Chosen Ciphertext Secure Broadcast Threshold Encryption (resp. Threshold-Traitor Tracing)

Victor K. Wei[1] and Fangguo Zhang[2]

[1] Chinese University of Hong Kong, Hong Kong, {kwwei}@ie.cuhk.edu.hk
[2] Sun Yat-Sen University, China isszhfg@mail.sysu.edu.cn

August 18, 2006

Abstract. Recently, Boneh, Gentry, and Waters '05 presented an efficient broadcast encryption, and Boneh, Sahai, and Waters '06 presented an efficient traitor tracing scheme. The former broadcast encryption result contains both a simpler chosen plaintext secure version and a more complicated but chosen ciphertext secure version. The latter traitor tracing scheme is only chosen plaintext secure. In this paper, we use the twin encryption technique of Naor and Yung '90 to add chosen ciphertext security to both papers. By"twinning", we extend the simpler chosen plaintext secure broadcast encryption to achieve chosen ciphertext security, and we extend the chosen plaintext secure traitor tracing to achieve chosen ciphertext security. We also extend both schemes to versions corresponding to threshold encryption which we call "broadcast threshold encryption" and "threshold-traitor tracing", i.e. tracing of threshold traitors. In these schemes, any θ un-revoked users can decrypt while $\theta - 1$ users cannot. The tracing is to a set of θ users. We call this set a "threshold-traitor". Our broadcast threshold encryption is collusion resistant. Our threshold-traitor tracing is collusion resistant in its traceability.

1 Introduction

In a *broadcast encryption*, a center publishes a public key and distributes private keys to a number of users, and then can efficiently broadcast confidential messages to all users. In *public key* broadcast encryption, anyone can use to public key to broadcast. For each broadcast, an arbitrary set of users can be revoked of their decryption priviledges.

Broadcast encryption schemes often come with a *traitor tracing* scheme, where an adversary capable of decryption can be traced to at least one of the broadcast decryption keys it actually possesses. In *public* traitor tracing, no trapdoor is needed to trace the traitor, so anyone can do it. In *black-box* traitor tracing, the adversary is modelled as a black box such that none of its intermediate computations are observable and only inputs and outputs can be observed and interacted with.

Recently, Boneh, Gentry, and Waters [1] presented an efficient broadcast encryption, and Boneh, Sahai, and Waters [2] presented an efficient traitor tracing scheme. The broadcast encryption result [1] contains both a simpler chosen plaintext secure version and a more complicated but chosen ciphertext secure version. The traitor tracing scheme [2] is only chosen plaintext secure. In this paper, we use the twin encryption technique of Naor and Yung [10] to add chosen ciphertext security to both papers. By *twinning*, we extend the simpler chosen plaintext secure broadcast encryption to achieve chosen ciphertext security, and we extend the chosen plaintext secure traitor tracing to achieve chosen ciphertext security. We also extend both schemes to versions corresponding to threshold encryption which we call *broadcast threshold encryption* and *threshold-traitor tracing*, i.e. tracing of threshold traitors. In these schemes, any θ users can decrypt while $\theta - 1$ users cannot. Tracing is to a set of un-revoked users. We call this set a *threshold-traitor*.

Our **Contributions** are (N is the number of users, θ the threshold):

1. Boneh, Gentry, and Waters [1] included a simple chosen plaintext secure broadcast encryption and a more complicated chosen ciphertext secure broadcast encryption. We use Naor and Yung's twin encryption technique to add chose ciphertext security to [1]'s first simpler broadcast encryption scheme. The ciphertext (resp. private key for each user, public key) size remains $O(1)$ (resp. $O(1)$, $O(N)$).
2. Boneh, Gentry, and Waters [1] presented a chosen plaintext secure traitor tracing scheme. We use Naor and Yung's twin encryption technique to add chosen ciphertext security to it. The ciphertext (resp. private key for each user, public key) size remains $O(\sqrt{N})$ (resp. $O(1)$, $O(\sqrt{N})$).
3. We introduce the study of broadcast threshold encryption where the decryption required an arbitrary set of θ unrevoked users. We also construct one by modifying out broadcast encryption above. The ciphertext (resp. private key for each user, public key) size is $O(1)$ (resp. $O(N^{\theta-1})$, $O(N^{\theta})$).
4. We introduce the study of threshold-traitor tracing (tracing of threshold traitor), where the decryption requires an arbitrary set of θ unrevoked users, and the tracing is to a set of θ users, which we call a *threshold-traitor*. We construct one by extending our traitor tracing scheme above. The ciphertext (resp. private key for each user, public key) size is $O(N^{\theta/2})$ (resp. $O(N^{\theta-1})$, $O(N^{\theta/2})$).

[1]'s broadcast encryption is collusion resistant against semantic indistinguishability attackers. So is our broadcast threhold encryption, in the single inconsistent player model. [2]'s traitor tracing is a non-revoking traitor tracing and is collusion resistant against traceability attackers. So is our threshold-traitor tracing, in the single inconsistent player model.

Our intuitions about twin encryption: Broadcast encryption [10] is an effective and versatile method for upgrading a chosen plaintex secure encryption scheme to a chosen ciphertext secure one. It is simple to use. Just prepare two key pairs, encrypt the plaintext twice using the two keys, and then include a (zero-knowledge) proof that the "twin ciphertexts" decrypt to the same plaintext. Its security reduction is also easy, piggybacking on that of the chose plaintext encryption it upgrades. The notorious drawback is inefficiency. The proof that twin ciphertexts decrypt to the same can be long. Much more than just doubling the ciphertext length and encoding and decryption complexities. The increase is especially large when proofs of the the commit-challenge-response type is used.

In this paper, we use the twin encryption technique to upgrade the chosen plaintext secure broadcast encryption in [1] (resp. the chosen plaintext secure traitor tracing in [2]) to chosen ciphertext secure ones. Fortunately, the ciphertext length (resp. encoding complexity, decoding complexity) only about doubles. We attribute our luck to two reasons: (1) The use of gap Diffie-Hellman (GDP) groups in the scheme. (2) The luck of the original schemes in [1, 2]. Elaborations: (1) The proof that twin ciphertexts decrypt to the same plaintext involves only proving DDH (Decisional Diffie-Hellman) tuples, i.e. proving that certain tuples (a, b, c, d) are of the form (g, g^x, h, h^x). In the GDH group, such relations can be verified without additional transmissions such as commits, challenges, or responses. (2) Encrypting twice carefully in the broadcast encryption of [1] already contains sufficient relations to ensure that the twin ciphertexts decrypt to the same plaintext. Encrypting twice in the traitor tracing scheme of [2] is not sufficient, but only a small amount of additional ciphertext makes it sufficient. Summarizing (1) and (2), the ciphertext length increase in our result is modest. Consequently,

the computational complexity to compute the encryption (resp. the decryption) only increases only modestly.

Our intuitions about thresholding: We construct our broadcast threshold encryption (resp. threshold-traitor tracing) by a piggyback technique on the non-threshold predecessor. Let $N' = \binom{N}{\theta}$. A non-threshold broadcast encryption for N' users is used to construct a broadcast threshold encryption of N users and threshold θ as follows. Each group of θ decryptors is given a distinct group serial number ϕ, $1 \leq \phi \leq N'$. The ϕ-th user secret key usk_ϕ in the N'-user non-threshold broadcast encryption is (θ, θ)-secret-shared to all members of group ϕ in the Setup Stage. To broadcast threshold encrypt to users in S, just broadcast (non-threshold) encryption to all users $\phi_1, \cdots, \phi_k$, $k = \binom{|S|}{\theta}$, where each ϕ_ℓ corresponds to a θ-size subset of S. Due to the structure of the user secret key in [1], $\mathsf{usk}_\phi = g^{\alpha^\phi}$, collusion attack on our broadcast threshold encryption is as hard as that on the (non-threshold) broadcast encryption that it is based on.

Related results. In 1993, Fiat and Naor [4] explored broadcast encryption. Naor, et al. [9] presented a revoking and tracing scheme. [7, 3, 6] presented further results. Other broadcast encryption results: please see the excellent Related Work in [1]. For traitor tracing see the excellent Related Work in [2]. For threshold encryption, see Helger Lipmaa's links at http://www.cs.ut.ee/ lipmaa/crypto/link/threshold. Some papers on twin encryption and its applications are [5, 8, 11]

2 Security Model

We follow the security model of broadcast encryption (resp. traitor tracing) from [1] (resp. [2], except to add the threshold version. Brief summaries below.

2.1 Broadcast threshold encryption

A *broadcast threshold encryption* is a tuple (*Setup*, *Encrypt*, *Decrypt*) where

- $Setup(\lambda_s, N, \theta) \mapsto (\mathsf{pk}, \mathsf{usk}_1, \cdots \mathsf{usk}_N)$. Upon inputs the security parameter λ_s, the number of users N, and the threshold θ, outputs a public key pk, and user private keys usk_i, $1 \leq i \leq N$.
- $Encrypt(\mathsf{pk}, M, S) \mapsto \mathsf{ctxt}$. Upon inputs message M, public key pk, and a set of users $S \subset \{1, \cdots, N\}$, encrypt.
- $Decrypt(S, i_1, \mathsf{usk}_{i_1}, \cdots, i_\theta, \mathsf{usk}_{i_\theta}, \mathsf{ctxt}, \mathsf{pk}) \mapsto M$ or $\mathsf{InvalidCiphertext}$. Upon inputs pk, a user set $S \subset \{1, \cdots, N\}$, θ distinct users in S and their private keys, decrypt.

Correctness. For any $S \subset \{1, \cdots, N\}$, any distinct $i_1, \cdots, i_\theta \in S$, any M, we have $Decrypt(S, i_1, \mathsf{usk}_{i_1}, \cdots, i_\theta, \mathsf{usk}_{i_\theta}, Encrypt(\mathsf{pk}, M, S), \mathsf{pk}) = M$

Game IND

1. **Init.** Adversary $\mathcal{A}$ outputs a set $S_{ga} \subset \{1, \cdots, N\}$ it will not corrupt.
2. **Setup**. Simulator $\mathcal{B}$ runs *Setup*. Gives all usk_i, $i \notin S_{ga}$, to $\mathcal{A}$.
3. **Queries** In arbitrary interleaf, $\mathcal{A}$ makes q_D queries to the Decryption Oracle $\mathcal{DO}$.
4. **Gauntlet** At a certain point and in arbitrary interleaf with queries to $\mathcal{DO}$, $\mathcal{A}$ issues a message M_1, and a set of $\theta - 1$ users $S'_{ga} = \{i_1, \cdots, i_{\theta-1}\} \subset \{1, \cdots, N\} \setminus S_{ga}$, to $\mathcal{B}$. $\mathcal{B}$ select random message M_0, flips a fair coin b, and sends the *gauntlet ciphertext* $\mathsf{ctxt}_{ga} = Encrypt(\mathsf{pk}, M_b, S_{ga} \cup S'_{ga})$.

5. **Guess** At the end, $\mathcal{A}$ sends $\hat{b}$, its estimate of b.

$\mathcal{A}$ wins if $\hat{b} = b$ and $\mathcal{A}$ has never queried ctxt_{ga} to $\mathcal{DO}$. Its *advantage* is its probability of winning minus 1/2.

Definition 1. *A* broadcast threshold encryption *is* q_D-CCA secure *if it is* correct*, and no PPT algorithm has a non-negligible advantage in Game IND. It is* CPA secure *if it is* 0-CCA secure*.*

2.2 Threshold-traitor tracing

A *(non-revoking) threshold-traitor tracing* is a tuple ($Setup$, $Encrypt$, $Decrypt$, $Trace$) where

- $Setup(\lambda_s, N, \theta) \mapsto$ (pk,TK, usk_1, $\cdots$ usk_N). Note TK is the tracing key.
- $Encrypt(\mathsf{pk}, M) \mapsto \mathsf{ctxt}$.
- $Decrypt(i_1, \mathsf{usk}_{i_1}, \cdots, i_\theta, \mathsf{usk}_{i_\theta}, \mathsf{ctxt}, \mathsf{pk}) \mapsto M$ or $\mathsf{InvalidCiphertext}$.
- $Trace^{\mathcal{D}}(\mathsf{TK}, \epsilon)$. Upon inputs a black-box pirate decoder $\mathcal{D}$, tracing key TK, and a probability parameter ϵ, $0 < \epsilon < 1$, output a family $S = \{TT_1, \cdots, TT_\ell\}$. $\in \{1, \cdots, N\}^\theta$. Each TT_i, consisting of θ distinct users, is called a *threshold-traitor*.

Remark: We adopt [2]'s model of traitor tracing which is non-revoking. Any set of θ distinct users can always decrypt.

Correctness. For any distinct $i_1, \cdots, i_\theta \in \{1, \cdots, N\}$, any M, we have $Decrypt(i_1, \mathsf{usk}_{i_1}, \cdots, i_\theta, \mathsf{usk}_{i_\theta}, Encrypt(\mathsf{pk}, M), \mathsf{pk}) = M$

Game IND for semantic security

1. **Init.** Adversary $\mathcal{A}$ outputs a set $S_{ga} \subset \{1, \cdots, N\}$ it will not corrupt.
2. **Setup**. Simulator $\mathcal{B}$ runs $Setup$. Gives all usk_i, $i \notin S_{ga}$, to $\mathcal{A}$.
3. **Queries** In arbitrary interleaf, $\mathcal{A}$ makes q_D queries to the Decryption Oracle $\mathcal{DO}$.
4. **Gauntlet** At a certain point and in arbitrary interleaf with queries to $\mathcal{DO}$, $\mathcal{A}$ issues a message M_1. $\mathcal{B}$ select random message M_0, flips a fair coin b, and sends the *gauntlet ciphertext* $\mathsf{ctxt}_{ga} = Encrypt(\mathsf{pk}, M_b)$.
5. **Guess** At the end, $\mathcal{A}$ sends $\hat{b}$, its estimate of b.

$\mathcal{A}$ *wins* if $\hat{b} = b$ and it has never queried ctxt_{ga} to $\mathcal{DO}$. Its *advantage* is its probability of winning minus 1/2.

Tracing Game

1. Adversary $\mathcal{A}$ outputs a colluder set $T \subset \{1, \cdots, N\}$.
2. Simulator $\mathcal{B}$ runs $Setup$. Gives all usk_i, $i \in T$, to $\mathcal{A}$.
3. $\mathcal{A}$ outputs a pirate decoder $\mathcal{D}$.
4. $\mathcal{B}$ runs $Trace^{\mathcal{D}}(\mathsf{TK}, \epsilon)$ to obtain S.

We say $\mathcal{A}$ wins if both following conditions hold:

1. $\mathcal{D}$ is useful, i.e. $\Pr\{\mathcal{D}(Encrypt(\mathsf{pk}, M)) = M\} \geq \epsilon$ for random M.
2. S is either empty or not a subset of T^θ.

The *advantage* of $\mathcal{A}$ is his probability of winning.

Definition 2. *A* threshold-traitor tracing *is* q_D-CCA-secure *if it is* correct*,* q_D-CCA secure*, no PPT algorithm has a non-negligible advantage in Game IND, and no PPT algorithm has a non-negligible advantage in the Tracing Game. It is* CPA-secure *if it is* 0-CCA secure*.*

3 Intractability assumptions

Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ be a bilinear map, where $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of prime order q_1, g is a generator of $\mathbb{G}_1$, and $\hat{\mathbf{e}}(u^a, v^b) = \hat{\mathbf{e}}(u, v)^{ab}$ for all $u, v \in \mathbb{G}_1$, and all $a, b \in \mathbb{Z}$.

Definition 3. *Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ be a pairing. The* decisional ℓ-BDHE (Bilinear Diffie-Hellman Exponent) problem *is, given $\mathbb{G}_1$-elements h, g, g^{α}, $\cdots$, $g^{\alpha^{\ell}}$, $g^{\alpha^{\ell+2}}$, $\cdots$, $g^{\alpha^{2\ell}}$, distinguish $\hat{\mathbf{e}}(g, h)^{\alpha^{\ell+1}}$ versus random. The* decisional ℓ-BDHE (Bilinear Diffie-Hellman Exponent) assumption *is that no PPT algorithm can solve a random instance of the ℓ-BDHE problem with probability non-negligibly over half.*

Definition 4. *Let $\mathbb{G}$ be a group of prime order p. The* decisional three-party Diffie-Hellman (D3DH) problem *is , given $\mathbb{G}_1$-elements g, $A = g^a$, $B = g^b$, $C = g^c$, distinguish g^{abc} from random. The* D3DH assumption *is that no PPT algorithm can solve a random instance of the D3DH problem with probability non-negligibly over half.*

Definition 5. *The* subgroup decisional (SD) problem *is , given a group $\mathbb{G}$, its order $n = pq$, where p, q are distinct unknown primes, $\mathbb{G}$-elements g_p, g, h where order$(g_p) = p$, order$(g) = n$, and order(h) has half-half probability of being p or n, distinguish whether order$(h) = p$. The* subgroup decisional (SD) assumption *is that no PPT algorithm can solve a random instance of the* SD problem *with probability non-negligibly over half. The* bilinear subgroup decisional (BSD) problem *(resp.* assumption*) is the version when there is a pairing $\mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ and h is replaced by an element of $\mathbb{G}_T$ whose order equals p or n with half-half probability.*

4 Chosen ciphertext secure broadcast encryption

We construct the captioned scheme, and give a security reduction.

Setup Upon input the security paramater λ_s, generate a pairing $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$, where order$(\mathbb{G}_1) = q_1$ which is a prime. Generate random $g \in \mathbb{G}_p$, $\alpha, \gamma_1, \gamma_2 \in Z_{q_1}$. Compute $v_1 = g^{\gamma_1}$, $v_2 = g^{\gamma_2}$. Denote N the number of users and θ the threshold. Let $s_1 = N + 1$ and $f_1(i) = i$, $f_2(j) = s_1 - j$, $f_3(i, j) = f_1(i) + f_2(j) = s_1 - j + i$. Let $y_i = g^{\alpha^i}$. The public key is $\mathsf{pk} = (\hat{\mathbf{e}}, N, \theta, g, v_1, v_2, y_1, \cdots, y_N, y_{N+2}, \cdots, y_{2N})$. The private decryption key for User i is $\mathsf{usk}_i = (g^{\gamma_1 \alpha^i}, g^{\gamma_2 \alpha^i})$

Encryption. Let Symm be a secure symmetric cipher, $N' \leq N$, $\mathcal{S} = \{i_1, \cdots, i_{N'}\}$, $1 \leq i_1 < \cdots < i_{N'} \leq N$. Upon inputs pk, M and a set of θ users S, randomly choose t and output the ciphertext $\mathsf{ctxt} = (\mathsf{ctxt}_0, \mathsf{ctxt}_1, \mathsf{ctxt}_2, \mathsf{ctxt}_3)$ where

$$\mathsf{ctxt}_0 = g^t,\ \mathsf{ctxt}_1 = v_1^t \textstyle\prod_{j \in S} y_{f_2(j)}^t,\ \mathsf{ctxt}_2 = v_2^t \prod_{j \in S} y_{f_2(j)}^t,$$
$$\mathsf{ctxt}_3 = \mathsf{Symm}_K(M) \text{ where } K = \hat{\mathbf{e}}(g, g)^{t\alpha^{s_1}} = \hat{\mathbf{e}}(g^{\alpha^{f_1(\ell)}}, g^{\alpha^{f_1(s_1 - 1 - \ell)}})^t$$

Decryption Upon inputs pk, $S \subset \{1, \cdots, N\}$, $i \in S$, usk_i, ctxt, confirm

$$\hat{\mathbf{e}}(\mathsf{ctxt}_0, v_1 \prod_{j \in S} y_{f_2(j)}) = \hat{\mathbf{e}}(g, \mathsf{ctxt}_1),\ \hat{\mathbf{e}}(\mathsf{ctxt}_0, v_2 \prod_{j \in S} y_{f_2(j)}) = \hat{\mathbf{e}}(g, \mathsf{ctxt}_2) \tag{1}$$

If not all confirmed, output $\mathsf{InvalidCiphertext}$ and abort. Else compute

$$K = \hat{\mathbf{e}}(g^{\alpha^{f_1(i)}}, \mathsf{ctxt}_1) / \hat{\mathbf{e}}(\mathsf{ctxt}_0, \mathsf{usk}_{i,1} \prod_{j' \in S \setminus \{i\}} y_{f_3(i, j')}) \tag{2}$$

and output $M = \mathsf{Symm}_K^{-1}(\mathsf{ctxt}_3)$.

The size of the ciphertext (resp. the private key for each user, the public key) is $O(1)$ (resp. $O(1)$, $O(N)$). Security reduction below.

Theorem 1. *The above broadcast encryption is correct. It is CCA secure provided the Decisional BDHE Assumption holds.*

Proof Sketch: The proof of the chosen ciphertext security of a twin encryption is typically simple, and it reduces to the same intractability assumption of the chosen plaintext secure encryption it is based on. In the initialization, the Simulator $\mathcal{S}$ is set up to have either of the *twin* decryption keys. To service the Decryption Oracle, $\mathcal{S}$ simply uses the key it has to decrypt any legitimate ciphertext. Due to symmetry, the adversary $\mathcal{A}$ cannot distinguish exactly which of the twin keys $\mathcal{S}$ possesses. So in the end-game extraction, $\mathcal{A}$ has a 50-50 chance of being extracted w.r.t. the other decryption key that $\mathcal{S}$ does not have.

The "twin ciphertexts" are (ctxt_0, ctxt_1, ctxt_3) and (ctxt_0, ctxt_2, ctxt_3). It is crucial to deduct that either of the twin ciphertexts decrypt to the same plaintext, as follows. Let $t = \log_g \mathsf{ctxt}_0$, then Equation (1) implies $\mathsf{ctxt}_1 = v_1 \prod_j y_{f_2(i,j)}$ and $\mathsf{ctxt}_2 = v_2 \prod_j y_{f_2(i,j)}$. Consequently the same session key K and plaintext M are recovered using Eq. (2) as is, or using it with ctxt_1 replaced by ctxt_2. Remaining proof details are left to the full paper. □

5 Broadcast threshold encryption

We construct the captioned scheme, and give a security reduction.

Setup Upon input the security paramater λ_s, generate a pairing $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$, where order$(\mathbb{G}_1) = q_1$ which is a prime. Generate random $g \in \mathbb{G}_1$, $\alpha, \gamma, \in Z_{q_1}$. Compute $v = g^\gamma$. Denote N the number of users and θ be the threshold value. Let $N^\theta < 2^{-4} q_1 < s_1 < 2^{-2} q_1$. Let $s_1 = N^\theta + 1$ and adopt the shorthand notations $\mathbf{i} = (i_1, \cdots, i_\theta)$ and $\mathbf{j} = (j_1, \cdots, j_\theta)$ for $1 \le i_1 < \cdots < i_\theta \le N$.

Let SS denote a secret sharing scheme and let $SS(\theta, \mathbf{i}, k, s)$ denotes the k-th participant's share of the secret s where the threshold equals θ, the participants are $\mathbf{i}$. We require that in SS the secret is recovered by the commonplace "linear combining"

$$s = \prod_{1 \le k \le \theta} SS(\theta, \mathbf{i}, k, s)^{\mathsf{cbn}(\theta, \mathbf{i}, k)}$$

where $\mathsf{cbn}(\cdot)$ is the combining coefficient. Let $f_1(\mathbf{i}) = \sum_{1 \le k \le \theta} i_k N^{k-1}$, $f_2(\mathbf{j}) = s_1 - f_1(\mathbf{j})$, $f_3(\mathbf{i}, \mathbf{j}) = f_1(\mathbf{i}) + f_2(\mathbf{j})$, $S_1 = \{\mathbf{i}\}$, $Y_1 = \{f_1(\mathbf{i})\}$, $Y_2 = \{f_2(\mathbf{j})\}$, $Y_3 = \{f_3(\mathbf{i}, \mathbf{j})\}$. The public key is

$$\mathsf{pk} = (\hat{\mathbf{e}}, q_0, N, \theta, s_1, g, v, \{g^{\alpha^k} : k \in Y_1 \cup Y_2 \cup Y_3, k \ne s_1\}, \mathsf{cbn}(\cdot))$$

The private decryption key for User i is

$$\{SS(\theta, \mathbf{i}, k, g^{\gamma_b \alpha^{f_1(\mathbf{i})}}) : b \in \{1, 2\}, i = i_k \text{ for some } k\}$$

i.e. all secret pieces that User i is a participant of SS.

Encryption Let $N' \le N^\theta$, $\mathcal{S} = \{i_1, \cdots, i_{N'}\}$, $1 \le i_1 < \cdots < i_{N'} \le N$, let Upon input the public key param and message M and a set of θ users S, randomly choose t and output

ciphertext $\mathsf{ctxt} = (\mathsf{ctxt}_0, \mathsf{ctxt}_1, \mathsf{ctxt}_2)$ where

$$\begin{aligned}\mathsf{ctxt}_0 &= g^t \\ \mathsf{ctxt}_b &= v_b^t \prod_{\mathbf{j}\in S_1\cap S^\theta} y_{f_2(\mathbf{j})}^t,\ b\in\{1,2\} \\ \mathsf{ctxt}_3 &= \mathsf{Symm}_K(M),\ K = y_{s_1}^t\end{aligned}$$

Decryption Upon input param, $S\subset\{1,\cdots,N\}$, $i\in S$, usk_i, and ctxt, verify $\hat{\mathbf{e}}(\mathsf{ctxt}_b, g) = \hat{\mathbf{e}}(\mathsf{ctxt}_0, v_b\prod_{\mathbf{j}\in S_1\cap S^\theta} y_{f_2(\mathbf{j})})$ for $b=1,2$. If not verified, output $\mathsf{InvalidCiphertext}$ and abort. Else output $M = \mathsf{Symm}_K^{-1}(\mathsf{ctxt}_2)$ where

$$K = \hat{\mathbf{e}}(y_{f_1(\mathbf{i})}, g^{t\gamma_1}\prod_{\mathbf{j}\in S_1\cap S} y_{f_2(\mathbf{j})}^t)\cdot\hat{\mathbf{e}}(\mathsf{ctxt}_0, y_{f_1(\mathbf{i})})^{-1}$$
$$\hat{\mathbf{e}}(\mathsf{ctxt}_0, \prod_{(\mathbf{j})\in(S_1\cap S^\theta)\setminus\{(\mathbf{i})\}} y_{f_3(\mathbf{i},\mathbf{j})})^{-1}$$

The second term above is computed by the recovery protocol of SS as follows

$$\hat{\mathbf{e}}(\mathsf{ctxt}_0, y_{f_1(\mathbf{i})}) = \prod_{1\le k\le\theta}\hat{\mathbf{e}}(\mathsf{ctxt}_0, SS(\theta,\mathbf{i},k,y_{f_1(\mathbf{i})}))^{\mathsf{cbn}(\theta,\mathbf{i},k)} \tag{3}$$

with each participating threshold decoder i_k contributing one $\mathbb{G}_T$ element $\hat{\mathbf{e}}(\mathsf{ctxt}_0, SS(\theta, \mathbf{i}, k, y_{f_1(\mathbf{i})})$.

The size of the ciphertext (resp. the private key for each user, the public key) is $O(1)$ (resp. $O(N^{\theta-1})$, $O(N^\theta)$). Note the above is a piggyback scheme on the non-threshold broadcast scheme of Section 4 in the following sense: Each group of θ users denoted by $\mathbf{i}$ is mapped to a distinct group serial number $f_1(\mathbf{i})$. Then the N^θ-user instantiation of Section 4's scheme is used. To revoke users $T\subset\{1,\cdots,N\}$ in the broadcast threshold encryption, simply revoke all groups $\mathbf{i}$ which intersects T. From this piggyback perspective, the following security theorem is relatively easy and details are left to the full paper.

Theorem 2. *Assume the secret sharing scheme is secure. The above threshold broadcast encryption is correct. It is CCA secure provided the Decisional BDHE Assumption holds.*

6 Chosen ciphertext secure (non-revoking) traitor tracing

We construct the captioned scheme, and give security reductions. We follow the convention in [2] to specify the PLBE (Private Linear Broadcast Encryption) which specializes to the broadcast encryption in the traitor tracing. PLBE is a sort of *linearly revoking* broadcast encryption, where users can be linearly ordered such that the only revocations allowed are on the first k users in that order.

Setup. $N=m^2$. Generate pairing $\hat{\mathbf{e}}:\mathbb{G}_1\times\mathbb{G}_1\to\mathbb{G}_T$, where $\mathrm{order}(\mathbb{G}_1)=n=pq$ and p, q are primes. Generate subgroups $\mathbb{G}_p,\mathbb{G}_q\lhd\mathbb{G}_1$ with $\mathrm{order}(\mathbb{G}_p)=p$ and $\mathrm{order}(\mathbb{G}_q)=q$. Select random $g_p,h_p\in\mathbb{G}_p$, $g_q,h_q\in\mathbb{G}_q$. Set $g=g_pg_q$, $h=h_ph_q\in\mathbb{G}_1$. Select random $r_1,\cdots,r_m$, $c_1,\cdots,c_m$, $\alpha_1,\cdots,\alpha_m$, $\bar{r}_1,\cdots,\bar{r}_m$, $\bar{c}_1,\cdots,\bar{c}_m\in\mathbb{Z}_n$, $\beta\in\mathbb{Z}_q$. The public key PK consists of:

$$\begin{gathered}\hat{\mathbf{e}}, n, g, h, E=g^\beta, G_1=\hat{\mathbf{e}}(g_q,g_q)^{\beta\alpha_1},\cdots,G_m=\hat{\mathbf{e}}(g_q,g_q)^{\beta\alpha_m}, \\ E_1=g_q^{\beta r_1},\cdots,E_m=g_q^{\beta r_m},\bar{E}_1=g_q^{\beta\bar{r}_1},\cdots,\bar{E}_m=g_q^{\beta\bar{r}_m}, \\ F_1=h_q^{\beta r_1},\cdots,F_m=h_q^{\beta r_m},\bar{F}_1=h_q^{\beta\bar{r}_1},\cdots,\bar{F}_m=h_q^{\beta\bar{r}_m}, \\ H_1=g^{c_1},\cdots,H_m=g^{c_m},\bar{H}_1=g^{\bar{c}_1},\cdots,\bar{H}_m=g^{\bar{c}_m},\end{gathered}$$

The private key for user (x, y) is $K_{x,y} = (g^{\alpha_x} g^{r_x c_y}, g^{\alpha_x} g^{\bar{r}_x \bar{c}_y})$. (To ensure honesty, publish a proof of the correct formation of PK, and send a proof of the correct formation of $K_{x,y}$ to user (x, y).)

TrEncrypt(PK,M, (i, j)): Select random t, w_1, $\cdots$, w_m, s_1, $\cdots$, s_m, $v_{1,1}$, $v_{1,2}$, $v_{1,3}$, $v_{1,4}$, $v_{1,5}$, $v_{1,6}$, $\cdots$, $v_{i-1,1}$, $v_{i-1,2}$, $v_{i-1,3}$, $v_{i-1,4}$, $v_{i-1,5}$, $v_{i-1,6}$, $\in \mathbb{Z}_n$. For each row x, compute and send row ciphertext components (R_x, $\tilde{R}_x$, A_x, B_x, $\bar{R}_x$, $\hat{R}_x$, D_x):

$$\begin{aligned}
\text{if } x > i:\ & R_x = g_q^{s_x r_x},\ \ \tilde{R}_x = h_q^{s_x r_x},\ \ A_x = g_q^{s_x t},\ \ B_x = M\hat{\mathbf{e}}(g_q, g)^{\alpha_x s_x t},\\
& \bar{R}_x = g_q^{s_x \bar{r}_x},\ \ \hat{R}_x = h_q^{s_x \bar{r}_x},\ D_x = g_q^{s_x}\\
\text{if } x = i:\ & R_x = g^{s_x r_x},\ \ \tilde{R}_x = h^{s_x r_x},\ \ A_x = g^{s_x t},\ \ B_x = M\hat{\mathbf{e}}(g, g)^{\alpha_x s_x t},\\
& \bar{R}_x = g^{s_x \bar{r}_x},\ \hat{R}_x = h^{s_x \bar{r}_x},\ D_x = g^{s_x}\\
\text{if } x < i:\ & R_x = g^{v_{x,1}},\ \tilde{R}_x = h^{v_{x,1}},\ \ A_x = g^{v_{x,2}},\ \ B_x = \hat{\mathbf{e}}(g, g)^{v_{x,3}},\\
& \bar{R}_x = g^{v_{x,4}},\ \hat{R}_x = g^{v_{x,5}},\ D_x = g^{v_{x,6}}
\end{aligned}$$

For each column y compute and send the following column ciphertext components

$$\begin{aligned}
\text{if } y \geq j:\ & C_y = g^{c_y t} h^{w_y},\ \ \tilde{C}_y = g^{w_y},\ \ \bar{C}_y = g^{\bar{c}_y t} h^{w_y}\\
\text{if } y < j:\ & C_y = g^{c_y t} h^{w_y} g_p^{z_{p,y}},\ \ \tilde{C}_y = g^{w_y},\ \ \bar{C}_y = g^{\bar{c}_y t} h^{w_y} g_p^{z_{p,y}}
\end{aligned}$$

Encrypt(PK,M): Select random $t, w_1, \cdots, w_m, s_1, \cdots, s_m \in \mathbb{Z}_n$. For each row x, compute and send the following row ciphertext components

$$\begin{gathered}
R_x = E_x^{s_x},\quad \tilde{R}_x = F_x^{s_x},\quad A_x = E^{s_x t},\quad B_x = MG_x^{s_x t},\\
\bar{R}_x = \bar{E}_x^{s_x},\quad \hat{R}_x = \bar{F}_x^{s_x},\quad D_x = g_q^{s_x},
\end{gathered}$$

For each column y compute and send the following column ciphertext components

$$C_y = H_y^t h^{w_y},\quad \tilde{C}_y = g^{w_y},\quad \bar{C}_y = \bar{H}_y^t h^{w_y}$$

Decrypt($x, y, K_{x,y}, \mathsf{ctxt}$): Verify, for each row x and each column y satisfying $x > i$, or $x = i$ and $y \geq j$, that

$$\begin{gathered}
\hat{\mathbf{e}}(R_x, F) = \hat{\mathbf{e}}(\tilde{R}_x, E_x),\ \hat{\mathbf{e}}(\bar{R}_x, \bar{F}_x) = \hat{\mathbf{e}}(\hat{R}_x, \bar{E}_x),\ \hat{\mathbf{e}}(R_x, \bar{E}_x) = \hat{\mathbf{e}}(\bar{R}_x, E_x),\\
\hat{\mathbf{e}}(C_y, h) = \hat{\mathbf{e}}(\tilde{C}_y, h),\ \hat{\mathbf{e}}(\bar{C}_y, h) = \hat{\mathbf{e}}(\bar{C}_y, h),\\
\hat{\mathbf{e}}(A_x, H_y) = \hat{\mathbf{e}}(D_x, C_y),\ \hat{\mathbf{e}}(A_x, \bar{H}_y) = \hat{\mathbf{e}}(D_x, \bar{C}_y)
\end{gathered} \tag{4}$$

If not all verified, output $\mathsf{InvalidCiphertext}$ and abort. Else output

$$M = B_x \hat{\mathbf{e}}(R_x, C_y) \hat{\mathbf{e}}(K_{x,y,1}, A_x)^{-1} \hat{\mathbf{e}}(\tilde{R}_x, \tilde{C}_y)^{-1}$$

Tracing. We use the black-box tracing in [2]. For each user (i, j), produce a number of encryptions and ask the adversary $\mathcal{A}$ to decrypt. If $\mathcal{A}$ can correctly decrypt with non-negligible probability, then (i, j) is a suspect. At the conclusion, select a random suspect to output. If no suspect is found, output $\mathsf{NoSuspect}$.

Theorem 3. *Assume the secret sharing scheme is CCA secure provided the D3DH (decisional 3-party Diffie Hellman) assumption, the SD (subgroup decision) assumption, and the (BCD) bilinear subgroup decision assumption all hold.*

Proof Sketch. The proof is similar to that in [2]. The modification for *twinning* is straightforward. Proof for twin encryption is typically simple. The security reduction is to the same assumption as that of the chosen plaintext scheme that it is based on. However, we do note that merely "twinning" [2]'s broadcast encryption part is not sufficient to ensure that twin ciphertexts decrypt to the same plaintext. More relations are needed. To combat this problem, we have added the ciphertext components denoted by D_x. With the inclusion of D_x's in the ciphertext, the verification relations Eq (4) imply that twin ciphertexts decrypt to the same. Another crucial aspect of the proof beyond twin encryption considerations is to show that traitor tracing is not compromised by adding D_x's. Details are left to our future full paper. □

Efficiency. The size of our ciphertext (resp. private key for each user, public key) remains the same as those in [2]. It is $O(\sqrt{N})$ (resp. $O(1)$, $O(\sqrt{N})$).

7 Threshold-traitor tracing

We construct the captioned scheme, without revoking capabilities, and reduce its chosen ciphertext security to intractability assumptions. Our technique is a piggyback technique on the non-threshold traitor tracing scheme of Section 6, much like the broadcast threshold encryption scheme in Section 5 is a piggyback ride on its non-threshold version in Section 4. Here is an outline:

Each group of θ distinct users $\mathbf{i} = (i_1, \cdots, i_\theta)$ is given a unique group serial number $f_1(\mathbf{i})$. The Setup of the (non-threshold) traitor tracing of Section 6 with $N' = n^\theta$ users is used to setup public and non-threshold private keys. The private key of the $f_1(\mathbf{i})$-th non-threshold user's private key is (θ, θ)-secret shared to members of the group corresponding to $\mathbf{i}$, namely $i_1, \cdots, i_\theta$. To encrypt for the threshold version with N users, use the non-threshold encryption with N' users. To decrypt for the threshold version, an arbitrary group of θ users $\mathbf{i} = (i_1, \cdots, i_\theta)$ collaborate to recover the group secret, which equals the $f_1(\mathbf{i})$-th non-threshold user's private key, and equals $g^{\alpha^{f_1(\mathbf{i})}}$ as in Section 5, is N'-user non-threshold scheme. The secret sharing scheme is required to have a "linear combining" as in Eq. (3) which allows each partipants to contribute its computations without revealing secrets. Therefore, the *Setup*, *Encrypt*, and *Decrypt* can be accomplished by the above piggyback technique.

Tracing For each θ-user group $\mathbf{i} = (i_1, \cdots, i_\theta)$, generate a number of messages, encrypt them and test if the black-box pirate decoder box can decrypt with probability above ϵ. If so, $\mathbf{i}$ is a suspect. Else, it is not. At the conclusion, output all suspect $\mathbf{i}$.

The size of the ciphertext (resp. the private key for each user, the public key) is $O(N^{\theta/2})$ (resp. $O(N^{\theta-1})$, $O(N^{\theta/2})$).

Theorem 4. *Assume the secret sharing scheme is CCA secure provided the D3DH (decisional 3-party Diffie Hellman) assumption, the SD (subgroup decision) assumption, and the (BCD) bilinear subgroup decision assumption all hold.*

8 Discussions and Conclusions

We have used the twin encryption technique [10] to upgrade the simpler broadcast encryption in [1] (resp. the (non-revoking) traitor tracing in [2]) from chosen plaintext security to chose ciphertext security. We have also introduced the studies of broadcast threshold encryption and threshold-traitor tracing.

It remains open problem to improve the private key size in our threshold schemes, and to upgrade to revoking traitor tracing.

Acknowledgements of sponsorship to Hong Kong Earmarked Grants 4232-03E and 4328-02E, including financial support for the second author's visit to Chinese University of Hong Kong where he conducted part of this research.

References

1. Dan Boneh, Craig Gentry, and Brent Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *CRYPTO 2005*, volume 3621 of *LNCS*, pages 258–275. Springer-Verlag, 2005.
2. Dan Boneh, Amit Sahai, and Brent Waters. Fully collusion resistant traitor tracing with short ciphertexts and private keys. In *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 573–592. Springer-Verlag, 2006.
3. Y. Dodis and N. Fazio. Public key broadcast encryption for stateless receivers. In *DRM 2002*, volume 2696 of *LNCS*, pages 61–80. Springer-Verlag, 2002.
4. A. Fiat and M. Naor. Boradcast encryption. In *Crypto 1993*, volume 773 of *LNCS*, pages 480–491. Springer-Verlag, 1993.
5. P. A. Fouque and D. Pointcheval. Threshold cryptosystems secure against chosen ciphertext attacks. In *Asiacrypt 2001*, volume 2248 of *LNCS*, pages 351–368. Springer-Verlag, 2001.
6. M. T. Goodrich, J. Z. Sun, and R. Tamassia. Efficient tree-based revocation in groups of low-state devices. In *Crypto 2004*, volume 3152 of *LNCS*, pages 511–527. Springer-Verlag, 2004.
7. D. Halevy and A. Shamir. The lsd broadcast encryption scheme. In *Crypto 2002*, volume 2442 of *LNCS*, pages 47–60. Springer-Verlag, 2002.
8. A. Kiayias and M. Yung. Group signatures: provable security, efficient constructions, and anonymity from trapdoor-holders. Cryptology ePrint Archive, Report 2004/076, 2004. http://eprint.iacr.org/.
9. D Naor, M. Naor, and J. Lotspiech. Revocation and tracing schems for stateless receivers. In *Crypto 2001*, volume 2139 of *LNCS*, pages 256–266. Springer-Verlag, 2001.
10. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC 1990*, pages 427–437. ACM Press, 1990.
11. Victor K. Wei. Short (resp. fast) CCA2-fully-anonymous group signatures using IND-CPA-encrypted escrows. Cryptology ePrint Archive, Report 2005/4104073, 2005. http://eprint.iacr.org/.