3-2018

An efficient and privacy-preserving biometric identification scheme in cloud computing

Liehuang ZHU

Chuan ZHANG

Chang XU

Ximeng LIU
Singapore Management University, xmliu@smu.edu.sg

Cheng HUANG

Citation

ZHU, Liehuang; ZHANG, Chuan; XU, Chang; LIU, Ximeng; and HUANG, Cheng. An efficient and privacy-preserving biometric identification scheme in cloud computing. (2018). *IEEE Access*. 6, 19025-19033.
Available at: https://ink.library.smu.edu.sg/sis_research/4863

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 14, 2018, accepted March 15, 2018, date of publication March 26, 2018, date of current version April 23, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2819166

An Efficient and Privacy-Preserving Biometric Identification Scheme in Cloud Computing

LIEHUANG ZHU[1], (Member, IEEE), CHUAN ZHANG[1], CHANG XU[1], XIMENG LIU[2], (Member, IEEE), AND CHENG HUANG[3]

[1]School of Computer Science and Technology, Beijing Institute of Technology, Beijing 10081, China
[2]School of Information Systems, Singapore Management University, Singapore 178902
[3]Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

Corresponding author: Chang Xu (xuchang@bit.edu.cn).

This work was supported in part by the National Natural Science Foundation of China under Grant 61402037 and Grant 61272512 and in part by the Guangxi Cooperative Innovation Center of Cloud Computing and Big Data under Grant YD16E14.

ABSTRACT Biometric identification has become increasingly popular in recent years. With the development of cloud computing, database owners are motivated to outsource the large size of biometric data and identification tasks to the cloud to get rid of the expensive storage and computation costs, which, however, brings potential threats to users' privacy. In this paper, we propose an efficient and privacy-preserving biometric identification outsourcing scheme. Specifically, the biometric To execute a biometric identification, the database owner encrypts the query data and submits it to the cloud. The cloud performs identification operations over the encrypted database and returns the result to the database owner. A thorough security analysis indicates that the proposed scheme is secure even if attackers can forge identification requests and collude with the cloud. Compared with previous protocols, experimental results show that the proposed scheme achieves a better performance in both preparation and identification procedures.

INDEX TERMS Biometric identification, data outsourcing, privacy-preserving, cloud computing.

I. INTRODUCTION

Biometric identification has raised increasingly attention since it provides a promising way to identify users. Compared with traditional authentication methods based on passwords and identification cards, biometric identification is considered to be more reliable and convenient [1]. Additionally, biometric identification has been widely applied in many fields by using biometric traits such as fingerprint [2], iris [3], and facial patterns [4], which can be collected from various sensors [5]–[9].

In a biometric identification system, the database owner such as the FBI who is responsible to manage the national fingerprints database, may desire to outsource the enormous biometric data to the cloud server (e.g., Amazon) to get rid of the expensive storage and computation costs. However, to preserve the privacy of biometric data, the biometric data has to be encrypted before outsourcing. Whenever a FBI's partner (e.g., the police station) wants to authenticate an individual's identity, he turns to the FBI and generates an identification query by using the individual's biometric traits (e.g., fingerprints, irises, voice patterns, facial patterns etc.). Then, the FBI encrypts the query and submits it to the cloud to find the close match. Thus, the challenging problem is how to design a protocol which enables efficient and privacy-preserving biometric identification in the cloud computing.

A number of privacy-preserving biometric identification solutions [10]–[17] have been proposed. However, most of them mainly concentrate on privacy preservation but ignore the efficiency, such as the schemes based on homomorphic encryption and oblivious transfer in [10] and [11] for fingerprint and face image identification respectively. Suffering from performance problems of local devices, these schemes are not efficient once the size of the database is larger than 10 MB. Later, Evans *et al.* [12] presented a biometric identification scheme by utilizing circuit design and ciphertext packing techniques to achieve efficient identification for a larger database of up to 1GB. Additionally, Yuan and Yu [13] proposed an efficient privacy-preserving biometric identification scheme. Specifically, they constructed three modules and designed a concrete protocol to achieve the security of fingerprint trait. To improve the efficiency, in their scheme, the database owner outsources identification matching tasks

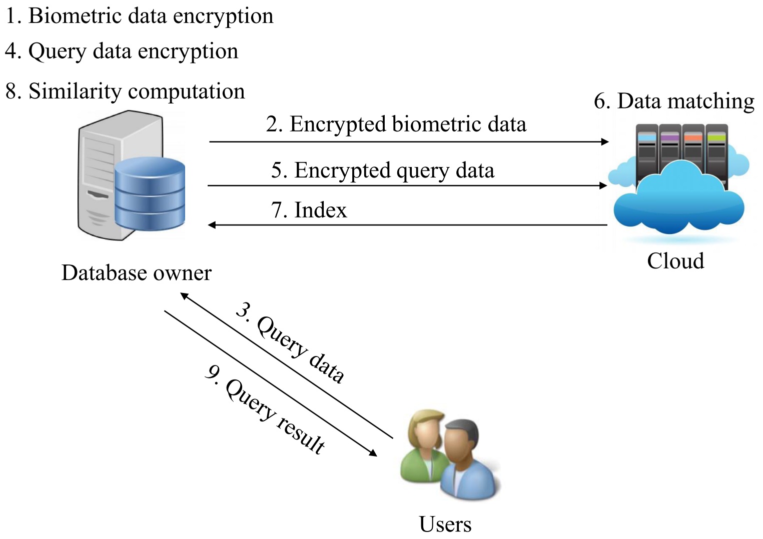


FIGURE 1. System model.

to the cloud. However, Zhu *et al.* [18] pointed out that Yuan and Yu's protocol can be broken by a collusion attack launched by a malicious user and cloud. Wang *et al.* [14] proposed the scheme CloudBI-II which used random diagonal matrices to realize biometric identification. However, their work was proven insecure in [15] and [16].

In this paper, we propose an efficient and privacy-preserving biometric identification scheme which can resist the collusion attack launched by the users and the cloud. Specifically, our main contributions can be summarized as follows:

- We examine the biometric identification scheme [13] and show its insufficiencies and security weakness under the proposed level-3 attack. Specifically, we demonstrate that the attacker can recover their secret keys by colluding with the cloud, and then decrypt the biometric traits of all users.
- We present a novel efficient and privacy-preserving biometric identification scheme. The detailed security analysis shows that the proposed scheme can achieve a required level of privacy protection. Specifically, our scheme is secure under the biometric identification outsourcing model and can also resist the attack proposed by [18].
- Compared with the existing biometric identification schemes, the performance analysis shows that the proposed scheme provides a lower computational cost in both preparation and identification procedures.

The remainder of this paper is organized as follows: section II presents the models and design goals. In section III, we provide an overview and the security analysis of the previous protocol proposed by Yuan and Yu. In section IV, we present an efficient and privacy-preserving biometric identification scheme. Security analysis is presented in section V, followed by performance evaluation in section VI. In section VII, we give the related work and we show our conclusions in section VIII.

II. MODELS AND DESIGN GOALS

This section introduces the system model, attack model, design goals and the notations used in the following sections.

A. SYSTEM MODEL

As shown in Fig.1, three types of entities are involved in the system including the database owner, users and the cloud. The database owner holds a large size of biometric data (i.e., fingerprints, irises, voice, and facial patterns etc.), which is encrypted and transmitted to the cloud for storage. When a user wants to identify himself/herself, a query request is be sent to the database owner. After receiving the request, the database owner generates a ciphertext for the biometric trait and then transmits the ciphertext to the cloud for identification. The cloud server figures out the best match for the encrypted query and returns the related index to the database owner. Finally, the database owner computes the similarity between the query data and the biometric data associated with the index, and returns the query result to the user.

In our scheme, we assume that the biometric data has been processed such that its representation can be used to execute biometric match. Without loss of generality, similar to [17] and [18], we target fingerprints and use FingerCodes [19] to represent the fingerprints. More specifically, a FingerCode consists of n elements and each element is a l-bit integer (typically $n = 640$ and $l = 8$). Given two FingerCodes $x = [x_1, x_2, \cdots, x_n]$ and $y = [y_1, y_2, \cdots, y_n]$, if their Euclidean distance is below a threshold ϵ, they are usually considered as a good match, which means the two fingerprints are considered from the same person.

B. ATTACK MODEL

First of all, the cloud server is considered to be "honest but curious" as described in [13]–[15] and [17]. The cloud strictly follows the designed protocol, but makes efforts to reveal privacy from both the database owner and the user. We assume that an attacker can observe all the data stored in the cloud including the encrypted biometric database, encrypted queries and matching results. Moreover, the attacker can act as a user to construct arbitrary queries.

Thus, we categorize the attack model into three levels as follows:

- Level 1: Attackers can only observe the encrypted data stored in the cloud. This follows the well-known ciphertext-only attack model [20].
- Level 2: In addition to the encrypted data stored in the cloud, attackers are able to get a set of biometric traits in the database D but do not know the corresponding ciphertexts in the database C, which is similar to the known-candidate attack model [21].
- Level 3: Besides all the abilities in level-2, attackers in level-3 can be valid users. Thus, attackers can forge as many identification queries as possible and obtain the corresponding ciphertexts. This attack follows the known-plaintext attack model [20].

A biometric identification scheme is secure if it can resist the level-$\alpha(\alpha \in \{1, 2, 3\})$ attack. Note that that if the proposed scheme can resist level-2 and level-3 attacks, it does not mean that the attacker can both be the valid user and observe some plaintexts of the biometric database simultaneously. This sophisticated attack is too strong and no effective methods is designed to defend against this kind of attack [14]. In this paper, we focus on the collusion attack between a malicious user and the cloud server. The relationship between the plaintexts of the biometric database and the ciphertexts is not known to the attacker, which is similar to the attack model proposed in [14].

C. DESIGN GOALS

In order to achieve practicality, both security and efficiency are considered in the proposed scheme. To be more specific, design goals of the proposed scheme are described as follows:

- Efficiency: Computational costs should be as low as possible at both the database owner side and the user side. To gain high efficiency, most biometric identification operations should be executed in the cloud.
- Security: During the identification process, the privacy of biometric data should be protected. Attackers and the semi-honest cloud should learn nothing about the sensitive information.

D. NOTATIONS

Here, we list the main notations used in the remaining section as follows.

- b_i – the i-th sample FingerCode, denoted as an n-dimensional vector $b_i = [b_{i1}, b_{i2}, \cdots b_{in}]$.
- B_i – the extended sample FingerCode of b_i, denoted as an $(n+1)$-dimensional vector $B_i = [b_{i1}, b_{i2}, \cdots b_{i(n+1)}]$, where $b_{i(n+1)} = -0.5(b_{i1}^2 + b_{i2}^2 + \cdots + b_{in}^2)$.
- b_c – the query FingerCode, denoted as an n-dimensional vector $b_c = [b_{c1}, b_{c2}, \cdots b_{cn}]$.
- B_c – the extended query FingerCode of b_c, denoted as an $(n+1)$-dimensional vector $B_c = [b_{c1}, b_{c2}, \cdots b_{c(n+1)}]$, where $b_{c(n+1)} = 1$.
- W – the secret keys collection, denoted as $W = (M_1, M_2, M_3, H, R)$, where M_1,M_2 and M_3 are $(n+1) \times (n+1)$ invertible matrices, and H, R are $(n+1)$-dimensional row vectors.
- I_i – the searchable index associated with the i-th sample FingerCode b_i.
- Γ – the query FingerCodes collection constructed by the attacker, denoted as $\Gamma = (\widetilde{b}_1, \widetilde{b}_2, \cdots \widetilde{b}_{t+1})$.
- $\widetilde{B}_i$ – the i-th extended query FingerCode constructed by the attacker, denoted as $\widetilde{B}_i = [\widetilde{b}_{i1}, \widetilde{b}_{i2}, \cdots \widetilde{b}_{i(n+1)}]$, where $\widetilde{b}_{i(n+1)} = 1$.

III. SECURITY ANALYSIS OF Yuan AND Yu's SCHEME

In this section, we firstly describe Yuan and Yu's scheme and then give the security analysis about their scheme. To facilitate understanding of the scheme, we use $*$ to denote the elements multiplication operations, and use $\times$ to denote the matrices or vectors multiplication operations.

A. Yuan AND Yu's SCHEME

Step 1: The database owner randomly generates an $(n+1) \times (n+1)$ matrix A where $H \times A_i^T = 1$ and A_i is a row vector in A, $1 \leq i \leq (n+1)$.

Then, the database owner generates a corresponding matrix $D_i = [A_1^T * b_{i1}, A_2^T * b_{i2}, \cdots A_{n+1}^T * b_{i(n+1)}]$ to hide B_i.

After that, the database owner performs the following operations:

$$C_i = M_1 \times D_i \times M_2, \tag{1}$$

$$C_h = H \times M_1^{-1}, \tag{2}$$

$$C_r = M_3^{-1} \times R^T. \tag{3}$$

Subsequently, the database owner uploads (C_i, C_h, C_r, I_i) to the cloud, where I_i is the index of B_i.

Step 2: After Step 1 is executed, the cloud has stored many tuples in its database C. When a user requests to identify his/her identity, he/she extends b_i and then submits the extended query B_i to the database owner. On receiving the request from the user, the database owner generates a random $(n+1) \times (n+1)$ matrix E such that $E_i \times R^T = 1$, where E_i is a row vector in matrix E and $1 \leq i \leq (n+1)$. The database owner then generates a corresponding matrix $F_c = [E_1^T * b_{c1}, E_2^T * b_{c2}, \cdots E_{n+1}^T * b_{c(n+1)}]^T$ to hide the query FingerCode B_c. The Database owner then performs the following operations:

$$C_f = M_2^{-1} \times F_c \times M_3. \tag{4}$$

Then, the database owner uploads C_f to the cloud.

Step 3: On receiving C_f, the cloud begins to search for the best match. Specifically, the cloud computes $P_i = C_h \times C_i \times C_f \times C_r$ for all encrypted biometric database to compare the Euclidean distances between b_c and b_i. Other details are eliminated since they are irrelevant for the security analysis we will describe.

B. SECURITY ANALYSIS OF Yuan AND Yu's SCHEME

In level-3 attack, an attacker has the ability to select query FingerCodes Γ of his/her interest as inputs and then tries to recover the privacy of B_i. Specifically, the attacker can compute the secret key M_2 by performing the following equation:

$$\begin{aligned} C_f \times C_r &= M_2^{-1} \times F_c \times M_3 \times M_3^{-1} \times R^T \\ &= M_2^{-1} \times F_c \times R^T \\ &= M_2^{-1} \times B_c^T. \end{aligned} \tag{5}$$

In equation 5, C_f is an $(n+1) \times (n+1)$ matrix and C_r is an $(n+1)$-dimensional vector which are both known to the attacker. B_c is an $(n+1)$-dimensional vector which can be constructed by the attacker. M_2^{-1} is one of the secret keys which is an $(n+1)\times(n+1)$ matrix but unknown to the attacker. Let S be $C_f \times C_r$. To recover M_2^{-1}, t query FingerCodes $\Gamma = [\widetilde{b}_1, \widetilde{b}_2, \cdots \widetilde{b}_t]$ which are extended to $[\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T]$ can be constructed, such that

$$[S_1, S_2, \cdots S_t] = M_2^{-1} \times [\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T]. \tag{6}$$

There are $(n+1) \times t$ known elements in $[S_1, S_2, \cdots S_t]$ and $(n+1) \times t$ known elements in $[\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T]$, M_2^{-1} is a matrix with $(n+1) \times (n+1)$ unknown elements. Suppose $M_2^{-1} = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1(n+1)} \\ q_{21} & q_{22} & \cdots & q_{2(n+1)} \\ \vdots & \vdots & \ddots & \vdots \\ q_{(n+1)1} & q_{(n+1)2} & \cdots & q_{(n+1)(n+1)} \end{bmatrix}$, we will show how to recover M_2^{-1} by constructing special FingerCodes.

For the first row vector $q_1 = [q_{11}, q_{12}, \cdots, q_{1(n+1)}]$ in M_2^{-1}, the adversary constructs two special vectors as $\widetilde{B}_1^T = [1, 0, \cdots, -0.5]$, and $\widetilde{B}_2^T = [2, 0, \cdots, -2]$. Then, the attacker can compute as follows:

$$\begin{cases} 1 * q_{11} - 0.5 * q_{1(n+1)} = S_{11}, \\ 2 * q_{11} - 2 * q_{1(n+1)} = S_{21}. \end{cases} \tag{7}$$

From equation 7, it is easy to compute q_{11} and $q_{1(n+1)}$. Following the same analysis, the attacker can obtain all the elements in M_2^{-1} by constructing other special vectors.

After recovering M_2^{-1}, the attacker can compute the biometric data as follows:

$$\begin{aligned} C_h \times C_i &= H \times M_1^{-1} \times M_1 \times D_i \times M_2 \\ &= H \times D_i \times M_2 \\ &= B_i \times M_2. \end{aligned} \tag{8}$$

In equation 8, C_h and C_i are known by the attacker. M_2 is the secret key which is recovered by the above foregoing. Therefore, the attacker can recover B_i.

IV. A NOVEL BIOMETRIC IDENTIFICATION SCHEME

In this section, we show the details of the proposed biometric identification scheme.

A. OVERVIEW

We construct a novel biometric identification scheme to address the weakness of Yuan and Yu's scheme [13]. To achieve a higher level of privacy protection, a new retrieval way is constructed to resist the level-3 attack. Moreover, we also reconstruct the ciphertext to reduce the amount of uploaded data and improve the efficiency both in the preparation and identification procedures.

In the remaining part of this section, we will introduce the preparation process and the identification process.

B. PREPARATION PROCESS

In the preparation process, b_i is the i-th sample feature vector derived from the fingerprint image using a feature extraction algorithm [19]. To be more specific, b_i is an n-dimensional vector with l bits of each element where $n = 640$ and $l = 8$.

For ease of identification, b_i is extended by adding an $(n+1)$-th element as B_i. Then, the database owner encrypts B_i with the secret key M_1 as follows:

$$C_i = B_i \times M_1. \tag{9}$$

The database owner further performs the following operation:

$$C_h = M_2^{-1} \times H^T. \tag{10}$$

Each FingerCode B_i is associated with an index I_i. After execute the encryption operations, the database owner uploads (C_i, C_h, I_i) to the cloud.

C. IDENTIFICATION PROCESS

The identification process includes the following steps:

Step 1: When a user has a query fingerprint to be identified, he/she first gets the query FingerCode b_c derived from the query fingerprint image. The FingerCode b_c is also an n-dimensional vector. Then, the user sends b_c to the database owner.

Step 2: After receiving b_c, the database owner extends b_c to B_c by adding an $(n+1)$-th element equals to 1. Then the database owner randomly generates an $(n+1)\times(n+1)$ matrix E. The i-th row vector $E_i = [E_{i1}, E_{i2}, \cdots E_{i(n+1)}]$ is set as a random vector, where the $(n+1)$-th element is $(1 - \sum_{j=1}^{n} E_{ij} * H_j)/H_{n+1}$, $1 \leq i \leq (n+1)$. After that, the database owner performs the following computation to hide B_c:

$$F_c = [E_1^T * b_{c1}, E_2^T * b_{c2}, \cdots E_{(n+1)}^T * b_{c(n+1)}]^T. \tag{11}$$

To securely send F_c to the cloud, the database owner needs to encrypt F_c with the secret keys and a random integer $r (r > 0)$. The computation is performed as follows:

$$C_f = M_1^{-1} \times r \times F_c \times M_2. \tag{12}$$

Then, the database owner sends C_f to the cloud for identification.

Step 3: After receiving C_f from the database owner, the cloud begins to search the FingerCode which has the minimum Euclidean distance with the query FingerCode B_c.

P_i denotes the *relative distance* between B_i and B_c as follows:

$$\begin{aligned} P_i &= C_i \times C_f \times C_h \\ &= B_i \times M_1 \times M_1^{-1} \times r \\ &\quad \times F_c \times M_2 \times M_2^{-1} \times H^T \\ &= B_i \times r \times F_c \times H^T \\ &= \sum_{j=1}^{n+1} r * b_{ij} * b_{cj}. \end{aligned} \tag{13}$$

In equation 13, the computation result is an integer, which can be used to compare two FingerCodes. For example, to compare the query b_c with two FingerCodes, say b_i and b_z, the cloud computes P_i and P_z, and performs the following operation, where $1 \leq i, z \leq t, i \neq z$:

$$\begin{aligned} P_i - P_z &= \sum_{j=1}^{n+1} r * b_{ij} * b_{cj} - \sum_{j=1}^{n+1} r * b_{zj} * b_{cj} \\ &= (\sum_{j=1}^{n} r * b_{ij} * b_{cj} - 0.5 \sum_{j=1}^{n} r * b_{ij}^2) \\ &\quad -(\sum_{j=1}^{n} r * b_{zj} * b_{cj} - 0.5 \sum_{j=1}^{n} r * b_{zj}^2) \\ &= 0.5r(dist_{zc}^2 - dist_{ic}^2). \end{aligned} \tag{14}$$

As shown in equation 14, if $P_i - P_z > 0$, the cloud learns that b_i matches the query FingerCode much better than b_z. After repeating the operations for the encrypted FingerCode database C in the cloud, the ciphertext C_i which has the minimum Euclidean distance with b_c can be found. The cloud further gets the corresponding index I_i according to the tuple (C_i, C_h, I_i) and sends it back to the database owner.

Step 4: After receiving the index I_i, the database owner gets the corresponding sample FingerCode b_i in the database D and calculates the accurate Euclidean distance between b_i and b_c as $dist_{ic} = \sqrt{\sum_{j=1}^{n}(b_{ij} - b_{cj})^2}$. Then, the database owner compares the Euclidean distance with the standard threshold. If the distance is less than the threshold value, the query is identified. Otherwise, the identification fails.

Step 5: Finally, the database owner returns the identification result to the user.

V. SECURITY ANALYSIS

In this part, we first prove that our scheme is secure under level-2 and level-3 attacks, and then we will show the proposed scheme can resist the attack proposed by Zhu *et al.* [18].

A. SECURITY ANALYSIS UNDER LEVEL-2 ATTACK

According to the attack scenario 2, an attacker can obtain some plaintexts of the biometric database, but does not know the corresponding ciphertexts.

We consider C_i which is obtained by multiplying B_i and M_1. Since the mapping relationship between B_i and C_i is not known, it is impossible for the attacker to compute B_i and M_1.

B. SECURITY ANALYSIS UNDER LEVEL-3 ATTACK

In the level-3 attack, besides the knowledge of encrypted data in the cloud, the attacker can forge a large number of query FingerCodes Γ as inputs. In the following, we will show the proposed scheme is secure by proving that the secret keys cannot be recovered.

When colluding with the cloud, the attacker gets C_f and C_h, and then performs the following operation:

$$\begin{aligned} C_f \times C_h &= M_1^{-1} \times r \times F_c \times M_2 \times M_2^{-1} \times H^T \\ &= M_1^{-1} \times r \times F_c \times H^T \\ &= M_1^{-1} \times r \times B_c^T. \end{aligned} \tag{15}$$

In equation 15, since r is a positive random integer in identification process, the attacker cannot compute the secret key M_1^{-1} directly.

Pretending a valid user, the attacker can construct t query FingerCodes $\Gamma = [\widetilde{b}_1, \widetilde{b}_2, \cdots \widetilde{b}_t]$ extended as $[\widetilde{B}_1, \widetilde{B}_2, \cdots \widetilde{B}_t]$ for identification, which introduces a set of positive random values r_j and C_{fj}, $1 \leq j \leq t$. Let $\widetilde{P}_j$ be the value of $C_{fj} \times C_h$. The attacker computes $\widetilde{P}_j$ as follows:

$$\widetilde{P}_j = M_1^{-1} \times r_j \times \widetilde{B}_j^T. \tag{16}$$

After constructing t equations, we have:

$$\begin{aligned} \widetilde{P} &= M_1^{-1} \times [\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T] \times \begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_t \end{bmatrix} \\ &= M_1^{-1} \times \widetilde{B} \times R. \end{aligned} \tag{17}$$

Here $[\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T]$ is denoted as $\widetilde{B}$, $\begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_t \end{bmatrix}$ is denoted as R. In this equation, $\widetilde{P}$ is an $(n+1) \times t$ matrix known to the attacker, $\widetilde{B}$ is an $(n+1) \times t$ matrix constructed by the attacker, R is an $t \times t$ matrix, since r_j is a random positive integer, it is unknown to the attacker.

We then demonstrate that the attacker cannot recover M_1 according to **Theorem 1.**

Theorem 1: Assume after t equations are constructed, M_1 cannot be computed in $\widetilde{P} = M_1^{-1} \times \widetilde{B} \times R$. When $(t+1)$ equations are constructed, the following equation holds, and M_1 cannot be recovered.

$$[\widetilde{P}|\widetilde{P}_{t+1}] = M_1^{-1} \times [\widetilde{B}|\widetilde{B}_{t+1}^T] \times \begin{bmatrix} R & 0 \\ 0 & r_{t+1} \end{bmatrix}. \tag{18}$$

Proof: This theorem is proven with *the inductive method*. When $t = 1$, M_1 cannot be computed in equation 16. Assume the equation 17 holds, where $(t > 1)$. When $(t+1)$ query FingerCodes are constructed, we obtain:

$$[\widetilde{P}, \widetilde{P}_{t+1}] = [M_1^{-1} \times \widetilde{B}, M_1^{-1} \times \widetilde{B}_{t+1}^T] \times \begin{bmatrix} R & 0 \\ 0 & r_{t+1} \end{bmatrix}. \tag{19}$$

For $(t+1)$-th query FingerCode $\widetilde{B}_{t+1}$, we have

$$\widetilde{P}_{t+1} = M_1^{-1} \times \widetilde{B}_{t+1}^T \times r_{t+1}. \tag{20}$$

TABLE 1. Security comparison with other schemes.

Schemes	Level 1 attack	Level 2 attack	Level 3 attack
Yuan and Yu's scheme [13]	Yes	Yes	No
Wang et al.'s scheme [14]	Yes	Yes	No
Our scheme	Yes	Yes	Yes

From equation 20, we have

$$\widetilde{B}_{t+1} \times (M_1^{-1})^T = (r_{t+1}^{-1})^T \times \widetilde{P}_{t+1}^T. \tag{21}$$

Let $(d_1, d_2, \cdots d_{n+1})$ be the vector $(r_{t+1}^{-1})^T \times \widetilde{P}_{t+1}^T$ where $d_j = (r_{t+1}^{-1})^T \times \widetilde{P}_{(t+1)j}^T$. Let $(M_1^{-1})^T = (m_1^T, m_2^T, \cdots, m_{n+1}^T)^T$, where m_j denotes a row vector in M_1^{-1}, $1 \le j \le (n+1)$. The following equations hold:

$$\widetilde{B}_{t+1} \times (m_1^T, m_2^T, \cdots m_{n+1}^T)^T = (d_1, d_2, \cdots, d_{n+1}), \tag{22}$$

$$\widetilde{B}_{t+1} \times m_j^T = d_j. \tag{23}$$

Equation 23 is a typical non-linear homogeneous equation. Since the rank of $\widetilde{B}_{t+1}$ is $r(\widetilde{B}_{t+1})$, we assume the result is $\alpha_1\beta_1 + \alpha_2\beta_2 + \cdots + \alpha_{(n-r(\beta_{t+1}))}\beta_{(n-r(\beta_{t+1}))}$. We further state the special solution of equation 23 is β^* which satisfies the formula $\widetilde{B}_{t+1} \times m_j^T = d_j$. Because $d_j = (r_{t+1}^{-1})^T \times \widetilde{P}_j^T$, $(r_{t+1}^{-1})^T$ is included in the special solution β^*. For m_j^T in matrix $(M_1^{-1})^T$, the particular solution of m_j^T is $\alpha_1\beta_1 + \alpha_2\beta_2 + \cdots + \alpha_{((n-r(\beta_{t+1})))}\beta_{((n-r(\beta_{t+1})))} + \beta^*$. Since r is a random integer, the special solution β^* is uncertain as well, which means the attacker cannot derive the exact particular solution for m_j^T in $(M_1^{-1})^T$. ■

Therefore, when $(t+1)$ query FingerCodes are constructed, the secret key M_1 cannot be computed by the attacker as well.

As discussed above, the attacker cannot recover the secret key even if he is a malicious user. Therefore, the attacker cannot recover the biometric data as well.

Moreover, we compare our scheme with the schemes proposed in [13] and [14]. According to Table 1, other schemes have some weaknesses, while our scheme is secure under all the three level attacks

C. SECURITY ANALYSIS UNDER THE ATTACK PROPOSED BY Zhu et al.

Zhu *et al.* [18] showed an attack for Yuan and Yu's scheme. In their attack, the attacker observes the cloud and gets the values of *relative distance*. According to the equation 1, 2, 3, 4, the *relative distance* in Yuan and Yu's scheme can be computed as follows:

$$\begin{aligned} P_i &= C_h \times C_i \times C_f \times C_r \\ &= H \times M_1^{-1} \times M_1 \times D_i \times M_2 \\ &\quad \times M_2^{-1} \times F_c \times M_3 \times M_3^{-1} \times R^T \\ &= H \times D_i \times F_c \times R^T \\ &= \sum_{j=1}^{n+1} b_{ij} * b_{cj} \\ &= B_i \times B_c^T. \end{aligned} \tag{24}$$

As shown in equation 24, P_i is an integer which the attacker can get in the cloud, B_c is the extended query FingerCode which can be constructed by the attacker pretending to be a user. B_i is the extended sample FingerCode which is sensitive and should not be leaked. To recover B_i, the attacker can construct t query FingerCodes $\Gamma = [\widetilde{b}_1, \widetilde{b}_2, \ldots \widetilde{b}_t]$ extended as $[\widetilde{B}_1, \widetilde{B}_2, \cdots \widetilde{B}_t]$ for identification. $\widetilde{P}_{ij}$ denotes the *relative distance* between the sample FingerCode B_i and the query FingerCode $\widetilde{B}_j$ where $1 \le j \le t$. Then, the attacker has:

$$[\widetilde{P}_{i1}, \widetilde{P}_{i2}, \cdots \widetilde{P}_{it}] = [b_{i1}, b_{i2}, \cdots b_{i(n+1)}] \times [\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T]. \tag{25}$$

In this equation, $\widetilde{P}_{ij}$ and $\widetilde{B}_j$ are known to the attacker. For each element in B_i, it can be recovered if t equations are built, where $t > (n+1)$.

Then, we demonstrate the proposed scheme is secure under the attack proposed by Zhu *et al.* In the proposed scheme, $\widetilde{P}_{ij}$ is set as the *relative distance* between B_i and $\widetilde{B}_j$.

$$\begin{aligned} \widetilde{P}_{ij} &= C_i \times C_{fj} \times C_h \\ &= r_j \times B_i \times \widetilde{B}_j^T. \end{aligned} \tag{26}$$

r_j is the j-th positive random integer in t identification processes. The attacker constructs t query FingerCodes and gets the equation as follows:

$$\begin{aligned} [\widetilde{P}_{i1}, \widetilde{P}_{i2}, \cdots \widetilde{P}_{it}] &= [b_{i1}, b_{i2}, \cdots b_{i(n+1)}] \\ &\quad \times [r_1\widetilde{B}_1^T, r_2\widetilde{B}_2^T, \cdots r_t\widetilde{B}_t^T] \\ &= [b_{i1}, b_{i2}, \cdots b_{i(n+1)}] \times [\widetilde{B}_1^T, \widetilde{B}_2^T, \cdots \widetilde{B}_t^T] \\ &\quad \times \begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_t \end{bmatrix}. \end{aligned} \tag{27}$$

In this equation, r_j is a positive random integer which is unknown to the attacker. For every element in B_i, after t computations, the attacker can only get the value of $r_j * b_{iq}$ where $t > (n+1)$, $1 \le q \le (n+1)$. For the reason that r_j is a random integer, $r_j * b_{iq}$ is also unexpected which means the attacker cannot acquire B_i. Thus, the proposed scheme can resist the attack proposed by Zhu *et al.*

VI. PERFORMANCE ANALYSIS

To evaluate the performance of the proposed scheme, we implement a cloud-based privacy-preserving fingerprint identification system. For the cloud, we use 2 nodes with 6-core 2.10 GHz Intel Xeous CPU and 32GB memory. We utilize a laptop with an Intel Core 2.40 GHz CPU and 8G. Similar to [13] and [14], the query FingerCodes are randomly

TABLE 2. A summary of complexity costs. In the table, *m* denotes the number of FingerCodes in the biometric database; *n* ≪ *m*.

		Phases	Yuan and Yu's scheme [13]	Wang et al.'s scheme [14]	Our scheme
Computation	Database owner	Preparation	$O(mn^3)$	$O(mn^3)$	$O(mn^2)$
		Identification	$O(n^3)$	$O(n^3)$	$O(n^3)$
		Retrieval	$O(n)$	$O(n)$	$O(n)$
	Cloud server	Identification	$O(mn^2 + m\log m)$	$O(mn^3 + m\log m)$	$O(mn^2 + m\log m)$
	User	Identification	/	/	/
Communication	Database owner	Preparation	$O(mn^2)$	$O(mn^2)$	$O(mn)$
		Identification	$O(n^2)$	$O(n^2)$	$O(n^2)$
		Retrieval	$O(1))$	$O(1)$	$O(1)$
	Cloud server	Identification	/	/	/
		Retrieval	$O(1)$	$O(1)$	$O(1)$
	User	Identification	$O(1)$	$O(1)$	$O(1)$

selected from the database which is constructed with random 640-entry vectors.

A. COMPLEXITY ANALYSIS

Table 2 summarizes the computation and communication costs on the data owner side, cloud server and users in our scheme and the schemes in [13] and [14]. In this work, each matrix multiplication costs $O(n^3)$, where n denotes the dimension of a FingerCode, and the sorting cost of fuzzy Euclidean distances has time complexity of $O(m \log m)$. As illustrated in Table 2, our scheme has lower complexities in the preparation phase. That is, more computation and bandwidth costs can be saved for the database owner. In the identification phase, the computation complexity of our scheme is lower than that in [14]. The reason is that our scheme performs vector-matrix multiplication operations to find the close match, while [14] needs to execute matrix-matrix multiplication operations. Although the complexity of our scheme is the same as that in [13], we emphasize that [13] sacrifices the substantial security to achieve such fast computation of P_i. Moreover, our scheme executes fewer multiplication operations, and thus obtains better performance.

B. EXPERIMENTAL EVALUATION

1) PREPARATION PHASE

Fig. 2 and Fig. 3 show the computation and communication costs in the preparation phase with the number of FingerCodes varying from 1000 to 5000. As shown in Fig.2, in our scheme, registering 5000 FingerCodes needs 29.37s, which can save about 88.85% and 90.58% time cost compared with [13] and [14] respectively. The reason is when encrypting a sample FingerCode, in our scheme, only one matrix is needed which leads to fewer matrix multiplication operations. Fig. 3 shows the bandwidth costs of the three schemes. Since the data outsourced to the cloud is in the form of vectors in comparison with matrices in the other two schemes, the communication cost in our scheme is much less than [13], [14].

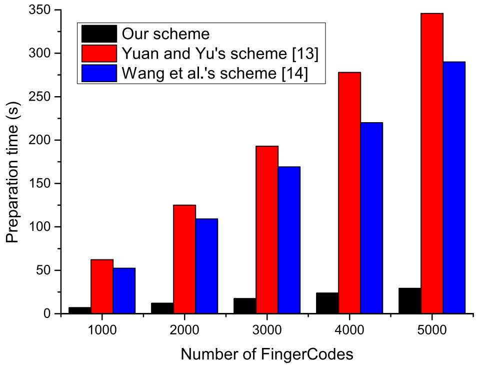


FIGURE 2. Time costs in the preparation phase.

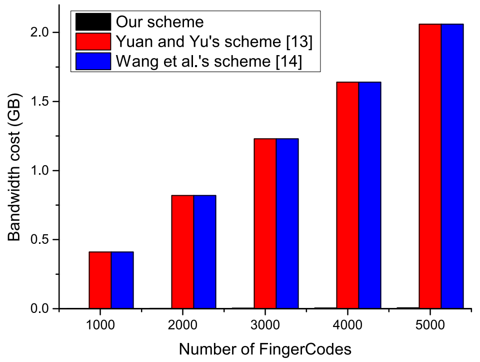


FIGURE 3. Bandwidth costs in the preparation phase.

2) IDENTIFICATION PHASE

Fig.4 and Fig. 5 show the computation and communication costs in the identification phase with the number of FingerCodes ranges from 1000 to 5000. As demonstrated in Fig. 4, all schemes grow linearly as the size of database increases. As in our scheme fewer matrix multiplication operations are used than [13], it can save about 56% time cost. Compared with [14], the identification time can be saved as much as 84.75%, since the vector-matrix multiplication rather than

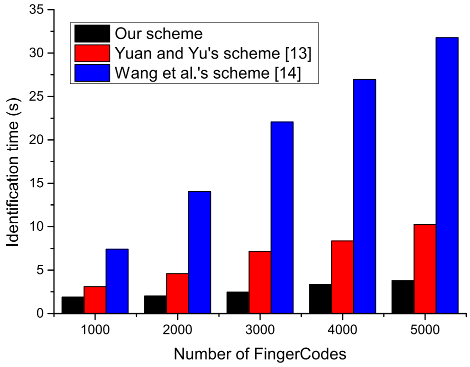


FIGURE 4. Time costs in the identification phase.

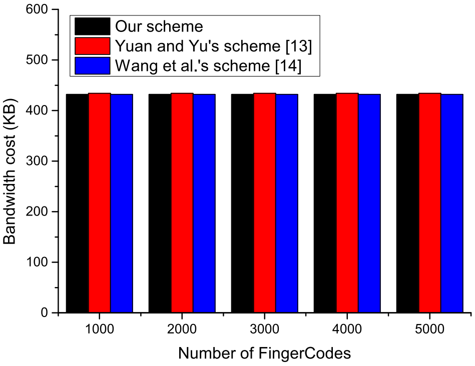


FIGURE 5. Bandwidth costs in the identification phase.

the matrix-matrix multiplication operation is executed. The bandwidth costs of the three schemes, as shown in Fig. 5, are almost the same. The reason is that all schemes need to transmit a matrix in the identification phase.

VII. RELATED WORKS

Related works on privacy-preserving biometric identification are provided in this section. Recently, some efficient biometric identification schemes have been proposed. Wang and Hatzinakos proposed a privacy-preserving face recognition scheme [22]. Specifically, a face recognition method is designed by measuring the similarity between sorted index numbers vectors. Wong and Kim [23] proposed a privacy-preserving biometric matching protocol for iris codes verification. In their protocol, it is computationally infeasible for a malicious user to impersonate as an honest user. Barni *et al.* [10] presented a FingerCode identification protocol based on the Homomorphic Encryption technique. However, all distances are computed between the query and sample Fingercodes in the database, which introduces too much burden as the size of fingerprints increases. To improve the efficiency, Evans *et al.* [12] proposed a novel protocol which reduces the identification time. They used an improved Homomorphic encryption algorithm to compute the Euclidean distance and designed novel garbled circuits to find the minimum distance. By exploiting a backtracking protocol, the best match FingerCode can be found. However, in [12], the whole encrypted database has to be transmitted to the user from the database server. Wong *et al.* [24] proposed an identification scheme based on kNN to achieve secure search in the encrypted database. However, their scheme assumes that there is no collusion between the client side and cloud server side. Yuan and Yu [13] proposed an efficient privacy-preserving biometric identification scheme. However, Zhu *et al.* [18] pointed out their protocol can be broken if a malicious user colludes with the cloud server in the identification process. Based on [13], Wang *et al.* presented a privacy-preserving biometric identification scheme in [14] which introduced random diagonal matrices, named CloudBI-II. However, their scheme has been proven insecure in [15] and [16]. Recently, Zhang *et al.* [17] proposed an efficient privacy-preserving biometric identification scheme by using perturbed terms.

VIII. CONCLUSION

In this paper, we proposed a novel privacy-preserving biometric identification scheme in the cloud computing. To realize the efficiency and secure requirements, we have designed a new encryption algorithm and cloud authentication certification. The detailed analysis shows it can resist the potential attacks. Besides, through performance evaluations, we further demonstrated the proposed scheme meets the efficiency need well.

REFERENCES

[1] A. Jain, L. Hong, and S. Pankanti, "Biometric identification," *Commun. ACM*, vol. 43, no. 2, pp. 90–98, 2000.

[2] R. Allen, P. Sankar, and S. Prabhakar, "Fingerprint identification technology," in *Biometric Systems*. London, U.K.: Springer, 2005, pp. 22–61.

[3] J. de Mira, Jr., H. V. Neto, E. B. Neves, and F. K. Schneider, "Biometric-oriented iris identification based on mathematical morphology," *J. Signal Process. Syst.*, vol. 80, no. 2, pp. 181–195, 2015.

[4] S. Romdhani, V. Blanz, and T. Vetter, "Face identification by fitting a 3D morphable model using linear shape and texture error functions," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 3–19.

[5] Y. Xiao *et al.*, "A survey of key management schemes in wireless sensor networks," *Comput. Commun.*, vol. 30, nos. 11–12, pp. 2314–2341, Sep. 2007.

[6] X. Du, Y. Xiao, M. Guizani, and H.-H. Chen, "An effective key management scheme for heterogeneous sensor networks," *Ad Hoc Netw.*, vol. 5, no. 1, pp. 24–34, 2007.

[7] X. Du and H. H. Chen, "Security in wireless sensor networks," *IEEE Wireless Commun. Mag.*, vol. 15, no. 4, pp. 60–66, Aug. 2008.

[8] X. Hei and X. Du, "Biometric-based two-level secure access control for implantable medical devices during emergencies," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 346–350.

[9] X. Hei, X. Du, J. Wu, and F. Hu, "Defending resource depletion attacks on implantable medical devices," in *Proc. IEEE GLOBECOM*, Dec. 2010, pp. 1–5.

[10] M. Barni *et al.*, "Privacy-preserving fingercode authentication," in *Proc. 12th ACM Workshop Multimedia Secur.*, 2010, pp. 231–240.

[11] M. Osadchy, B. Pinkas, A. Jarrous, and B. Moskovich, "SCiFI—A system for secure face identification," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2010, pp. 239–254.

[12] D. Evans *et al.*, "Efficient privacy-preserving biometric identification," in *Proc. 17th Conf. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2011, pp. 1–40.

[13] J. Yuan and S. Yu, "Efficient privacy-preserving biometric identification in cloud computing," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2652–2660.

[14] Q. Wang, S. Hu, K. Ren, M. He, M. Du, and Z. Wang, "CloudBI: Practical privacy-preserving outsourcing of biometric identification in the cloud," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2015, pp. 186–205.

[15] Y. Zhu, Z. Wang, and J. Wang, "Collusion-resisting secure nearest neighbor query over encrypted data in cloud," in *Proc. IEEE/ACM 24th Int. Symp. Quality Ser. (IWQoS)*, Jun. 2016, pp. 1–6.

[16] S. Pan, S. Yan, and W. Zhu, "Security analysis on privacy-preserving cloud aided biometric identification schemes," in *Proc. Australasian Conf. Inf. Secur. Privacy*, 2016, pp. 446–453.

[17] C. Zhang, L. Zhu, and C. Xu, "PTBI: An efficient privacy-preserving biometric identification based on perturbed term in the cloud," *Inf. Sci.*, vols. 409–410, pp. 56–67, Oct. 2017.

[18] Y. Zhu, T. Takagi, and R. Hu, "Security analysis of collusion-resistant nearest neighbor query scheme on encrypted cloud data," *IEICE Trans. Inf. Syst.*, vol. E97.D, no. 2, pp. 326–330, 2014.

[19] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, May 2000.

[20] H. Delfs, H. Knebl, and H. Knebl, *Introduction to Cryptography*. Berlin, Germany: Springer, 2002.

[21] K. Liu, C. Giannella, and H. Kargupta, "An attacker's view of distance preserving maps for privacy preserving data mining," in *Knowledge Discovery in Databases: PKDD* (Lecture Notes in Computer Science). Heidelberg, Germany: Springer, 2006, pp. 297–308.

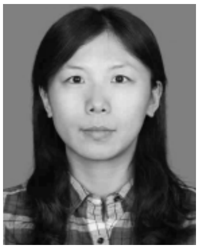
[22] Y. Wang and D. Hatzinakos, "Face recognition with enhanced privacy protection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 885–888.

[23] K.-S. Wong and M.-H. Kim, "A privacy-preserving biometric matching protocol for iris codes verification," in *Proc. 3rd FTRA Int. Conf. Mobile, Ubiquitous, Intell. Comput. (MUSIC)*, Jun. 2012, pp. 120–125.

[24] W. K. Wong, D. W.-L. Cheung, B. Kao, and N. Mamoulisa, "Secure kNN computation on encrypted databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2009, pp. 139–152.

LIEHUANG ZHU (M'16) received the Ph.D. degree in computer science from the Beijing Institute of Technology, Beijing, China, in 2004. He is currently a Professor with the School of Computer Science and Technology, Beijing Institute of Technology. His research interests include security protocol analysis and design, group key exchange protocol, wireless sensor network, and cloud computing.

CHUAN ZHANG received the bachelor's degree in network engineering from the Dalian University of Technology, Dalian, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Beijing Institute of Technology. His current research interests include secure data services in cloud computing, security and privacy in VANET, and big data security.



CHANG XU received the bachelor's and master's degrees from the School of Computer Science and Technology, Jilin University, in 2005 and 2008, respectively, and the Ph.D. degree in computer science from Beihang University in 2013. She is currently an Assistant Professor with the School of Computer Science and Technology, Beijing Institute of Technology. Her current research interests include security and privacy in VANET and big data security.

XIMENG LIU (S'13–M'16) received the B.Sc. degree in electronic engineering and the Ph.D. degree in cryptography from Xidian University, Xi'an, China, in 2010 and 2015, respectively. He is currently a Research Fellow at the School of Information System, Singapore Management University, Singapore, and a Qishan Scholar with the College of Mathematics and Computer Science, Fuzhou University. He has authored over 80 research articles, including the IEEE TIFS, the IEEE TDSC, the IEEE TC, the IEEE TSC, and the IEEE TCC. His research interests include cloud security, applied cryptography, and big data security.

CHENG HUANG received the B.Eng. and M.Eng. degrees from Xidian University, China, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. He was a Project Officer with the INFINITUS Laboratory, School of Electrical and Electronic Engineering, Nanyang Technological University, until 2016. His current research interests include applied cryptography, cyber security, and privacy.