KICS The Korean Institute of Communications and Information Sciences

Available online at www.sciencedirect.com

ScienceDirect

ICT Express 2 (2016) 135–139

www.elsevier.com/locate/icte

Efficient and privacy-preserving biometric identification in cloud☆

Changhee Hahn, Junbeom Hur*

Department of Computer Science and Engineering, Korea University, Republic of Korea

Received 28 July 2016; accepted 14 August 2016
Available online 24 August 2016

Abstract

With the rapid growth in the development of smart devices equipped with biometric sensors, client identification system using biometric traits are widely adopted across various applications. Among many biometric traits, fingerprint-based identification systems have been extensively studied and deployed. However, to adopt biometric identification systems in practical applications, two main obstacles in terms of efficiency and client privacy must be resolved simultaneously. That is, identification should be performed at an acceptable time, and only a client should have access to his/her biometric traits, which are not revocable if leaked. Until now, multiple studies have demonstrated successful protection of client biometric data; however, such systems lack efficiency that leads to excessive time utilization for identification. The most recently researched scheme shows efficiency improvements but reveals client biometric traits to other entities such as biometric database server. This violates client privacy. In this paper, we propose an efficient and privacy-preserving fingerprint identification scheme by using cloud systems. The proposed scheme extensively exploits the computation power of a cloud so that most of the laborious computations are performed by the cloud service provider. According to our experimental results on an Amazon EC2 cloud, the proposed scheme is faster than the existing schemes and guarantees client privacy by exploiting symmetric homomorphic encryption. Our security analysis shows that during identification, the client fingerprint data is not disclosed to the cloud service provider or fingerprint database server.

Keywords: Privacy; Biometrics; Identification; Cloud

1. Introduction

Biometric identification is one of the most prominent methods for identifying an individual. All biometric traits, such as fingerprint, iris, and retina, share the important factors of universality (people have their own fingerprint), uniqueness (the probability that two persons have the same fingerprint is negligible), and permanence (biometric traits usually do not change over time) [1]. Such properties have certain pros and cons. Although they make the usage of biometric traits easy and client identification precise, they raise concerns for client privacy. For example, suppose Alice identifies herself using her fingerprint to access some web services, such as a health-care service and social network service (SNS), the service providers may track the transmission of her fingerprint and discern her private information including health condition and registered SNS identity. This severely violates client privacy. Furthermore, if Alice's fingerprint data is revealed to the public, anyone can masquerade as Alice by simply submitting her fingerprint data, thereby invalidating the entire identification system [2,3]. As biometric traits are unique and cannot be changed during the lifetime, once leaked, they cannot be revoked and re-generated.

Recently, several studies [4,5] have proposed privacy-preserving fingerprint identification systems, which use an asymmetric homomorphic encryption algorithm to encrypt the fingerprint data so that only key owners can access their fingerprints. Although the systems guarantee privacy-preserving identification, the computation cost of the encryption algorithm is considerable. Thus, they are not scalable considering the growing number of clients.

Yuan et al. [6] introduced an efficiency-improved fingerprint identification scheme that exploits matrix operations to encrypt fingerprint data, thus avoiding heavy computations compared with the schemes using the asymmetric encryption algorithm.

* Corresponding author.
E-mail addresses: hahn850514@korea.ac.kr (C. Hahn), jbhur@korea.ac.kr (J. Hur).

Peer review under responsibility of The Korean Institute of Communications Information Sciences.

☆ This paper is part of a special issue entitled ICT Convergence in the Internet of Things (IoT) guest edited by Yacine Ghamri-Doudane, Yeong Min Jang, Daeyoung Kim, Hossam Hassanein and JaeSeung Song.

http://dx.doi.org/10.1016/j.icte.2016.08.006

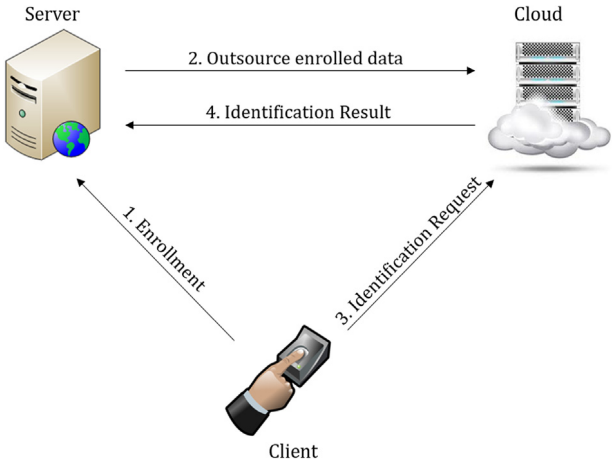

Fig. 1. The system model of the proposed scheme.

Further, most computations are shifted from a server onto a third-party entity (e.g., a cloud) to exploit their resources. Nevertheless, there is a trade-off between efficiency and privacy, that is, they must assume that the server has the authority to access the fingerprint database. We stress that such assumption must be alleviated considering irrevocability of biometric data. For example, a malicious employee with access to the fingerprint database may sell a copy of the data to someone, or the server could be compromised, thus rendering the data unrecoverable.

In this paper, we propose an efficient privacy-preserving fingerprint-based identification scheme. Compared with Yuan et al.'s scheme [6], our scheme exploits a symmetric homomorphic encryption algorithm in biometric identification, thus achieving both security and efficiency. We allow the server to outsource most computations to a cloud to save storage cost and improve efficiency so that fingerprints are secure.

2. System description

2.1. System model

The proposed scheme consists of three entities: a client, data server (a server for short), and cloud (see Fig. 1). The client encrypts and enrolls his/her fingerprint. For identification, the client encrypts and sends a newly scanned fingerprint to the cloud. Note that the previous key used to enroll is not re-used but a fresh key is generated at every identification to encrypt the fingerprint.

We adopt a filterbank-based fingerprint matching system [7], which is also used in other biometric identification schemes. This system [7] promises high accuracy by using FingerCode: a chain of N-independent feature codes that are typically 8-bit integers. This is used to measure the Euclidean distance between two fingerprints.

2.2. Threat model

We assume that attackers reside outside the system and attempt to eavesdrop on the data sent from a client. The goal of these attackers is to obtain a client's raw biometric data, in this case, the fingerprint. The attackers can then bypass the identification process and successfully access the data server. As mentioned earlier, biometric traits are incapable of being revoked when leaked. Therefore, it is important that the biometric data is secured from attackers.

We define the cloud as an honest-but-curious entity, implying that it behaves properly in most cases but attempts to harvest biometric information. In addition, we postulate that the cloud may collude with an outside adversary to recover a client's fingerprint data to gain illegal profits. We assume that the data server is also curious about fingerprint data. A data server providing service to a client does not necessarily imply that it is allowed to access the client's fingerprint data.

2.3. Design goal

Our goal is threefold. First, during enrollment and identification, fingerprint data should not be revealed to any entities including the server and the cloud. Next, the proposed scheme should be able to filter out malicious clients who submit random values similar to legitimate clients' FingerCodes. Lastly, the identification regarding computation and communication should be efficient.

3. The proposed scheme

3.1. Preliminaries

Let $Enc(\cdot)$ be a homomorphic encryption function. Then, for any given encryption key k, the encryption function satisfies $Enc_k(m_1 \lhd m_2) \longleftarrow Enc_k(m_1) \rhd Enc_k(m_2)$, for some operators $\lhd$ and $\rhd$ on input messages m_1 and m_2. The encryption scheme is said to be additively homomorphic if the following equation holds, given two encrypted messages, $Enc_{k_1}(m_1)$ and $Enc_{k_2}(m_2)$:

$$Enc_{(k_1+k_2)}(m_1+m_2) = Enc_{k_1}(m_1) + Enc_{k_2}(m_2)\,. \tag{1}$$

We use a secret key $k = [k_1, \ldots, k_N]$ to encrypt the FingerCode $m = [x_1, \ldots, x_N]$ as follows:

$$Enc_{k_i}(x_1) = x_1 + k_1 \bmod M, \tag{2}$$

$$\vdots$$

$$Enc_{k_N}(x_N) = x_N + k_N \bmod M, \tag{3}$$

where M is a randomly-chosen large integer satisfying $0 \leq x_i < M$.

3.2. Initial client enrollment

The first client has FingerCode $m_1 = [x_{11}, \ldots, x_{1N}]$ and a random key $k_1 = [k_{11}, \ldots, k_{1N}]$. He encrypts m_1 such that $Enc_{k_1}(m_1) = [x_{11} + k_{11} \bmod M, \ldots, x_{1N} + k_{1N} \bmod M]$. The client then sends the encrypted file to the server, which re-encrypts the file by using the key pair (k_{s1}, k'_{s1}). Note that neither key is permanent, instead they are used only for the enrollment of the first client. The server computes $Enc_{k_{s1}+k'_{s1}}(0)$ and re-encrypts $Enc_{k_1}(m_1)$ to generate $Enc_{k_1+k_{s1}+k'_{s1}}(m_1)$,