IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 14, 2019, accepted March 28, 2019, date of current version April 24, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2909040

Security Analysis and Enhancement of a Certificateless Searchable Public Key Encryption Scheme for IIoT Environments

TSU-YANG WU[1,2], CHIEN-MING CHEN[3], KING-HANG WANG[4], AND JIMMY MING-TAI WU[3]

[1]Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350118, China
[2]National Demonstration Center for Experimental Electronic Information and Electrical Technology Education, Fujian University of Technology, Fuzhou 350118, China
[3]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[4]Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong

Corresponding author: Jimmy Ming-Tai Wu (wmt@wmt35.idv.tw)

The work of T.-Y. Wu was supported in part by the Science and Technology Development Center, Ministry of Education, China, under Grant 2017A13025 and in part by the Natural Science Foundation of Fujian Province under Grant 2018J01636.

ABSTRACT With the fast development of Industrial Internet of Things (IIoT), IIoT storage, which provides a lower cost, higher reliability of data, remote access services, and expandable storage space, is received much attention among enterprises and individual users. However, providing a secure IIoT storage service could be a challenging task for some situations, for example, how to search encrypted data in the IIoT. In this paper, we examine the security of a recent proposed certificateless searchable public key encryption scheme for IIoT environments by Ma *et al.* We find that their scheme is insecure against an off-line keyword guessing attack. Then, we propose an enhancement based on the scheme and prove the security of our enhancement with a formal model.

INDEX TERMS IIoT, searchable public key encryption, certificateless, data storage, provably secure.

I. INTRODUCTION

Internet of Things (IoT) [1]–[4] is the network of physical devices which contain embedded technology such as sensors [5], [6], RFID and network connectivity [7], [8] to communicate with other devices or external environment. This has led to new avenues for connecting technologies and businesses in various areas such as health care, transportation, commerce, data mining [9]–[13] etc. The phrase IoT was firstly proposed by MIT Auto-ID Center in 1999 [14]. Now IoT has become a popular topic of research in both academia and industry.

Recently, Industrial Internet of Things (IIoT) [15], the IoT used in industry for advanced manufacturing, receives increasing attention. Along the continuous development of IIoT, IIoT cloud storage, which excels in remote access service, low cost, high data reliability, large and expandable storage space, is becoming more and more popular among enterprises and individual users. A typical network environment for IIoT storage is depicted in Fig. 1. In this environment, the enterprise collects the data during the industrial production and operating status of equipments.

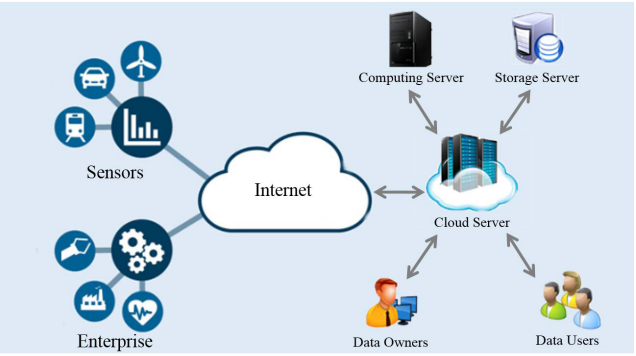


FIGURE 1. A typical network environment for IIoT storage.

Meanwhile, other outside information are collected by sensors. Then, these data are sent to the cloud via the Internet. The cloud server can interact with the computing server and the storage server and is responds to computing and storing of the data.

However, due to the fact that cloud data management is out of the supervision of enterprise and individual users, many sensitive information is easily leaked. Therefore, cloud

The associate editor coordinating the review of this manuscript and approving it for publication was Jiafeng Xie.

service providers must ensure that the data collected by the sensors are securely stored. In order to prevent privacy data leakage, the most straightforward method is let sensors encrypt their data. However, it would lead to another problem. That is, when a user needs to retrieve the encrypted data, cloud server cannot response the request since the server does not have the encryption key.

Aiming at the search problem of encrypted data, Boneh *et al.* [16] proposed the first searchable public key encryption (SPKE) scheme in traditional pairing-based cryptosystems. This scheme allows a data owner (or a sensor) has their encrypted data hosted on a cloud server while a data user, who is capable in decrypting the data, can access a particular piece of data by making a query to the server. This query, which is also encrypted, however allows the server to retrieve a set of encrypted data fulfilling the query. In principle both data user and data owner do not need to trust the cloud server, which is indeed not always trustworthy, but at the same time no one other than them shall know the content of the data nor the query of the data user. This framework was used in some subsequent works [17], [18].

In 2008, Baek *et al.* [19] proposed a new SPKE scheme with designated server, namely SCF-MPEKS. Beak et al.'s framework first introduced the server role and pointed out that the attacker can be divided into malicious servers and external attackers. However, Rhee *et al.* [20] pointed out SCF-MPEKS suffered from an off-line keyword guessing attack. They also proposed an enhanced framework. In 2010, Rhee *et al.* [21] proposed a variant of SCF-MPEKS scheme called SCF-dMPEKS. Meanwhile, they defined a new security notion called ''trapdoor indistinguishability'' which provides formal provably secure of SPKE schemes to against off-line keyword guessing attacks [22]–[31].

In 2003, Al-Riyami and Paterson [32] proposed the concept of certificateless public key cryptosystem (CLPKC) to solve the key escrow problem in identity (ID)-based public key cryptosystems [33]–[36]. After that SPKE schemes designed in CLPKC were received much attentions by cryptographers in [37]–[39]. Recently, Ma *et al.* [40] proposed a new SPKE scheme based on CLPKC called SCF-MCLPEKS which not only eliminates the certificate management but also prevents the key escrow problems. Their scheme is the first certificateless-based SPKE scheme applied to IIoT environment and provides a higher efficiency and security (provable secure) as well as solving the search problem of encrypted data and providing a powerful guarantee in IIoT cloud storage.

In this paper, we inspect Ma et al.'s SCF-MCLPEKS scheme and found that SCF-MCLPEKS is in fact insecure against an off-line keyword guessing attack. Under this attack a malicious server, or an attacker having control over the public channel can retrieve the content of query issued by the data user, which is supposed to be private. This implies the attacker will learn the attributes of the encrypted data. Take an concrete example, say a data user wish to retrieve documents with the meta-data ''confidential''. The attacker will first gain the knowledge of the query is about searching ''confidential'' documents. From the reply of the server he also know which encrypted documents is confidential. This attack brings severe problem and cannot be ignored.

In order to overcome the proposed attack, we present a modified security model and a modified SCF-MCLPEKS (called SCF-MCLPEKS^{+}) based on their scheme. Finally, we prove that our enhancement can resist off-line keyword guessing attacks.

The rest of this paper is organized as follows. In Section II, we briefly review Ma et al's scheme and then in Section III, point out the drawbacks of their scheme. We also propose our enhancement. In Section VI, we formally prove the security of our enhancement and demonstrate its performance in Section V. Finally, we conclude the paper in Section VI.

II. REVIEW OF MA et al.'s SCF-MCLPEKS SCHEME

In this section, we briefly review Ma et al.'s SCF-MCLPEKS scheme [40], including the underlying mathematical method (pairing) and the detailed scheme.

A. BILINEAR PAIRING AND HARD PROBLEM

Here, we refer literatures [33]–[36], [39], [41] to present a review of the pairings and hard problem that plays an important role in this paper.

Definition 1 (Pairings): Let $E(F_p)$ be an elliptic curve over a finite field F_p. A pairing (or called bilinear pairing) $e : G_1 \times G_1 \rightarrow G_T$ is a map. In general, G_1 is set an additive group of $E(F_p)$ and G_T is set a multiplicative group of F_p with the same large prime order q. This map satisfies the following three properties:

1) Bilinear: For all $x, y \in Z_q^*$ and $P, Q \in G_1$, $e(xP, yQ) = e(P, Q)^{xy}$.
2) Non-degenerate: $e(\epsilon_{G_1}, \epsilon_{G_1})$ is an identity of G_T, for any identity $\epsilon_{G_1} \in G_1$.
3) Computable: For all $P, Q \in G_1$, there exist several algorithms to compute $e(P, Q)$.

Definition 2 (Bilinear Diffie-Hellman (BDH) Problem): Given $P, xP, yP, zP \in G_1$ for some $x, y, z \in Z_q^*$ are unknown, the BDH problem is to compute $e(P, P)^{xyz}$ in the group G_2.

Definition 3 (BDH Assumption): No a probabilistic polynomial time (PPT) algorithm that can be used to solve the BDH problem.

B. DETAILED SCF-MCLPEKS SCHEME

The detailed SCF-MCLPEKS scheme can be described as follows:

Setup. Inputting a security parameter k, the KGC executes the following steps to generate the required system parameters.

1) To generate a bilinear map $e : G_1 \times G_1 \rightarrow G_T$.
2) To select a master key $s \in_R Z_q^*$. Then, computing $P_{pub} = s \cdot P$, where P is a generator of G_1.
3) To select four hash functions $H_1, H_2, H_3 : \{0, 1\}^* \rightarrow G_1$, and $H_4 : G_T \rightarrow \{0, 1\}^{\log_q}$.

Finally, the public parameter *param* is defined by $param = \{k, G_1, G_2, e, q, P, P_{pub}, H_1, H_2, H_3, H_4\}$.

Partial-Private-Key-Extract. Inputting server's identity $ID_S \in \{0,1\}^*$, it returns server's partial private key $D_S = s \cdot Q_S$, where $Q_S = H_1(ID_S)$. Similarly, receiver's partial private key D_R is computed by $D_R = s \cdot Q_R$, where $Q_R = H_1(ID_R)$ and $ID_R \in \{0,1\}^*$.

Set-Secret-Value. Server and receiver select x_S and $x_R \in Z_q^*$ as their secret value, respectively.

Set-Private-Key. Server's private key and receiver's private key are defined by $SK_S = (x_S, D_S)$ and $SK_R = (x_R, D_R)$, respectively.

Set-Public-Key. Server's public key and receiver's public key are defined by $PK_S = x_S \cdot P$ and $PK_R = x_R \cdot P$, respectively.

MCLPKES. Inputting *param*, ID_S, PK_S, ID_R, PK_R, and n keywords w_i for $i = 1, 2, \ldots, n$, data owner generates ciphertext $C_i = (U_i, v_i)$ of w_i as follows.

1) Selecting $r_i \in Z_q^*$ and computing $U_i = r_i \cdot P$.
2) Computing $T_i = e(r_i \cdot H_2(w_i), PK_R + PK_S) \cdot e(r_i \cdot Q_R, P_{pub}) \cdot e(r_i \cdot H_3(w_i), P)$ and $v_i = H_4(T_i)$, where $Q_R = H_1(ID_R)$.

Trapdoor. Inputting *param*, w_i, and SK_R, receiver computes trapdoor $T_w = x_R \cdot H_2(w) + D_R$.

Test. Inputting *param*, T_w, SK_S, and C_i, server verifies $H_4(e(T_w + x_S \cdot H_2(w_i) + H_3(w_i), U_i)) \stackrel{?}{=} v_i$. If it is true, output "1." Otherwise, output "0."

III. SECURITY ANALYSIS AND OUR ENHANCEMENT OF SCF-MCLPEKS

In this section, we first point out that Ma et al.'s SCF-MCLPEKS scheme suffered from an off-line keyword guessing attacks. In order to overcome this attack, we propose a modified framework and an enhancement based on SCF-MCLPEKS.

A. DRAWBACKS OF SCF-MCLPEKS

Here, we demonstrate that Ma et al.'s SCF-MCLPEKS scheme suffered from an off-line keyword guessing attack. Note that, "off-line keyword guessing attack" means that attacker can test the linkability between keyword and trapdoor while the attacker captured a trapdoor. In other words, this attack will lead violating "trapdoor indistinguishability", if it success. We assume a PPT external attacker A or a malicious sever S can capture a valid trapdoor T_w. The goal of A (or S) is to recover $w_i \in W$ from T_w, where W is a set of all possible keywords. The details are described as follows:

1) Guessing a keyword $w' \in W$.
2) Verifying $e(T_w, P) \stackrel{?}{=} e(H_2(w'), PK_R) \cdot e(H_1(ID_R), P_{pub})$.

It the verification is true, it returns w'. Obviously, if $w' = w$, $e(T_w, P) = e(x_R \cdot H_2(w) + D_R, P) = e(x_R \cdot H_2(w), P) \cdot e(s \cdot H_1(ID_R), P) = e(H_2(w), P)^{x_R} \cdot e(H_1(ID_R), P)^s = e(H_2(w'), PK_R) \cdot e(H_1(ID_R), P_{pub})$.

B. OUR MODIFIED FRAMEWORK OF SCF-MCLPEKS

In this subsection, we refer literatures [39], [42] to propose a modified framework of SCF-MCLPEKS. In the new framework, we involve the authentication functionality and modify the definition of MCLPKES and trapdoor generating algorithms. The new definition of two algorithms are defined as follows.

$C \leftarrow MCLPEKS(param, SK_O, ID_S, PK_S, ID_R, PK_R, w_i)$: Inputting *param*, data owner's private key SK_O, data user's identity ID_R, server's identity ID_S, data user's public key PK_R, server's public key PK_S, and keyword w_i, this algorithm returns ciphertext C.

$T_w \leftarrow Trapdoor(param, w, SK_R, PK_S, PK_O)$: Inputting *param*, w_i, data user's private key SK_R, PK_S, and data owner's public key PK_O, this algorithm returns a trapdoor T_w for w_i.

Note that, for the original framework of SCF-MCLPEKS please refer to [40].

C. OUR ENHANCEMENT

In this section, we propose an enhancement named as SCF-MCLPEKS$^+$. Our enhanced scheme is proposed as follows.

Setup. Given security parameter k, it generates the needed parameters $param = \{k, G_1, G_2, e, q, P_1, P_2, P_{pub}, H_1, H_2\}$.

1) Generating a bilinear map $e : G_1 \times G_1 \rightarrow G_T$.
2) Selecting $s \in_R Z_q^*$ as master key and computing $P_{pub} = s \cdot P_1$, where $P_1, P_2 \in_R G_1$ are generators in G_1.
3) Choosing hash functions $H_1 : \{0,1\}^* \rightarrow G_1$ and $H_2 : \{0,1\}^* \rightarrow Z_q^*$.

Partial-Private-Key-Extract. Data owner's partial private key D_O, data user's partial private key D_R, and server's partial private key D_S are computed by the same way in SCF-MCLPEKS.

Set-Secret-Value. Data owner's secret value x_O, data user's secret value x_R, and server's secret value x_S are chosen by the same way in SCF-MCLPEKS.

Set-Private-Key. Data owner's private key $SK_O = (x_O, D_O)$, data user's private key $SK_R = (x_R, D_R)$, and server's private key $SK_S = (x_S, D_S)$ are set by the same way in SCF-MCLPEKS.

Set-Public-Key. Data owner's public key $PK_O = x_O \cdot P_1$, data user's public key $PK_R = x_R \cdot P_1$, and server's public key $PK_S = x_S \cdot P_1$ are computed by the same way in SCF-MCLPEKS.

MCLPKES. Inputting *param*, SK_O, ID_S, PK_S, ID_R, PK_R, and n keywords w_i for $i = 1, 2, \ldots, n$, data owner returns ciphertext $\{C_1, C_2, C_{3,1}, C_{3,2}, \ldots, C_{3,n}\}$ for w_i as follows.

1) Selecting $r_i \in Z_q^*$.
2) Computing $C_1 = r_i \cdot P_1$, $C_2 = r_i \cdot P_2$, and $C_{3,i} = e(C_1, PK_R)^{x_O \cdot H_2(w_i)} \cdot e(H_1(ID_R) + H_1(ID_S), P_{pub})^{r_i}$.

Trapdoor. Inputting *param*, w_i, SK_R, PK_S, and PK_O, data user executes the following steps to generate trapdoor T_{w_i}.

1) Selecting $r \in_R Z_q^*$.
2) Computing $T_1 = D_R + x_R \cdot H_2(w_i) \cdot PK_O + r \cdot P_2$ and $T_2 = r \cdot PK_S$.

The trapdoor T_{w_i} of w_i is defined by $T_{w_i} = (T_1, T_2)$.

Test. Inputting *param*, T_{w_i}, the server's private key SK_S, and ciphertext C_W, the server verifies $e(C_1, D_S + T_1) \stackrel{?}{=} C_{3,i} \cdot e(T_2, C_2)^{x_S^{-1}}$. If it is true, outputting "1". Otherwise, outputting "0".

The correctness of SCF-MCLPEKS$^+$ can be seen as below:

$$\begin{aligned} & e(C_1, D_S + T_1) \\ &= e(r_i \cdot P_1, D_S + D_R + x_R \cdot H_2(w) \cdot PK_O + r \cdot P_2) \\ &= e(r_i \cdot P_1, D_S + D_R) \cdot e(r_i \cdot P_1, x_R \cdot H_2(w) \cdot PK_O) \\ &\quad \cdot e(r_i \cdot P_1, r \cdot P_2) \\ &= e(H_1(ID_R) + H_1(ID_S), P_{pub})^{r_i} \cdot e(C_1, PK_R)^{x_O \cdot H_2(w_i)} \\ &\quad \cdot \cdot e(r \cdot P_1, r_i \cdot P_2) \\ &= C_{3,i} \cdot e(T_2, C_2)^{x_S^{-1}}. \end{aligned}$$

IV. SECURITY ANALYSIS

In this section, we first refer literatures [39], [40], [42], [43] to define a new security model of our enhancement SCF-MCLPEKS$^+$. Then, we show that the security of our enhancement.

A. SECURITY MODEL OF SCF-MCLPEKS$^+$

We assume that there are adversary $\mathcal{A}$ and challenger $\mathcal{C}$ in the following games, where $\mathcal{A}$ may make some queries to $\mathcal{C}$ and it answers $\mathcal{A}$ honestly.

Note that there are two types of adversaries named type I ($\mathcal{A}_I$) and type II ($\mathcal{A}_{II}$) in certificateless public key cryptosystems. The ability of $\mathcal{A}_I$ is set to disallow querying master key and the ability of $\mathcal{A}_{II}$ hasn't this limitation. In other words, $\mathcal{A}_I$ is to simulate an external attacker and $\mathcal{A}_{II}$ is to simulate a malicious server in SCF-MCLPEKS$^+$.

1) CIPHERTEXT INDISTINGUISHABILITY

1) System setup. $\mathcal{C}$ first generates a master key x and *param*. If $\mathcal{A} = \mathcal{A}_I$, $\mathcal{C}$ returns *param* to $\mathcal{A}_I$. Otherwise, $\mathcal{C}$ sends *param* and x to $\mathcal{A}_{II}$.
2) Trapdoor query. $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ may make this query with w, SK_R, PK_S, PK_O to $\mathcal{C}$. Upon receiving this request, it returns a trapdoor T_w to $\mathcal{A}$. Note that, the definitions about Hash query, Partial-Private-Key-Extract query, Private-Key-Extract query, Public-Key-Retrieve query, and Public-Key-Replacing request are same as in [39].
3) Challenge. $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ may send SK_O^*, ID_S^*, PK_S^*, ID_R^*, PK_R^*, w_0^*, and w_1^* to $\mathcal{C}$. Upon receiving this request, it returns a ciphertext $C_{w_b^*}$ to $\mathcal{A}$, where $b \in \{0, 1\}$.
4) More queries. $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ may continue to make more trapdoor queries for w', SK_R', PK_S', and PK_O'. The restrictions are $w' \neq \{w_0^*, w_1^*\}$, $SK_R' \neq SK_R^*$, $PK_S' \neq PK_S^*$, and $PK_O' \neq PK_O$.
5) Guess. Finally, $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ returns $b' = \{0, 1\}$ to show $C_{w_b^*}$ is the result of $C_{w_0^*}$ or $C_{w_1^*}$.

We call that $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ wins the above game, if the advantage of $\mathcal{A}$, $Adv^{Cipher-ind}_{\mathcal{A}=\{\mathcal{A}_I,\mathcal{A}_{II}\}}(t) = |\Pr[b' = b] - 1/2|$, is non-negligible.

Definition 4 (Ciphertext Indistinguishability): We call that a SCF-MCLPEKS$^+$ scheme satisfies ciphertext indistinguishability under adaptive chosen keyword attacks, if the advantage $Adv^{Cipher-ind}_{\mathcal{A}=\{\mathcal{A}_I,\mathcal{A}_{II}\}}(t)$ is negligible for any PPT adversary $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$.

2) TRAPDOOR INDISTINGUISHABILITY

Note that, the definitions about the system setup phase, query issuing phase, Public-Key Replacing request, and more queries phase are same as in the above game.

1) Challenge. $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ may send w_0^*, w_1^*, SK_R^*, PK_S^*, and PK_O^* to $\mathcal{C}$. Upon receiving this request, it returns $T_{w_b^*}$ to $\mathcal{A}$ for $b = \{0, 1\}$.
2) Guess. Finally, $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ returns $b' = \{0, 1\}$ to show $T_{w_b^*}$ is the result of $T_{w_0^*}$ or $T_{w_1^*}$.

We call that $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$ wins the trapdoor indistinguishability game, if the advantage of $\mathcal{A}$, $Adv^{Trap-ind}_{\mathcal{A}=\{\mathcal{A}_I,\mathcal{A}_{II}\}}(t) = |\Pr[b' = b] - 1/2|$, is non-negligible.

Definition 5 (Trapdoor Indistinguishability): We call that a SCF-MCLPEKS$^+$ scheme satisfies trapdoor indistinguishability under adaptive chosen keyword attacks, if the advantage $Adv^{Trap-ind}_{\mathcal{A}=\{\mathcal{A}_I,\mathcal{A}_{II}\}}(t)$ is negligible for any PPT adversary $\mathcal{A} = \{\mathcal{A}_I, \mathcal{A}_{II}\}$.

B. SECURITY PROOF OF SCF-MCLPEKS$^+$

In the following proofs, we follow our best knowledge [39] to demonstrate the security of SCF-MCLPEKS$^+$.

1) CIPHERTEXT INDISTINGUISHABILITY

Here, we prove that our enhancement SCF-MCLPEKS$^+$ satisfies ciphertext indistinguishability.

Lemma 1: In the random oracle model and based on the BDH problem, SCF-MCLPEKS$^+$ scheme provides ciphertext indistinguishability under an adaptive chosen keyword attack for any $\mathcal{A}_I$ adversary.

Proof: In order to prove this lemma, we design a simulation, called Game 1, to simulate SCF-MCLPEKS$^+$. In Game 1, we assume that there exists a PPT algorithm $\mathcal{C}$ which interacts with $\mathcal{A}_I$. Note that the goal of $\mathcal{A}_I$ is to break SCF-MCLPEKS$^+$ scheme and the goal of $\mathcal{C}$ is to solve the BDH problem. Game 1 is set as follows.

1) System setup. $\mathcal{C}$ generates master key $s \in Z_q^*$ and *param*. Then, $\mathcal{C}$ returns *param* to $\mathcal{A}_I$.
2) Trapdoor query. $\mathcal{A}_I$ may make this query with $w^{(i)}$, $SK_R^{(i)}$, $PK_S^{(i)}$, and $PK_O^{(i)}$ to $\mathcal{C}$. Upon receiving this request, it computes $T_1^{(i)} = D_R^{(i)} + x_R^{(i)} \cdot \beta^{(i)} \cdot PK_O^{(i)} + r^{(i)} \cdot P_2$, and $T_2^{(i)} = r^{(i)} \cdot PK_S^{(i)}$, where $r^{(i)}$ is a random value and $\beta^{(i)}$ from H_2 query. Then, $\mathcal{C}$ returns trapdoor $T_{w^{(i)}}^{(i)} = (T_1^{(i)}, T_2^{(i)})$ to $\mathcal{A}_I$. Note that, the descriptions about H_1 query, H_2 query, Partial-Private-Key-Extract query, Private-Key-Extract query, Public-Key-Retrieve query, and Public-Key-Replacing request are similar to [39].
3) Challenge. $\mathcal{A}_I$ may send SK_O^*, ID_S^*, PK_S^*, ID_R^*, PK_R^*, w_0^*, and w_1^* to $\mathcal{C}$. Upon receiving this request, it selects w_b^* for some $b = \{0, 1\}$ and computes $C_1^* = r^* \cdot P_1$, $C_2^* = r^* \cdot P_2$, and $C_3^* = e(C_1^*, PK_R^*)^{x_O^* \cdot \beta^*} \cdot e(Q_R^* + Q_S^*, P_{pub})^{r^*}$, where r^* is a random value, β^* from H_2

query, Q_R^* and Q_S^* from H_1 query. Then, $\mathcal{C}$ returns $C_{w_b^*} = (C_1^*, C_2^*, C_3^*)$ to $\mathcal{A}_I$.

4) More queries. $\mathcal{A}_I$ may continue to make more trapdoor queries for $w^{(i)}$, $SK_R^{(i)}$, $PK_S^{(i)}$, and $PK_O^{(i)}$. The restrictions are $w^{(i)} \neq \{w_0^*, w_1^*\}$, $SK_R^{(i)} \neq SK_R^*$, $PK_S^{(i)} \neq PK_S^*$, and $PK_O^{(i)} \neq PK_O^*$.
5) Guess. Finally, $\mathcal{A}_I$ returns some $b' = \{0, 1\}$ to show $C_{w_b^*}$ is the result of $C_{w_0^*}$ or $C_{w_1^*}$. In the same time, $\mathcal{C}$ chooses (C_1', C_2', C_3') and outputs $\frac{C_3'}{e(x_R' \cdot C_1', PK_O')^{\beta'} \cdot e(C_1', D_S')}$ as its guess, where β' is a value in the challenge phase.

If $\mathcal{A}_I$'s guess is correct, then it must compute $C_3' = e(C_1', PK_R')^{x_O' \cdot H_2(w_b)} \cdot e(Q_R' + Q_S', P_{pub})^{r'}$ for either $b = 0$ or 1. Therefore, setting $P_{pub} = aP_1$, $C_1' = bP_1$, and $Q_R' = cP_1$ for some a, b, c unknown, $\mathcal{C}$ can compute $e(P_1, P_1)^{abc}$ by $\mathcal{A}_I$ as follows.

$$\begin{aligned}
&\frac{C_3'}{e(x_R' \cdot C_1', PK_O')^{\beta'} \cdot e(C_1', D_S')} \\
&= \frac{e(C_1', x_R' \cdot P_1)^{x_O' \cdot \beta'} \cdot e(Q_R', P_{pub})^{r'} \cdot e(Q_S', P_{pub})^{r'}}{e(x_R' \cdot C_1', PK_O')^{\beta'} \cdot e(C_1', D_S')} \\
&= \frac{e(x_R' \cdot C_1', PK_O)^{\beta'} \cdot e(Q_R', P_{pub})^{r'} \cdot e(C_1', D_S')}{e(x_R' \cdot C_1', PK_O')^{\beta'} \cdot e(C_1', D_S')} \\
&= e(Q_R', P_{pub})^{r'} = e(P_1, P_1)^{abc}.
\end{aligned}$$

Lemma 2: In the random oracle model and based on the BDH problem, SCF-MCLPEKS$^+$ scheme provides ciphertext indistinguishability under an adaptive chosen keyword attack for any $\mathcal{A}_{II}$ adversary.

Proof: In order to prove this lemma, we design a simulation, called Game 2, to simulate SCF-MCLPEKS+. In Game 2, we also assume that there exists a PPT algorithm $\mathcal{C}$ which interacts with $\mathcal{A}_{II}$. Note that the goal of $\mathcal{A}_{II}$ is to break SCF-MCLPEKS$^+$ scheme and the goal of $\mathcal{C}$ is to solve the BDH problem. Game 2 is similar to Game 1 described as follows.

1) System setup. $\mathcal{C}$ generates master key $s \in Z_q^*$ and *param*. Then, $\mathcal{C}$ sends them to $\mathcal{A}_{II}$.
2) Queries. $\mathcal{A}_{II}$ may make H_1 query, H_2 query, Private-Key-Extract query, Public-Key Retrieve query, and Trapdoor query to $\mathcal{C}$. The descriptions of these queries are similar to Game 1.
3) Partial-Private-Key Computation. $\mathcal{A}_{II}$ can compute $D^{(i)}$ for any $ID^{(i)} \in \{0, 1\}^*$ by itself.
4) Challenge. The description of this phase is similar to Game 1.
5) More query. The description of this phase is similar to Game 1.
6) Guess. Finally, $\mathcal{A}_{II}$ returns some $b' = \{0, 1\}$ to show $C_{w_b^*}$ is the result of $C_{w_0^*}$ or $C_{w_1^*}$. In the same time, $\mathcal{C}$ chooses (C_1', C_2', C_3') and outputs $\frac{C_3'}{e(D_S' + D_R', C_1')}$ as its guess.

If $\mathcal{A}_{II}$'s guess is correct, then it must compute $C_3' = e(C_1', PK_R')^{x_O' \cdot H_2(w_b)} \cdot e(Q_R' + Q_S', P_{pub})^{r'}$ for either $b = 0$ or 1. Therefore, setting $PK_R' = aP_1$, $PK_O' = bP_1$, and $H_2(w_b) \cdot C_1' = cP_1$ for some a, b, c unknown, $\mathcal{C}$ can compute $e(P_1, P_1)^{abc}$ by $\mathcal{A}_{II}$ as follows.

$$\begin{aligned}
\frac{C_3'}{e(D_S' + D_R', C_1')} &= \frac{e(C_1', PK_R')^{x_O' \cdot H_2(w_b)} \cdot e(Q_R' + Q_S', P_{pub})^{r'}}{e(D_S' + D_R', C_1')} \\
&= e(H_2(w_b) \cdot C_1', PK_R')^{x_O'} = e(P_1, P_1)^{abc}.
\end{aligned}$$

According to the results of Lemmas 1 and 2, we can conclude that our enhancement SCF-MCLPEKS$^+$ satisfies ciphertext indistinguishability in Theorem 1.

Theorem 1: In the random oracle model and based on the bilinear Diffie-Hellman (BDH) problem, our enhancement SCF-MCLPEKS$^+$ satisfies ciphertext indistinguishability under an adaptive chosen keyword attack for the two types of adversaries $\mathcal{A}_I$ and $\mathcal{A}_{II}$.

2) TRAPDOOR INDISTINGUISHABILITY

Here, we prove that our enhancement SCF-MCLPEKS$^+$ satisfies trapdoor indistinguishability.

Lemma 3: In the random oracle model and based on the BDH problem, SCF-MCLPEKS$^+$ provides trapdoor indistinguishability under an adaptive chosen keyword attack for any $\mathcal{A}_I$ adversary.

Proof: The proof of Lemma 3 is similar to Lemma 1 except challenge and guess phases.

1) Challenge. $\mathcal{A}_I$ may send SK_R^*, PK_S^*, PK_O^*, w_0^*, and w_1^* to $\mathcal{C}$. Upon receiving this request, $\mathcal{C}$ chooses a keyword w_b^* for some $b = \{0, 1\}$ and computes $T_1^* = D_R^* + x_R^* \cdot \beta^* \cdot PK_O + r^* \cdot P_2$ and $T_2^* = r^* \cdot PK_S^*$, where r^* is a random value and β^* from H_2 query. Then, $\mathcal{C}$ returns $T_{w_b^*} = (T_1^*, T_2^*)$ to $\mathcal{A}_I$.
2) Guess. Finally, $\mathcal{A}_I$ returns some $b' = \{0, 1\}$ to show $T_{w_b^*}$ is the result of $T_{w_0^*}$ or $T_{w_1^*}$. In the same time, $\mathcal{C}$ chooses (T_1', T_2') and outputs $\frac{e(T_1', PK_S')}{e(x_R' \cdot PK_O', PK_S')^{\beta'} \cdot e(T_2', P_2)}$ as its guess, where β' is a value in the challenge phase.

If $\mathcal{A}_I$'s guess is correct, then it must compute $T_1' = D_R' + x_R' \cdot \beta' \cdot PK_O + r' \cdot P_2$ and $T_2' = r' \cdot PK_S'$ for either $b = 0$ or 1. Therefore, setting $P_{pub} = aP_1$, $Q_R' = bP_1$, and $PK_S' = cP_1$ for some a, b, and c are unknown, $\mathcal{C}$ can compute $e(P_1, P_1)^{abc}$ by $\mathcal{A}_I$ as follows.

$$\begin{aligned}
&\frac{e(T_1', PK_S')}{e(x_R' \cdot PK_O', PK_S')^{\beta'} \cdot e(T_2', P_2)} \\
&= \frac{e(D_R', PK_S') \cdot e(x_R' \cdot \beta' \cdot PK_O, PK_S') \cdot e(r' \cdot P_2, PK_S')}{e(x_R' \cdot PK_O', PK_S')^{\beta'} \cdot e(T_2', P_2)} \\
&= e(D_R', PK_S') = e(P_1, P_1)^{abc}.
\end{aligned}$$

Lemma 4: n the random oracle model and based on the BDH problem, SCF-MCLPEKS$^+$ scheme provides trapdoor indistinguishability under an adaptive chosen keyword attack for any $\mathcal{A}_{II}$ adversary.

Proof: The proof of Lemma 4 is similar to Lemma 2 and Lemma 3. Finally, $\mathcal{A}_{II}$ returns some $b' = \{0, 1\}$ to show $T_{w_b^*}$ is the result of $T_{w_0^*}$ or $T_{w_1^*}$. In the same time, $\mathcal{C}$ chooses (T_1', T_2')

and outputs $\frac{e(T_1', PK_s')}{e(D_R', PK_S') \cdot e(T_2', P_2)}$ as its guess. If $\mathcal{A}_{II}$'s guess is correct, than it must compute $T_1' = D_R' + x_R' \cdot H_2(w_b) \cdot PK_O' + r' \cdot P_2$ and $T_2' = r' \cdot PK_S'$ for either $b = 0$ or 1. Therefore, setting $PK_R' = aP_1$, $H_2(w_b) \cdot PK_O' = bP_1$, and $PK_S' = cP_1$ for some a, b, c are unknown, $\mathcal{C}$ can compute $e(P_1, P_1)^{abc}$ by $\mathcal{A}_{II}$ as follows.

$$\frac{e(T_1', PK_s')}{e(D_R', PK_S') \cdot e(T_2', P_2)}$$
$$= \frac{e(D_R', PK_s') \cdot e(x_R' \cdot H_2(w_b) \cdot PK_O', PK_S') \cdot e(r' \cdot P_2, PK_S')}{e(D_R', PK_S') \cdot e(T_2', P_2)}$$
$$= e(x_R' \cdot H_2(w_b) \cdot PK_O', PK_S') = e(P_1, P_1)^{abc}.$$

According to the results of Lemmas 3 and 4, we can conclude that our enhancement SCF-MCLPEKS$^+$ satisfies trapdoor indistinguishability in Theorem 2.

Theorem 2: In the random oracle model and based on the bilinear Diffie-Hellman (BDH) problem, our enhancement SCF-MCLPEKS$^+$ satisfies trapdoor indistinguishability under an adaptive chosen keyword attack for the two types of adversaries $\mathcal{A}_I$ and $\mathcal{A}_{II}$.

3) SECURE AGAINST OFF-LINE KEYWORD GUESSING ATTACKS

According to the previous studies in [21], [39], they described the linkage between "off-line keyword guessing attacks" and "trapdoor indistinguishability". In Theorem 2, we have shown that our enhancement SCF-MCLPEKS$^+$ satisfies trapdoor indistinguishability. Thus, we can conclude that SCF-MCLPEKS$^+$ is secure against off-line keyword guessing attacks launched by external attacker based on the results in [21], [39].

In other aspect, our SCF-MCLPEKS$^+$ adopts data owner's private key in keyword encryption phase (MCLPKES) and data owner's public key in the trapdoor generation phase. For a malicious server, it cannot generate legal keyword ciphertext and then tests the relationship between the ciphertext and the captured trapdoor. Furthermore, this malicious server cannot launch off-line keyword guessing attacks to captured trapdoor without data owner's private key. Thus, our SCF-MCLPEKS$^+$ is secure against off-line keyword guessing attacks launched by malicious server.

V. PERFORMANCE EVALUATION

In this section, we evaluate the computation performance through a series of experiments.

A. THEORETICAL PERFORMANCE COMPARISONS

First, we implement each of the primitive computation cost and summarized them in TABLE 1. This result is similar to the result given in other literature [40], [44]. The experimental environment is set using the JPBC library [45] and Java 8.0. A *Type A* curve - The curve $y^2 = x^3 + x$ over the field $\mathcal{F}_q$ (where q is 512-bits) is chosen since it is optimized for computation time. On the server side the database PostgreSQL 9.6 is used to record a single table storing <fileID, ownerID, userID, cipher> where "cipher" is a byte array (*bytea*[ ]) type field to store a dynamic size of keywords. The experiment is conducted on a typical PC with CPU Intel i5 @3.20GHz, 4GB ram running Windows 10. We think this configuration is a proper modeling of a data owner and a data user. Depends on the scale of the service, a lightweight small scale server might take this as a baseline configuration.

TABLE 1. Notations.

Notation	Definition	Times
TG_e	The execution time of one bilinear pairing operation e	7.704ms
TG_{mul}	The execution time of one scalar multiplication operation in G_1	6.232ms
TG_H	The execution time of one map-to-point hash function	26.708ms
TG_{add}	The execution time of one addition operation in G_1	0.06ms
T_{exp}	The execution time of one modular exponentiation operation	0.047ms
T_h	The execution time of hash function	0.015ms

In TABLE 2, we directly compare the required computation cost between Ma *et al.*'s scheme [40] and our enhancement in terms of the keyword encryption, the trapdoor generation, and the test phases for conducting one of these operations under the assumption that the file contains only one keyword and the trapdoor contains only one keyword. The result shows that our performance is also improving along with the security.

TABLE 2. Performance comparisons.

Phase	Ma et al.'s scheme [40] SCF-MCLPEKS	Our enhancement SCF-MCLPEKS$^+$
Keyword Encryption	$3TG_e + 4TG_{mul} + TG_{add} + 3TG_H + T_h$ 128.239ms	$2TG_e + 2TG_{mul} + TG_{add} + 2TG_H + 2T_{exp} + T_h$ 81.457ms
Trapdoor Generation	$TG_{mul} + TG_H + TG_{add}$ 33ms	$3TG_{mul} + 2TG_{add} + T_h$ 18.831ms
Test	$TG_e + TG_{mul} + 2TG_H + 2TG_{add} + T_h$ 67.487ms	$2TG_e + TG_{add} T_{exp}$ 15.483ms
Security	Vulnerable against Off-line keyword guessing attacks	Provably secure

B. THE SCALABILITY OF OUR ENHANCEMENT

A comprehensive experiment has been conducted to study the scalability of protocol where database I/O is also considered as a factor. In a baseline setting we assume each data owner has a file to shared with three data users. Each file contains five keywords. Three testing keywords will be issued by a

data user on each query. We study the impact of the following variations against the running time: number of keywords labeled in a file, the number of sharees (data users) per each shared file, and the number of testing keywords queried by the users. The results are shown in Fig 2, 3, 4.

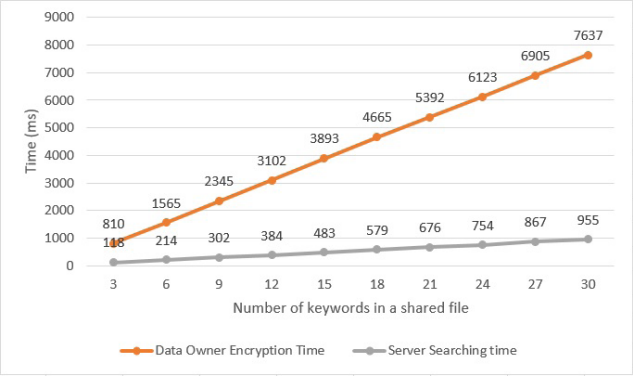


FIGURE 2. Running time of Data owner and the Server against the number of keywords.

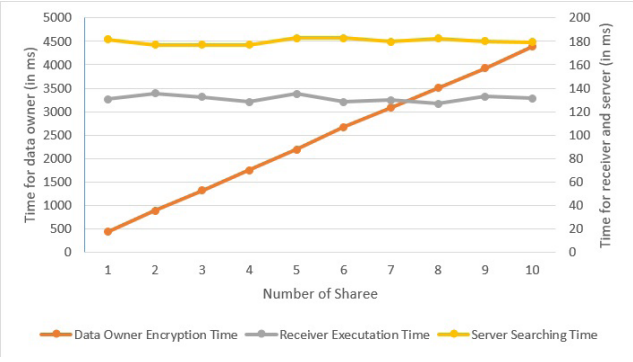


FIGURE 3. Running time against number of data users per each shared file.

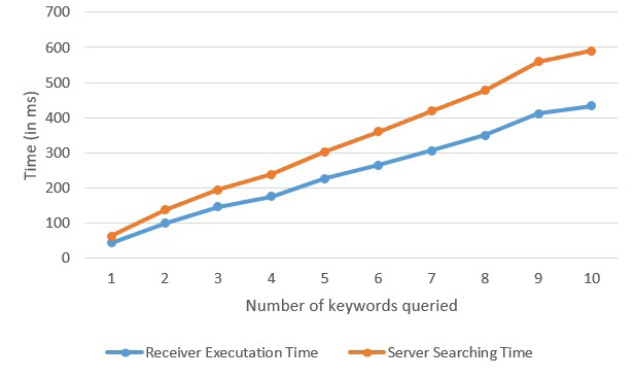


FIGURE 4. Running time of Data owner and the Server against the number of testing keywords queried by a user.

As expected the running time of the data owner and the server grow in linear with the growth of the number of keywords. We can also observe that the computation effort is mainly residing at the client side which is favorable in scalability. In the second experiment we change the number of sharees from 1 to 10 and the running time of the data owner growth linearly while the running time of the server and the owner remains the same. This is expected for the data user since the trapdoor generation does not involve anything about the keyword encryption. With the help of the database, encrypted keywords can be retrieved very fast and cost almost zero overhead on the growth of the number of sharees. In the third experiment the number of testing keywords queried is changed. We again observe that the running time of the server and the data users growth steadily.

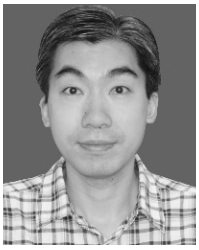
VI. CONCLUSION

In this paper we identify the vulnerability of Ma et al's SCF-MCLPEKS scheme and presented an enhancement SCF-MCLPEKS$^+$ over their scheme. The enhanced scheme is formally proven and data user's privacy can be protected. The performance of SCF-MCLPEKS$^+$ suggested that is practical for some IIoT environments, especially for those has strong need in data storage security.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[2] K.-H. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, "A secure authentication scheme for Internet of Things," *Pervasive Mobile Comput.*, vol. 42, pp. 15–26, Dec. 2017.
[3] K.-H. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, "On the security of a new ultra-lightweight authentication protocol in IoT environment for RFID tags," *J. Supercomput*, vol. 74, no. 1, pp. 65–70, 2018.
[4] J. C.-W. Lin, J. M.-T. Wu, P. Fournier-Viger, Y. Djenouri, C.-H. Chen, and Y. Zhang, "A sanitization approach to secure shared data in an IoT environment," *IEEE Access*, vol. 7, pp. 25359–25368, 2019.
[5] J.-S. Pan, L. Kong, T.-W. Sung, P.-W. Tsai, and V. Snášel, "A clustering scheme for wireless sensor networks based on genetic algorithm and dominating set," *J. Internet Technol.*, vol. 19, no. 4, pp. 1111–1118, 2018.
[6] J.-S. Pan, L. Kong, T.-W. Sung, P.-W. Tsai, and V. Snášel, "α-fraction first strategy for hierarchical model in wireless sensor networks," *J. Internet Technol.*, vol. 19, no. 6, pp. 1717–1726, 2018.
[7] J. Sun, Y. Bao, X. Nie, and H. Xiong, "Attribute-hiding predicate encryption with equality test in cloud computing," *IEEE Access*, vol. 6, pp. 31621–31629, 2018.
[8] H. Xiong, H. Zhang, and J. Sun, "Attribute-based privacy-preserving data sharing for dynamic groups in cloud computing," *IEEE Syst. J.*, to be published.
[9] J. C.-W. Lin, L. Yang, P. Fournier-Viger, and T.-P. Hong, "Mining of skyline patterns by considering both frequent and utility constraints," *Eng. Appl. Artif. Intell.*, vol. 77, pp. 229–238, Jan. 2019.
[10] J. M.-T. Wu *et al.*, "Applying an ensemble convolutional neural network with Savitzky–Golay filter to construct a phonocardiogram prediction model," *Appl. Soft Comput.*, vol. 78, pp. 29–40, May 2019.
[11] J. M.-T. Wu, J. C.-W. Lin, P. Fournier-Viger, Y. Djenouri, C.-H. Chen, and Z. Li, "The density-based clustering method for privacy-preserving data mining," *Math. Biosci. Eng.*, vol. 16, no. 3, pp. 1718–1728, 2019.
[12] W. Gan, J. C.-W. Lin, P. Fournier-Viger, H.-C. Chao, and P. S. Yu, "HUOPM: High-utility occupancy pattern mining," *IEEE Trans. Cybern.*, to be published.
[13] J. C.-W. Lin, Y. Zhang, B. Zhang, P. Fournier-Viger, and Y. Djenouri, "Hiding sensitive itemsets with multiple objective optimization," *Soft Comput.*, pp. 1–19, Feb. 2019. doi: 10.1007/s00500-019-03829-3.
[14] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelfflé, "Vision and challenges for realising the Internet of Things," in *Cluster of European Research Projects on the Internet of Things*. Brussels, Belgium: European Commision, 2010.
[15] H. Xiong, Q. Mei, and Y. Zhao, "Efficient and provably secure certificateless parallel key-insulated signature without pairing for IoT environments," *IEEE Syst. J.*, to be published.
[16] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. EUROCRYPT*, vol. 3027. Berlin, Germany: Springer, 2004, pp. 506–522.

[17] D. J. Park, K. Kim, and P. J. Lee, "Public key encryption with conjunctive field keyword search," in *Proc. Int. Workshop Inf. Secur. Appl.* Berlin, Germany: Springer, 2004, pp. 73–86.
[18] Y. H. Hwang and P. J. Lee, "Public key encryption with conjunctive keyword search and its extension to a multi-user system," in *Pairing-Based Cryptography–Pairing*. Berlin, Germany: Springer, 2007, pp. 2–22.
[19] J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," in *Computational Science and its Applications*. Berlin, Germany: Springer, 2008, pp. 1249–1259.
[20] H. S. Rhee, W. Susilo, and H.-J. Kim, "Secure searchable public key encryption scheme against keyword guessing attacks," *IEICE Electron. Exp.*, vol. 6, no. 5, pp. 237–243, 2009.
[21] H. S. Rhee, J. H. Park, W. Susilo, and D. H. Lee, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," *J. Syst. Softw.*, vol. 83, no. 5, pp. 763–771, 2010.
[22] J. W. Byun, H. S. Rhee, H.-A. Park, and D. H. Lee, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in *Proc. Workshop Secure Data Manage.* Berlin, Germany: Springer, 2006, pp. 75–83.
[23] H. S. Rhee, J. H. Park, W. Susilo, and D. H. Lee, "Improved searchable public key encryption with designated tester," in *Proc. 4th Int. Symp. Inf., Comput., Commun. Secur.*, 2009, pp. 376–379.
[24] W. Bingjian, C. Tzungher, and J. Fuhgwo, "Security improvement against malicious server's attack for a dPEKS scheme," *Int. J. Inf. Educ. Technol.*, vol. 1, no. 4, pp. 350–353, 2011.
[25] C. Hu and P. Liu, "An enhanced searchable public key encryption scheme with a designated tester and its extensions," *J. Comput.*, vol. 7, no. 3, pp. 716–723, 2012.
[26] W.-C. Yau, R. C.-W. Phan, S.-H. Heng, and B.-M. Goi, "Keyword guessing attacks on secure searchable public key encryption schemes with a designated tester," *Int. J. Comput. Math.*, vol. 90, no. 12, pp. 2581–2587, 2013.
[27] C.-T. Li, C.-C. Lee, C.-Y. Weng, T.-Y. Wu, and C.-M. Chen, "Cryptanalysis of 'an efficient searchable encryption against keyword guessing attacks for shareable electronic medical records in cloud-based system'," in *Proc. Int. Conf. Inf. Sci. Appl.* Singapore: Springer, 2017, pp. 282–289.
[28] T.-Y. Wu, C. Meng, C.-M. Chen, K.-H. Wang, and J.-S. Pan, "On the security of a certificateless public key encryption with keyword search," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.* Cham, Switzerland: Springer, 2017, pp. 191–197.
[29] T.-Y. Wu, C. Meng, K.-H. Wang, C.-M. Chen, and J.-S. Pan, "Comments on recent proposed Cui Et Al.'s KASE and Lu Et Al.'s dIBEKS schemes," *J. Inf. Hiding Multimedia Signal Process.*, vol. 9, no. 1, pp. 162–169, 2018.
[30] T.-Y. Wu *et al.*, "Security analysis of Rhee et al.'s public encryption with keyword search schemes: A review," *J. Netw. Intell.*, vol. 3, no. 1, pp. 16–25, Feb. 2018.
[31] T.-Y. Wu, C.-M. Chen, K.-H. Wang, J. M.-T. Wu, and J.-S. Pan, "Security analysis of a public key authenticated encryption with keyword search scheme," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.* Cham, Switzerland: Springer, 2018, pp. 178–183.
[32] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Advances in Cryptology*. Berlin, Germany: Springer, 2003, pp. 452–473.
[33] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 2001, pp. 213–229.
[34] T.-Y. Wu and Y.-M. Tseng, "An efficient user authentication and key exchange protocol for mobile client–server environment," *Comput. Netw.*, vol. 54, no. 9, pp. 1520–1530, 2010.
[35] T.-Y. Wu and Y.-M. Tseng, "An ID-based mutual authentication and key exchange protocol for low-power mobile devices," *Comput. J.*, vol. 53, no. 7, pp. 1062–1070, 2010.
[36] T.-Y. Wu, T.-T. Tsai, and Y.-M. Tseng, "Efficient searchable ID-based encryption with a designated server," *Ann. Telecommun.*, vol. 69, nos. 7–8, pp. 391–402, 2014.
[37] P. Yanguo, C. Jiangtao, P. Changgen, and Y. Zuobin, "Certificateless public key encryption with keyword search," *China Commun.*, vol. 11, no. 11, pp. 100–113, Nov. 2014.
[38] M. Ma, D. He, M. K. Khan, and J. Chen, "Certificateless searchable public key encryption scheme for mobile healthcare system," *Comput. Elect. Eng.*, vol. 65, pp. 413–424, Jan. 2018.
[39] T.-Y. Wu, C.-M. Chen, K.-H. Wang, C. Meng, and E. K. Wang, "A provably secure certificateless public key encryption with keyword search," *J. Chin. Inst. Eng.*, vol. 42, no. 1, pp. 20–28, 2019.
[40] M. Ma, D. He, N. Kumar, K.-K. R. Choo, and J. Chen, "Certificateless searchable public key encryption scheme for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 759–767, Feb. 2018.
[41] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *Int. J. Inf. Secur.*, vol. 6, no. 4, pp. 213–241, 2007.
[42] Q. Huang and H. Li, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," *Inf. Sci.*, vols. 403–404, pp. 1–14, Sep. 2017.
[43] H. Xiong, Y. Zhao, L. Peng, H. Zhang, and K.-H. Yeh, "Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing," *Future Gener. Comput. Syst.*, vol. 97, pp. 453–461, Aug. 2019.
[44] Y. Rouselakis and B. Waters, "Efficient statically-secure large-universe multi-authority attribute-based encryption," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Berlin, Germany: Springer, 2015, pp. 315–332.
[45] A. De Caro and V. Iovino, "jPBC: Java pairing based cryptography," in *Proc. 16th IEEE Symp. Comput. Commun. (ISCC)*, Kerkyra, Greece, Jun./Jul. 2011, pp. 850–855. [Online]. Available: http://gas.dia.unisa.it/projects/jpbc/



TSU-YANG WU received the Ph.D. degree from the Department of Mathematics, National Changhua University of Education, Taiwan. He is currently an Assistant Professor with the Harbin Institute of Technology at Shenzhen. He is also an Associate Professor with the College of Information Science and Engineering, Fujian University of Technology, China. His research interests include cryptography and network security. He serves as an Executive Editor of the *Journal of Network Intelligence* (JNI) and as an Associate Editor of *Data Science and Pattern Recognition* (DSPR).

CHIEN-MING CHEN received the Ph.D. degree from National Tsing Hua University, Taiwan. He is currently an Associate Professor with the College of Computer Science and Technology, Shandong University of Science and Technology, Shandong, China. His current research interests include network security, the mobile Internet, wireless sensor networks, and cryptography.

KING-HANG WANG received the Ph.D. degree from National Tsing Hua University, Taiwan. He was a Lecturer with the Hong Kong Institute of Technology, in 2010. In 2015, he joined the Hong Kong University of Science and Technology for teaching. His research interests include cryptography, mobile security, and steganography.

JIMMY MING-TAI WU received the Ph.D. degree in computer science and engineering from National Sun Yat-sen University, Kaohsiung, Taiwan. He was a Research Scholar with the Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, and with the Department of Computer Science, College of Engineering, University of Nevada at Las Vegas, and an Assistant Professor with the Harbin Institute of Technology at Shenzhen, China. He is currently an Assistant Professor with the College of Computer Science and Engineering, Shandong University of Science and Technology, China. He worked in an IC design company in Taiwan as a firmware developer and an information technology manager in two years. His current research interests include big data, cloud computing, and the Internet of Things (IoT).