \;\; [1,n] \setminus \tilde{S} \end{array}$$

Formally, $\mathcal{B}$ sets $\alpha \leftarrow y_0 \cdot a^{n+1}$. It sets PK to include a description of $\langle \mathbb{G}, \mathbb{G}_T, e \rangle$, as well as

$$g \;,\;\; e(g,g)^\alpha \;,\;\; h_1 \;,\; \ldots,\;\; h_n$$

where $e(g,g)^\alpha$ can be computed as $e(g^a, g^{a^n})^{y_0}$. $\mathcal{B}$ sends PK to $\mathcal{A}$.

Private Key Queries $\mathcal{A}$ is allowed to query the private key only for indices $i \in [1,n] \setminus \tilde{S}$. To answer the query, $\mathcal{B}$ generates $z_i \xleftarrow{R} \mathbb{Z}_p$ and formally sets $r_i \leftarrow z_i - y_0 \cdot a^{n+1-i}$. It outputs

$$d_i \leftarrow \langle d_{i,0}, \ldots, d_{i,n} \rangle \quad \text{where} \quad d_{i,0} \leftarrow g^{-r_i} \;, \quad d_{i,i} \leftarrow g^\alpha h_i^{r_i} \;, \quad \forall_{j \neq i} \; d_{i,j} \leftarrow h_j^{r_i}$$

Notice that $\mathcal{B}$ can compute all these terms from the instance; in particular

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

which can be computed since the a^{n+1} term in the exponent cancels out.

Challenge $\mathcal{A}$ chooses a subset $S^* \subset \tilde{S}$. $\mathcal{B}$ sets

$$\text{Hdr} \leftarrow \langle C_1, C_2 \rangle \quad \text{where} \quad C_1 \leftarrow g^s \;, \quad C_2 \leftarrow (\prod_{j \in S^*} h_j)^s$$

It sets $K \leftarrow Z^{y_0}$. It sends $\langle \text{Hdr}, K \rangle$ to $\mathcal{A}$.

Notice that $\mathcal{B}$ can compute these terms from the instance. C_1 and K come directly from the instance. $\mathcal{B}$ can compute C_2 since it knows $\mathrm{DL}_g(h_i)$ for all $i \in S^*$; in particular,

$$C_2 = (\prod_{j \in S^*} h_j)^s = (\prod_{j \in S^*} g^{y_j})^s = (g^s)^{\sum_{j \in S^*} y_j}$$

Guess Eventually, $\mathcal{A}$ outputs a bit b'. $\mathcal{B}$ sends b' to the challenger.

Perfect Simulation From $\mathcal{A}$'s perspective, $\mathcal{B}$'s simulation has exactly the same distribution as the semi-static game defined in Section 2.2. The public and private keys are appropriately distributed, since α and the values $\{\mathrm{DL}_g(h_i)\}$ and $\{r_i\}$ are uniformly random and independent.

When $b = 0$ in the semi-static game, $\langle \mathrm{Hdr}, K\rangle$ is generated according to the same distribution as in the real world. This is also true in $\mathcal{B}$'s simulation: when $b = 0$, $K = e(g, g)^{\alpha \cdot s}$, and so the challenge is valid ciphertext under randomness s. When $b = 1$ in the semi-static game, $\langle \mathrm{Hdr}, K'\rangle$ is generated as in the real world, but K' is replaced by $K \xleftarrow{R} \mathcal{K}$, and $\langle \mathrm{Hdr}, K\rangle$ is sent to the adversary. This distribution is identical to that of $\mathcal{B}$'s simulation, where Hdr is valid for randomness s, but $K = Z$ is a uniformly random element of $\mathbb{G}_T$.

From this, we see that $\mathcal{B}$'s advantage in deciding the BDHE instance is precisely $\mathcal{A}$'s advantage against BE_{SS}.

B Proof of Theorems 3 and 4

First, a lemma. Let $p(x)q(x)|i$ denote the coefficient of x^i in $p(x)q(x)$.

Lemma 1. *Let $f_1(x), f_2(x) \in \mathbb{F}_p[x]$ be polynomials of degrees d_1 and d_2, respectively, whose resultant is nonzero. Let $d_3 \leftarrow d_1 + d_2 - 1$ and $i \in \{d_1, \ldots, d_3\}$. There exists a polynomial $t(x) \in \mathbb{F}_p[x]$ of degree d_3 such that $t(x)f_1(x)|_i = 1$, $t(x)f_1(x)|_j = 0$ for $j \in \{d_1, \ldots, d_3\} \setminus \{i\}$, and $t(x)f_2(x)|_j = 0$ for $j \in \{d_2, \ldots, d_3\}$.*

Proof. (Lemma 1) Consider the Sylvester matrix S of $f_1(x)$ and $f_2(x)$. The condition on $t(x)$ is equivalent to $S \cdot (t_0, \ldots, t_{d_3})^T = (0, \ldots, 0, 1, 0, \ldots, 0)^T$, where $t_i = t(x)|_i$. Since the resultant of $f_1(x)$ and $f_2(x)$ is nonzero, the Sylvester matrix is invertible. Set $(t_0, \ldots, t_{d_3})^T \leftarrow S^{-1} \cdot (0, \ldots, 0, 1, 0, \ldots, 0)^T$ and $t(x) = \sum_i t_i x^i$.

The complexity of computing $t(x)$ is $\mathcal{O}(d_2(d_1 + d_2))$ arithmetic operations over $\mathbb{Z}_p$.

Proof. (Theorems 3 and 4) This is given in the full version.