Distributed Broadcast Encryption from Bilinear Groups

Dimitris Kolonelos

IMDEA Software Institute
& Universidad Politecnica de Madrid

Giulio Malavolta

Bocconi University
& MPI for Security and Privacy

Hoeteck Wee

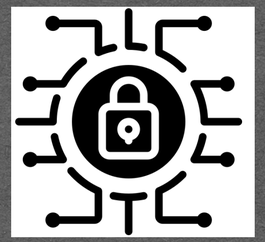
NTT Research
& École Normale Supérieure - PSL

ePrint: https://eprint.iacr.org/2023/874

Asiacrypt 2023, Guangzhou
6 December 2023

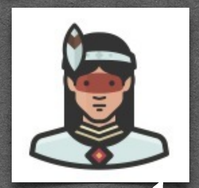
‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator



Encryptor

Decryptors


user 1


user 2


user 3

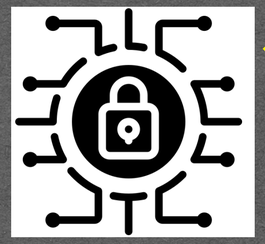

user 4


user 5


user 6

‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator

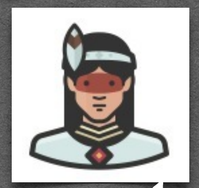


1 $\text{Setup}(1^\lambda, N) \rightarrow (\text{msk}, \text{mpk})$

mpk

Encryptor

Decryptors


user 1


user 2


user 3

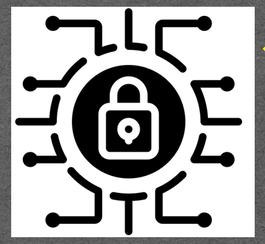

user 4


user 5


user 6

‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator



1 $\text{Setup}(1^\lambda, N) \rightarrow (\text{msk}, \text{mpk})$

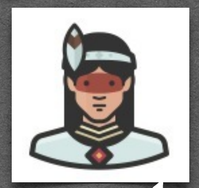
mpk

Encryptor

2 $\text{KeyGen}(\text{msk}, i) \rightarrow \text{sk}_i$

sk_1 sk_2 … sk_n

Decryptors


user 1


user 2


user 3

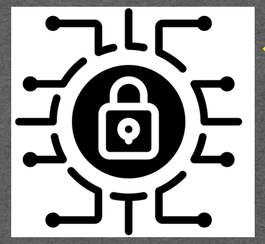

user 4


user 5


user 6

‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator



1 Setup$(1^\lambda, N) \rightarrow$ (msk, mpk)

mpk

Encryptor

message m,
Subset S

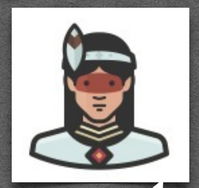
2 KeyGen(msk, i) $\rightarrow$ sk$_i$

3 Encrypt(mpk, S, m) $\rightarrow$ ct

sk$_1$ sk$_2$ … sk$_n$

ct

Decryptors


user 1


user 2


user 3

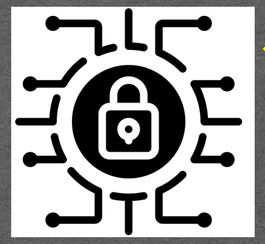

user 4


user 5


user 6

‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator



1 $\text{Setup}(1^\lambda, N) \rightarrow (\text{msk}, \text{mpk})$

mpk

Encryptor

message m, Subset S

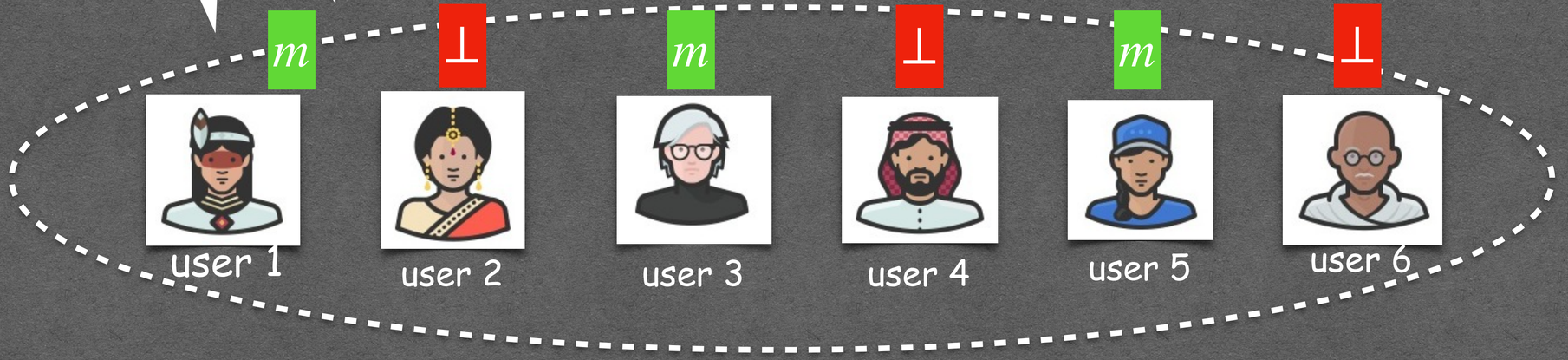
2 $\text{KeyGen}(\text{msk}, i) \rightarrow \text{sk}_i$

3 $\text{Encrypt}(\text{mpk}, S, m) \rightarrow \text{ct}$

sk_1 sk_2 … sk_n

ct

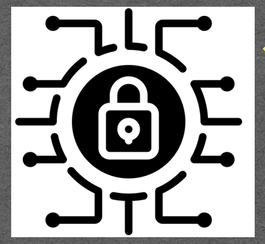
Decryptors



4 $\text{Decrypt}(\text{mpk}, \text{sk}_i, S, i, \text{ct}) \rightarrow m$ iff $i \in S$

‘Traditional’ Broadcast Encryption (BE) [FN93]

Private Key Generator



1 Setup$(1^\lambda, N) \rightarrow$ (msk, mpk)

mpk

Encryptor

message m, Subset S

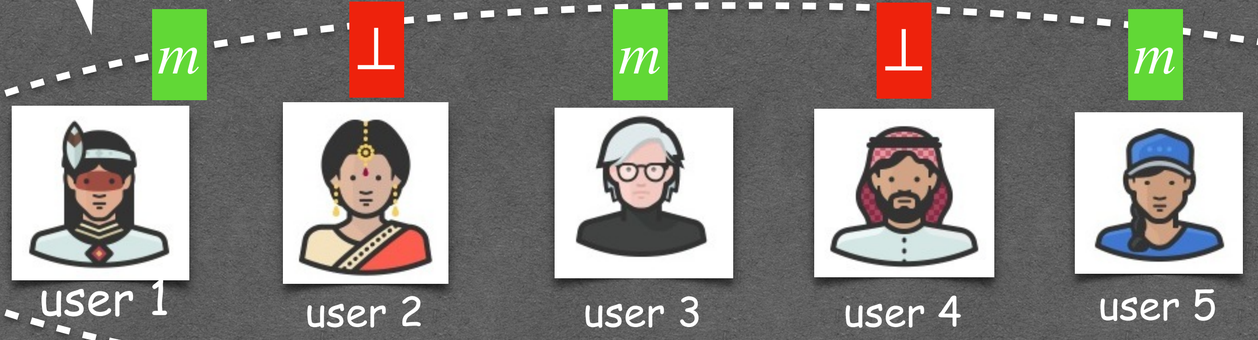
2 KeyGen(msk, i) $\rightarrow$ sk_i

3 Encrypt(mpk, S, m) $\rightarrow$ ct

sk_1 sk_2 … sk_n

ct

Decryptors



Non-triviality:

[BS03,BGW05]

$|\text{ct}| \ll |S|$

(Collusion Resistant) Security:

[NNL01]

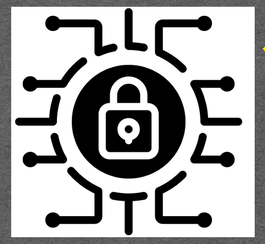
Even if all users $\notin S$ collude

ct is semantically secure

4 Decrypt(mpk, $\text{sk}_i, S, i, \text{ct}) \rightarrow m$ iff $i \in S$

Key-Escrow Problem

Private Key Generator



1 Setup$(1^\lambda, N) \rightarrow$ (msk, mpk)

mpk

Encryptor

message m, Subset S

2 KeyGen(msk, i) $\rightarrow$ sk_i

3 Encrypt(mpk, S, m) $\rightarrow$ ct

sk_1 sk_2 $\ldots$ sk_n

ct

Decryptors

m $\perp$ m $\perp$ m $\perp$

4 Decrypt(mpk, $\text{sk}_i, S, i,$ ct) $\rightarrow m$ iff $i \in S$

Key-Escrow Problem

Private Key Generator

1 Setup(1

PKG can decrypt all messages!

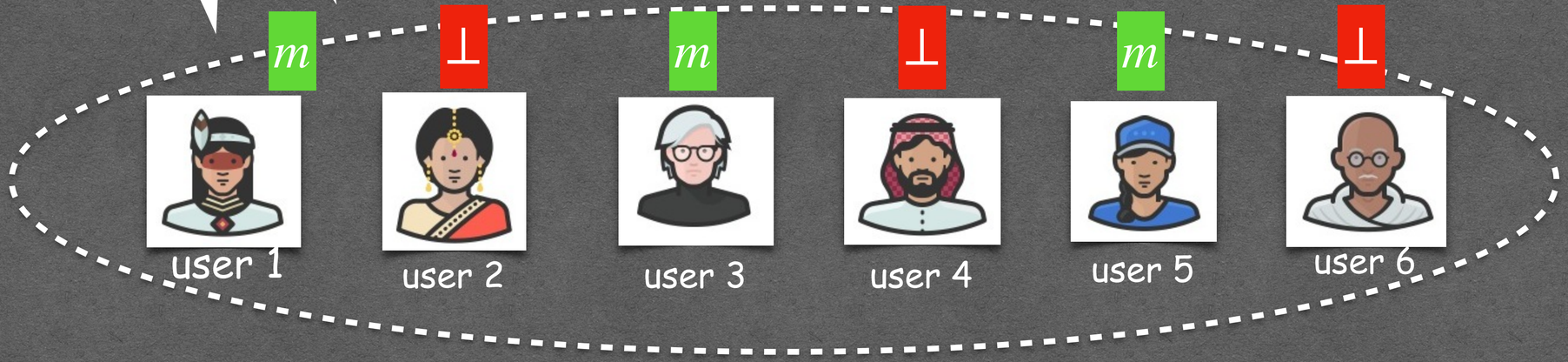
2 KeyGen(msk, i) → sk_i

3 Encrypt(mpk, S, m) → ct

sk_1 sk_2 … sk_n

ct

Decryptors



4 Decrypt(mpk, $\mathsf{sk}_i, S, i,$ ct) → m iff $i \in S$

Existing Approaches

Existing Approaches

- Mitigating the power of PKG:
  - Multiple PKGs+ secret sharing [BF01]
  - Acountable PKGs [G07, GLSW08]

Existing Approaches

- Mitigating the power of PKG:
  - Multiple PKGs+ secret sharing [BF01]
  - Acountable PKGs [G07, GLSW08]
- Decentralized BE [PPS12]
  - No PKG
  - Users interact upon new registration
  - $O(|S|)$ size ciphertexts (in the worst-case)

Existing Approaches

- Mitigating the power of PKG:
  - Multiple PKGs+ secret sharing [BF01]
  - Acountable PKGs [G07, GLSW08]
- Decentralized BE [PPS12]
  - No PKG
  - Users interact upon new registration
  - $O(|S|)$ size ciphertexts (in the worst-case)
- Distributed BE [WQZD10, BZ14]
  - No PKG
  - No interaction (CRS+Bulletin Board)
  - O(1) size ciphertexts

Existing Approaches

- Mitigating the power of PKG:
  - Multiple PKGs+ secret sharing [BF01]
  - Acountable PKGs [G07, GLSW08]
- Decentralized BE [PPS12]
  - No PKG
  - Users interact upon new registration
  - $O(|S|)$ size ciphertexts (in the worst-case)
- Distributed BE [WQZD10, BZ14]
  - No PKG
  - No interaction (CRS+Bulletin Board)
  - O(1) size ciphertexts

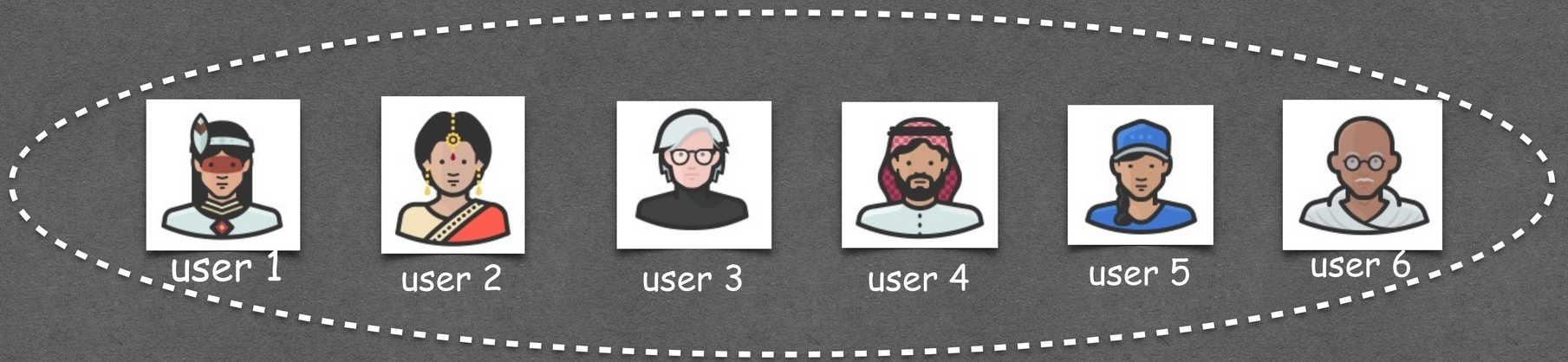
This Work

Distributed Broadcast Encyption [WQZD10,BZ14]

Bulletin Board

Encryptor

Decryptors



Distributed Broadcast Encyption [WQZD10,BZ14]

Bulletin Board

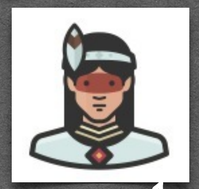
$\mathsf{pk}_1, \mathsf{pk}_2, \ldots, \mathsf{pk}_n$

Encryptor

pk_1 pk_2 $\ldots$ pk_n

2 $\mathsf{KeyGen}(\mathsf{crs}, i) \rightarrow (\mathsf{sk}_i, \mathsf{pk}_i)$

Decryptors


user 1


user 2


user 3


user 4


user 5


user 6

Distributed Broadcast Encyption [WQZD10,BZ14]

Bulletin Board

$\mathsf{pk}_1, \mathsf{pk}_2, \ldots, \mathsf{pk}_n$

Encryptor

message m,
Subset S

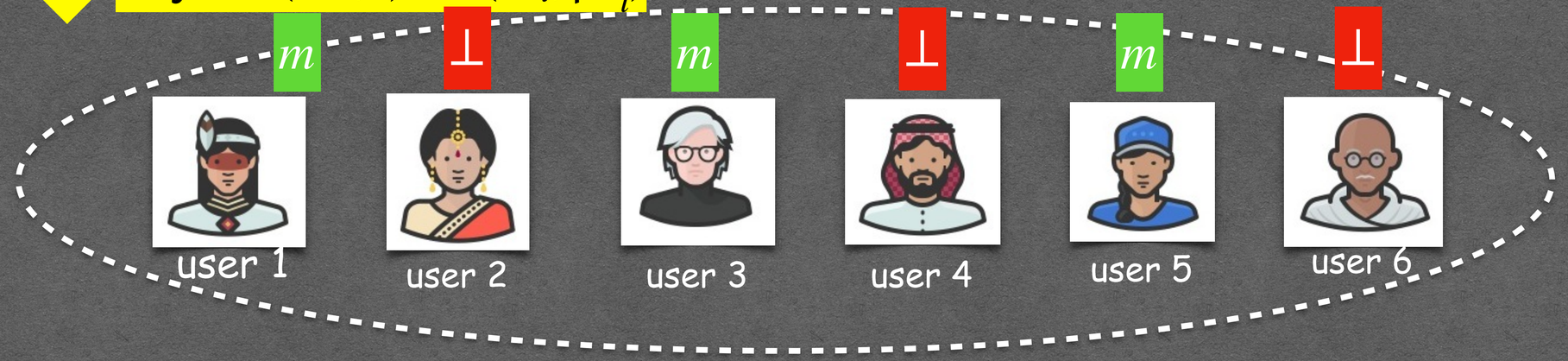
pk_1 pk_2 $\ldots$ pk_n

3 $\mathsf{Encrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j\in S}, S, m) \to \mathsf{ct}$

ct

2 $\mathsf{KeyGen}(\mathsf{crs}, i) \to (\mathsf{sk}_i, \mathsf{pk}_i)$

Decryptors



4 $\mathsf{Decrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j\in S}, \mathsf{sk}_i, S, i, \mathsf{ct}) \to m$ iff $i \in S$

Distributed Broadcast Encyption [WQZD10,BZ14]

Bulletin Board

$\mathsf{pk}_1, \mathsf{pk}_2, \ldots, \mathsf{pk}_n$

Encryptor

message m, Subset S

pk_1 pk_2 $\ldots$ pk_n

3 Encrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}, S, m$

ct

Non-triviality:

[BS03,BGW05]

$|\mathsf{ct}| \ll |S|$

2 KeyGen(crs, i) $\rightarrow (\mathsf{sk}_i, \mathsf{pk}_i)$

Decryptors

m $\perp$ m $\perp$ m

user 1 user 2 user 3 user 4 user 5

(Collusion Resistant) Security:

[NNL01]

Even if all users $\notin S$ collude

ct is semantically secure

4 Decrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}, \mathsf{sk}_i, S, i, \mathsf{ct}) \rightarrow m$ iff $i \in S$

Distributed Broadcast Encyption [WQZD10,BZ14]

Bulletin Board

$\mathsf{pk}_1, \mathsf{pk}_2, \ldots, \mathsf{pk}_n$

1 $\mathsf{Setup}(1^\lambda, N) \to \mathsf{crs}$

crs

Encryptor

message m, Subset S

3 $\mathsf{Encrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j\in S}, S, m$

ct

pk_1 pk_2 $\ldots$ pk_n

2 $\mathsf{KeyGen}(\mathsf{crs}, i) \to (\mathsf{sk}_i, \mathsf{pk}_i)$

Decryptors

Non-triviality:

[BS03,BGW05]

$|\mathsf{ct}| \ll |S|$

(Collusion Resistant) Security:

[NNL01]

Even if all users $\notin S$ collude

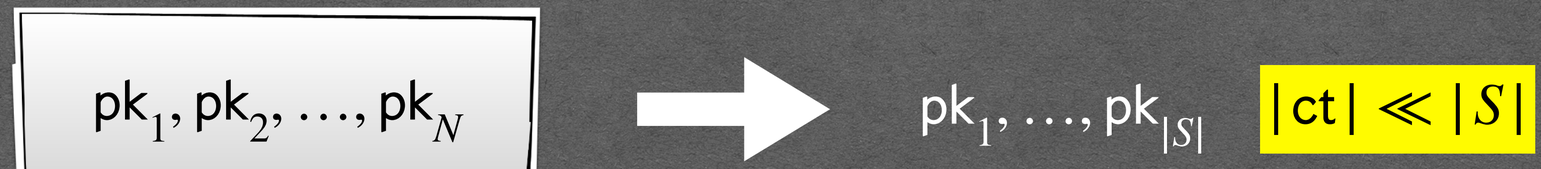
ct is semantically secure

4 $\mathsf{Decrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j\in S}, \mathsf{sk}_i, S, i, \mathsf{ct}) \to m$ iff $i \in S$

Connections with Registered Encryption
(RBE, R-ABE,…)

- **Registration-based Encryption** [GHMRS18]**:** Compress $\{\mathsf{pk}_1, \ldots, \mathsf{pk}_n\}$.

- **Distributed-Broadcast Encryption** [WQZD10,BZ14]**:** Compress $\{\mathsf{ct}_1, \ldots, \mathsf{ct}_{|S|}\}$.



- **Registrered ABE** [HLWW23]**:** Compress $\{\mathsf{pk}_1, \ldots, \mathsf{pk}_n\}$ + policy-based decryption.

Distributed Broadcast Encyption
Prior Constructions

- [WQZD10]: Pairings, $O(n^3)$ parameters, no formal security argument.
- [BZ14]: from indistinguishability obfuscation.

Distributed Broadcast Encyption
Prior Constructions

- [WQZD10]: Pairings, $O(n^3)$ parameters, no formal security argument.
- [BZ14]: from indistinguishability obfuscation.

Concurrent work:
[FWW23]: Flexible BE from WE

Our Work

★ **Thorough Definitions**

- ➡ Dynamic Joins
- ➡ Malicious Keys

★ **Two Efficient Constructions from Bilinear Groups**

- ➡ q-type assumption (BDHE)
- ➡ k-Lin (e.g. SXDH)

E.g. $|\mathsf{ct}| = 13\text{KB}$ and 7KB respectively (for 1024 users).

Our Definition of DBE

Syntax:

1. Setup$(1^\lambda, N) \to$ crs
2. KeyGen(crs, j) $\to$ (sk$_j$, pk$_j$)
3. Encrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}, S, m$) $\to$ ct
4. Decrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, sk$_i$, ct, S, i) $\to m$

Our Definition of DBE

Syntax:

1. Setup$(1^\lambda, N) \to$ crs
2. KeyGen(crs, j) $\to$ (sk$_j$, pk$_j$)
3. Encrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, S, m) $\to$ ct
4. Decrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, sk$_i$, ct, S, i) $\to m$

*Only the public keys of the ciphertext-set are necessary.

Our Definition of DBE

Syntax:

1. Setup$(1^\lambda, N) \to$ crs
2. KeyGen(crs, j) $\to$ (sk$_j$, pk$_j$)
3. Encrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, S, m) $\to$ ct
4. Decrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, sk$_i$, ct, S, i) $\to m$

*Only the public keys of the ciphertext-set are necessary.

Additional requirements:

Our Definition of DBE

Syntax:

1. Setup$(1^\lambda, N) \to$ crs
2. KeyGen(crs, j) $\to$ (sk$_j$, pk$_j$)
3. Encrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, S, m) $\to$ ct
4. Decrypt(crs, $\{\mathsf{pk}_j\}_{j\in S}$, sk$_i$, ct, S, i) $\to m$

*Only the public keys of the ciphertext-set are necessary.

Additional requirements:

★ Dynamic Joins: decryptors check the bulletin board for new pk$_j$'s at most polylog(N) times instead of N.

Our Definition of DBE

Syntax:

1. $\mathsf{Setup}(1^\lambda, N) \to \mathsf{crs}$
2. $\mathsf{KeyGen}(\mathsf{crs}, j) \to (\mathsf{sk}_j, \mathsf{pk}_j)$
3. $\mathsf{Encrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j \in S}, S, m) \to \mathsf{ct}$
4. $\mathsf{Decrypt}(\mathsf{crs}, \{\mathsf{pk}_j\}_{j \in S}, \mathsf{sk}_i, \mathsf{ct}, S, i) \to m$

*Only the public keys of the ciphertext-set are necessary.

Additional requirements:

★ Dynamic Joins: decryptors check the bulletin board for new pk_j's at most $\mathsf{polylog}(N)$ times instead of N.

★ Malicious Key-Registration: there exists an $\mathsf{isValid}$ algorithm for pk_j s.t. Correctness hold as long as $\mathsf{isValid}(\mathsf{crs}, \mathsf{pk}_j, j) = 1$ even for malformed public keys.

Our first DBE construction

Starting point [BGW05]

- Implicit notation
- Symmetric pairings for simplicity

1. Setup$(1^\lambda, N)$: $\quad \mathsf{mpk} = \left([\alpha], [\alpha^2], \ldots, [\alpha^N], [\alpha^{N+2}], [\alpha^{2N}], [t]\right), \quad \mathsf{msk} = t$

2. KeyGen(msk, i) : $\quad \mathsf{sk}_i = [t\alpha^{N+1-i}]$

3. Enc(mpk, S, m) : $\quad \mathsf{ct} = \left([s], [s(t + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right), \quad s \leftarrow_\$ \mathbb{Z}_p$

4. Dec(mpk, $\mathsf{sk}_i, i, S, \mathsf{ct}$) :

$$e\left([s(t + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}]\right) = e\left([s], [t\alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})]\right) \cdot e\left([s], [\alpha^{N+1}]\right)$$

$\mathsf{ct}_2 \quad \mathsf{mpk}_{N+1-i} \quad \mathsf{ct}_1 \quad \mathsf{sk}_i \quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j} \quad \mathsf{ct}_3 / m$

Our first DBE construction
Starting point [BGW05]

- Implicit notation
- Symmetric pairings for simplicity

1. Setup$(1^\lambda, N)$: $\mathsf{mpk} = \left([\alpha], [\alpha^2], \ldots, [\alpha^N], [\alpha^{N+2}], [\alpha^{2N}], [t]\right), \quad \mathsf{msk} = t$

2. KeyGen(msk, i) : $\mathsf{sk}_i = [t\alpha^{N+1-i}]$

3. Enc(mpk, S, m) : $\mathsf{ct} = \left([s], [s(t + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right), \quad s \leftarrow_\$ \mathbb{Z}_p$

4. Dec(mpk, $\mathsf{sk}_i, i, S, \mathsf{ct}$) :

$$e\left([s(t + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}]\right) = e\left([s], [t\alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})]\right) \cdot e\left([s], [\alpha^{N+1}]\right)$$

ct_2 $\quad \mathsf{mpk}_{N+1-i}$ $\quad \mathsf{ct}_1$ $\quad \mathsf{sk}_i$ $\quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j}$ $\quad \mathsf{ct}_3/m$

Our first DBE construction

Starting point [BGW05]

- Implicit notation
- Symmetric pairings for simplicity

1. Setup$(1^\lambda, N)$: $\mathsf{mpk} = \left([\alpha], [\alpha^2], \ldots, [\alpha^N], [\alpha^{N+2}], [\alpha^{2N}], [t]\right), \quad \mathsf{msk} = t$

2. KeyGen(msk, i) : $\mathsf{sk}_i = [t\alpha^{N+1-i}]$

3. Enc(mpk, S, m) : $\mathsf{ct} = \left([s], [s(t + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right), \quad s \leftarrow_\$ \mathbb{Z}_p$

4. Dec(mpk, $\mathsf{sk}_i, i, S, \mathsf{ct}$) :

$$e\left([s(t + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}]\right) = e\left([s], [t\alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})]\right) \cdot e\left([s], [\alpha^{N+1}]\right)$$

ct_2 $\quad \mathsf{mpk}_{N+1-i}$ $\quad \mathsf{ct}_1$ $\quad \mathsf{sk}_i$ $\quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j}$ $\quad \mathsf{ct}_3/m$

Our first DBE construction

Starting point [BGW05]

- Implicit notation
- Symmetric pairings for simplicity

1. Setup$(1^\lambda, N)$: $\mathsf{mpk} = \left([\alpha], [\alpha^2], \ldots, [\alpha^N], [\alpha^{N+2}], [\alpha^{2N}], [t]\right), \quad \mathsf{msk} = t$

2. KeyGen(msk, i) : $\mathsf{sk}_i = [t\alpha^{N+1-i}]$

Keys are correlated

3. Enc(mpk, S, m) : $\mathsf{ct} = \left([s], [s(t + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right), \quad s \leftarrow_\$ \mathbb{Z}_p$

4. Dec(mpk, $\mathsf{sk}_i, i, S, \mathsf{ct}$) :

$$e\left([s(t + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}]\right) = e\left([s], [t\alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})]\right) \cdot e\left([s], [\alpha^{N+1}]\right)$$

ct_2 $\quad \mathsf{mpk}_{N+1-i}$ $\quad \mathsf{ct}_1$ $\quad \mathsf{sk}_i$ $\quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j}$ $\quad \mathsf{ct}_3/m$

Our first DBE construction

Main idea: Distribute t

$t \to \Sigma_{j \in S} t_j$ in the ciphertext: user j samples their own secret t_i and publishes $[t_j]$

Our first DBE construction

Main idea: Distribute t

$t \to \Sigma_{j \in S} t_j$ in the ciphertext: user j samples their own secret t_i and publishes $[t_j]$

$$\mathsf{ct} = \left([s], [s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right)$$

$$e\left([s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}] \right) = e\left([s], [t_i \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} t_j \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})] \right) \cdot e\left([s], [\alpha^{N+1}] \right)$$

Our first DBE construction

Main idea: Distribute t

$t \to \Sigma_{j \in S} t_j$ in the ciphertext: user j samples their own secret t_i and publishes $[t_j]$

$$\mathsf{ct} = \left([s], [s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right)$$

$$e\left([s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}] \right) = e\left([s], [t_i \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} t_j \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})] \right) \cdot e\left([s], [\alpha^{N+1}] \right)$$

ct_2 $\quad \mathsf{mpk}_{N+1-i}$ $\quad \mathsf{ct}_1$ $\quad \mathsf{sk}_i$ $\quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j}$ $\quad \mathsf{ct}_3 / m$

Our first DBE construction

Main idea: Distribute t

$t \to \Sigma_{j \in S} t_j$ in the ciphertext: user j samples their own secret t_i and publishes $[t_j]$

$$\mathsf{ct} = \left([s], [s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right)$$

$$e\left([s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], [\alpha^{N+1-i}] \right) = e\left([s], [t_i \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} t_j \alpha^{N+1-i}] \cdot [\sum_{j \in S, j \neq i} \alpha^{N+1-i+j})] \right) \cdot e\left([s], [\alpha^{N+1}] \right)$$

ct_2 $\quad \mathsf{mpk}_{N+1-i}$ $\quad \mathsf{ct}_1$ $\quad \mathsf{sk}_i$ $\quad \prod_{j \in S, j \neq i} \mathsf{mpk}_{N+1-i+j}$ $\quad \mathsf{ct}_3 / m$

Cross-terms published along with $[t_i]$ from user i

Our first DBE construction: wrapped up

$$\mathsf{crs} = \left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], [\alpha^{2N}]\right\}$$

user 1: $\mathsf{pk}_1 = [t_1], \cancel{[t_1\alpha^{N+1-1}],}\ [t_1\alpha^{N+1-2}], \ldots, [t_1\alpha^{N+1-N}]$

user 2: $\mathsf{pk}_2 = [t_2], [t_2\alpha^{N+1-1}], \cancel{[t_2\alpha^{N+1-2}],} \ldots, [t_2\alpha^{N+1-N}]$

$\vdots$

user N: $\mathsf{pk}_N = [t_N], [t_N\alpha^{N+1-1}], [t_N\alpha^{N+1-2}], \ldots, \cancel{[t_N\alpha^{N+1-N}]}$

$\mathsf{sk}_1 = [t_1\alpha^{N+1-1}]$

$\mathsf{sk}_2 = [t_2\alpha^{N+1-2}]$

$\vdots$

$\mathsf{sk}_N = [t_N\alpha^{N+1-N}]$

$$\mathsf{ct} = \left([s], [s(\sum_{j\in S} t_j + \sum_{j\in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m\right)$$

$$m = \frac{\mathsf{ct}_3 \cdot e\left(\mathsf{ct}_1,\ [t_i\alpha^{N+1-i}] \cdot \prod_{j\in S, j\neq i} [t_j\alpha^{N+1-i}] \cdot \prod_{j\in, j\neq i} [\alpha^{N+1-i+j}]\right)}{e\left(\mathsf{ct}_2,\ [\alpha^{N+1-i}]\right)}$$

Our first DBE construction: wrapped up

$\mathsf{crs} = \{[\alpha], \ldots, [\alpha^N],$ Encryptor: $O(N)$

user 1: $\mathsf{pk}_1 = [t_1], [t_1\alpha^{N+1-1}], [t_1\alpha^{N+1-2}], \ldots, [t_1\alpha^{N+1-N}]$

user 2: $\mathsf{pk}_2 = [t_2], [t_2\alpha^{N+1-1}], [t_2\alpha^{N+1-2}], \ldots, [t_2\alpha^{N+1-N}]$

$\vdots$

user N: $\mathsf{pk}_N = [t_N], [t_N\alpha^{N+1-1}], [t_N\alpha^{N+1-2}], \ldots, [t_N\alpha^{N+1-N}]$

$\mathsf{sk}_1 = [t_1\alpha^{N+1-1}]$

$\mathsf{sk}_2 = [t_2\alpha^{N+1-2}]$

$\vdots$

$\mathsf{sk}_N = [t_N\alpha^{N+1-N}]$

$$\mathsf{ct} = \left([s], [s(\sum_{j\in S} t_j + \sum_{j\in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right)$$

$$m = \frac{\mathsf{ct}_3 \cdot e\left(\mathsf{ct}_1,\ [t_i\alpha^{N+1-i}] \cdot \prod_{j\in S, j\neq i} [t_j\alpha^{N+1-i}] \cdot \prod_{j\in, j\neq i} [\alpha^{N+1-i+j}]\right)}{e\left(\mathsf{ct}_2,\ [\alpha^{N+1-i}]\right)}$$

Our first DBE construction: wrapped up

$\mathsf{crs} = \{[\alpha], \ldots, [\alpha^N], [\alpha^{N+1}], \ldots, [\alpha^{2N}]\}$

Encryptor: $O(N)$

Decryptor i: $O(N)$

user 1: $\mathsf{pk}_1 = [t_1], [t_1\alpha^{N+1-1}], [t_1\alpha^{N+1-2}], \ldots, [t_1\alpha^{N+1-N}]$

user 2: $\mathsf{pk}_2 = [t_2], [t_2\alpha^{N+1-1}], [t_2\alpha^{N+1-2}], \ldots, [t_2\alpha^{N+1-N}]$

$\vdots$

user N: $\mathsf{pk}_N = [t_N], [t_N\alpha^{N+1-1}], [t_N\alpha^{N+1-2}], \ldots, [t_N\alpha^{N+1-N}]$

$\mathsf{sk}_1 = [t_1\alpha^{N+1-1}]$

$\mathsf{sk}_2 = [t_2\alpha^{N+1-2}]$

$\vdots$

$\mathsf{sk}_N = [t_N\alpha^{N+1-N}]$

$$\mathsf{ct} = \left([s], [s(\sum_{j \in S} t_j + \sum_{j \in S} \alpha^j)], e\left([s\alpha], [\alpha^N]\right) \cdot m \right)$$

$$m = \frac{\mathsf{ct}_3 \cdot e\left(\mathsf{ct}_1,\ [t_i\alpha^{N+1-i}] \cdot \prod_{j \in S, j \neq i} [t_j\alpha^{N+1-i}] \cdot \prod_{j \in, j \neq i} [\alpha^{N+1-i+j}]\right)}{e\left(\mathsf{ct}_2,\ [\alpha^{N+1-i}]\right)}$$

Our first DBE construction
Semi-Selective Security

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathscr{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathscr{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathscr{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathscr{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

- crs: directly

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathcal{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

- crs: directly
- sk_j for $j \notin S^*$: $t_j \leftarrow_\$ \mathbb{Z}_p^*$ honestly

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathcal{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

- crs: directly
- sk_j for $j \notin S^*$: $t_j \leftarrow_\$ \mathbb{Z}_p^*$ honestly
- sk_j for $j \in S^*$: $t_j = \tilde{t}_j - \alpha^j$ implicitly, for $\tilde{t}_j \leftarrow_\$ \mathbb{Z}_p^*$

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathcal{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

- crs: directly
- sk_j for $j \notin S^*$: $t_j \leftarrow_\$ \mathbb{Z}_p^*$ honestly
- sk_j for $j \in S^*$: $t_j = \tilde{t}_j - \alpha^j$ implicitly, for $\tilde{t}_j \leftarrow_\$ \mathbb{Z}_p^*$
- Then for any $S^{**} \subseteq S^*$:

Our first DBE construction
Semi-Selective Security

BDHE assumption [BGW05]

Given $\left\{[\alpha], \ldots, [\alpha^N], [\alpha^{N+2}], \ldots, [\alpha^{2N}], [s]\right\}$ (where $\alpha \leftarrow_\$ \mathbb{Z}_p^*, R \leftarrow_\$ \mathbb{G}_T$):

$$[s\alpha^{N+1}]_T \approx_c R$$

Semi-Selective Security:

$\mathscr{A}$ chooses a target set S^*, then can pick any challenge subset $S^{**} \subseteq S^*$

Reduction to BDHE, simulate:

- crs: directly
- sk_j for $j \notin S^*$: $t_j \leftarrow_\$ \mathbb{Z}_p^*$ honestly
- sk_j for $j \in S^*$: $t_j = \tilde{t}_j - \alpha^j$ implicitly, for $\tilde{t}_j \leftarrow_\$ \mathbb{Z}_p^*$
- Then for any $S^{**} \subseteq S^*$:

$$\mathsf{ct} = \left([s],\ [s \sum_{j \in S^{**}} (t_j + \alpha^j)],\ e\left([s\alpha], [\alpha^N]\right) \cdot m\right) = \left([s],\ [s \sum_{j \in S^{**}} \tilde{t}_j],\ e\left([s\alpha], [\alpha^N]\right) \cdot m\right)$$

Our first DBE construction
More in the paper

- ★ Malicious keys registration: pairing checks
- ★ $\log(N)$ number of key updates updates [GHMRS18]
- ★ Adaptive security in the RO model [GW09]

Our first DBE construction
Concrete Efficiency

For the BLS12-381 curve and $N = 1024$ users:

- $|\mathsf{crs}| = 288\text{KB}$
- $|\mathsf{pk}_j| = 191.9\text{KB}$
- $|\mathsf{sk}_j| = 0.19\text{KB}$
- $|\mathsf{BB}| = 191.9\text{MB}$
- $|\mathsf{Encryptor}| = 96\text{KB}$
- $|\mathsf{Decryptor}| = 95.1\text{KB}$
- $|\mathsf{ct}| = 13\text{KB}$
- #Decryptor updates $= 10$

Our first DBE construction
Summary

★ Start from [BGW05].

★ ‘Distribute’ it ➜ the way keys are generated.

★ Apply the [GHMRS18] transformation for logarithmic updates.

★ Use the pairing to define **isValid** for malicious key registration.

★ Use [GW09] to achieve adaptive security.

➡ Concretely efficient DBE from Bilinear Groups.

Our second DBE construction
Summary

★ Start from [GKW18].

★ ‘Distribute’ it ➜ the way keys are generated.

★ Apply the [GHMRS18] transformation for logarithmic updates.

★ Use the pairing to define **isValid** for malicious key registration.

★ ~~Use [GW09] to achieve adaptive security.~~ Adaptive Security directly via DSEM.

➡ Concretely efficient DBE from Bilinear Groups from k-Lin (SXDH for k=1).

Efficiency Comparison with traditional BE

The cost of 'distributing'

From BDHE:

	Distributed	Security	|Enc|	|Dec|	|ct|	|BB|	#updates
BGW05	No	Selective	O(N)	O(N)	3	–	–
WQZD10	Yes	Selective	$O(N^2)$	$O(N^2)$	3	$O(N^3)$	log(N)/N
Constr. 1	Yes	Selective	O(N)	O(N)	3log(N)/3	$O(N^2)$	log(N)/N
Constr. 1	Yes	Adaptive	O(N)	O(N)	6log(N)/6	$O(N^2)$	log(N)/N

From k-Lin (k=1):

	Distributed	Security	|Enc|	|Dec|	|ct|	|BB|	#updates
GKW18	No	Adaptive	O(N)	O(N)	4	–	–
Constr. 2	Yes	Adaptive	O(N)	O(N)	4log(N)/4	$O(N^2)$	log(N)/N

Conclusion: Trade the trusted authority for an $O(N^2)$-BB

CRS vs Private Key Generator

Concurrent work:
[FFW23]

Levels of trust:

PKG **>** structured CRS **>** Updatable CRS **>** Transparent CRS **>** No CRS

Traditional BE

Our second Construction

Our first Construction

[WQZD10]
[BZ14]

Summary & Open Problems

What was there:

- ❖ [WQZD10] definition, (inefficient) construction from Pairings, (without a formal proof).
- ❖ [BZ14] definition, construction from iO.

What we did:

- ★ Enriched definition, two simple and practical constructions from Pairings.

Open problems:

- DBE from Lattices.
- Optimal (**polylog**(N)) parameters for DBE.
- Efficient DBE with trasparent setup.
- Sublinear |Decryptor|, |Encryptor|, subquadratic |BB| ➜ very important for messaging.
- Anonymous DBE ➜ Meta-data hiding!

Thank you!

ePrint: https://eprint.iacr.org/2023/874