Hindawi
Security and Communication Networks
Volume 2019, Article ID 6293970, 12 pages
https://doi.org/10.1155/2019/6293970

Research Article

An Approach Enabling Various Queries on Encrypted Industrial Data Stream

Tao Wang,[1,2,3] **Bo Yang**,[1,2] **Guoyong Qiu**,[1] **Lina Zhang**,[1,4] **Yong Yu**,[1] **Yanwei Zhou**,[1] **and Juncai Guo**[1]

[1] *School of Computer Science, Shaanxi Normal University, Xi'an 710119, China*
[2] *State Key Laboratory of Information Security (Institute of Information Engineering, Chinese Academy of Sciences), Beijing 100093, China*
[3] *Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an, Shaanxi 710016, China*
[4] *Department of Computing Science and Technology, Xi'an University of Science and Technology, Xi'an 710054, China*

Correspondence should be addressed to Bo Yang; byang@snnu.edu.cn

Received 14 March 2019; Accepted 11 June 2019; Published 3 July 2019

Guest Editor: Lein Harn

Massive data are generated and collected by devices in the industrial Internet of Things. Data sources would encrypt the data and send them to the data center through the gateway. For some supervision purpose, the gateway needs to observe the encrypted data stream and label the suspicious data. Instead of decrypting ciphertext at the gateway, which is not efficient, this paper presents a Φ-searchable functional encryption scheme that supports inner product evaluations on encrypted data. Based on this scheme, an approach enabling various queries on the encrypted industrial data stream is proposed. The adaptive security of our proposed underlying functional encryption scheme can be proven under general subgroup decision assumptions, and our scheme has the smaller public key, the smaller secret key, and the smaller ciphertext size compared to the related schemes. In addition, the experimental results show that our proposed scheme is efficient. Especially for the gateway, querying on the encrypted data only needs less than 20ms, which is practical for industrial data stream auditing scenario.

1. Introduction

1.1. Motivation. While manufacturers, mobile end systems, security cameras, wearable devices, and so forth have been generating highly distributed data from various systems, devices, and applications in industrial Internet of Things (iIoT), more and more data are gathered and intensively exploited by many organizations to extract valuable information either to make marketing decisions, track specific behaviours, or detect threat attacks. Big Data gives a huge opportunity to industries and decisions-makers, but it also represents a big risk for users. Due to data breaches, private information is leaked now and then [1, 2]. It is clear that safeguarding private data to protect manufacturers, sensitive customers, or patients is paramount [3, 4]. But, for massive manufacturers and health and financial organizations, actually implementing the best controls and security is challenging, especially when troves of data originate from multiple sources and are stored across singular or multiple databases and data warehouses.

Using encryption for sensitive information can effectively protect privacy [5]. But, paradoxically, encryption will destroy the usability of data. Especially for real-time industrial application's traffic, how to monitor or audit the encrypted data stream is a key problem. For example, as shown in Figure 1, a card payment gateway would observe the transaction stream, which often includes encrypted data between acquiring banks and issuing banks. The payment gateway needs to audit all encrypted data streams to label some suspicious transactions, say, whose value is over $10000. One solution is to encrypt all transactions under the card company's public key and give the private key to the payment gateway, which can decrypt the transactions stream to do auditing. This solution has two obvious drawbacks. One is being not efficient, because decryption needs to be done for every transaction passing by. Another drawback is that it is

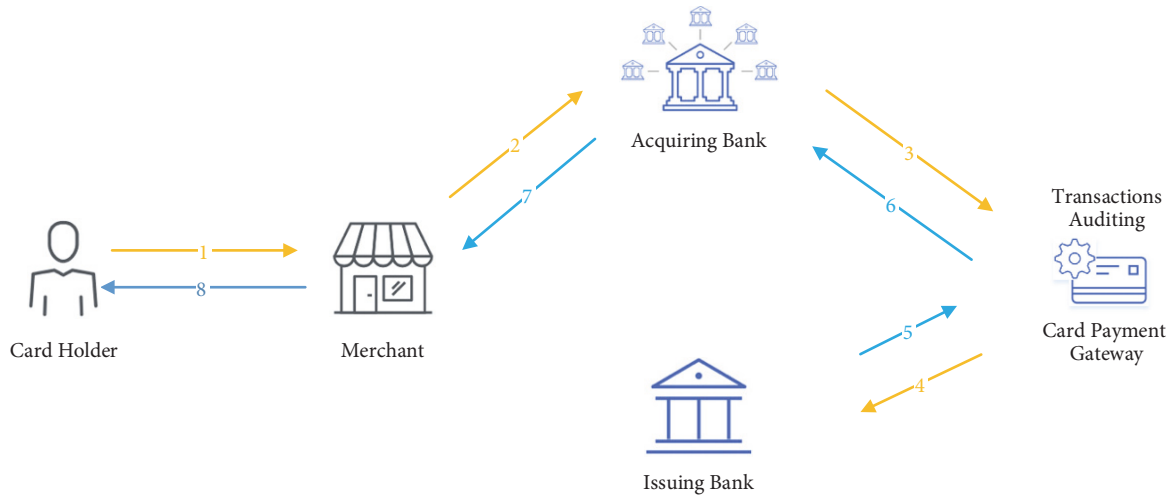

Figure 1: Card payment gateway observes transactions.

bad for both security and privacy concerns, because when the gateway holds the card company's private key, he can see everything he wants.

This work proposes a Φ-searchable functional encryption scheme that supports various types of queries on the encrypted data such as conjunctions, disjunctions, DNFs/CNFs, polynomial equations, and inner products. By utilizing this scheme, we show an approach that can enable the auditing gateways to query or to evaluate the encrypted data stream passing by.

1.2. Related Work. Querying on encrypted data is a long-term interesting open problem in applications with secure and privacy-preserved concerns. Encryption schemes supporting queries on ciphertext are called searchable encryption (SE) schemes, which have two different types of research roadmap. One is searchable symmetric encryption (SSE) [6–12]. The other is public key encryption with keyword search (PEKS) [13–16]. SSE is more efficient but less expressive, and it is hard to achieve privacy for search pattern and access pattern. PEKS is easy to support phrase search and even more complicated evaluations on encrypted data, such as conjunctive, subset, range [14–16], DNF/CNF, polynomial equation, inner product [15, 16], and negation [16]. However, PEKS often has much more computation overheads than SSE due to doing paring computations. There are more recent works for improving security [17, 18], improving functionality [19, 20], and improving performance [21] for the PEKS. Boneh et al. showed that PEKS implies IBE [13]. So, in some sense, an SE scheme is a special form of functional encryption. Brent Waters first publicly used the notion of functional encryption in his talk *Functional Encryption: Beyond Public Key Cryptography*, and Boneh et al. formally defined functional encryption [22]. There was a long-standing open feasibility problem in cryptography: *Does there exist a functional encryption scheme supporting all polynomial-size circuits?* Until 2013, Garg et al. [23] had shown a method to construct functional encryption schemes for polynomial-size circuits based on the indistinguishability obfuscation [24]. Functional encryption has most powerful expressive ability, which can express identity-based encryption (IBE) [25, 26], ABE [27, 28], predicate encryption [14, 15], and inner product encryption [15, 29]. In functional encryption systems, a user who has a decryption key can learn a function of ciphertext. Roughly speaking, in a functional encryption system for functionality $F(\cdot,\cdot)$, an authority holding a master secret key can generate a key $sk_{\mathbf{K}}$ in which the key attribute $\mathbf{K}$ is encoded and that enables the computation of the function $F(\mathbf{K},\cdot)$ on encrypted data which encoded the ciphertext attribute $\mathbf{A}$. More specifically, the user can compute $F(\mathbf{K}, M)$ using $sk_{\mathbf{K}}$ from encryption of plaintext M. We will show the formal definition and security notion of the functional encryption in Section 2.

1.3. Our Approaches and Results. Boneh, Sahai, and Waters have firstly presented formal syntax and put forth a general framework of functional encryption [22]. They also defined two subclasses of functional encryption which are predicate encryption and predicate encryption with public index. In predicate encryption subclass, a functional encryption scheme is defined in terms of a polynomial-time predicate $P : \Sigma_{\mathbf{K}} \times \Sigma_{\mathbf{A}} \longrightarrow \{0,1\}$, where $\Sigma_{\mathbf{K}}$ is key attributes space and $\Sigma_{\mathbf{A}}$ is ciphertext attributes space. Formally, the functional encryption is defined as

$$F\left(\mathbf{K} \in \Sigma_{\mathbf{K}}, (\mathbf{A}, M)\right) := \begin{cases} M & if\ P_{\mathbf{K}}(\mathbf{A}) = 1 \\ \perp & if\ P_{\mathbf{K}}(\mathbf{A}) = 0 \end{cases} \quad (1)$$

Consequently, if $P_{\mathbf{K}}(\mathbf{A}) = 1$, decryption algorithm can recover plaintext and, otherwise, get nothing about the plaintext.

Inspired by the Φ-searchable public key system proposed by Boneh and Waters [9], this paper describes a new functional encryption construction, that is, Φ-searchable functional encryption system providing security against adaptive adversaries and supporting conjunctive, subset,

range, DNF/CNF, polynomial equation, and inner product on encrypted data. We use the predicate encryption subclass to express our functional encryption scheme. We will show the formal definition and security notion of the Φ-searchable functional encryption system in Section 3.

For encoding the key attribute into the secret key and encoding the ciphertext attribute into ciphertext, we follow the inner product encryption (IPE) methodology [15, 29] to realize the predicate $P_{\mathbf{K}}(\mathbf{A})$, which means $P_{\mathbf{K}}(\mathbf{A}) = 1$ if the inner product of key attribute vector $\mathbf{K}$ and ciphertext attribute vector $\mathbf{A}$ is 0 and $P_{\mathbf{K}}(\mathbf{A}) = 0$ otherwise. Formally, for some $\mathbf{K} \in \Sigma_{\mathbf{K}}$ and $\mathbf{A} \in \Sigma_{\mathbf{A}}$, a predicate P over $\Sigma_{\mathbf{K}} \times \Sigma_{\mathbf{A}}$ is defined as

$$P_{\mathbf{K}}(\mathbf{A}) \coloneqq \begin{cases} 1 & if \ \langle \mathbf{K}, \mathbf{A} \rangle = 0 \\ 0 & Otherwise \end{cases} \tag{2}$$

Thanks to "inner-product" style construction, our scheme supports any kind of inner product queries on encrypted data. Clearly, our scheme supports the equality test directly. To achieve this, for the attribute $\mathbf{A}'$ set $\mathbf{A} \coloneqq (-\mathbf{A}', 1)$ and encrypt a message M with $\mathbf{A}$. In order to generate a secret key for the attribute $\mathbf{K}'$, set $\mathbf{K} \coloneqq (1, \mathbf{K}')$. Since $\langle \mathbf{K}, \mathbf{A} \rangle = 0$ if and only if $\mathbf{K}' = \mathbf{A}'$, correctness and security follow. Our scheme also supports the polynomial evaluation after we encode the coefficient of a univariate polynomial into secret keys and encode the univariate into ciphertexts. As a positive result, we can use the polynomial evaluation to achieve supporting conjunctions, disjunctions, CNF, and DNF formulas. We defer the details of applications of our scheme to Section 6.

Our construction relies on general subgroup decision assumptions in composite-order groups which are described in Section 2. We follow the standard Lewko-Waters [30] proof methodology to prove adaptive security of our construction. We propose a Φ-searchable functional encryption system which supports various evaluations on encrypted data including equality, comparison, subset tests, and polynomial evaluations as well as conjunctions, disjunctions, CNF, and DNF formulas. Moreover, compared to the prior constructions that are built for inner product evaluations on encrypted data in composite-order bilinear groups [15, 31], our scheme not only has the adaptive security but also has a smaller public key, smaller secret key, and smaller ciphertext size.

From our proposed searchable encryption scheme, we present an approach enabling various queries on encrypted industrial data for general data flow structure, which includes data sources, gateway, and data center. Our proposed approach makes the gateway easily observe the encrypted data stream passing by sent by the data sources to the data center without decryption. Moreover, if the encrypted data passing by is not matching with some condition, the gateway will learn nothing about the data. From performance evaluation results, the gateway's overhead is less than 20ms which is practical for application in the scenario of querying on the encrypted industrial data stream. We will show our proposed approach in Section 6.1.

2. Preliminaries

2.1. Notations. Given two vectors $\mathbf{U} = (u_1, u_2, ..., u_d) \in \mathbb{Z}_N^d$ and $\mathbf{V} = (v_1, v_2, ..., v_d) \in \mathbb{Z}_N^d$, we use the notation $\langle \mathbf{U}, \mathbf{V} \rangle$ to denote dot product $\mathbf{U}^T\mathbf{V}$. For a group element g, we use $g^{\mathbf{U}}$ to denote a vector $(g^{u_1}, g^{u_2}, ..., g^{u_d})$.

2.2. Syntax of Functional Encryption. We now describe the definition of functional encryption for a functionality $F : \Sigma_{\mathbf{A}} \times \Sigma_{\mathbf{K}} \longrightarrow \{0, 1\}$, where $\Sigma_{\mathbf{A}}$ denotes the ciphertext attributes space and $\Sigma_{\mathbf{K}}$ denotes the key attributes space [22].

Definition 1. For F, a functional encryption scheme consists of four PPT algorithms (Setup, Keygen, Enc, and Dec): for all $\mathbf{A} \in \Sigma_{\mathbf{A}}$ and $\mathbf{K} \in \Sigma_{\mathbf{K}}$, the algorithm Setup$(1^\lambda)$ generates public parameters pp and master secret key mk, the algorithm Keygen$(mk, \mathbf{K})$ outputs secret key for $\mathbf{K}$, the algorithm Enc$(pp, M, \mathbf{A})$ generates ciphertext for a message $M \in \mathscr{M}$, and Dec$(sk_{\mathbf{K}}, c)$ uses $sk_{\mathbf{K}}$ to compute $y = F(\mathbf{K}, M)$ from c.

2.3. Security Notion of Functional Encryption. Before defining the security of functional encryption, we need to describe a restriction for the adversary. Observe that, after the adversary gets the secret keys he wants, he will submit two distinct messages $M_0, M_1 \in \mathscr{M}$. The challenger randomly chooses one to encrypt and sends the ciphertext c to the adversary. Therefore, we need to restrict M_0 and M_1 chosen by the adversary and for all $\mathbf{K}$ that the adversary has $sk_{\mathbf{K}}$, we require that

$$F(\mathbf{K}, M_0) = F(\mathbf{K}, M_1) \tag{3}$$

Clearly, if this restriction is not satisfied, that is, if the adversary has $sk_{\mathbf{K}}$ for some $\mathbf{K}$, he can trivially break the semantic security of the scheme by testing whether Dec$(sk_{\mathbf{K}}, c) = F(\boldsymbol{K}, M_0)$ or not.

For a functional encryption scheme $\mathscr{E}$, $b \in \{0, 1\}$ and, for an adversary $\mathscr{A}$, define an experiment as follows:

(i) *Setup*. Run Setup(1^λ), get (pp, mk), and send pp to $\mathscr{A}$.

(ii) *Query phase1*. $\mathscr{A}$ adaptively makes queries by submitting $\mathbf{K}_i \in \Sigma_{\mathbf{K}}$, where $i = 1, 2, ...$, and receives $sk_{\mathbf{K}_i} \longleftarrow$ Keygen$(mk, \mathbf{K}_i)$.

(iii) *Challenge*. $\mathscr{A}$ outputs two messages $M_0, M_1 \in \mathscr{M}$ satisfying the above restriction and receives Enc(pp, M_b).

(iv) *Query phase2*. $\mathscr{A}$ continues to make queries for some $\mathbf{K}_i$ as query phase1 subject to the restriction and finally outputs a bit.

For b, define

$$\mathrm{Adv}_{\mathscr{A}}^{\mathscr{E}}(\lambda) \coloneqq \left| \Pr\left[b' = b\right] - \frac{1}{2} \right| \tag{4}$$

Definition 2. If, for all PPT $\mathscr{A}$, $\mathrm{Adv}_{\mathscr{A}}^{\mathscr{E}}(\lambda)$ is negligible, we say a functional encryption scheme $\mathscr{E}$ is secure.

2.4. Assumptions over Composite-Order Bilinear Groups

Bilinear Groups of Composite Order. Composite-order bilinear groups were first introduced by Boneh et al. [32] and used by many researchers [15, 33, 34]. Let $\mathcal{G}$ be a group generator that takes a security parameter 1^n as input and outputs $(p, q, r, \mathbb{G}, \mathbb{G}_T, \hat{e})$, where $\mathbb{G}$ and $\mathbb{G}_T$ are two cyclic groups of order $N = p \times q \times r$, where p, q, r are three distinct primes, and $\hat{e} : \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$ is a function satisfying the following properties:

(i) *Bilinear*: $\forall g_1, g_2 \in \mathbb{G}$, $x, y \in \mathbb{Z}_N$, and $\hat{e}(g_1^x, g_2^y) = \hat{e}(g_1, g_2)^{xy}$.

(ii) *Nondegenerate*: $\exists g \in \mathbb{G}$ such that $\hat{e}(g, g)$ has order N in $\mathbb{G}_T$.

(iii) *Cancellation*: let $\mathbb{G}_p$, $\mathbb{G}_q$, and $\mathbb{G}_r$ be subgroups of $\mathbb{G}$ with order p, q, and r, respectively. For some elements h_1 and h_2 from distinct subgroups, we have

$$\hat{e}(h_1, h_2) = 1 \tag{5}$$

To see this, we note that $\mathbb{G} = \mathbb{G}_p \times \mathbb{G}_q \times \mathbb{G}_r$ and also note that if $g \in \mathbb{G}$ is a generator of $\mathbb{G}$, then g^{pq} generates $\mathbb{G}_r$; g^{pr} generates $\mathbb{G}_q$; g^{qr} generates $\mathbb{G}_p$. Hence, for some elements h_1 and h_2 from distinct subgroups (e.g., $h_1 = h_p \in \mathbb{G}_p$ and $h_2 = h_q \in \mathbb{G}_q$), $h_p = (g^{qr})^{\beta}$ (for some β) and $h_q = (g^{pr})^{\gamma}$ (for some γ). So, we note that $\hat{e}(h_p, h_q) = \hat{e}((g^{qr})^{\beta}, (g^{pr})^{\gamma}) = \hat{e}(g^{\beta}, g^{r\gamma})^{pqr} = 1$.

Cryptographic Assumptions. Our construction relies on the general subgroup decision assumptions in composite-order groups [33]. We now give the following three assumptions.

Assumption 3. Let $\mathcal{G}$ be a group generator as above, and define the following distribution:

$$\begin{aligned}
&(p, q, r, \mathbb{G}, \mathbb{G}_T, \hat{e}) \longleftarrow_{\$} \mathcal{G}(1^n),\\
&N = p \times q \times r,\\
&g \longleftarrow_{\$} \mathbb{G}_p,\\
&X_3 \longleftarrow_{\$} \mathbb{G}_r,\\
&D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g, X_3),\\
&T_0 \longleftarrow_{\$} \mathbb{G}_{pq},\\
&T_1 \longleftarrow_{\$} \mathbb{G}_p,
\end{aligned} \tag{6}$$

where $\mathbb{G}_{pq}$ is the subgroup of $\mathbb{G}$ with order pq. Define $\mathcal{A}$'s advantage in breaking Assumption 3 as

$$\begin{aligned}
&Adv_{\mathcal{G},\mathcal{A}}^{SD1}(\lambda)\\
&\quad := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|
\end{aligned} \tag{7}$$

Definition 4. For any ppt algorithm $\mathcal{A}$, if $\mathcal{A}$'s advantage in breaking Assumption 3 and $\mathrm{Adv}_{\mathcal{G},\mathcal{A}}^{\mathrm{SD1}}(\lambda)$ is negligible, we say that $\mathcal{G}$ satisfies Assumption 3.

Assumption 5. Let $\mathcal{G}$ be a group generator as above, and define the following distribution:

$$\begin{aligned}
&(p, q, r, \mathbb{G}, \mathbb{G}_T, \hat{e}) \longleftarrow_{\$} \mathcal{G}(1^n),\\
&N = p \times q \times r,\\
&g, X_1 \longleftarrow_{\$} \mathbb{G}_p,\\
&X_2, Y_2 \longleftarrow_{\$} \mathbb{G}_q,\\
&X_3 Y_3 \longleftarrow_{\$} \mathbb{G}_r,\\
&D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g, X_1 X_2, X_3, Y_2 Y_3),\\
&T_0 \longleftarrow_{\$} \mathbb{G},\\
&T_1 \longleftarrow_{\$} \mathbb{G}_{pr},
\end{aligned} \tag{8}$$

where $\mathbb{G}_{qr}$ is the subgroup of $\mathbb{G}$ with order qr. Define $\mathcal{A}$'s advantage in breaking Assumption 5 as

$$\begin{aligned}
&Adv_{\mathcal{G},\mathcal{A}}^{SD2}(\lambda)\\
&\quad := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|
\end{aligned} \tag{9}$$

Definition 6. For any ppt algorithm $\mathcal{A}$, if $\mathcal{A}$'s advantage in breaking Assumption 5 and $\mathrm{Adv}_{\mathcal{G},\mathcal{A}}^{\mathrm{SD2}}(\lambda)$ is negligible, we say that $\mathcal{G}$ satisfies Assumption 5.

Assumption 7. Let $\mathcal{G}$ be a group generator as above, and define the following distribution:

$$\begin{aligned}
&(p, q, r, \mathbb{G}, \mathbb{G}_T, \hat{e}) \longleftarrow_{\$} \mathcal{G}(1^n),\\
&N = p \times q \times r,\\
&\alpha, s \longleftarrow_{\$} \mathbb{Z}_N\\
&g \longleftarrow_{\$} \mathbb{G}_p,\\
&X_2, Y_2, Z_2 \longleftarrow_{\$} \mathbb{G}_q,\\
&X_3 \longleftarrow_{\$} \mathbb{G}_r,\\
&D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g, g^{\alpha} X_2, X_3, g^{s} Y_2, Z_2),\\
&T_0 = \hat{e}(g, g)^{\alpha s},\\
&T_1 \longleftarrow_{\$} \mathbb{G}_T.
\end{aligned} \tag{10}$$

Define $\mathcal{A}$'s advantage in breaking Assumption 7 as

$$\begin{aligned}
&Adv_{\mathcal{G},\mathcal{A}}^{SD3}(\lambda)\\
&\quad := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|
\end{aligned} \tag{11}$$

Definition 8. For any ppt algorithm $\mathcal{A}$, if $\mathcal{A}$'s advantage in breaking Assumption 7 and $\mathrm{Adv}_{\mathcal{G},\mathcal{A}}^{\mathrm{SD3}}(\lambda)$ is negligible, we say that $\mathcal{G}$ satisfies Assumption 7.

2.5. Dual System Encryption.

Brent Waters firstly introduced a methodology to build adaptively secure IBE and HIBE

which is dual system encryption [35], and a lot of work relied on this powerful proof tool [33, 36–38]. In dual system encryption schemes, there are two forms of ciphertext and key: normal and semifunctional. The semifunctional ciphertext and key are only used in the hybrid security proof, while normal ciphertext and key are used in the real system. A normal ciphertext can be decrypted correctly by either normal key or semifunctional key. But semifunctional key cannot decrypt a semifunctional ciphertext, whereas only normal key can. The hybrid security games advance one by one, and the first one is real security game, while in the last one the ciphertext is replaced by encryption of a random message. The most important part of the proof is to show two consecutive games are indistinguishable.

3. Definition

We first show a definition of a Φ-searchable functional encryption system inspired by the Φ-searchable public key system proposed by Boneh and Waters [14]. Then, we show the definition of the security notion.

3.1. Φ-Searchable Functional Encryption System. We use Σ to denote a finite set of binary strings and let Φ be a set of predicates over attributes space Σ. A predicate $P \in \Phi$ is a map: $P : \Sigma \longrightarrow \{0, 1\}$. For two attribute vectors $\mathbf{A}, \mathbf{K} \in \Sigma$, we use the notion $P_{\mathbf{K}}(\mathbf{A}) = 1$ to denote that $\mathbf{A}$ satisfies P which is related to $\mathbf{K}$. We also follow Boneh et al. [14] to use the term *GenToken* to denote the algorithm to generate a search or query token instead of the term *GenKey*, and we use the term *Query* to denote the algorithm to query rather than the term *Decrypt*.

Definition 9. For a predicate $P \in \Phi$, a Φ-searchable functional encryption system comprises four algorithms, $Setup(1^\lambda)$, $Encrypt(PK, (\mathbf{A}, M))$, $GenToken(SK, P_{\mathbf{K}})$, and $Query(TK_{\mathbf{K}}, \mathrm{C})$ such that

(i) $Setup(1^\lambda)$: a probabilistic algorithm that takes as input a security parameter λ and outputs the public parameters PP along with the public key PK and the master secret key SK.

(ii) $Encrypt(PK, (\mathbf{A}, M))$: a probabilistic algorithm that takes as input the public key PK and a plaintext pair $(\mathbf{A}, M)$. We consider $\mathbf{A}$ as the searchable attribute vector of the data M. The algorithm outputs a searchable encryption of $(\mathbf{A}, M)$ under the public key PK.

(iii) $GenToken(SK, P_{\mathbf{K}})$: a probabilistic algorithm that takes the secret key SK and a description of a predicate $P_{\mathbf{K}}$ as input and outputs a search token $TK_{\mathbf{K}}$.

(iv) $Query(TK_{\mathbf{K}}, \mathrm{C})$: a deterministic algorithm that takes a token $TK_{\mathbf{K}}$ and a ciphertext C as input and outputs $F(\mathbf{K}, M)$.

For correctness, we require that, for all λ and all $(PP, PK, SK) \longleftarrow \mathrm{Setup}(1^\lambda)$, all $P_{\mathbf{K}} \in \Phi$, any token $TK_{\mathbf{K}} \longleftarrow GenToken(SK, P_{\mathbf{K}})$, and all $\mathbf{A} \in \Sigma$:

(i) If $P_{\mathbf{K}}(\mathbf{A}) = 1$, then $F(\mathbf{K}, M) = M$.

(ii) If $P_{\mathbf{K}}(\mathbf{A}) = 0$, then $F(\mathbf{K}, M) = \perp$ with all but negligible probability.

3.2. Security Notion. We now show a security notion definition of a Φ-searchable functional encryption system.

Definition 10. A Φ-searchable functional encryption system $\mathscr{E}$ defined as above is adaptive secure if, for all PPT adversaries $\mathscr{A}$, the advantage of $\mathscr{A}$ in the following game is negligible in the security parameters λ:

(i) *Setup*. The challenger runs $Setup(1^\lambda)$ and gives the adversary $\mathscr{A}$ the PP and PK.

(ii) *Query phase 1.* $\mathscr{A}$ outputs descriptions of predicates $P_{\mathbf{K}_1}, P_{\mathbf{K}_2}, \ldots, P_{\mathbf{K}_{\ell_1}} \in \Phi$. The challenger responds with the corresponding tokens:

$$TK_{\mathbf{K}_j} \longleftarrow GenToken\left(SK, P_{\mathbf{K}_j}\right). \tag{12}$$

(iii) *Challenge*. $\mathscr{A}$ outputs two pairs of messages $((\mathbf{A}_0, M_0), (\mathbf{A}_1, M_1)) \in \Sigma \times \mathscr{M}$ subject to the following restrictions:

$$P_{\mathbf{K}_j}(\mathbf{A}_0) = P_{\mathbf{K}_j}(\mathbf{A}_1) \tag{13}$$

for all $P_{\mathbf{K}_j}$ in predicates list queried at query phase 1 and if $M_0 \neq M_1$ then

$$P_{\mathbf{K}_j}(\mathbf{A}_0) = P_{\mathbf{K}_j}(\mathbf{A}_1) = 0 \tag{14}$$

These two restrictions ensure that the tokens given to the adversary do not trivially break the challenge. The first restriction ensures that tokens given to the adversary do not directly distinguish $\mathbf{A}_0$ from $\mathbf{A}_1$. The second restriction ensures that the tokens do not directly distinguish M_0 from M_1.

The challenger randomly chooses $b \in \{0, 1\}$ and gives $C \longleftarrow Encrypt(PK, (\mathbf{A}_b, M_b))$ to $\mathscr{A}$.

(iv) *Query phase 2.* $\mathscr{A}$ continues to output adaptively descriptions of predicates $P_{\mathbf{K}_{\ell_1+1}}, \ldots, P_{\mathbf{K}_\ell} \in \Phi$, subject to the two restrictions (13) and (14). The challenger responds with the corresponding tokens $TK_{\mathbf{K}_j} \longleftarrow GenToken(SK, P_{\mathbf{K}_j})$.

(v) *Guess*. $\mathscr{A}$ outputs a bit b' and wins if $b' = b$.

The adversary $\mathscr{A}$'s advantage in breaking $\mathscr{E}$ is $\mathrm{Adv}_{\mathscr{A}} = |\Pr[b' = b] - 1/2|$.

4. Our Construction

Our construction is based on general subgroup decision assumptions in composite-order groups, that is, Assumptions 3, 5, and 7. Let Φ be a set of predicates over Σ (in our construction, $\Sigma := \mathbb{Z}_N^d$); for a predicate $P \in \Phi$, a Φ-searchable functional encryption scheme for P is defined as follows:

(i) *Setup*(1^λ): this algorithm takes the security parameter λ as input. First, it runs $\mathcal{G}(1^n)$ and gets $(p, q, r, \mathbb{G}, \mathbb{G}_T, \hat{e})$, where $\mathbb{G} = \mathbb{G}_p \times \mathbb{G}_q \times \mathbb{G}_r (N = p \times q \times r)$. Then, it computes g_p, g_q, g_r as a generator of $\mathbb{G}_p, \mathbb{G}_q, \mathbb{G}_r$, respectively. In addition, it chooses random α, $x \in_R \mathbb{Z}_N^*$, and $\mathbf{X} \in_R \mathbb{Z}_N^d$. Finally, it outputs public parameters $PP := (\mathbb{G}, g_p, g_r)$ along with the public key:

$$PK := \left(g_p^x, g_p^{\mathbf{X}}, \hat{e}\left(g_p, g_p\right)^\alpha\right) \tag{15}$$

It keeps $SK := (g_p^\alpha, x, \mathbf{X})$ private as the master secret key.

(ii) *Encrypt*$(PK, (\mathbf{A}, M))$: let $\mathbf{A} = (a_1, ..., a_d) \in \Sigma$; this algorithm takes the public key PK and a pair $(\mathbf{A}, M)$ as input and chooses random exponent $s \in_R \mathbb{Z}_N^*$; then it outputs C as the ciphertext, where

$$C := \left\{C_0 := \hat{e}\left(g_p, g_p\right)^{\alpha s} \bullet M,\ C_1 := g_p^{s(x\mathbf{A}+\mathbf{X},1)}\right\} \tag{16}$$

(iii) *GenToken*$(SK, P_{\mathbf{K}})$: let $\mathbf{K} = (k_1, ..., k_d) \in \Sigma$; this algorithm takes as input master secret key SK and a predicate $P_{\mathbf{K}}$, in our case that is $\mathbf{K}$ itself, and chooses random $y \in_R \mathbb{Z}_N^*$ and $\mathbf{W} \in_R \mathbb{Z}_N^{d+1}$. Finally, it outputs a token:

$$TK_{\mathbf{K}} := g_p^{(y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle)} g_r^{\mathbf{W}} \tag{17}$$

(iv) *Query*$(TK_{\mathbf{K}}, \mathrm{C})$: this algorithm takes a token $TK_{\mathbf{K}}$ for a predicate $P_{\mathbf{K}}$ and a ciphertext C as input; it outputs

$$F(\mathbf{K}, M) := \begin{cases} M := \dfrac{C_0}{\hat{e}\left(C_1, TK_{\mathbf{K}}\right)} & P_{\mathbf{K}}(\mathbf{A}) = 1 \\ \perp & P_{\mathbf{K}}(\mathbf{A}) = 0 \end{cases} \tag{18}$$

where

$$P_{\mathbf{K}}(\mathbf{A}) := \begin{cases} 1 & if\ \ \langle \mathbf{K}, \mathbf{A}\rangle = 0 \\ 0 & Otherwise \end{cases} \tag{19}$$

Correctness. Let C and $TK_{\mathbf{K}}$ be as above. Then

$$\begin{aligned} M &= \frac{C_0}{\hat{e}\left(C_1, TK_{\mathbf{K}}\right)} = \frac{\hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M}{\hat{e}\left(C_1, TK_{\mathbf{K}}\right)} \\ &= \frac{\hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M}{\hat{e}\left(g_p{}^{s(x\mathbf{A}+\mathbf{X},1)}, g_p{}^{(y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle)} g_r{}^{\mathbf{W}}\right)} \\ &= \frac{\hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M}{\hat{e}\left(g_p{}^{s(x\mathbf{A}+\mathbf{X},1)}, g_p{}^{(y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle)}\right) \cdot \hat{e}\left(g_p{}^{s(x\mathbf{A}+\mathbf{X},1)}, g_r{}^{\mathbf{W}}\right)} \\ &= \frac{\hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M}{\hat{e}\left(g_p{}^{s(x\mathbf{A}+\mathbf{X},1)}, g_p{}^{(y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle)}\right)} \\ &= \frac{\hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M}{\hat{e}\left(g_p^s, g_p\right)^{(x\mathbf{A}+\mathbf{X},1)^T \bullet (y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle)}} \end{aligned} \tag{20}$$

where

$$\begin{aligned} &(x\mathbf{A} + \mathbf{X}, 1)^T \bullet \left(y\mathbf{K}, \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle\right) \\ &= (x\mathbf{A} + \mathbf{X}, 1)^T \bullet y\mathbf{K} + \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle \\ &= xy\langle \mathbf{X}, \mathbf{K}\rangle + y\langle \mathbf{X}, \mathbf{K}\rangle + \alpha - y\langle \mathbf{X}, \mathbf{K}\rangle \\ &= xy\langle \mathbf{A}, \mathbf{K}\rangle + \alpha \end{aligned} \tag{21}$$

For all $(\mathbf{A}, \mathbf{K}) \in \mathcal{A} \times \mathcal{K}$ such that $P_{\mathbf{K}}(\mathbf{A}) = 1$, which means $\langle \mathbf{K}, \mathbf{A}\rangle = 0 \bmod N$, the data M will be recovered correctly.

5. Security Proof

To prove the security of our Φ-searchable functional encryption scheme, it depends on the above-mentioned assumptions. Before that, we need to define the semifunctional ciphertext and semifunctional token. These additional structures will only be used in our proof, not in the real system.

(i) *Semifunctional ciphertext*: first, we choose randomly $\mathbf{Z}_c \in_R \mathbb{Z}_N^{d+1}$; then we can use the algorithm *Encrypt*$(PK, (\mathbf{A}, M))$ to construct the normal ciphertext as follows:

$$C' := \left\{C_0' := \hat{e}\left(g_p, g_p\right)^{\alpha s} \cdot M,\ C_1' := g_p^{s(x\mathbf{A}+\mathbf{X},1)}\right\}, \tag{22}$$

and we let the semifunctional ciphertext be

$$\widehat{C} := \left\{\widehat{C_0} := C_0',\ \widehat{C_1} := C_1' \cdot g_q^{\mathbf{Z}_c}\right\} \tag{23}$$

(ii) *Semifunctional token*: we also use the algorithm *GenToken*$(SK, P_{\mathbf{K}})$ to generate the normal token $TK_{\mathbf{K}}'$ and choose a random exponent $\mathbf{Z}_k \in_R \mathbb{Z}_N^{d+1}$. Then we can construct the semifunctional token as follows:

$$\widehat{TK_{\mathbf{K}}} = TK_{\mathbf{K}}' \cdot g_q^{\mathbf{Z}_k} \tag{24}$$

Remark. About query (decryption) capabilities, we observe that a normal ciphertext can be decrypted correctly by either normal key or semifunctional key. But semifunctional key cannot decrypt a semifunctional ciphertext, whereas only normal key can because there is an additional blinding factor of $\hat{e}(g_q, g_q)^{\langle \mathbf{Z}_c, \mathbf{Z}_k\rangle}$. But if $\mathbf{Z}_c$ and $\mathbf{Z}_k$ are orthotropic, the query will still work. We use the term nominally semifunctional token following the definition used by Katz et al. [10] which has components of the subgroup $\mathbb{G}_q$.

Now following the dual system encryption methodology presented by Lewko et al. [30], the proof proceeds with a game sequence starting from Game_{Real}, which is the real security game, followed by a restricted game $\mathrm{Game}_{Restricted}$ which is the same as Game_{Real} except that the adversary cannot query for the tokens for attributes that are equal to the challenge attributes $\mathbf{A}_0, \mathbf{A}_1$. Let ℓ be the number of times of token generation queries that adversary makes. Then we define the following Game_k $(0 \le k \le \ell)$ as follows:

(i) Game_0 is the real game except that the challenge ciphertext is semifunctional.

(ii) Game_k ($0 \le k \le \ell$) is the same as Game_0 except that the first i token generation queries are answered by a semifunctional token, and the last $\ell - i$ token generation queries are answered by a normal token.

Following Game_k, last game is Game_{Final}, which is identical to Game_ℓ, but the challenge ciphertext is not for one of the two messages submitted by the adversary but semifunctional encryption of a random message instead. In the following lemmas, we will prove the indistinguishability between two consecutive games and prove that the adversary $\mathcal{A}$'s view in Game_{Final} is statistically independent of challenge bit b'.

Lemma 11. *If there is an algorithm $\mathcal{A}$ that can distinguish $\text{Game}_{Restricted}$ from Game_{Real} with advantage of $Adv^{\mathcal{A}}_{\text{Game}_{Restricted}} - Adv^{\mathcal{A}}_{\text{Game}_{Real}} = \epsilon$, then we can build an algorithm $\mathfrak{B}$ to break Assumption 3 or Assumption 5 with advantage $\epsilon/2$.*

Proof. If there exists an adversary $\mathcal{A}$ whose advantage is a nonnegligible ϵ, we can find a nontrivial factor of N with nonnegligible probability and break Assumption 5. The proof methodology is similar to the proof of Lemma 1 in [33].

$\mathfrak{B}$ sets up the environment for $\mathcal{A}$ according to Game_{Real}. Suppose that $\mathcal{A}$ produces a ciphertext attribute $\mathbf{A}$ such that it is not equal to the challenge attributes $\mathbf{A}_0^*$ and $\mathbf{A}_1^*$. Since $\mathbf{A} \neq \mathbf{A}_0^*$, there exists at least one pair of components a_i and a_i^* such that $a_i \neq a_i^* \bmod N$, and $a_i - a_i^*$ can be divided by q, where a_i is a component of $\mathbf{A}$ and a_i^* is a component of $\mathbf{A}_0^*$. $\mathfrak{B}$ can compute $d = \gcd(a_i - a_i^*, N)$; set $d' = N/d$. Note that q divides d and $N = dd' = p \cdot q \cdot r$. With probability $\epsilon/2$, one of these two cases must occur; that is, p divides d' or $d = p \cdot q$ and $d' = r$. In the case of p dividing d', $g^{d'}$ is the identity. Then, given g, $\mathfrak{B}$ can test whether $T^{d'}$ is the identity. If not, $T \in \mathbb{G}_{pq}$ holds. Otherwise, $T \in \mathbb{G}_p$. Then $\mathfrak{B}$ breaks Assumption 3. In case of $d = p \cdot q$ and $d' = r$, given g, X_1X_2, X_3, Y_2, Y_3, $\mathfrak{B}$ can verify that $(X_1X_2)^d$ is the identity and determine that $d = p \cdot q$. Then $\mathfrak{B}$ can test whether $\hat{e}((Y_2Y_3)^{d'}, T)$ is the identity. If not, then $T \in \mathbb{G}$ holds. Otherwise, $T \in \mathbb{G}_{pr}$. Then $\mathfrak{B}$ breaks Assumption 5. □

Lemma 12. *If there is an algorithm $\mathcal{A}$ that can distinguish Game_0 from $\text{Game}_{Restricted}$ with advantage of $Adv^{\mathcal{A}}_{\text{Game}_0} - Adv^{\mathcal{A}}_{\text{Game}_{Restricted}} = \epsilon$, then we can build an algorithm $\mathfrak{B}$ to break Assumption 3 with advantage $\epsilon/2$.*

Proof. On input $D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_p, X_3)$ and $T \in \{T_0, T_1\}$, where $T_0 \longleftarrow_{\$} \mathbb{G}_p$ and $T_1 \longleftarrow_{\$} \mathbb{G}_{pq}$, $\mathfrak{B}$ simulates $\mathcal{A}$ as follows.

Setup. Choose random $\alpha, x \in_R \mathbb{Z}_N$, and $\mathbf{X} \in_R \mathbb{Z}_N^d$; set $SK := (g^\alpha, x, \mathbf{X})$ and output

$$PP := \left(\mathbb{G}, g_p, g_r\right),$$
$$PK := \left(g_p^x, g_p^{\mathbf{X}}, \hat{e}\left(g_p, g_p\right)^\alpha\right) \tag{25}$$

Token Queries. Each time $\mathfrak{B}$ is asked to provide a token for a predicate $P_{\mathbf{K}_j}$, it chooses random $y_j' \in_R \mathbb{Z}_N^*$ and $\mathbf{W}_j' \in_R \mathbb{Z}_N^{d+1}$ and outputs a token:

$$TK_{\mathbf{K}_j} := g_p^{(y_j'\mathbf{K}_j, \alpha - y_j'\langle \mathbf{X}, \mathbf{K}_j\rangle)} X_3^{\mathbf{W}_j'} \tag{26}$$

Challenge Ciphertext. After receiving two pair of messages $(\mathbf{A}_0, M_0), (\mathbf{A}_1, M_1)$, $\mathfrak{B}$ chooses random $\beta \in_R \{0, 1\}$ and then forms the ciphertext:

$$C_0 := \hat{e}\left(g_p, T\right)^\alpha \cdot M_\beta,$$
$$C_1 := T^{(x\mathbf{A}_\beta + \mathbf{X}, 1)} \tag{27}$$

It sets the $\mathbb{G}_p$ part of T equal g_p^s to implicitly. The correctness of decryption follows clearly. Observe that if $T = T_0 \longleftarrow_{\$} \mathbb{G}_p$, this is a normal ciphertext and we are in $\text{Game}_{Restricted}$. If $T = T_1 \longleftarrow_{\$} \mathbb{G}_{pq}$, this is a semifunctional ciphertext; then we are in Game_0. □

Lemma 13. *If there is an algorithm $\mathcal{A}$ that can distinguish Game_k from Game_{k-1} with advantage of $Adv^{\mathcal{A}}_{\text{Game}_k} - Adv^{\mathcal{A}}_{\text{Game}_{k-1}} = \epsilon$, then we can build an algorithm $\mathfrak{B}$ to break Assumption 5 with advantage $\epsilon/2$.*

Proof. On input $D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_p, X_1X_2, X_3, Y_2Y_3)$ and $T \in \{T_0, T_1\}$, where $T_0 \longleftarrow_{\$} \mathbb{G}$ and $T_1 \longleftarrow_{\$} \mathbb{G}_{pr}$, $\mathfrak{B}$ simulates $\mathcal{A}$ as follows.

Setup. Choose random $\alpha, x \in_R \mathbb{Z}_N$, and $\mathbf{X} \in_R \mathbb{Z}_N^d$; set $SK := (g_p^\alpha, x, \mathbf{X})$ and output

$$PP := \left(\mathbb{G}, g_p, g_r\right),$$
$$PK := \left(g_p^x, g_p^X, \hat{e}\left(g_p, g_p\right)^\alpha\right) \tag{28}$$

Token Queries. When $\mathcal{A}$ requests the ith token for the predicate $P_{\mathbf{K}_i}$, $\mathfrak{B}$ answers the tokens differently according to the following cases:

(i) Case $i < k$: $\mathfrak{B}$ chooses $y_i' \in_R \mathbb{Z}_N^*$ and $\mathbf{W}_i' \in_R \mathbb{Z}_N^{d+1}$ randomly and creates the semifunctional tokens:

$$TK_{\mathbf{K}_i} := g_p^{(y_i'\mathbf{K}_i, \alpha - y_i'\langle \mathbf{X}, \mathbf{K}_i\rangle)} \left(Y_2Y_3\right)^{\mathbf{W}_i'} \tag{29}$$

Observe that this is an identical distribution from semifunctional tokens.

(ii) Case $i > k$: $\mathfrak{B}$ chooses $y_i' \in_R \mathbb{Z}_N^*$ and $\mathbf{W}_i' \in_R \mathbb{Z}_N^{d+1}$ randomly and creates the normal tokens:

$$TK_{\mathbf{K}_i} := g_p^{(y_i'\mathbf{K}_i, \alpha - y_i'\langle \mathbf{X}, \mathbf{K}_i\rangle)} \left(X_3\right)^{\mathbf{W}_i'} \tag{30}$$

(iii) Case $i = k$: $\mathfrak{B}$ chooses $y_k' \in_R \mathbb{Z}_N^*$, $\mathbf{Z}_k \in_R \mathbb{Z}_N^{d+1}$, and $\mathbf{W}_k' \in_R \mathbb{Z}_N^{d+1}$ randomly and creates the normal tokens:

$$TK_{\mathbf{K}_k} := g_p^{(y_k'\mathbf{K}_k, \alpha - y_k'\langle \mathbf{X}, \mathbf{K}_k\rangle)} T^{\mathbf{Z}_k} \left(X_3\right)^{\mathbf{W}_k'} \tag{31}$$

Challenge Ciphertext. After receiving two pair of messages $(\mathbf{A}_0, M_0), (\mathbf{A}_1, M_1)$, $\mathfrak{B}$ chooses $\beta \in_R \{0,1\}$ randomly and sets $\mathbf{Z}_c \in \mathbb{Z}_N^{d+1}$ such that $\langle \mathbf{Z}_c, \mathbf{Z}_k \rangle = 0$ and then forms the ciphertext:

$$C_0 \coloneqq \hat{e}\left(g_p, X_1X_2\right)^{\alpha} \cdot M_\beta,$$
$$C_1 \coloneqq \left(X_1X_2\right)^{(x\mathbf{A}_\beta + \mathbf{X}, 1)} \tag{32}$$

Recall that $X_1X_2 \in \mathbb{G}_{pq}$ and X_1 is the $\mathbb{G}_p$ part in it. This implicitly sets $X_1 \coloneqq g_p^s$. Furthermore, X_2 is the $\mathbb{G}_q$ part in X_1X_2, so there exits some δ such that $X_2 \coloneqq g_q^\delta$ and $\langle \delta(x\mathbf{A}_\beta + \mathbf{X}, 1), \mathbf{Z}_k \rangle = 0$. Then this implicitly sets $\mathbf{Z}_c \coloneqq \delta(x\mathbf{A}_\beta + \mathbf{X}, 1)$. This relationship between $\mathbf{Z}_c$ and $\mathbf{Z}_k$ makes one thing happen; that is, if $\mathfrak{B}$ wants to test whether the token $TK_{\mathbf{K}}$ is semifunctional by creating a semifunctional ciphertext for predicate $P_{\mathbf{K}}$ and trying to decrypt and finish the query, the decryption will succeed no matter what $TK_{\mathbf{K}}$ is due to $\langle \mathbf{Z}_c, \mathbf{Z}_k \rangle = 0$. So, if $T \in \mathbb{G}_{pr}$, then we are in Game_{k-1}. If $T \in \mathbb{G}$, then we are in Game_k. □

Lemma 14. *If there is an algorithm $\mathscr{A}$ that can distinguish Game_{Final} from Game_ℓ with advantage of $Adv^{\mathscr{A}}_{\text{Game}_{Final}} - Adv^{\mathscr{A}}_{\text{Game}_\ell} = \epsilon$, then we can build an algorithm $\mathfrak{B}$ to break Assumption 7 with advantage $\epsilon/2$.*

Proof. On input $\alpha, s \longleftarrow_\$ \mathbb{Z}_N$, $D = ((N, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_p, g_p^\alpha X_2, X_3, g_p^s Y_2, Z_2)$ and $T \in \{T_0, T_1\}$, where $T_0 = \tilde{e}(g_p, g_p)^{\alpha s}$ and $T_1 \longleftarrow_\$ \mathbb{G}_T$; $\mathfrak{B}$ simulates $\mathscr{A}$ as follows.

Setup. Choose random $\alpha, x \in_R Z_N$ and $\mathbf{X} \in_R Z_N^d$; set $SK \coloneqq (g_p^\alpha, x, \mathbf{X})$ and output

$$PP \coloneqq \left(\mathbb{G}, g_p, g_r\right),$$
$$PK \coloneqq \left(g_p^x, g_p^{\mathbf{X}}, \hat{e}\left(g_p, g_p\right)^\alpha = \hat{e}\left(g_p^\alpha X_2, g_p\right)\right) \tag{33}$$

Token Queries. Each time $\mathfrak{B}$ is asked to provide a token for a predicate $P_{\mathbf{K}_j}$, it randomly chooses $y_j' \in_R \mathbb{Z}_N$, $\mathbf{W}_j' \in_R \mathbb{Z}_N^{d+1}$, and $\mathbf{Z}_k' \in_R \mathbb{Z}_N^{d+1}$ and outputs a semifunctional token:

$$TK_{\mathbf{K}_j} \coloneqq g_p^{(y_j'\mathbf{K}_j, \alpha - y_j'\langle \mathbf{X}, \mathbf{K}_j \rangle)} Z_2^{Z_k'} X_3^{\mathbf{W}_j'} \tag{34}$$

Challenge Ciphertext. After receiving two pairs of messages $(\mathbf{A}_0, M_0), (\mathbf{A}_1, M_1)$, $\mathfrak{B}$ randomly chooses $\beta \in_R \{0,1\}$ and forms the ciphertext:

$$C_0 \coloneqq T \cdot M_\beta,$$
$$C_1 \coloneqq \left(g_p^s Y_2\right)^{(x\mathbf{A}_\beta + \mathbf{X}, 1)} \tag{35}$$

This implicitly sets $\mathbf{Z}_c \coloneqq (x\mathbf{A}_\beta + \mathbf{X}, 1)$ and this has a proper distribution of semifunctional ciphertexts. So, if $T = \hat{e}(g_p, g_p)^{\alpha s}$, then we are in Game_ℓ. If $T = T_1 \longleftarrow_\$ \mathbb{G}_T$, then we are in Game_{Final}. □

Theorem 15. *Under Assumptions 3, 5, and 7, our Φ-searchable functional encryption scheme described in Section 4 is adaptively secure.*

Proof. When Assumptions 3, 5, and 7 hold, we have shown the indistinguishability between two consecutive games by the previous lemmas, which means that the real security game is indistinguishable from the simulated game in Game_{Final}, in which β is theoretically hidden from the adversary. So, the adversary has no advantage in breaking our scheme. □

6. Applications for Various Queries on Encrypted Data

In this section, we show a candidate system structure that enables the auditor to do various queries on the encrypted industrial data stream and discuss how to implement these query types.

6.1. General Structure for Querying on the Encrypted Industrial Data Stream. Based on our proposed Φ-searchable functional encryption scheme, one can easily enable the gateway to query encrypted industrial data stream. Specifically, as shown in Figure 2, data are collected from various sources such as manufactures, security cameras, and GPS chips. Data sources would send data to the data center through the gateway. The data center stores and analyzes these data, and the gateway observes and audits data stream for supervision purpose. For both security and privacy-preserving concerns, data sources encrypt the data stream under the data center's public key PK with the ciphertext attribute $\mathbf{A}$ by invoking the *Encrypt* algorithm. For a predicate P, the gateway sends the key attribute $\mathbf{K}$ to the data center, and the data center invokes the *GenToken* algorithm to delegate a query token $TK_{\mathbf{K}}$ to the gateway instead of sending its secret key SK to the gateway, which is a bad idea. The gateway who has the query token $TK_{\mathbf{K}}$ can make tests on the encrypted data stream by invoking the *Query* algorithm without decryption. If the output is 1, which means $P_{\mathbf{K}}(\mathbf{A}) = 1$, that is, the encrypted data passing by is matched with the conditions, then the gateway can decrypt that data correctly and take further actions, say, label it. If the output is 0, the gateway can learn nothing about that data; this is guaranteed by the security level of our proposed scheme.

6.2. Equality, Comparison, and Subset Queries. Let $\Sigma_{\mathbf{A}} \coloneqq \mathbb{Z}_N^d$, $\mathbf{A}' \coloneqq (a_1, a_2, ..., a_d) \in \Sigma_{\mathbf{A}}$, and a data $M \in \mathscr{M}$; we encrypt a pair $(\mathbf{A}', M)$ using the *Encrypt* algorithm of our scheme. For example, $\mathscr{M}$ is a personal bank transaction, a_1 is the transaction value, a_2 is the card expiration date, and so on. Also, let $\Sigma_{\mathbf{K}} \coloneqq \mathbb{Z}_N^d$ and $\mathbf{K}' \coloneqq (k_1, k_2, ..., k_d) \in \Sigma_{\mathbf{K}}$. We interpret the predicate $P_{\mathbf{K}'}(\mathbf{A}')$ (i.e., if $k_i = a_i$ or not) for equality test. We interpret the predicate $P_{\mathbf{K}'}(\mathbf{A}')$ as a comparison predicate (i.e., if $k_i \geq a_i$ or not) for a comparison test. We interpret $\mathbf{A}'$ as a set and interpret the predicate $P_{\mathbf{K}'}(\mathbf{A}')$ as a subset predicate (i.e., if $k_i \in \mathbf{A}'$ or not) for subset test. Then, to achieve above-mentioned three kinds of tests, for the attribute $\mathbf{A}'$, set $\mathbf{A} \coloneqq (-\mathbf{A}', 1)$ and encrypt a data M using $\mathbf{A}$. To generate a token for the attribute $\mathbf{K}'$,

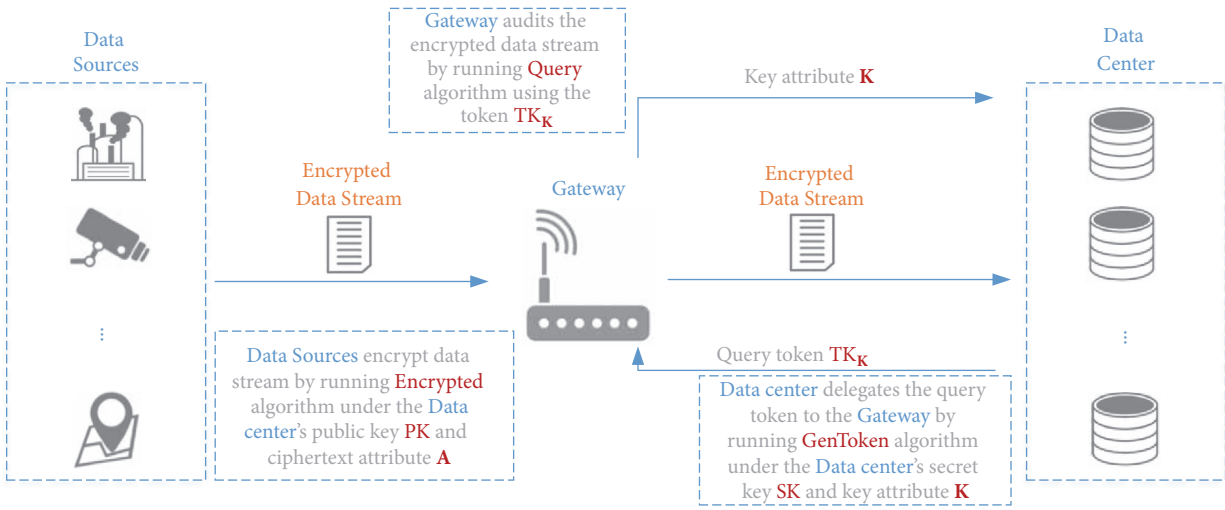

Figure 2: Logical structure for querying on the encrypted industrial data stream.

set $\mathbf{K} := (1, \mathbf{K}')$. Since $\langle \mathbf{K}, \mathbf{A} \rangle = 0$, if and only if $\mathbf{K}' = \mathbf{A}'$, correctness and security follow.

6.3. Polynomial Evaluation. Similar to the predicate encryption scheme presented by Katz et al. [15], we can also support the polynomial evaluation by defining the classes of predicates accordingly. A Φ-searchable functional encryption for polynomials of degree $\leq d(p(x) = k_0 + k_1 x^1 + \cdots + k_d x^d)$ can be defined as follows. Let key attributes space $\Sigma_{\mathbf{K}}^{Poly_{\leq d}} := \mathbb{Z}_p^{d+1}$; we map the polynomial $p(x) = k_0 + k_1 x_1 + \cdots + k_d x^d$ to $\mathbf{K} := (k_0, k_1, \ldots, k_d)$. For ciphertext attribute, each element $w \in \mathbb{Z}_p$ is mapped onto a ciphertext attribute vector $\mathbf{A} := (w^0 \bmod N, w^1 \bmod N, \ldots, w^d \bmod N)$. We also need to define the predicate set $\Phi_{\leq d}^{Poly} := \{P_{\mathbf{K}}^{Poly}(\mathbf{A}) | p \in \mathbb{Z}_N[x], \deg(p) \leq d\}$, where

$$P_{\mathbf{K}}^{Poly}(\mathbf{A}) := \begin{cases} 1 & if\ p(x) = 0 \bmod N \\ 0 & Otherwise \end{cases} \tag{36}$$

for $x \in \mathbb{Z}_N$.

Then, for predicate $P_{\mathbf{K}}^{Ploy}(\mathbf{A}) \in \Phi_{\leq d}^{Ploy}$, correctness and security of our Φ-searchable functional encryption hold, since $p(w) = 0$ whenever $\langle \mathbf{K}, \mathbf{A} \rangle = 0$.

6.4. Conjunctions, Disjunctions, CNF, and DNF Formulas. Based on our Φ-searchable functional encryption for $P_{\mathbf{K}}^{Ploy}(\mathbf{A}) \in \Phi_{\leq d}^{Ploy}$, we can easily support the conjunctions, disjunctions, and their extensions CNF/DNF. We show this ability using an example of conjunctions of equality tests. To do this, for some $\mathbf{K} := (k_0, k_1, ..., k_d)$ and $\mathbf{A} := (a_0, a_1, ..., a_d)$ we define the conjunction predicate as $P_{k_1,k_2}^{\text{AND}}(a_1, a_2)$, where $P_{k_1,k_2}^{\text{AND}}(a_1, a_2) = 1$ if both $k_1 = a_1$ and $k_2 = a_2$. This predicate can be a polynomial as

$$p(x_1, x_2) = r \cdot (x_1 - k_1) + (x_2 - k_2), \tag{37}$$

where $r \longleftarrow_{\$} \mathbb{Z}_N$. If $P_{k_1,k_2}^{\text{AND}}(a_1, a_2) = 1$, then $p(a_1, a_2) = 0$. Otherwise, with all but negligible probability over the choice of r, it will hold that $p(a_1, a_2) \neq 0$.

In a similar fashion, we can define the predicate for disjunction of equality tests. For some $\mathbf{K} := (k_0, k_1, ..., k_d)$ and $\mathbf{A} := (a_0, a_1, ..., a_d)$, we define the disjunction predicate as $P_{k_1,k_2}^{OR}(a_1, a_2)$, where $P_{k_1,k_2}^{OR}(a_1, a_2) = 1$ if either $k_1 = a_1$ or $k_2 = a_2$. This predicate also can be a polynomial as

$$p(x_1, x_2) = (x_1 - k_1) \cdot (x_2 - k_2) \tag{38}$$

If $P_{k_1,k_2}^{OR}(a_1, a_2) = 1$, then $p(a_1, a_2) = 0$; otherwise $p(a_1, a_2) \neq 0$.

We can combine disjunctions, conjunctions, and Boolean variables to handle arbitrary CNF or DNF formulas.

7. Comparison and Evaluation

7.1. Comparison. We compare our construction to prior constructions which are built for inner product evaluations on encrypted data in composite-order bilinear groups [15, 31]. We show the comparison of the basic parameters' performance between these schemes in Table 1.

We use KSW12 to denote the scheme proposed by Katz, Sahai, and Waters [15] and LL18 to denote the scheme proposed by Lee and Lee [31]. As shown in Table 1, our scheme has been proven to be secure against adaptive adversaries, whereas the other two schemes just have selective security. The lengths of the public key are $(3 + 2d)|\mathbb{G}|$ for KSW12 and LL18, whereas $(2 + d)|\mathbb{G}| + |\mathbb{G}_T|$ for our proposal. Obviously, our construction has a smaller public key than others. For the length of the search token (i.e., the private key), our construction has nearly half elements of others. Our construction also gets smaller ciphertext size, $(1+d)|\mathbb{G}|+|\mathbb{G}_T|$, which has just one more group element than LL18 but nearly half elements of KSW12.

Table 1: Comparison of basic parameters∗.

Scheme	Len_{PK}	Len_{TK}	Len_C	SecLev
KSW12 [15]	$(3+2d)\|\mathbb{G}\|$	$(1+2d)\|\mathbb{G}\|$	$(1+2d)\|\mathbb{G}\|$	selectively
LL18 [31]	$(3+2d)\|\mathbb{G}\|$	$2d\|N\|$	$(1+d)\|\mathbb{G}\|$	selectively
Our proposal	$(2+d)\|\mathbb{G}\|+\|\mathbb{G}_T\|$	$(1+d)\|\mathbb{G}\|$	$(1+d)\|\mathbb{G}\|+\|\mathbb{G}_T\|$	*adaptively*

∗ Let Len_{PK} denote the length of the public key including public parameters, let Len_{TK} denote the length of the search token (in some schemes, i.e., the private key), let Len_C denote the length of the ciphertext, and let SecLev denote the security level. Let $|\mathbb{G}|$ and $|\mathbb{G}_T|$ denote the length of the element in groups $\mathbb{G}$ and $\mathbb{G}_T$, respectively, let $|N|$ denote the length of the element in the field $\mathbb{Z}_N^*$, and let d denote the dimension of the ciphertext attribute and the key attribute.

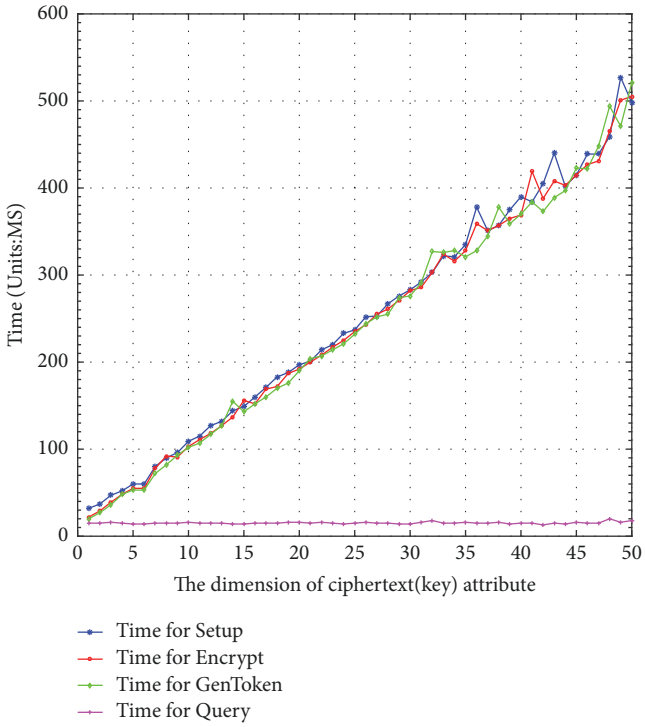

Figure 3: Time cost of the algorithms.

7.2. Performance Evaluation. We implement the algorithms of proposed functional encryption scheme using pairing-based cryptography library pbc-0.5.14 with pbc wrapper-0.8.0 [39] on a PC with 3.3GHz Intel, i5-6600 CPU, and 8GB memory. In our implementation, we made use of parameter *a.param*, one of the standard parameter settings of pbc library. The implementation time overheads are demonstrated as shown in Figure 3. We would like to observe the impact of the dimension of ciphertext (key) attribute vector in terms of the time cost of the algorithms. Obviously, the bigger dimension of ciphertext (key) attribute will make the attributes more expressive, which means the scheme will support more complex predicates. We can see that the time consumptions of the *Setup* algorithm, *Encrypt* algorithm, and *GenToken* algorithm are linearly increasing with the increase from 1 to 50 of the dimension of ciphertext (key) attribute d. The *Setup* algorithm is run by the trusted authority which usually can be executed once and offline. The *Encrypt* algorithm is run by the data source which also can be executed offline. The *GenToken* algorithm is run by the data center that has powerful computing ability. So, the time cost of these three algorithms is considerably acceptable. Fortunately, the *Query* algorithm's time cost is nearly constant (less than 20ms) with the increase of the dimension of ciphertext (key) attribute d. This merit makes the gateway able to effectively test the encrypted data passing by without a significant reduction in processing speed.

8. Conclusions

In this paper, we have put forth an Φ-searchable functional encryption scheme. We have built our scheme on the composite-order bilinear groups and have proven the adaptive security by utilizing dual system encryption proof technology. By using our proposed scheme as the underlying encryption scheme, we present an approach that supports the fact that the gateway effectively audits the encrypted data stream. According to the comparison and performance evaluation results, our proposed encryption scheme has the smaller public key, the smaller query token, and the smaller ciphertext. Moreover, our proposed approach can enable the gateway to effectively test the encrypted data stream, which is practical for industrial data stream auditing scenario.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by National Key R&D Program of China (no. 2017YFB0802000), the National Natural Science Foundation of China (61572303, 61772326, 61802241, and 61802242), National Cryptography Development Fund during the 13th Five-year Plan Period (MMJJ20180217), the Foundation of State Key Laboratory of Information Security (2017-MS-03), the Fundamental Research Funds for the Central Universities (no. GK201903089, no. GK201702004, and no. GK201603084), and the Natural Science Basic Research Plan in Shaanxi Province of China (2019JM-552).

References

[1] Q. Jiang, X. Huang, N. Zhang, K. Zhang, X. Ma, and J. Ma, "Shake to communicate: secure handshake acceleration-based pairing mechanism for wrist worn devices," *IEEE Internet of Things Journal*, 2019.

[2] Q. Jiang, J. Ma, and C. Yang, "Efficient End-To-End Authentication Protocol for Wearable Health Monitoring Systems," *Computers Electrical Engineering*, vol. 63, pp. 182–195, 2017.

[3] X. Li, Y. Zhu, J. Wang, Z. Liu, Y. Liu, and M. Zhang, "On the soundness and security of privacy-preserving SVM for outsourcing data classification," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 906–912, 2018.

[4] Q. Jiang, Y. Qian, J. Ma, X. Ma, Q. Cheng, and F. Wei, "User centric three-factor authentication protocol for cloud-assisted wearable devices," *International Journal of Communication Systems*, vol. 32, no. 6, Article ID e3900, 2019.

[5] Z. Liu, X. Huang, Z. Hu, M. K. Khan, H. Seo, and L. Zhou, "On emerging family of elliptic curves to secure internet of things: ECC comes of age," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 237–248, 2017.

[6] D. X. Song, D. A. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proceedings of IEEE Symposium on Security and Privacy. (S&P '2000)*, pp. 44–55, USA, May 2000.

[7] E. J. Goh, "Secure Indexes," IACR Cryptology ePrint Archive: Report 2003/216, 2003.

[8] C. Bösch, A. Peter, B. Leenders et al., "Distributed searchable symmetric encryption," in *Proceedings of the Twelfth Annual International Conference on Privacy, Security and Trust, PST '2014*, pp. 330–337, Canada, July 2014.

[9] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," *Journal of Computer Security*, vol. 19, no. 5, pp. 895–934, 2011.

[10] P. Han, C. Liu, B. Fang et al., "Revisiting the practicality of search on encrypted data: from the security brokers perspective , scientific programming," *Scientific Programming*, vol. 2016, 9 pages, 2016.

[11] S. Kamara, C. Papamanthou, and T. Roeder, "Dynamic searchable symmetric encryption," in *Proceedings of the ACM conference on Computer and Communications Security, CCS '2012*, pp. 965–976, USA, October 2012.

[12] Z. Wang, Z. Fu, and X. Sun, "Semantic contextual search based on conceptual graphs over encrypted cloud," *Security and Communication Networks*, vol. 2018, pp. 1–10, 2018.

[13] D. Boneh, G. Di Crescenzo, R. Ostrovsky, G. Persiano et al., "Public key encryption with keyword search," in *Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT '2004*, vol. 3027 of *Lecture Notes in Computer Science*, pp. 506–522, Springer.

[14] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Proceedings of Fourth IACR Theory of Cryptography Conference, TCC '2007*, vol. 4392 of *Lecture Notes in Computer Science*, pp. 535–554, Springer, 2007.

[15] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," *Journal of Cryptology*, vol. 26, no. 2, pp. 191–224, 2013.

[16] N. Attrapadung and B. Libert, "Functional encryption for inner product achieving constant-sizeciphertexts with adaptive security or support for negation," in *Proceedings of the 13th International Conference on Practice and Theory in Public Key Cryptography PKC '2010*, vol. 6056 of *Lecture Notes in Computer Science*, pp. 384–402, Springer Berlin Heidelberg, France, 2010.

[17] R. Xie, C. He, D. Xie, C. Gao, and X. Zhang, "A secure ciphertext retrieval scheme against insider kgas for mobile devices in cloud storage," *Security and Communication Networks*, vol. 2018, Article ID 7254305, 7 pages, 2018.

[18] D. Sharma and D. C. Jinwala, "Multiuser searchable encryption with token freshness verification," *Security and Communication Networks*, vol. 2017, 16 pages, 2017.

[19] S. Kamara and T. Moataz, "SQL on structurally-encrypted databases," IACR Cryptology ePrint Archive Report 2016/453, 2016.

[20] D. N. Wu, Q. Q. Gan, and X. M. Wang, "Verifiable public key encryption with keyword search based on homomorphic encryption in multi-user setting," *IEEE Access*, vol. 6, 9 pages, 2018.

[21] G. Asharov, M. Naor, G. Segev et al., "earchable symmetric encryption -optimal locality in linear space via two-dimensional balanced allocations," in *Proceedings of the 48th Annual Symposium on the Theory of Computing, STOC '2016*, pp. 1101–1114, ACM, New York, NY, USA, 2016.

[22] D. Boneh, A. Sahai, and B. Waters, "Functional encryption: definitions and challenges," IACR Cryptology ePrint Archive Report 2010/543, 2010.

[23] S. Garg, C. Gentry, S. Halevi et al., "Candidate indistinguishability obfuscation and functional encryption for all circuits," in *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science, FOCS '2013*, pp. 40–49, USA, October 2013.

[24] M. Zhang, Y. Zhang, Y. Jiang, and J. Shen, "Obfuscating eves algorithm and its application in fair electronic transactions in public cloud systems," *IEEE Systems Journal*, 2019.

[25] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques, CRYPTO '1984*, vol. 196 of *Lecture Notes in Computer Science*, pp. 47–53, Springer, November 2000.

[26] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Proceedings of the Annual International Cryptology Conference, CRYPTO '2001*, vol. 2139 of *Lecture Notes in Computer Science*, pp. 213–229, USA, August 2001.

[27] A. Sahai and B. Waters, "Fuzzy identity-based encryption," Cryptology ePrint Archive Report 2004/086, 2004.

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

[29] T. Okamoto and K. Takashima, "Adaptively Attribute-Hiding (Hierarchical) Inner Product Encryption," in *Advances in Cryptology – EUROCRYPT 2012*, vol. 7237 of *Lecture Notes in Computer Science*, pp. 591–608, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[30] A. Lewko, "Tools for simulating features of composite order bilinear groups in the prime order setting," in *Advances in Cryptology—EUROCRYPT 2012*, vol. 7237 of *Lecture Notes in Computer Science*, pp. 318–335, Springer, Berlin, Heidelberg, Germany, 2012.

[31] K. LEE and D. H. LEE, "Two-Input functional encryption for inner products from bilinear maps," *IEICE Transactions on*

Fundamentals of Electronics, Communications and Computer Sciences, vol. E101.A, no. 6, pp. 915–928, 2018.

[32] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Proceedings of the Second Theory of Cryptography Conference, TCC '2005*, vol. 3378 of *Lecture Notes in Computer Science*, pp. 325–341, Springer, USA.

[33] A. Lewko and B. Waters, "New techniques for dual system encryption and fully secure HIBE with short ciphertexts," in *Theory of Cryptography*, vol. 5978 of *Lecture Notes in Computer Science*, pp. 455–479, Springer, Berlin, Germany, 2010.

[34] D. Boneh, A. Sahai, and B. Waters, "Fully collusion resistant traitor tracing with short ciphertexts and private keys," in *Advances in Cryptology - EUROCRYPT 2006*, vol. 4004 of *Lecture Notes in Computer Science*, pp. 573–592, Springer, Russia, 2006.

[35] B. Waters, "Dual system encryption: realizing fully secure ibe and hibe under simple assumptions," in *Advances in Cryptology—CRYPTO 2009*, vol. 5677 of *Lecture Notes in Computer Science*, pp. 619–636, Springer, 2009.

[36] T. Okamoto and K. Takashima, "Fully secure functional encryption with general relations from the decisional linear assumption," in *Advances in Cryptology—CRYPTO 2010*, T. Rabin, Ed., vol. 6223 of *Lecture Notes in Computer Science*, pp. 191–208, Springer, Berlin, Germany, 2010.

[37] A. Lewko, Y. Rouselakis, and B. Waters, "Achieving leakage resilience through dual system encryption," in *Theory of Cryptography—TCC 2011*, vol. 6597 of *Lecture Notes in Computer Science*, pp. 70–88, Springer, Berlin, Heidelberg, Germany, 2011.

[38] J. Zhang, J. Chen, A. Ge et al., "Shorter decentralized attribute-based encryption via extended dual system groups," *Security and Communication Networks*, vol. 2017, 19 pages, 2017.

[39] B. Lynn, "The pairing-based cryptography library (0.5.13)," http://crypto.stanford.edu/pbc/.