Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems | School of Information Systems

3-2016

CCA-secure keyed-fully homomorphic encryption

Junzuo LAI

DENG, Robert H.
Singapore Management University, robertdeng@smu.edu.sg

Changshe MA

Kouichi SAKURAI

Jian WENG

DOI: https://doi.org/10.1007/978-3-662-49384-7_4

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

 Part of the Information Security Commons

Citation
LAI, Junzuo; DENG, Robert H.; MA, Changshe; SAKURAI, Kouichi; and WENG, Jian. CCA-secure keyed-fully homomorphic encryption. (2016). *Proceedings of the 19th International Conference on the Theory and Practice of Public-Key Cryptography (PKC 2016)*. 70-98. Research Collection School Of Information Systems.
Available at: https://ink.library.smu.edu.sg/sis_research/3352

This Conference Proceeding Article is brought to you for free and open access by the School of Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email libIR@smu.edu.sg.

CCA-Secure Keyed-Fully Homomorphic Encryption

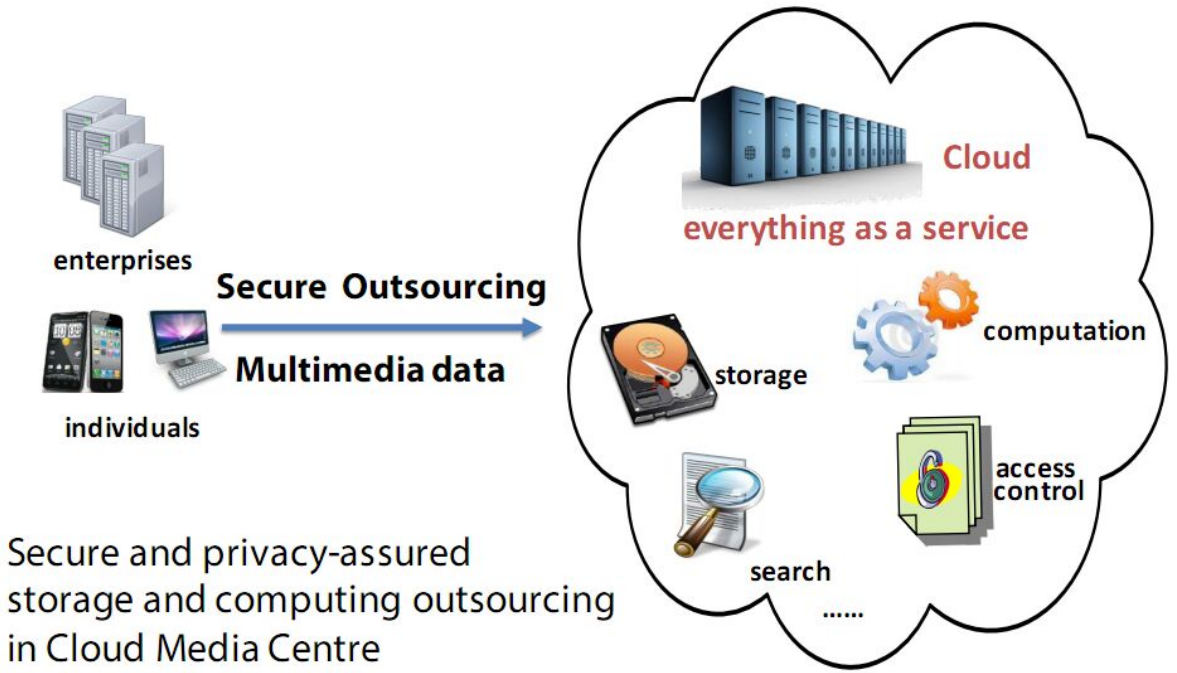
Junzuo Lai, Robert H. Deng, Changshe Ma, Kouichi Sakurai and Jian Weng

Outline

- Background
- Related Work
- CCA-Secure Keyed-Fully Homomorphic Encryption
- Conclusion

Background

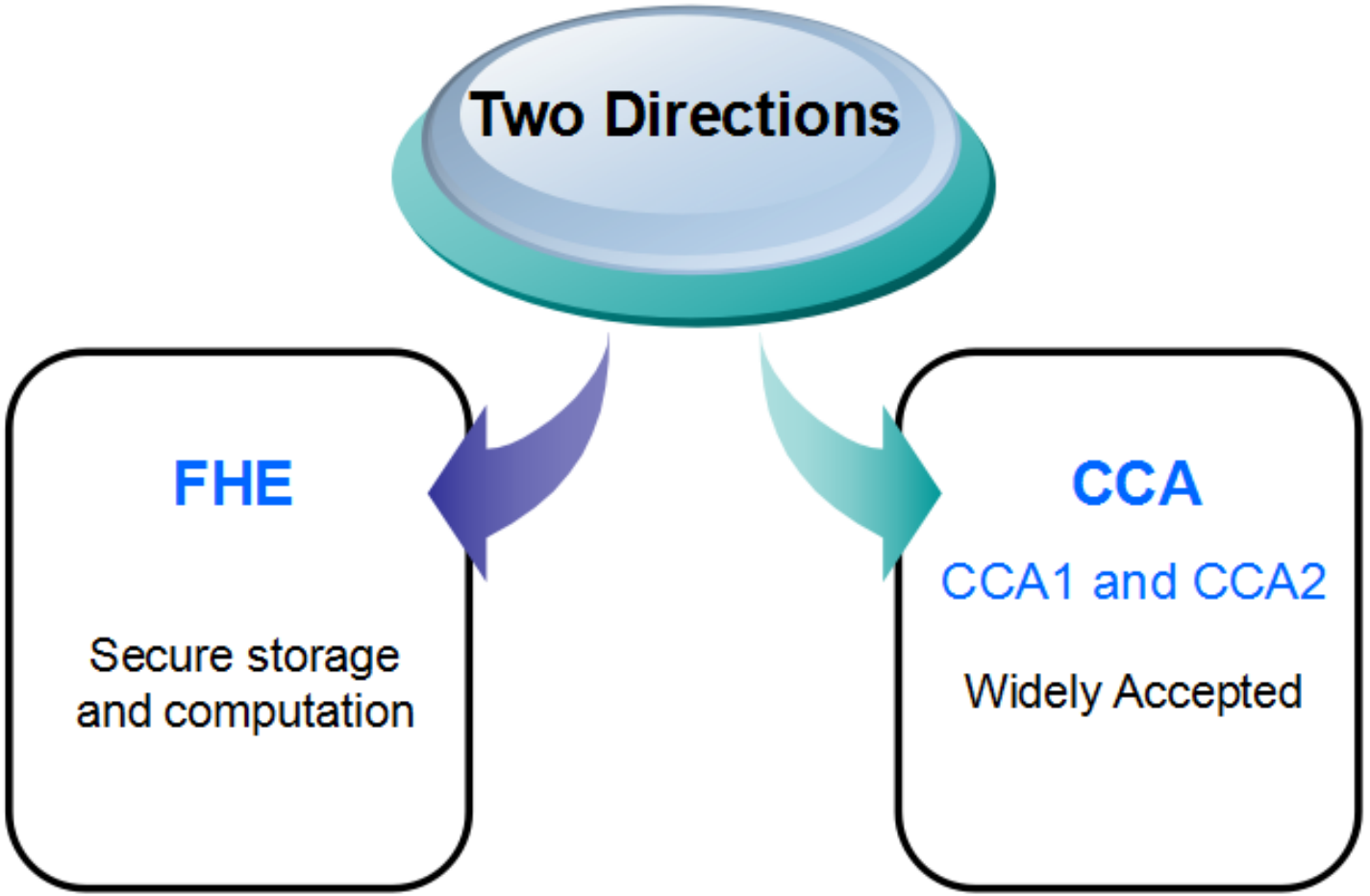
- Cloud storage and computing provide a set of resources and services through networks.
- One approach with privacy-preserving computation is fully homomorphic encryption (The Holy Grail of Cryptography).



Background

- In 1978, Rivest *et al.* left an open problem of constructing a fully homomorphic encryption scheme.
- In the early researches, additive homomorphism [GM82, Pail99], multiplicative homomorphism [RSA78, ElG84], additive homomorphism and one-time multiplication [BGN05].
- In 2009, Craig Gentry presented the first fully homomorphic encryption scheme, which opens the curtain for the study of fully homomorphic encryption.

Related Work



FHE's Current Research

- So far, most FHE schemes satisfy IND-CPA secure.
- Zhang et al.[ZPS12] present a CCA1 attack for the IND-CPA secure fully homomorphic encryption [DGH+10] proposed in EUROCRYPT 2010.
- It is well-known that CCA security and the homomorphic property cannot be achieved simultaneously.
- In present, constructing CCA1 secure fully homomorphic encryption scheme is still open.

CCA Fully Homomorphic Encryption

- Prabhakaran- Rosulek [PR08] proposed a new notion called homomorphic CCA which only allows some specified computations on encrypted data.
- Boneh-Segev-Waters[BSW12] also proposed a similar concept: targeted malleability.
- Emura et al. [EHO+13] suggested a new primitive called keyed-homomorphic encryption, where homomorphic ciphertext manipulations are only possible to a party holding a devoted evaluation key EK which, by itself, does not enable decryption.

Keyed-Homomorphic PKE [EHO+13]

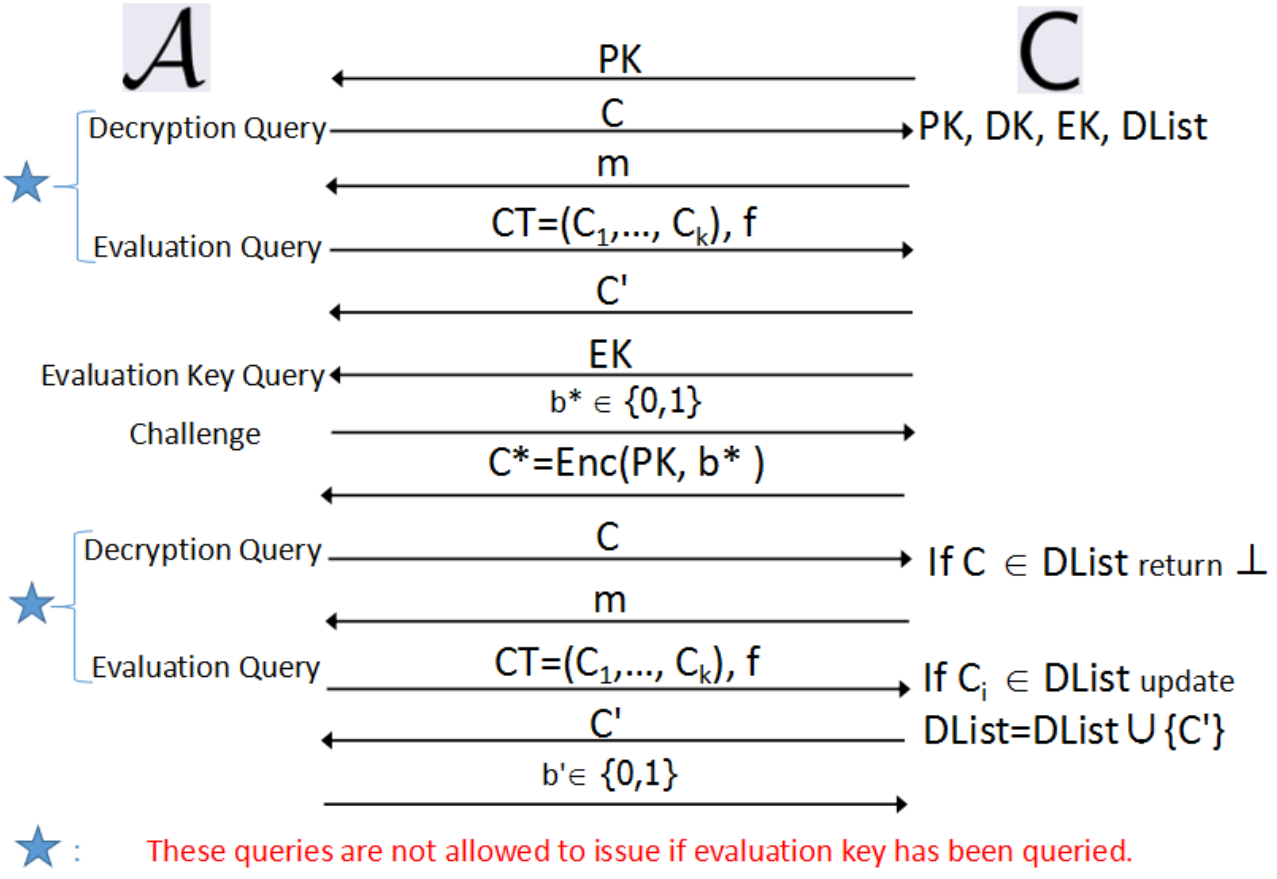
- Main ideas:
  - Cramer-Shoup [CS02b] show that IND-CCA2 secure PKE and IND-CCA1 secure PKE can be constructed by using universal-2 Hash Proof Systems (HPS) and universal-1 hash proof systems respectively.
  - Emura *et al.* show： a trapdoor can degenerate universal-2 HPS to homomorphic universal-1 HPS; In turn, universal-1 HPS can be transformed into universal-2 HPS with the same trapdoor.
  - Based on the above specified uiversal-2 HPS, they proposed a generic construction of keyed-HE.
- In present, constructing HPS that supports additive homomorphism and multiplicative homomorphism simultaneously is still open. Emura *et al.*'s approach cannot be employ to construct keyed FHE.

Keyed-Fully Homomorphic Encryption

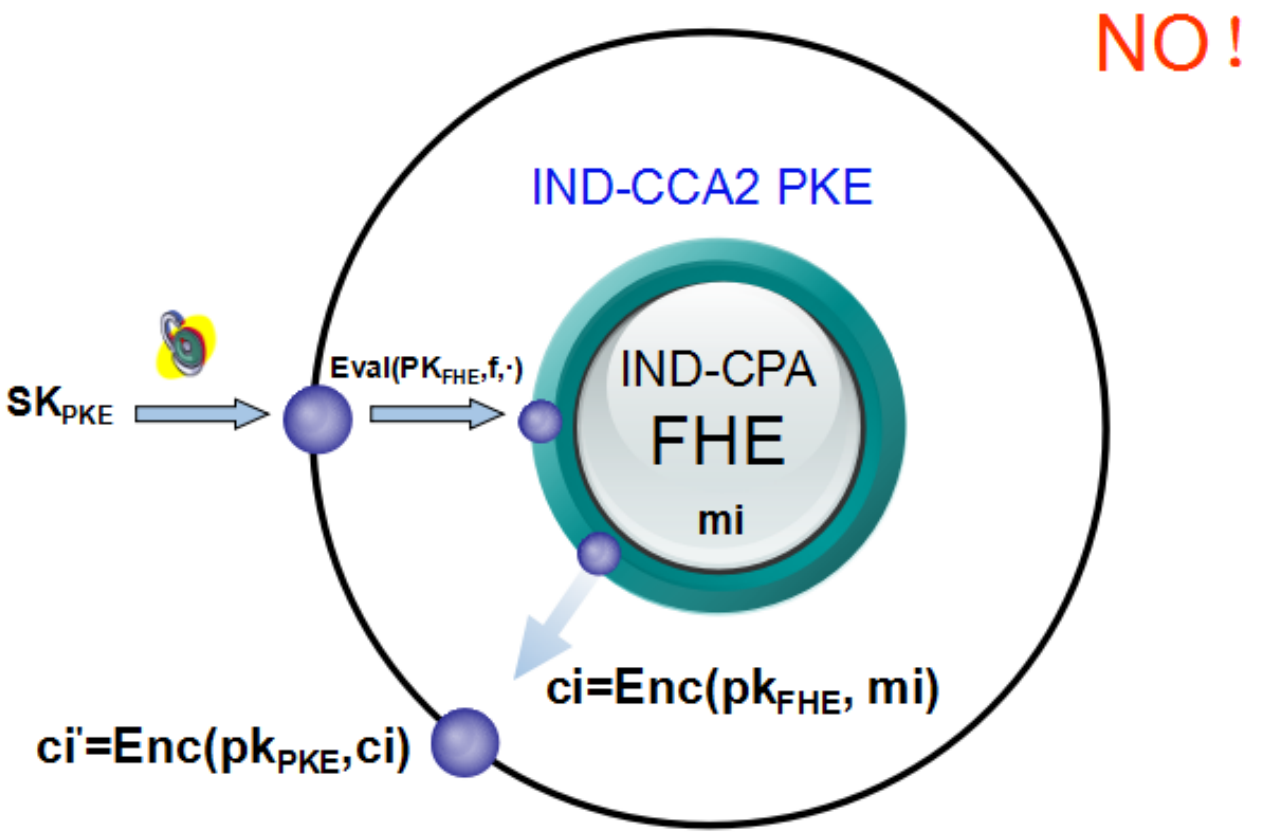
- $\mathsf{Setup}(1^k)$: outputs a decryption key DK and an evaluation key EK.
- $\mathsf{Enc}(\mathsf{PK}, b)$: takes as input a public key PK and a message bit b. It outputs a ciphertext C。
- $\mathsf{Dec}(\mathsf{PK}, \mathsf{DK}, C)$: takes as input a public key PK, a decryption key DK and a ciphertext C. It outputs a message bit b or $\perp$.
- $\mathsf{Eval}(\mathsf{PK}, \mathsf{EK}, \vec{CT}, f)$： takes as input a public key PK, an evaluation key EK, a tuple of ciphertexts $\vec{CT}$ and a Boolean circuit f. It outputs a ciphertext C.

Keyed FHE's Security Model

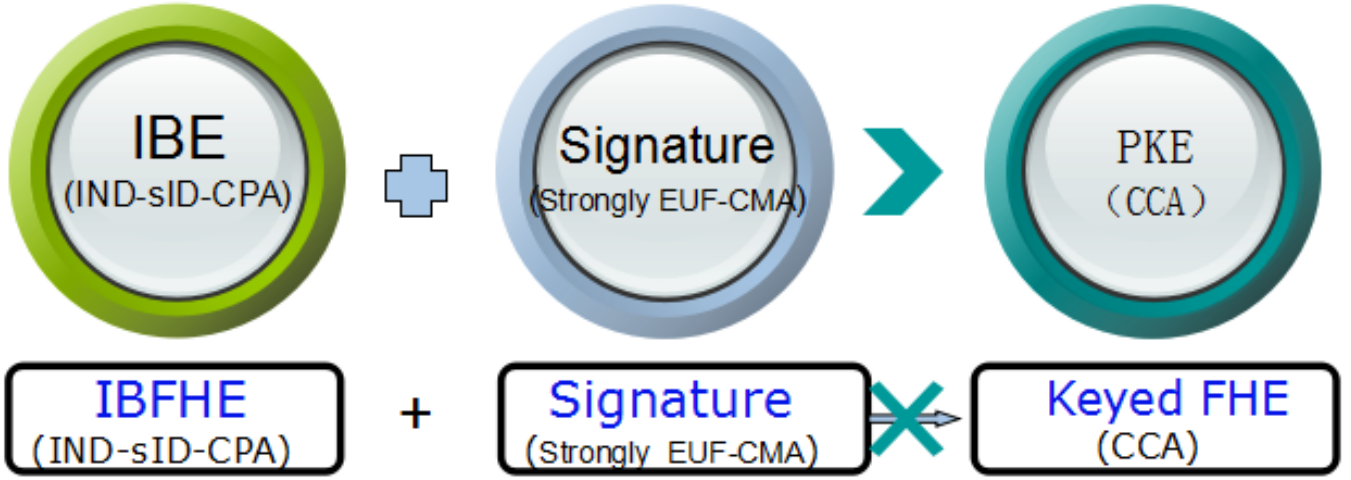
CCA Security



Double Encryption Methodology: First Attempt



CHK Transformation: Second Attempt

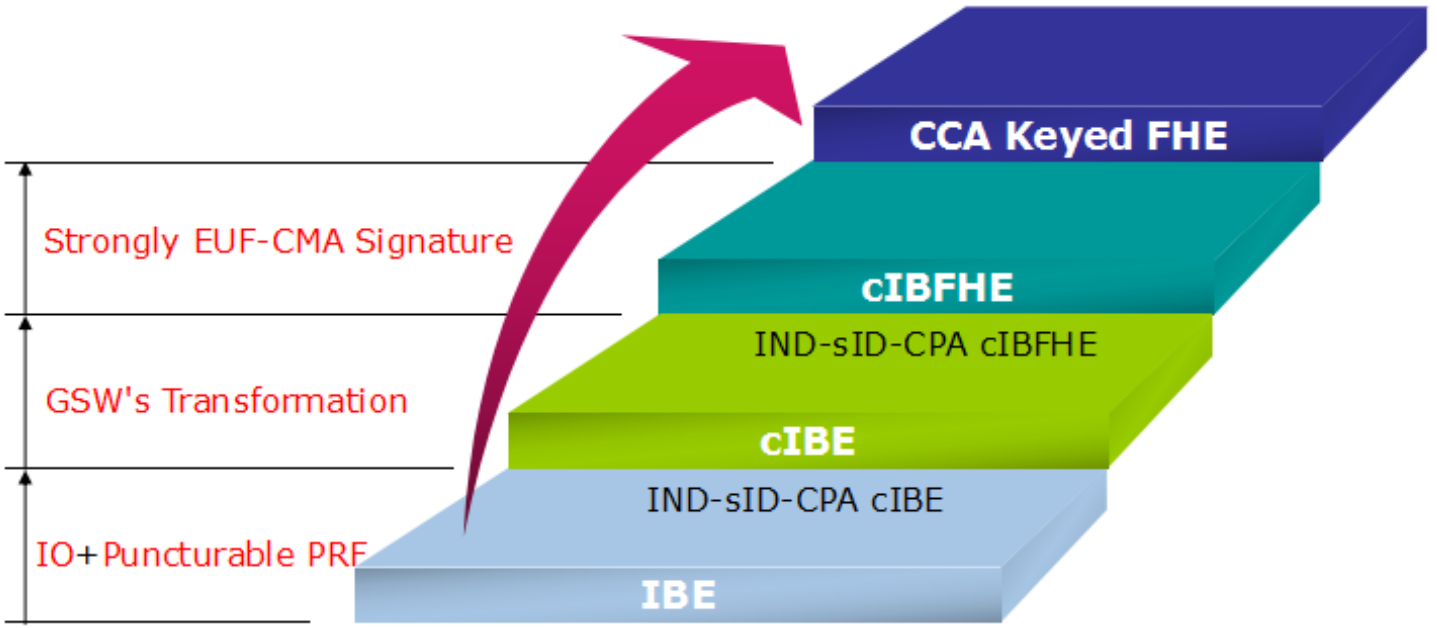


The transformation of CHK generates different user's ciphertext.

Our Solution

- We provide an approach to converting a ciphertext CT under any identity ID into a ciphertext $\widetilde{CT}$ under the designated identity $\widetilde{\mathsf{ID}}$.
- For transformation correctness, we need be able to check whether a ciphertext is well-formed. We resort to the recent advances in indistinguishability obfuscation to overcome the obstacle.
- We define a new primitive named convertible identity-based fully homomorphic encryption (cIBFHE).

Our Construction: Main Idea



[ABB10b] adaptively secure IBE

cIBFHE: Definition and Security

cIBFHE = (Setup, Extract, GenerateTK, Encrypt, Transform, Decrypt, Evaluate).

Two algorithms

- $\mathsf{GenerateTK}(\mathsf{PP}, \mathsf{MK}, \widetilde{\mathsf{ID}}) \rightarrow \mathsf{TK}_{\mapsto \widetilde{\mathsf{ID}}}$ for identity $\widetilde{\mathsf{ID}}$.
- $\mathsf{Transform}(\mathsf{PP}, \mathsf{TK}_{\mapsto \widetilde{\mathsf{ID}}}, \mathsf{ID}, \mathsf{CT}) \rightarrow \widetilde{\mathsf{CT}}$ under identity $\widetilde{\mathsf{ID}}$.

Security

- Setup: Send PP to the adversary $\mathcal{A}$.
- Query phase 1: $\mathcal{A}$ adaptively issues the following queries:
  - **GetSK**$\langle \mathsf{ID} \rangle$: $\mathcal{C}$ returns $\mathsf{SK}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID})$.
  - **GetTK**$\langle \mathsf{ID} \rangle$: $\mathcal{C}$ returns $\mathsf{TK}_{\mapsto \mathsf{ID}} \leftarrow \mathsf{GenerateTK}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID})$.
- Challenge: $\mathcal{C}$ returns $\mathsf{CT}^* \leftarrow \mathsf{Encrypt}(\mathsf{PP}, \mathsf{ID}^*, b^*)$.
- Query phase 2
- Guess

Keyed FHE: General Construction

A cIBFHE and a signature $\mathcal{S} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$.

- $\mathsf{Setup}(1^\kappa) : (\mathsf{PP}, \mathsf{MK}) \leftarrow \mathsf{cIBE.Setup}(1^\kappa)$, $(\widetilde{vk}, \widetilde{sk}) \leftarrow \mathcal{S}.\mathsf{Gen}(1^\kappa)$, $\mathsf{TK}_{\mapsto \widetilde{vk}} \leftarrow \mathsf{cIBE.GenerateTK}(\mathsf{PP}, \mathsf{MK}, \widetilde{vk})$.
  $\mathsf{PK} = \mathsf{PP}$, $\mathsf{DK} = \mathsf{MK}$, $\mathsf{EK} = (\widetilde{vk}, \widetilde{sk}, \mathsf{TK}_{\mapsto \widetilde{vk}})$.
- $\mathsf{Enc}(\mathsf{PK}, b \in \{0,1\})$: It proceeds as follows.
  1. Run $\mathcal{S}.\mathsf{Gen}(1^\kappa)$ to obtain a key pair (vk, sk).
  2. Compute $\mathsf{CT} \leftarrow \mathsf{cIBE.Encrypt}(\mathsf{PP}, vk, b)$ and $\sigma \leftarrow \mathcal{S}.\mathsf{Sign}(sk, \mathsf{CT})$ and output $C = (vk, \mathsf{CT}, \sigma)$.
- $\mathsf{Dec}(\mathsf{PK}, \mathsf{DK}, C)$: $\mathcal{S}.\mathsf{Vrfy}(vk, \mathsf{CT}, \sigma) = 1$, $\mathsf{SK}_{vk} \leftarrow \mathsf{cIBE.Extract}(\mathsf{PP}, \mathsf{MK}, vk)$,$b \leftarrow \mathsf{cIBE.Decrypt}(\mathsf{PP}, \mathsf{SK}_{vk}, \mathsf{CT})$.

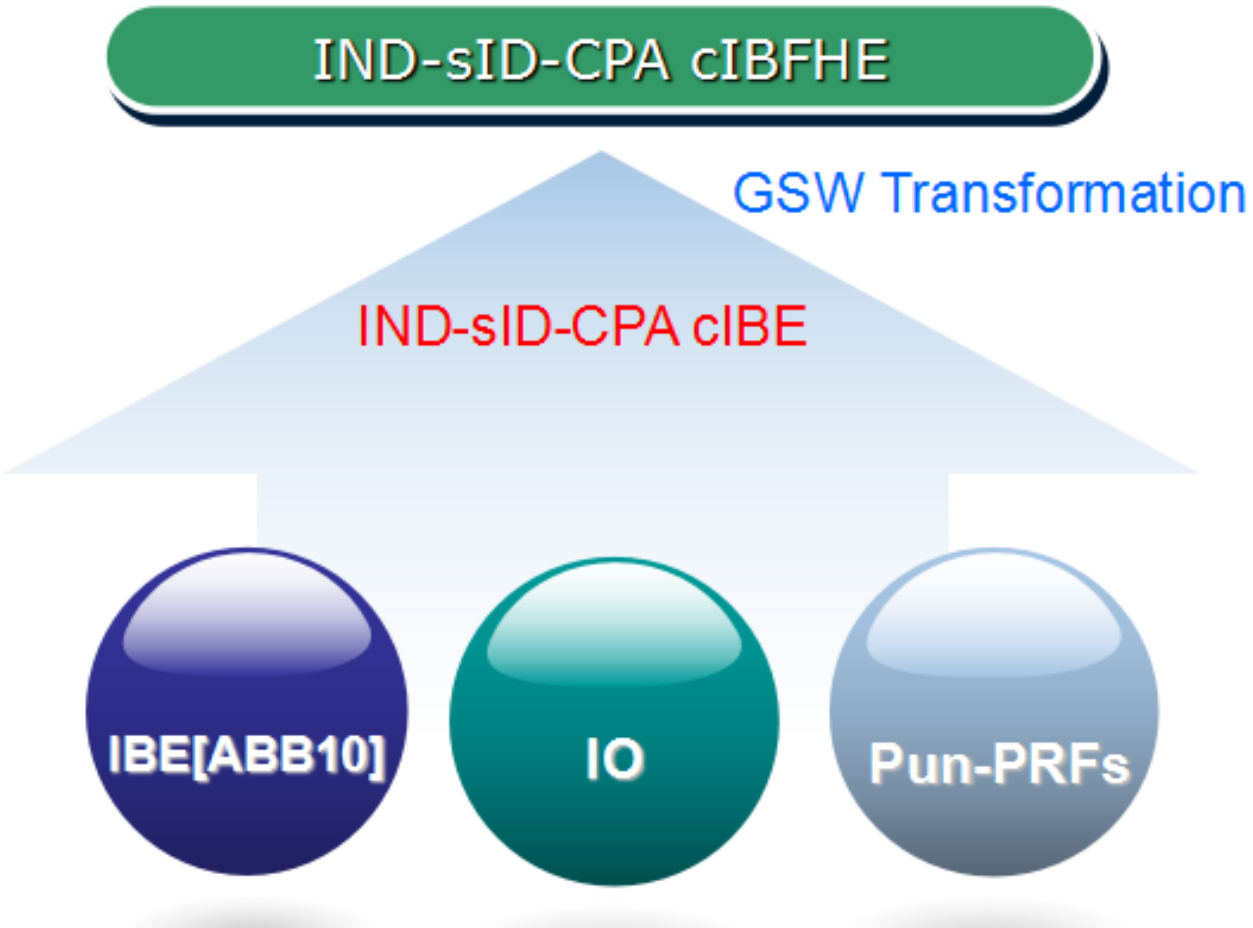
- $\mathsf{Eval}(\mathsf{PK}, \mathsf{EK}, \vec{C}, f)$: For $i = 1, \ldots, k$, it proceeds as follows.
  1. Check whether $\mathcal{S}.\mathsf{Vrfy}(vk_i, \mathsf{CT}_i, \sigma_i) = 1$. If not, it outputs $\perp$.
  2. Compute $\widetilde{\mathsf{CT}}_i \leftarrow \mathsf{cIBE.Transform}(\mathsf{PP}, \mathsf{TK}_{\mapsto \widetilde{vk}}, vk_i, \mathsf{CT}_i)$.

  Compute $\widetilde{\mathsf{CT}} \leftarrow \mathsf{cIBE.Evaluate}(\mathsf{PP}, \widetilde{vk}, (\widetilde{\mathsf{CT}}_1, \ldots, \widetilde{\mathsf{CT}}_k), f)$,
  $\tilde{\sigma} \leftarrow \mathcal{S}.\mathsf{Sign}(\widetilde{sk}, \mathsf{CT})$ and outputs the ciphertext $C = (\widetilde{vk}, \widetilde{\mathsf{CT}}, \tilde{\sigma})$.

Theorem

If the underlying convertible IBFHE scheme is IND-sID-CPA secure, and the signature scheme $\mathcal{S}$ is strongly EUF-CMA secure, then our proposed keyed-FHE scheme is CCA-secure.

cIBFHE's Construction



cIBE's Construction

[ABB10] Adaptively-secure IBE

Ciphertext $c_0 = u^\top s + x + b\lfloor \frac{q}{2} \rfloor \in \mathbb{Z}_q$,

$$c_1 = F_{\mathsf{ID}}^\top s + \begin{pmatrix} y \\ R_{\mathsf{ID}}^\top y \end{pmatrix} \in \mathbb{Z}_q^{2m}$$

where $F_{\mathsf{ID}} = A \mid B_0 + \sum_{i=1}^{\ell} d_i B_i$, $R_{\mathsf{ID}} = \sum_{i=1}^{\ell} d_i R_i$

cIBE

Property: To provide an approach to converting a ciphertext CT under any identity ID from [ABB10] into a ciphertext $\widetilde{CT}$ under the designated identity $\widetilde{\mathsf{ID}}$.

Methods: $i\mathcal{O}$ and Puncturable PRFs.

Security: IND-sID-CPA secure based on LWE assumption.

Indistinguishability Obfuscator ($i\mathcal{O}$)

A uniform probabilistic polynomial time (PPT) machine $i\mathcal{O}$ is called an indistinguishability obfuscator for a circuit class $\{\mathcal{C}_\lambda\}_{\lambda\in\mathbb{N}}$ if the following conditions are satisfied:

1. Correctness: For all security parameters $\lambda \in \mathbb{N}$, for all $C \in \mathcal{C}_\lambda$, and for all input x, we have that
$$\Pr[C'(x) = C(x) : C' \leftarrow i\mathcal{O}(\lambda, C)] = 1.$$
2. Security: For any (not necessarily uniform) PPT distinguisher D, for all pairs of circuits $C_0, C_1 \in \mathcal{C}_\lambda$ such that $C_0(x) = C_1(x)$ on all inputs x the following distinguishing advantage is negligible:
$$\mathsf{Adv}^{D}_{i\mathcal{O},C_0,C_1}(\lambda) := |\Pr[D(i\mathcal{O}(\lambda, C_0)) = 1] - \Pr[D(i\mathcal{O}(\lambda, C_1)) = 1]|.$$

Puncturable PRFs

A puncturable pseudorandom function (PRF):

- Correctness: For every PPT algorithm which on input a security parameter λ outputs a set $S \subseteq \{0,1\}^n$, for all $x \in \{0,1\}^n \backslash S$, we have that

$$\Pr[\mathsf{Eval}_F(K\{S\}, x) = F(K, x) : K \leftarrow \mathcal{K}, K\{S\} \leftarrow \mathsf{Puncture}_F(K, S)] = 1.$$

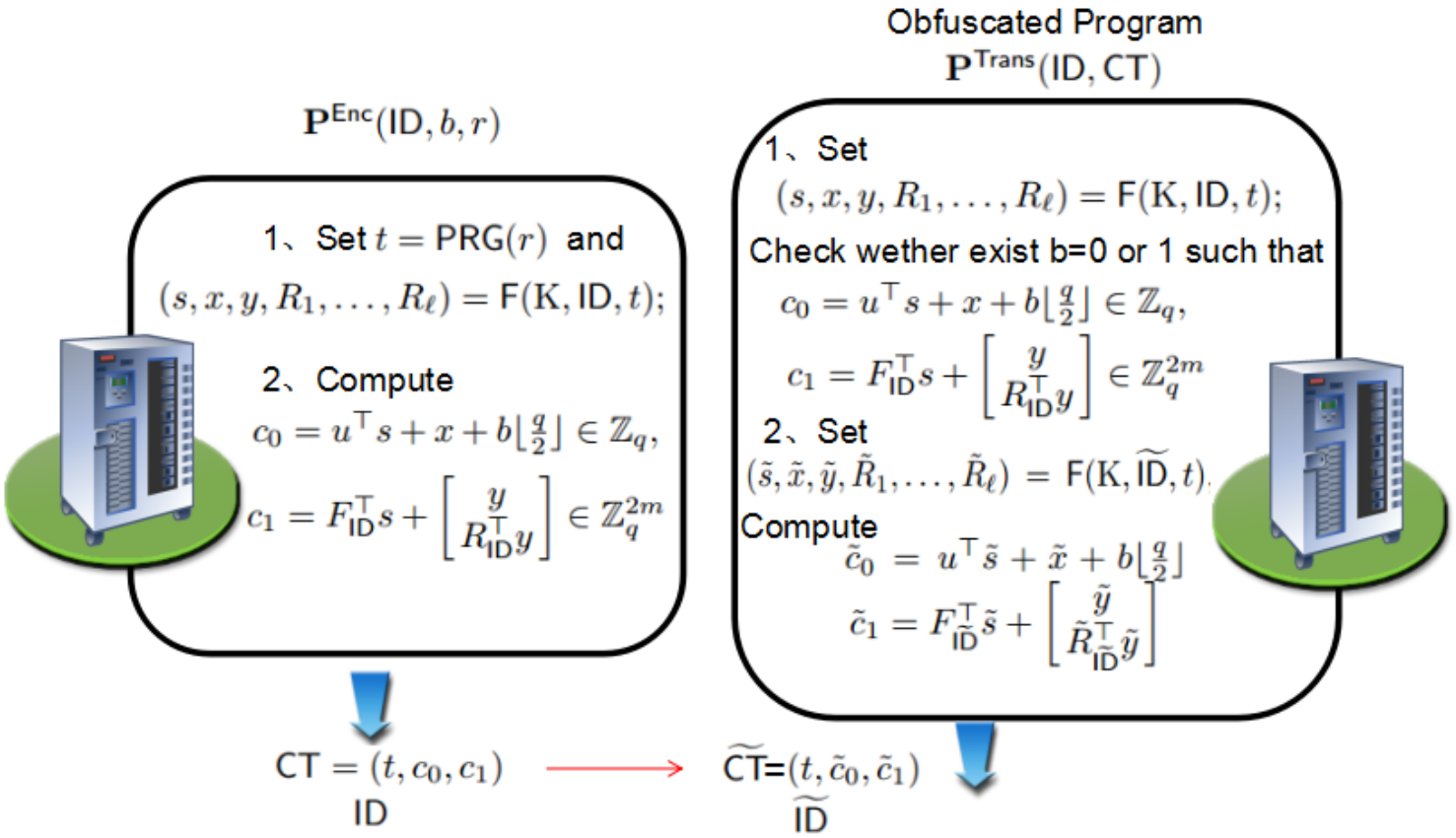
Puncturable PRFs

- Security: For any PPT algorithm $\mathcal{A}$, the following distinguishing advantage is negligible:

$$\begin{aligned}\mathsf{Adv}_F^{\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(S, K\{S\}, F(K,S)) = 1 : S \leftarrow \mathcal{A}(\lambda),\\ K\{S\} \leftarrow \mathsf{Puncture}_F(K,S)] -\\ \Pr[\mathcal{A}(S, K\{S\}, U_{\bar{\ell}\cdot|S|}) = 1 : S \leftarrow \mathcal{A}(\lambda),\\ K\{S\} \leftarrow \mathsf{Puncture}_F(K,S)]|,\end{aligned}$$

where $F(K,S)$ denotes the concatenation of $F(K,x_1), \cdots, F(K,x_k)$, $S = \{x_1, \cdots, x_k\}$ is the enumeration of the elements of S in lexicographic order, $\bar{\ell}$ denotes the bit-length of the output $F(K,x)$, and U_ℓ denotes the uniform distribution over $\bar{\ell}$ bits.

cIBE's Construction



Conclusion

- We define a new primitive cIBFHE and its IND-ID-CPA and IND-sID-CPA security.
- We propose a generic paradigm of constructing CCA-secure keyed-FHE by modifying CHK transformation slightly.
- We construct a leveled cIBFHE scheme based on the adaptively-secure IBE scheme [ABB10a].

Interesting Problems

- How to construct a verifiable FHE.
- Generic construction from identity based leveled FHE to identity based pure FHE.
- How to construct IND-CCA1 secure FHE.

THANKS

Appendix

Theorem

If the $(\mathbb{Z}_q, n, \bar{\Psi}_\alpha)$-LWE assumptions holds, the proposed convertible IBFHE scheme is IND-sID-CPA secure.

Proof Sketch:

- As for the IND-sID-CPA security of the convertible IBE scheme, we follow the line of [ABB10], i.e., utilizing the partitioning strategy.
- We define a sequence of games where the first game is the original IND-sID-CPA security game. Then we show that any PPT adversary's advantage in each game must be negligible close of that of the previous game, and the adversary's advantage in the final game is zero.
- Please see the full paper for the details.