CORE: Cooperative Encryption with Its Applications to Controllable Security Services

Ruei-Hau Hsu, Chun-I Fan
Department of Computer Science and Engineering
National Sun Yat-sen University
Kaohsiung, Taiwan, 80424
Email: rhhsu@mail.cse.nsysu.edu.tw

Jemin Lee
Department of Information and Communication Engineering
Daegu Gyeongbuk Institute of Science and Technology
Daegu, Korea, 43016
Email: jmnlee@dgist.ac.kr

Tony Q.S. Quek
Information Systems Technology and Design Pillar
Singapore University of Technology and Design
Singapore, 487372
Email: tonyquek@sutd.edu.sg

***Abstract*—This article introduces a new concept of encryption, the *cooperative encryption* (CORE), to control encryption capability, i.e., only permitted (not any) users can encrypt messages with their public keys. In CORE, any message encrypted with a user public key can be decrypted only after it is matured by the security mediator (SEM). Compared with the other security controllable encryption systems, CORE can be more flexibly applied to various encryption systems, e.g., identity-based and attribute-based encryptions, supported by encrypting user private keys tightly. Moreover, CORE can directly revoke the corrupted SEM without re-issuing user private keys and prevent from producing a complete ciphertext, which is not cooperatively computed with the SEM, by users with the disclosed security mediated keys. Generally, CORE is of unique interest in fully controlling user encryption/decryption behaviors in controllable security services, and supporting distributed SEMs for large scale networks. This work also develops the framework of CORE, and proposes two constructions of CORE to identity-based and attribute-based encryptions, which are empowered to achieve two essential security properties, i.e., immediate revocation and unforgeable encryption. In addition, this work analyzes the security and performance of the proposed schemes. Overall, CORE provides a new notion of controllable security without involving any trust authority or key management server.**

I. Introduction

The public key encryption has been proposed as an elaborate encryption system for secure communication in public networks. In this system, anyone can encrypt a message with the public key of a specified user, and the encrypted message can only be decrypted with the corresponding private key of the user. Typically, the trust of users' public keys relies on a so-called public key infrastructure (PKI), where every public key is certified by a certificate authority (CA). In PKI, certificate management incurs costs in synchronizing certificates of users on both CA and user sides. To simplify certificate management, the identity-based encryption (IBE) [1], [2] is proposed, where the public key of a user is her/his identity and the corresponding user private key is issued with the identity and master secret key of the IBE system. For example, a user Alice can simply send an encrypted email to Bob with his email address (bob@abc.123.com) in a secure email system using IBE system, where every user's identity is his/her email address.

As we consider public key encryption system as a security service running on an online system (e.g., secure email or secure messenger systems), how to manage the security service with immediate control on user security capability is of utmost importance. The *security capability control* brings two particular interests in public key system: 1) immediate revocation on users to avoid further disclosure of encrypted messages with compromised user private keys and prevent from man-in-the-middle attacks, which lead to the certificate verification of out-of-date revocation lists from attackers, to public key certificates, and 2) authorization to user encryption behavior to control users fully. The immediate revocation for IBE has been resolved by security mediated cryptography [3]–[5], where the decryption of users need to contact a so-called security mediator (SEM) to complete. Through controlling decryption capability of users, the SEM can check first whether the user is revoked before finishing decryption.

However, the security mediated system has some security issues. First, the SEM based encryption systems [3]–[5] rely on a strong security assumption that the SEM is fully trusted, where he private key of a user $d_i = d_{SEM,i} + d_{u,i}$ and $d_{SEM,i}$ for each user is stored on the SEM. This will lead the system corruption by a corrupted SEM. In [5], users can directly produce a complete ciphertext by $C = (M||\sigma) \oplus H(r \cdot P_A)$ and it can be decrypted by $C \oplus H(x_A \cdot U) = M||\sigma$, where $P_A = x_A \cdot P$, $U = r \cdot P$, x_A is the user private key, and r is randomly selected. It is inevitable that users may simply encrypt messages with the public key and result in the control of encryption and decryption capabilities becoming infeasible. Another critical issue in security mediated system is the corruption of SEM. As the user mediated keys are stored

978-1-5386-5790-4/18/$31.00

on an SEM, the corruption of the SEM will arise the failure of the entire security system to mediated security function, i.e., the decryption can be accomplished by the receiver itself. This makes the architecture of distributed SEM impractical. Most of the security mediated encryption system cannot simply resolve this issue since the user mediated keys are separated from the full user private key. As the user mediated keys are exposed, the entire system public/private keys need to be re-generated [5], or the user private and mediated keys have to be re-issued [3], [4], [6], [7].

The security capability control is also urgently required for attribute-based encryption (ABE) [8]. The ABE is a foundation of attribute-based access control (ABAC) mechanism, where a sender encrypts data with a specified policy and a receiver decrypts the data with the private key, associated with the attributes that satisfy the policy on the ciphertext. In some critical applications such as electronic health record [9] and disruption-tolerant military networks [10], attributes should be revocable to prevent any confidential data, which can be decrypted by some disclosed private decryption keys, from being exposed to unwanted receivers. Another application requiring security capability control in ABE is device-to-device (D2D) communication. In D2D communications, infrastructure needs to control issuing device discovery requests, such that only permitted devices can issue a device discovery request with ABAC based on ABE system. If the device discovery request is not controlled, the attribute information of users may be exposed by the following attack. The initiator, i.e., the discoverer, may issue two device discovery requests with different policies, $(x_1 \vee x_2 \vee x_3, x_2 \vee x_3)$ on the attributes (x_1, x_2, x_3) and check whether a device satisfies the policies by verifying whether they share the same session keys. If the responses of the device are $(true, false)$, the initiator can derive that the attributes of this device are $(x_1 = true, x_2 = false, x_3 = false)$. Even each device uses an anonymous protection technique to hide its position information, it can be revealed by some positioning techniques such as [11], [12], since the transmitted signals are received at a certain device with some correlation. Once a malicious discoverer gets the attribute information of the device, it will be able to guess whether responses to a discovery request are from the same device. Therefore, the ability of issuing device discovery requests should be controlled.

User behaviors such as device discovery request and decryption can be controlled by forcing users to connect to security servers for checking the critical information, such as revoked user or attribute list. Moreover, when cloud portal service providers (e.g., Google, Microsoft, and Yahoo) wants to support application programming interface (API) to be customized by users, an inattentive programmer may simply use the security keys to support the security of their applications. Therefore, it is essential to support the controllable security mechanism to avoid the possible human mistakes in this sense.

The main contributions of the paper can be summarized as follows: this paper

1) presents a novel *cooperative encryption* framework, the function definitions of CORE to IBE and ABE, and their security definitions;
2) shows the paradigm of constructing CORE based on the existing encryption systems, such as IBE and ABE;
3) provides how to support immediate control on the encryption/decryption capabilities of senders and receivers;
4) discusses how to support distributed SEMs and the revocation on the corrupted SEM; and
5) introduces the applications of the proposed *cooperative IBE* (CoIBE) and CoABE, i.e., a concrete device discovery protocol with fine-grained access control based on CoABE, which demonstrates the power of CORE.

The organization of this paper for the remaining sections are shown as follows. Section II introduces the related works. Section III shows the definitions of cooperative IBE and cooperative ABE. Section IV presents the constructions of a cooperative IBE and a cooperative ABE. Section V analyzes the security of the proposed schemes and their performance. After that, section VI concludes this work.

II. Related Works

This section discusses the state of the art related to the concept of CORE as follows. Use of SEM to control the security capability of users to achieve immediate user revocation is first proposed based on RSA [4], [6]. Another RSA-based security mediated encryption [3] is proposed by utilizing a common modulus to create user identities and private keys. However, they are claimed to be insecure to the common modulus attack and CCA Security model by [7]. Hence, two identity-based security mediated schemes [5], [7] are proposed by generic constructions. In [5], the message is encrypted by two secrets, which are encrypted an IBE and a public key encryption. The secrets can be decrypted by the user's private key and the private key of the SEM, respectively. This generic construction suffers from malicious encryptors, who can simply produce a ciphertext that is not cooperatively computed by the SEM and can be decrypted by its receiver. Thus, the SEM loses the control on security capabilities. Moreover, performance will be an issue when the SEM is required for every decryption. This is especially impractical when a security mediated encryption applies to a multi-receiver encryption, i.e., broadcast, attribute-based encryptions, etc. Distributing security mediated keys to multiple SEM may relieve the performance issue, but it will make the mediated key easier to be exposed. Hence, performance becomes critical when distributed SEMs is impractical. As well, the encryption system need to re-issue the mediated keys for all users once the SEM is compromised.

In ABE, immediate revocation to attributes is also urgently required in some critical applications. A security mediated ciphertext-policy ABE (mCP-ABE) [13] is proposed to support revocation on the attributes of each user private key. However, the performance of decryption is a bottleneck when each encryption may involve multiple users to contact the SEM for decryption. Besides that, for the corruption of the SEM the system also need to re-issue all mediated keys for all users.

Revocable identity-based encryptions (R-IBE) [14], [15] are proposed to control the decryption capabilities of users and revoke compromised identities. By updating the public key and user private keys with the corresponding revocation list of identities, only the private keys of unrevoked identities can decrypt ciphertexts successfully. To this end, R-IBE subjects the key authority to compute a key update information and all unrevoked users to update their private key upon this key update information. Regardless of the private key update procedure, users can still encrypt messages and decrypt them with the out-of-date private keys. Hence, R-IBE cannot achieve immediate user revocation.

Time-released encryption (TRE) [16]–[18] based on trust server approach is similar to the concept of security mediated encryption, where the trust time server will release time keys for the decryption of the ciphertexts encrypted with time information. The difference is that the time key is for all users and the immediate revocation is impractical in TRE.

III. Definitions of Cooperative Identity-based and Attribute-based Encryptions

This section defines cooperative identity-based encryption (CoIBE) and attribute-based encryption (CoABE), respectively.

A. Cooperative Identity-based Encryption

The function definition of CoIBE is given as follows.

Definition 1. *A CoIBE consistes of the following algorithms:*

- ***Setup***(λ) *takes a security parameter* λ *as an input, and outputs a public parameter* $params$*, a master secret key* msk*, and a cooperative secret key* csk*.*
- ***KeyGen***$(\mathsf{ID}, params, msk, csk)$ *takes an identity* ID*,* $params$*,* msk*, and* csk *as inputs, and outputs a user private key* d_{ID}*.*
- ***PreEnc***$(params, \mathsf{ID}, M)$ *takes* $params$*,* ID*, and a message* M *as inputs, and outputs a partial ciphertext* PC*.*
- ***CoEnc***$(params, PC, csk, \mathsf{ID})$ *take* $params$*,* PC*,* csk*, and* ID *as inputs, and outputs a complete ciphertext* C*.*
- ***Dec***$(params, C, d_{ID})$ *takes* $params$*,* C*, and* d_{ID} *as inputs, and outputs a message* M*.*

A CoIBE is correct if

$$M = \textbf{\textit{Dec}}(params, C, d_{ID})$$

for all $(params, msk, csk)$*, generated by* ***Setup***(λ)*, all* d_{ID}*, generated by* ***KeyGen***$(\mathsf{ID}, params, msk, csk)$*, and any message* M*, where*

$$C = \textbf{\textit{CoEnc}}(params, PC, csk, \mathsf{ID}),$$
$$PC = \textbf{\textit{PreEnc}}(params, \mathsf{ID}, M).$$

The security requirements of CoIBE are to control the security capability of each user, In CoIBE, a message is first encrypted for a designated identity as a partial ciphertext and it cannot be decrypted by the user private key. The security of the partial ciphertext should be guaranteed against the outsider and the SEM. The complete ciphertext that can be decrypted with the user key can only be produced with the cooperative key in the SEM. Moreover, no one can produce a complete ciphertext without the help of the SEM. That is, the unforgeability of the ciphertext against encryptor can be provided.

The security definitions of CoIBE that fulfill the security requirements are shown by the following games, played by an adversary who is a passive adversary and to break the security of ciphertext for CoIBE.

Definition 2 (IBE Semantic Security)**.** *A CoIBE is semantically secure if no probabilistic polynomial-time (PPT) adversary* $\mathfrak{A}$*, who issues at most* q *private key queries in time* t*, can win the following game with at least* ϵ *advantage by interacting with a challenger* $\mathfrak{C}$*.*

Simulation Setup. $\mathfrak{C}$ runs the **Setup** algorithm to generate the required parameters, $params, mpk, msk$, and csk, for Co-IBE system. It then gives mpk, msk, and csk to $\mathfrak{A}$.

Phase 1. $\mathfrak{A}$ is allowed to make polynomial amount of queries for d_{ID}s with any given IDs.

Challenge. $\mathfrak{A}$ submits a target identity ID^*, and two selected messages, M_0 and M_1, where ID^* has not been queried for the corresponding user private key in Phase 1. $\mathfrak{C}$ then returns $C = \textbf{CoEnc}(params, PC, csk, \mathsf{ID}^*)$, and $PC = \textbf{PreEnc}(params, \mathsf{ID}^*, M_b)$ according to a random bit $b \in \{0, 1\}$.

Phase 2. Phase 1 is repeated with the restriction that $\mathfrak{A}$ cannot request d_{ID} when $\mathsf{ID} = \mathsf{ID}^*$.

Output. $\mathfrak{A}$ output a guess b' of b.
$\mathfrak{A}$ wins the game if $b' = b$ and d_{ID^*} has never been queried.

In the security definition of IBE semantic security for CoIBE, $\mathfrak{A}$ is additionally given csk compared to the security definition of a traditional type of IBE.

Besides IBE semantic security, the security of partial ciphertext, i.e., PC, against the intended receiver (i.e., the user of ID^*) should also be guaranteed in CoIBE. This models that the partial ciphertext, PC, for ID^* cannot be decrypted until it is computed with csk as a complete ciphertext C. The security definition is given as follows.

Definition 3 (IBE Semantic Security to Partial Ciphertext)**.** *A CoIBE is semantically secure to partial ciphertext against receivers if no PPT adversary* $\mathfrak{A}$*, who issues at most* q *private key queries in time* t*, can win the following game with at least* ϵ *advantage by interacting with a challenger* $\mathfrak{C}$*.*

Simulation Setup. $\mathfrak{C}$ runs the **Setup** algorithm to generate the required parameters, $params, mpk, msk$, and csk, for Co-IBE system. It then gives mpks to $\mathfrak{A}$.

Phase 1. $\mathfrak{A}$ is allowed to make polynomial amount of queries for d_{ID}'s with any given ID's.

Challenge. $\mathfrak{A}$ submits a target identity ID^*, and two selected messages, M_0 and M_1. $\mathfrak{C}$ then returns $PC = \textbf{PreEnc}(params, \mathsf{ID}^*, M_b)$ according to a random bit $b \in \{0, 1\}$.

Phase 2. Phase 1 is repeated.

Output. $\mathfrak{A}$ output a guess b' of b.

$\mathfrak{A}$ wins the game if $b' = b$ and d_{ID^*} has never been queried. The advantage of $\mathfrak{A}$ in this game is defined as $\mathbf{Adv}_{\mathfrak{A}} = \left|\Pr[b = b'] - \frac{1}{2}\right|$.

In CoIBE, we additionally consider the adversary, who breaks the control of security capability. That is, the adversary can simply produce a complete ciphertext C of any target ID and C can be decrypted by the user private key of ID without calling **CoEnc**. The security definition for the adversary of this type is given as follows.

Definition 4. *A CoIBE is complete-ciphertext-unforgeable if no PPT adversary $\mathfrak{A}$, who issues polynomial amount of private key queries, **PreEnc**, and **CoEnc** queries, can win the following game with at least ϵ advantage by interacting with a challenger $\mathfrak{C}$.*

Simulation Setup. $\mathfrak{C}$ runs the **Setup** algorithm to generate the required parameters,$params, mpk, msk$, and csk, for Co-IBE system. It then gives mpks to $\mathfrak{A}$.

Phase 1. $\mathfrak{A}$ is allowed to make polynomial amount of queries for d_{ID}'s with any given ID's. $\mathfrak{A}$ is also allowed to make polynomial amount of queries for any PC encrypted with queried ID to **CoEnc**.

Challenge. $\mathfrak{C}$ give a target identity ID^* to $\mathfrak{A}$.

Phase 2. Phase 1 is repeated, except the queries on ID^*.

Output. $\mathfrak{A}$ output C for the target ID^* and the corresponding message M.

$\mathfrak{A}$ wins the game if $M = \textbf{Decrypt}(\mathsf{params}, \mathsf{C}, \mathsf{ID}^*)$.

We separate the adversaries of CoIBE into three types as follows:

- **Type-1 Adversary**: This kind of adversaries obtain no knowledge of the master secret key, all user secret keys, and the cooperative secret key, and try to break the semantic security of CoIBE.
- **Type-2 Adversary**: This kind of adversaries obtain no knowledge of the master secret key and all user secret keys, and try to break the semantic security of CoIBE.
- **Type-3 Adversary**: This kind of adversaries try to produce a ciphertext, which can be decrypted simply with the target user private key, without knowing the cooperative.

Those types of adversaries are considered in security analysis of proposed CoIBE in Sec. V.

B. Cooperative Attribute-based Encryption

In this subsection, the CoABE is defined with attribute-matching based on inner product operation. Let F be the class of boolean functions $F : \Sigma_k \times \Sigma_e \to \{0, 1\}$, where Σ_k and Σ_e denote *key attribute* and *ciphertext attribute* spaces.

Definition 5. *A CoABE for the class of F consists of the following algorithms.*

- ***Setup***(λ) *takes a security parameter λ as an input, and outputs a master public key mpk and a master secret key msk.*
- ***KeyGen***$(msk, mpk, \vec{x})$ *takes the master secret key msk, the master public key mpk, and a key attribute set $\vec{x}$ as inputs, and outputs a private decryption key $sk_{\vec{x}}$ associated with $\vec{x}$.*
- ***PreEnc***$(mpk, \vec{y}, M)$ *takes mpk, a ciphertext attribute set $\vec{y}$, and a message M in an associated message space as inputs, and outputs a partial ciphertext $PC_{\vec{y}}$.*
- ***CoEnc***$(mpk, PC_{\vec{y}}, csk, \vec{y})$ *takes mpk, $PC_{\vec{y}}$, csk, and $\vec{y}$ as inputs, and outputs a complete ciphertext $C_{\vec{y}}$.*
- ***Dec***$(sk_{\vec{x}}, \vec{y}, C_{\vec{y}})$ *takes a private decryption key $sk_{\vec{x}}$ and a ciphertext $C_{\vec{y}}$ as inputs, and if $\langle \vec{x} \cdot \vec{y} \rangle = 0$, outputs the message M, otherwise, outputs meaningless symbol $\perp$.*

*For the correctness of CoABE, the following conditions need to be satisfied: for all λ, $(mpk, msk, csk) \leftarrow$ **Setup**(λ), $\vec{x} \in \Sigma_k$, $sk_{\vec{x}} \leftarrow$ **KeyGen**$(msk, mpk, \vec{x})$, and $\vec{y} \in \Sigma_e$, where*

$$PC_{\vec{y}} = \textbf{PreEnc}(mpk, \vec{y}, M),$$
$$C_{\vec{y}} = \textbf{CoEnc}(mpk, PC_{\vec{y}}, csk, \vec{y}).$$

- *If $F(\vec{x}, \vec{y}) = 1 \implies \langle \vec{x} \cdot \vec{y} \rangle = 0$, then **Dec**$(sk_{\vec{x}}, \vec{y}, C_{\vec{y}}) = M$.*
- *If $F(\vec{x}, \vec{y}) = 0 \implies \langle \vec{x} \cdot \vec{y} \rangle \neq 0$, then **Dec**$(sk_{\vec{x}}, \vec{y}, C_{\vec{y}}) = \perp$ with all but only negligible probability.*

The implementation of F is to realize access control on decrypting ciphertexts of CoABE for a specific policy and the corresponding attributes. As shown in [19]–[21], those encryption schemes realize the relations or functions corresponding to the evaluation of polynomial or CNF/DNF formulae based on inner product, such that $\vec{x} \cdot \vec{y} = 0$ is embedded in the operation of decryption.

A CoABE for is said to be semantically secure if it satisfies the following security definition.

Definition 6. *A CoAB is semantically secure if no PPT adversary $\mathfrak{A}$ wins the following game with non-negligible probability ϵ.*

Simulation Setup. $\mathfrak{C}$ *runs* $(msk, mpk) \leftarrow$ ***Setup***(λ) *and gives mpk to $\mathfrak{A}$.*

Phase 1. $\mathfrak{A}$ *submits polynomial amount of private key queries for any given key attribute set $\vec{x}$ and get $sk_{\vec{x}} \leftarrow$ **KeyGen**$(msk, mpk, \vec{x})$.*

Challenge. $\mathfrak{A}$ *submits messages M_0, M_1, and a target policy set $\vec{y}^*$ such that $F(\vec{x}, \vec{y}^*) = 0$ for all queried key attribute sets $\vec{x}$. The challenger computes a challenge ciphertext $C^* \leftarrow$ **CoEnc**$(mpk, PC_{\vec{y}} =$ **PreEnc**$(mpk, \vec{y}, M_b), csk, \vec{y})$ according to random bits $b \in \{0, 1\}$.*

Query Phase 2. $\mathfrak{A}$ *submits private key queries for the given key attribute $\vec{x}$, which is different from the key attribute sets submitted in Phase 1, such that $F(\vec{x}, \vec{y}^*) = 0$.*

Guess. $\mathfrak{A}$ *outputs guesses $b' \in \{0, 1\}$ and wins the game if $b' = b$.*

The security definitions of CoABE against Type-2 and Type-3 adversaries are similar to those of the CoIBE, except that for CoABE case, the adversaries may submit private decryption key queries on the selected attributes excluding the target attributes. In CoABE, the security against Type-2 adversary is to demonstrate if the SEM is able to decrypt

the encrypted messages by any given policy, and the security against Type-3 adversary is to demonstrate if any sender can produce a complete ciphertext with any given policy without the cooperation of the SEM and the ciphertext can be simply decrypted with the private decryption key associated with the attributes, that satisfy the policy.

IV. Construction of Cooperative Encryption

This section presents two constructions of collaborative encryption, i.e., CoIBE and CoABE.

A. Construction of CoIBE

Before introducing the construction of CoIBE, we review the cryptographic preliminary first. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two bilinear groups of prime order $p > 2^\lambda$ with an efficiently computable mapping $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, such that for any $(g, h) \in \mathbb{G}^2$, $(a, b) \in \mathbb{Z}^2$, and $e(g, h) \neq 1$, $e(g^a, h^b) = e(g, h)^{ab}$ [22].

To construct a cooperative IBE, we choose an selective-ID secure IBE by Boneh and Boyen (BB-IBE) [23] as a basis, and constraint the encryption capability using *linear encryption (LE)* [24], which is defined as follows.

Definition 7 (Linear Encryption)**.** *The public key and the private key of a user are $pk = (u, v, h)$ and $sk = (x', y')$ such that $u^{x'} = v^{y'} = h$, respectively, where $(u, v, h) \in \mathbb{G}^3$ and $(a, b) \in \mathbb{Z}_p^2$. The ciphertext of the Linear encryption on a message M is $C_2 = (\hat{C}, T_1, T_2) = (M \cdot h^{\alpha+\beta}, u^\alpha, v^\beta)$. The user can decrypt the ciphertext by $M = \hat{C}/(T_1^{x'} \cdot T_2^{y'})$.*

Specifically, the encryption capability of a sender is constrained by encrypting the user private key by LE with public parameters. The secret key to decrypt the LE on the user private key is kept as a cooperative secret key on the SEM. To do so, the CoIBE system provides two types of secrets, the *master secret key* and the *cooperative secret key*, and keeps them secure on the key generation center (KGC) and the SEM, respectively.

The construction of CoIBE is shown as follows:

- **Setup**(λ) : The system administrator runs this algorithm and generates $msk = (\alpha, \beta, x, y, \delta)$, $csk = (x', y')$, and $params = (\mathcal{X} = g^x,\ \mathcal{Y} = g^y,\ \tilde{\mathcal{X}} = g^{\delta \cdot x},\ \tilde{\mathcal{Y}} = g^{\delta \cdot y},\ \Delta = g^\delta,\ g,\ u,\ v,\ u^\alpha,\ v^\beta,\ u^\delta,\ h)$ according to the given security parameter λ, where $(x, y, x', y', \alpha, \beta, \delta) \in \mathbb{Z}_p^7$ and $(g, u, v) \in \mathbb{G}^3$ are randomly selected, and $h = u^{x'} = v^{y'}$. Then, it publishes $params$ and gives msk to the KGC and csk to the SEM.
- **KeyGen**$(\mathsf{ID}, msk, csk, params)$: The KGC runs this algorithm by taking as input ID, msk, csk and $params$, and generating user private key $d_{\mathsf{ID}} = (K = g^{\frac{1}{\mathsf{ID}+x+ry}} \cdot h^{(\alpha+\beta)\cdot\delta}, r)$, where $r \in \mathbb{Z}_p$ is randomly selected. It gives d_{ID} to the user ID.
- **PreEnc**$(params, \mathsf{ID}, M)$: A sender, who wants to send a message M to the user ID, runs this algorithm to produce the first partial ciphertext as follows:

$$PC = \Big\{ A = g^{s\cdot\mathsf{ID}} \cdot \mathcal{X}^s, B = \mathcal{Y}^s, \\ D = (M \cdot e(g,g)^s), E = \big(e_1 = (u^\alpha)^s, e_2 = (v^\beta)^s\big) \Big\} \tag{1}$$

  where $s \in \mathbb{Z}_p$ is randomly selected.
- **CoEnc**$(params, PC, csk, \mathsf{ID})$: The SEM runs this algorithm by taking $params$, PC, csk, and ID as inputs. It then computes a complete ciphertext as follows:

$$C = \Big\{ A' = A \cdot g^{s'\cdot\mathsf{ID}} \cdot \mathcal{X}^{s'}, B' = B \cdot \mathcal{Y}^{s'}, \\ D' = D \cdot e(g,g)^{s'}, E' = \big(e_1' = (e_1 \cdot (u^\alpha)^{s'})^{x'}, \\ e_2' = (e_2 \cdot (v^\beta)^{s'})^{y'}\big) \Big\} \tag{2}$$

- **Dec**$(params, C, d_{\mathsf{ID}})$: The user ID takes mpk, d_{ID} and C as inputs, and decrypts C to obtain M as

$$M = \frac{D' \cdot e(e_1' \cdot e_2', \Delta^{\mathsf{ID}} \cdot \tilde{\mathcal{X}} \cdot \tilde{\mathcal{Y}}^r)}{e(K, A' \cdot B'^r)}. \tag{3}$$

B. Construction of CoABE

The proposed CoABEs supports to encrypt a message associated with a specified policy and decrypt a ciphertext by the key embedding given attributes. The attribute set of a secret key is defined as $\vec{x} = (x_1, ..., x_n)$ and the vector of a ciphertext policy is defined as $\vec{y} = (y_1, ..., y_n)$ where n is the number of attributes. AND-gate and OR-gate policy can be realized by applying inner product of two vectors on decryption as shown in [19]. For the construction of CoABE, we choose the zero inner-product functional encryption by Attrapadung, Libert, and Panafieu (ALP-FE) [25], and the CoABE consists of the followings algorithms.

- **Setup**(λ) : This algorithm takes λ as input to obtain $msk = \{e(g,g)^\alpha, \alpha, \alpha', \beta', \delta\}$, $csk = \{x', y'\}$, and $mpk = \{g, e(g,g)^\alpha, h_0, \vec{H} = g^{\vec{\alpha}}, u^{\alpha'}, v^{\beta'}, \Delta = g^\delta\}$, where $\alpha, \alpha', \beta', \delta, x', y', \vec{\alpha} = \{\alpha_1, ..., \alpha_n\}$ are randomly selected from $\mathbb{Z}_p$, $\vec{H} = \{h_1, ..., h_n\}$, and $h_i = g^{\alpha_i}$.
- **KeyGen**$(msk, csk, mpk, \vec{x})$: It takes msk, csk, mpk, and $\vec{x}$ as the inputs and generates $sk_{\vec{x}} = (D_1, D_2, \{K_i\}_{i\in[2,n]})$ where

$$D_1 = g^\alpha \cdot h_0^r \cdot h^{(\alpha'+\beta')\cdot\delta}, D_2 = g^r, K_i = (h_1^{\frac{-x_i}{x_1}} \cdot h_i)^r.$$

- **PreEnc**$(mpk, \vec{y}, M)$: This algorithm takes mpk, $\vec{y}$, and the message M as inputs and computes the partial ciphertext $PC = (C_1, C_2, C_3, e_1, e_2)$ where

$$C_1 = g^s, C_2 = (h_0 \cdot h_1^{y_1} \cdots h_n^{y_n})^s, C_3 = M \cdot e(g,g)^{\alpha\cdot s}, \\ e_1 = (u^{\alpha'})^s, e_2 = (v^{\beta'})^s. \tag{4}$$

- **CoEnc**$(mpk, csk, PC, \vec{y})$: This algorithm computes a complete ciphertext $C = (C'_1, C'_2, C'_3, e'_1, e'_2)$, where

$$C'_1 = C_1 \cdot g^{s'}, C'_2 = C_2 \cdot (h_0 \cdot h_1^{y_1} \cdots h_n^{y_n})^{s'},$$
$$C'_3 = C_3 \cdot e(g,g)^{\alpha \cdot s'}, \quad (5)$$
$$e'_1 = (e_1 \cdot (u^{\alpha'})^{s'})^{x'}, e'_2 = (e_2 \cdot (v^{\beta'})^{s'})^{y'}$$

- **Dec**$(sk_{\vec{x}}, \vec{y}, C)$: This algorithm decrypts C as

$$M = \frac{e(C'_1, D_1 \cdot K_2^{y_2} \cdots K_n^{y_n})}{e(C'_2, D_2) \cdot e(\Delta, e'_1 \cdot e'_2)}. \quad (6)$$

.

The private decryption key can be encrypted by the same way of CoIBE. Then the ciphertext will be embedded the secret key of the LE in **CoEnc** by the SEM. Hence, the private decryption key, protected by LE, can be used to decrypt the complete ciphertext. Moreover, we observe that the paradigm, encrypting on user private key, of providing cooperative encryption in IBE is easy to be adopt to ABE as well.

C. Compromised SEM Revocation

In this subsection, we discuss how the revocation to a compromised SEM can be supported in CORE. For this, we first present how the proposed CoIBE can be extended for distributed SEMs, and then discuss how to revoke a compromised SEM. Note that we present the revocation process for CoIBE only as it is the same in CoABE.

1) Distributed SEMs: The proposed CoIBE can be extended for distributed SEMs. For the master public keys, secret keys, and cooperative secret keys, the system generates $mpk' = mpk \cup \{u_i, v_i, u_i^{\alpha_i}, v_i^{\beta_i}, u_i^{\delta}\}_{i\in[1,n]}$, $\{csk_i = (x'_i, y'_i)\}_{i\in[1,n]}$, and $msk' = msk \cup \{\alpha_i, \beta_i\}_{i\in[1,n]}$, where $u_i^{x'_i} = v_i^{y_i}$. The system also generates $tk_i = (h^{(\alpha+\beta)\cdot\delta})^{-1} \cdot h^{(\alpha_i+\beta_i)\cdot\delta}$ as a transform key to transform the user private key for the specific SEM i and $csk_i = csk_i \cup \{tk_i\}$. The user private key d_{ID} is kept the same. For **PreEnc**, the encryptor takes ID, M, and mpk' as inputs, and computes $PC = \{A, B, D, E = (e_1 = (u_i^{\alpha_i})^s, e_2 = (v_i^{\beta_i})^s)\}$ for the SEM i, where A, B, and D are computed in the same way as that in the proposed CoIBE. For **CoEnc**, the SEM i takes mpk', csk_i, ID, and PC as inputs, and generates $C = \{A', B', D', E' = (e'_1 = (e_1 \cdot (u_i^{\alpha_i})^{s'})^{x'_i}, e'_2 = (e_2 \cdot (v_i^{\beta_i})^{s'})^{y'_i}, tk_i)\}$, where A', B', and D' are generated as the same way in the proposed CoIBE. Finally, the user ID takes mpk', d_{ID}, and C as inputs, and decrypts C as

$$M = \frac{C' \cdot e(e'_1 \cdot e'_2, \Delta^{\mathsf{ID}} \cdot \tilde{\mathcal{X}} \cdot \tilde{\mathcal{Y}})}{e(d_{\mathsf{ID}} \cdot tk_i, A' \cdot B'^r)}. \quad (7)$$

2) Revoke Compromised SEM in distributed SEMs: Once a SEM is compromised, csk can be public known. The system should renew the partial public parameters without re-issuing all the user private keys. The key update for the disclosure of csk can be done as follows. If the SEM i is compromised, the system announces the SEM i is compromised and stops encrypting message with the part of master public key, $(u_i, v_i, u^{\alpha_i}, v^{\beta_i})$, related to the SEM i. Since d_{ID} and csk_i are encrypted by msk, the security of them are kept secure, except for the compromised csk_i.

V. Security and Performance Analyses

This section analyzes the security of the proposed CoIBE and CoABE against the Type-1, Type-2 and Type-3 adversaries, and also present their performance in terms of computation cost.

A. Security Analysis

As presented in Sec. IV, BB-IBE [23], ALP-FE [25], and linear encryption (LE) [24] are used to construct CoIBE and CoABE. Hence, the security of CoIBE and CoABE are related to the security of those encryption schemes, as presented in following two theorems.

Theorem 1. *Assume BB-IBE and LE are selectively secure and semantic secure against PPT adversaries, respectively. Then, the proposed CoIBE is also selectively secure and complete ciphertext unforgeable against the Type-1, Type-2, and Type-3 adversaries, respectively.*

Proof. The proof is divided into three parts for Type-1, Type-2, and Type-3 adversaries, respectively. We first prove the security against Type-1 adversary as follows.

Security against Type-1 Adversary. We assume that there is a CoIBE Type-1 attacker $\mathfrak{A}_{CoIBE}$. We first construct a CoIBE simulator $\mathfrak{C}_{CoIBE}$, which can be considered as a challenger of CoIBE and an attacker against the security of BB-IBE. The advantage of breaking CoIBE is ϵ' and that of BB-IBE is ϵ. Before simulation, $\mathfrak{A}$ will select a target identity ID^* to attack. C_{CoIBE} and C_{BB-IBE} will simulate the schemes based on ID^*. Then, $\mathfrak{C}_{CoIBE}$ can generate the required CoIBE system parameters as follows. $\mathfrak{C}_{CoIBE}$ first obtains $g, \mathcal{X} = g^x, \mathcal{Y} = g^y$ from the BB-IBE challenger $\mathfrak{C}_{BB-IBE}$ and generates the remaining parameters, α, β, δ, and $csk = \{x', y'\}$, $params = \{\mathcal{X}, \mathcal{Y}, \tilde{\mathcal{X}} = g^{\delta \cdot x'}, \tilde{\mathcal{Y}} = g^{\delta \cdot y'}, \Delta = g^{\delta}, u, v, u^{\alpha}, v^{\beta}, u^{\delta}, h\}$, where $h = g^{\xi}$, $u = g^{\xi \cdot x'^{-1}} = h^{x'^{-1}}$, $v = g^{\xi \cdot y'^{-1}} = h^{y'^{-1}}$ such that $u^{x'} = v^{y'} = h$. For the user private key query on a given ID_i, $\mathfrak{C}_{CoIBE}$ submits the query to $\mathfrak{C}_{BB-IBE}$ and responds the query with the private key $d_{ID_i} = (K, r)$ returned from $\mathfrak{C}_{BB-IBE}$. After all queries finished, $\mathfrak{A}_{CoIBE}$ submits two messages M_0 and M_1 to $\mathfrak{C}_{CoIBE}$. Then, $\mathfrak{C}_{CoIBE}$ forwards (M_0, M_1) to $\mathfrak{C}_{BB-IBE}$, and $\mathfrak{C}_{BB-IBE}$ returns a ciphertext $C = (A = g^{s \cdot \mathsf{ID}^*} \cdot X^s, B = Y^s, D = M_b \cdot e(g,g)^s)$. Once received C from $\mathfrak{C}_{BB-IBE}$, $\mathfrak{C}_{CoIBE}$ will generates the CoIBE ciphertext $C_{CoIBE} = \{A \cdot g^{s' \cdot \mathsf{ID}} \cdot \mathcal{X}^{s'}, B \cdot \mathcal{Y}^{s'}, D \cdot e(g,g)^{s'}, (u^{\alpha \cdot s} \cdot (u^{\alpha})^{s'})^{x'}, (v^{\beta \cdot s} \cdot (v^{\beta})^{s'})^{x'}\}$ for $\mathfrak{A}_{CoIBE}$, where $u^s = g^{\xi \cdot x'^{-1} \cdot s}$ and $v^s = g^{\xi \cdot y'^{-1} \cdot s}$. In BB-IBE simulation, s is formed as l/a and g^a can be simulated by $\mathfrak{C}_{BB-IBE}$ with unknown a. Hence, $\mathfrak{C}$ can simulate $\xi \cdot x'^{-1} = a \cdot \psi_1$ and $\xi \cdot y'^{-1} = a \cdot \psi_2$, such that $u^s = g^{\psi_1 \cdot l}$ and $v^s = g^{\psi_2 \cdot l}$ by the given g^l. Finally, $\mathfrak{A}_{CoIBE}$ outputs the guess $b' \in \{0, 1\}$ on b and $\mathfrak{C}_{CoIBE}$ gives b' as the answer of the challenge from $\mathfrak{C}_{BB-IBE}$. If $b = b'$, then $\mathfrak{A}_{CoIBE}$ breaks the semantic security of CoIBE and $\mathfrak{C}_{CoIBE}$ also breaks the semantic security of BB-IBE. From the above, $\epsilon' = \Pr[b = b'] - 1/2$

and $\epsilon \geq \epsilon'$. Conclusively, ϵ' is negligible since ϵ is negligible based on the security of BB-IBE.

Security against Type-2 Adversary. The Type-2 adversary is going to break the security of partial ciphertext. It is obvious that the partial ciphertext is a complete ciphertext of BB-IBE. Hence, we can simply conclude that the partial ciphertext is semantically secure against outsiders based on the security of BB-IBE. The receiver of the target identity can be a Type-2 adversary as well. For the attacker, who has user private keys and breaks the semantic security of partial ciphertexts, we can prove the security of CoIBE based on the security of LE as follows. Assume there is a LE challenger $\mathfrak{C}_{LE}$, who generates (h, u, v), such that $h = u^{x'} = v^{y'}$. $\mathfrak{C}_{CoIBE}$ generates the essential parameters, $params$ and msk, except (α, β) is unknown, to build the encryption system, and let $csk = \{x', y'\}$, which is unknown to $\mathfrak{C}_{CoIBE}$. As $\mathfrak{A}_{CoIBE}$ submit a private key query for the target ID^*, $\mathfrak{C}_{CoIBE}$ prepares two messages M_0 and $M_1 = g^{\frac{1}{\mathsf{ID}^* + x + ry}}$, and submits to $\mathfrak{C}_{LE}$. $\mathfrak{C}_{LE}$ then encrypts one of the messages as $C = M_b \cdot h^{\alpha,\beta}$ and send it as a challenge to $\mathfrak{C}_{CoIBE}$, where α and β are randomly selected. Afterward, $\mathfrak{C}_{CoIBE}$ sends C as the answer of the private key query on ID^*. We observe that, if C is a LE ciphertext on M_1, it is a well-formed private key of ID^*. Otherwise, it is considered as a random element. Then, $\mathfrak{C}_{CoIBE}$ simulates CoIBE with $\mathfrak{A}_{CoIBE}$ for ID^*. If $\mathfrak{A}_{CoIBE}$ can decrypt the given ciphertext to ID^*, then $\mathfrak{C}_{CoIBE}$ guesses $b' = 1$. Otherwise, guessing on b randomly. If $\mathfrak{A}_{CoIBE}$ has non-negligible advantage, which is the probability of guessing the challenge from $\mathfrak{C}_{CoIBE}$ subtracts 1/2, then $\mathfrak{C}_{CoIBE}$ will also has non-negligible advantage to distinguish C being a LE ciphertext on M_0 or M_1 from $\mathfrak{C}_{Co}$. From the above, $\epsilon' \geq \epsilon$, where ϵ' is the advantage of breaking LE security and ϵ is the advantage of breaking the semantic security for partial ciphertext against receivers in CoIBE. Conclusively, ϵ is negligible since ϵ' is negligible based on LE Security.

Security against Type-3 Adversary. The Type-3 adversary is going to generate a ciphertext, which can be decrypted by the target receiver ID^*, with $params$ only. As the security for the Type-2 adversary has been proven, only the ciphertext C on a given message M for the target identity ID^* is well-formed can be decrypted by d_{ID^*} overwhelmingly. Support LE tuples (h, u, v) and (u^α, v^β) are obtained for unknown α and β. We can simulate all the required parameters for CoIBE successfully. As the adversary $\mathfrak{A}_{CoIBE}$ can send out a complete ciphertext $C = \{A', B'D', E'\}$ without the help of the SEM with $csk = \{x', y'\}$. C must contain the well-formed $E' = (e'_1, e'_2)$ and it can be used to compute $h^{(\alpha+\beta)\cdot(s+s')}$. This breaks the security of LE system. Hence, we conclude that the probability of computing a complete ciphertext without csk is negligible.

□

Theorem 2. *Assume ALP-FE is selective secure and LE is semanic secure. Then, the proposed CoABE is also semantically secure and complete ciphertext unforgeable against the Type-1, Type-2, and Type-3 adversaries, respectively.*

TABLE I
THE COMPUTATION AND COMMUNICATION COSTS

	CoIBE			
	KeyGen	**PreEnc**	**CoEnc**	**Dec**
Computation Costs	$T^{\mathbb{G}}_{\mathsf{mul}}+T^{\mathbb{G}}_{\mathsf{exp}}$ (0.934 ms)	$2T^{\mathbb{G}}_{\mathsf{exp}}$ (1.848 ms)	$4T^{\mathbb{G}}_{\mathsf{mul}} + 7T^{\mathbb{G}}_{\mathsf{exp}} + T_{\mathsf{p}}$ (7.994 ms)	$3T^{\mathbb{G}}_{\mathsf{mul}} + 2T^{\mathbb{G}}_{\mathsf{exp}} + T_{\mathsf{p}}$ (3.364 ms)
	Partial Ciphertext		Complete Ciphertext	
Length (bits)	$4L_{\mathbb{G}}$+$L_{\mathbb{G}_T}$ (1020 bits)		$4L_{\mathbb{G}}$+$L_{\mathbb{G}_T}$ (1020 bits)	
	CoABE			
	KeyGen	**PreEnc**	**CoEnc**	**Dec**
Computation Cost	$T^{\mathbb{G}}_{\mathsf{mul}}+T^{\mathbb{G}}_{\mathsf{exp}}$ (0.934 ms)	$2T^{\mathbb{G}}_{\mathsf{exp}}$ (1.848 ms)	$4T^{\mathbb{G}}_{\mathsf{mul}} + 6T^{\mathbb{G}}_{\mathsf{exp}} + T_{\mathsf{p}}$ (7.07 ms)	$T^{\mathbb{G}}_{\mathsf{mul}} + T_{\mathsf{p}}$ (1.496 ms)
	Partial Ciphertext		Complete Ciphertext	
Length (bits)	$4L_{\mathbb{G}}$+$L_{\mathbb{G}_T}$ (1020 bits)		$4L_{\mathbb{G}}$+$L_{\mathbb{G}_T}$ (1020 bits)	

$T^{\mathbb{G}}_{\mathsf{mul}}, T^{\mathbb{G}}_{\mathsf{exp}}, T_{\mathsf{p}}, T^{\mathbb{G}_T}_{\mathsf{mul}}, T^{\mathbb{G}_T}_{\mathsf{exp}}$ represent the computation times of multiplication in $\mathbb{G}$, exponentiation in $\mathbb{G}$, pairing operation, multiplication, and exponentiation in $\mathbb{G}_T$, where $T^{\mathbb{G}}_{\mathsf{mul}} \approx 0.01$ ms, $T^{\mathbb{G}}_{\mathsf{exp}} \approx 0.924$ ms, $T_{\mathsf{p}} \approx 1.486$ ms, $T^{\mathbb{G}_T}_{\mathsf{mul}} \approx 0.002$ ms, $T^{\mathbb{G}_T}_{\mathsf{exp}} \approx 0.126$ ms; ** n is the number of attributes for each UE; $L_{\mathbb{G}}$ and $L_{\mathbb{G}_T}$ are the sizes of the elements in $\mathbb{G}$ and $\mathbb{G}_T$, respectively. $L_{\mathbb{G}} = 170$ bits and $L_{\mathbb{G}_T} = 340$ bits as we build the pairing mapping by MNT curves [26] for 80 bits security. For **KeyGen**, **PreEnc**, and **Dec**, we compared the additional costs in CoIBE and CoABE to the **KeyGen**, **Enc**, and **Dec** in the original encryption systems, i.e., BB-IBE and ALP-FE.

Proof. We can observe that the structure of the construction for cooperative encryption in CoABE, i.e., e'_1 and e'_2, is similar to that in CoIBE. Hence, we can readily prove the security of CoABE by the security proofs for the Type-1, Type-2, and Type-3 adversaries in CoABE since the design of the cooperative encryption can be easily adopted to the public key encryption systems built by pairing-based arithmetics. □

B. The performance

This section evaluates the performance of the proposed two schemes, CoIBE and CoABE, by comparing with the computation costs of the original BB-IBE and ALP-FE. Table I shows the additional computation costs of every algorithm in the proposed schemes, respectively. We consider **PreEnc** in CoIBE and CoABE is the same function as **Enc** in BB-IBE and ALP-FE and **CoEnc**, running on the SEM, is an additional algorithm, which is not required in the original encryption schemes. We evaluate the performance by running Pairing-based Cryptography Library (PBC) [27] on a laptop computer, which equips with Intel Core i5 1.8 GHz CPU and 4 Gigabytes 1600 MHz DDR3 RAM, as a testbed. Table I shows the computation and communication costs of two proposed schemes. We specially aim at the computation times of two **CoEnc** algorithms in CoIBE and CoABE since they run on the SEM and the reasonable computation times are required to process the numerous amount of requests from users. The computation times of them are 7.994 ms

and 7.07 ms, respectively. It is expected that the computation times of **CoEnc** can be extremely shorter on multi-core high performance computing systems (HPC). According to the number of searches per second on Google being approximately 40000 times, it is easy to support the controllable security services by CORE with distributed SEMs for the comparable amount of users. Regarding communication costs, the sizes of the partial ciphertexts in CoIBE and CoABE are both 1020 bits, and that of the complete ciphertexts in CoABE are also 1020 bits, where the bilinear mapping is constructed by MNT curves [26]. Note that, we do not count the size of policy into the size of the partial and complete ciphertexts in CoABE since its size depends on the number of attributes in a policy. Hence, the communication costs of the partial and complete ciphertext in CoIBE and CoABE are comparable to the size of RSA ciphertext, i.e., 1024 bits for 80-bit security.

VI. Conclusions

This work introduces a novel cooperative encryption framework, i.e., CORE, to control encryption and decryption capabilities, and demonstrates the feasibility by two constructions to IBE and ABE. We define the security of CoIBE and CoABE against three types of adversaries, i.e., outsiders, SEM, and receivers, and analyzes the security based on the definitions. The extension of distributed and revocable SEM for CoIBE and CoABE is also introduced in supporting large scale networks. The performance analysis demonstrates that the computation and communication costs are comparable to the standard encryption system, e.g., RSA encryption. This framework paves the way to control security capability, and will be useful for the applications, requiring controllable security, such secure e-mail, secure cloud storage, edge/fog computing, decentralized computing model, etc. Nevertheless, CORE is flexible and applicable to various encryption systems, e.g., IBE, ABE, broadcast encryption, proxy re-encryption, etc.

References

[1] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. of CRYPTO 1984*, vol. 196. LNCS, 1984, pp. 47–53.

[2] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Proc. of CRYPTO 2001*, vol. 2139. LNCS, 2001, pp. 213–229.

[3] X. Ding and G. Tsudik, "Simple identity-based cryptography with mediated RSA," in *Proc. of CT-RSA 2003*, vol. 2612. LNCS, 2003, pp. 193–210.

[4] D. Boneh, X. Ding, and G. Tsudik, "Fine-grained control of security capabilities," *ACM Transactions on Internet Technology*, vol. 4, no. 1, pp. 60–82, 2004.

[5] S. Chow, C. Boyd, N. Gonzalez, and M. Juan, "Security-mediated certificateless cryptography," in *Proc. of Public Key Cryptology - PKC 2006*, vol. 3958. LNCS, 2006, pp. 508–524.

[6] D. Boneh, X. Ding, G. Tsudik, and M. Wong, "A method for fast revocation of public key certificates and security capabilities," in *Proc. of the 10th USENIX Security Symposium*, 2001, pp. 297–308.

[7] B. Libert and J.-J. Quisquater, "Efficient revocation and threshold pairing based cryptosystems," in *Proc. of the 22nd Annual Symposium on Principles of Distributed Computing (PODC'03)*. ACM, 2003, pp. 163–171.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. of the 13th ACM Conference on Computer and Communication Security - CCS'06*. ACM, 2006, pp. 89–98.

[9] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013.

[10] J. Hur and K. Kang, "Secure data retrieval for decentalized disruption-tolerant military networks," *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 16–26, 2014.

[11] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y. L. Boudec, "Protecting location privacy: Optimal strategy against localization attacks," in *Proc. of the 2012 ACM Conference on Computer and Communication Security*, 2012, pp. 617–627.

[12] I. Polakis, G. Argyros, T. Petsios, S. Sivakorn, and A. Keromytis, "Where's Wally? precise user discovery attacks in location proximity services," in *Proc. of the 2015 ACM Conference on Computer and Communication Security*, 2015, pp. 817–828.

[13] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, "Mediated ciphertext-policy attribute-based encryption and its application," in *Proc. of 10th International Workshop, WISA 2009*. Springer, 2009, pp. 309–323.

[14] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proc. of the 2008 ACM Conference on Computer and Communication Security*. ACM, 2008, pp. 417–426.

[15] B. Libert and D. Vergnaud, "Adaptive-id secure revocable identity-based encryption," in *Proc. of The Cryptographers' Track at the RSA Conference 2009*. Springer, 2009, pp. 1–15.

[16] A.-F. Chan and I. Blake, "Scalable, server-passive, user-anonymous time release cryptography," in *Proc. of the 25th IEEE International Conference on Distributed Computing System (ICDCS'05)*. IEEE, 2005, pp. 504–513.

[17] J. Cathalo, B. Libert, and J.-J. Quisquater, "Efficient and non-interactive timed-release encryption," in *Proc. of the 7th International Conference of Information and Communications Security (ICICS'05)*, vol. 3783. LNCS, 2005, pp. 291–303.

[18] Y. Hwang, D. Yum, and P. Lee, "Timed-release encryption with pre-open capability and its application to certified e-mail system," in *Proc. of the 8th International Conference of Information Security*, vol. 3650. LNCS, 2005, pp. 344–358.

[19] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proc. of Advances in Cryptology - EUROCRYPT 2008*. LNCS, 2008, pp. 146–162.

[20] N. Attrapadung and B. Libert, "Functional encryption for inner product: Achieving constant-size ciphertext with adaptive security or support for negation," in *Proc. of Public Key Cryptography - PKC 2010*. LNCS, 2010, pp. 384–402.

[21] N. Attrapadung, B. Libert, and E. Panafieu, "Expressive key-policy attribute-based encryption with constant-size ciphertexts," in *Proc. of Public Key Cryptography - PKC 2011*. LNCS, 2011, pp. 90–108.

[22] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchy identity based encryption with constant size ciphertext," in *Proc. of Advances in Cryptology - EUROCRYPT 2005*. LNCS, 2005, pp. 440–456.

[23] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles," in *Proc. of EUROCRYPT 2004*, vol. 3027. LNCS, 2004, pp. 223–238.

[24] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. of the Advances in Cryptology - CRYPTO 2004*, vol. 3152. LNCS, 2004, pp. 41–55.

[25] N. Attrapadung, B. Libert, and E. Panafieu, "Expressive key-policy attribute-based encryption with constant-size ciphertexts," in *Proc. of the 14th International Conference on Practice and Theory in Public Key Cryptography (PKC 2011)*, vol. 6571. LNCS, 2011, pp. 90–108.

[26] A. Miyaji, M. Nakabayashi, and S. Takano, "Characterization of elliptic curve traces under fr-reduction," in *Proc. of Information Security and Cryptology (ICISC 2000)*, vol. 2015. LNCS, 2001, pp. 90–108.

[27] Ben Lynn, "Pairing-based Cryptography Library (PBC)," https://crypto.stanford.edu/pbc/.