Journal of Information Security and Applications 35 (2017) 106–118

Contents lists available at ScienceDirect

Journal of Information Security and Applications

journal homepage: www.elsevier.com/locate/jisa

Sequence aware functional encryption and its application in searchable encryption

Tran Viet Xuan Phuong[a,*], Guomin Yang[a], Willy Susilo[a], Fuchun Guo[a], Qiong Huang[b]

[a] School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia
[b] College of Mathematics and Informatics, South China Agricultural University, China

ARTICLE INFO

Article history:

Keywords:
Functional encryption
Hamming distance
Privacy preserving
Searchable encryption

ABSTRACT

As a new broad vision of public-key encryption systems, functional encryption provides a promising solution for many challenging security problems such as expressive access control and searching on encrypted data. In this paper, we present two Sequence Aware Function Encryption (SAFE) schemes. Such a scheme is very useful in many forensics applications where the order (or pattern) of the attributes forms an important characteristic of an attribute sequence. Our first scheme supports the matching of two bit strings, while the second scheme can support the matching of general characters. These two schemes are constructed based on the standard Decision Linear and Decision Bilinear Diffie-Hellman assumptions. In addition, we show that our SAFE schemes can also provide the additional feature of attribute-hiding, which is desirable in forensics applications. Moreover, we give an interesting application of SAFE schemes in constructing Sequential Aware Keyword Search (SAKS) schemes.

1. Introduction

Functional encryption, which is a general term for a range of new public-key encryption systems including Identity-based Encryption [1,6,7,19,21], Attribute-Based Encryption [4,5,11], Predicate Encryption [8,13,14,22], and Inner-Product Encryption [3,16,17], gives a new way of viewing encryption. It has provided a promising solution for many challenging security problems such as expressive access control and searchable encryption, which traditional public-key encryption cannot solve.

This paper focuses on a special type of functional encryption named *Sequence Aware Functional Encryption (SAFE)*. The essential property of a SAFE scheme is that a ciphertext *CT* encrypted under a string *V* can be decrypted by a secret key corresponding to another string *X* if and only if *V* and *X* have a ‘similar’ pattern. For example, if *V* is a DNA sequence ‘GACTCAGT’, then both “GACTCACC’ and “TTCTCAGT” can be considered similar if we allow different symbols to appear in at most two positions. The term “Sequence Aware” comes from that fact that the order of the symbols in the sequences forms a key attribute in measuring the similarity.

SAFE schemes can be defined and constructed based on different similarity metrics such as Hamming Distance, Edit Distance, Longest Common Subsequence, etc. In this paper, we focus on the constructions based on the Hamming Distance and propose two SAFE schemes that can support matching test between binary strings and general character strings, respective. As user privacy is another important issue that must be addressed in many forensics applications, privacy-preserving is another essential requirement in the design of a SAFE scheme. In SAFE, this means if the decryption sequence does not match the encryption sequence, then no information about the encrypted data or the encryption sequence (except the fact of mismatch) should be revealed.

Similar to the “Fuzzy Identity-Based Encryption” proposed by Sahai and Waters [19], a SAFE scheme naturally implies a ‘fault-tolerant” IBE, and hence can be used in the application of IBE with biometric-based identity. Since biometric data such as fingerprint or DNA sequence is very sensitive personal information, it is desirable that this information associated with a ciphertext is hidden from any party who cannot successfully decrypt the ciphertext. Therefore, to implement an IBE using biometric-based identity, it is more preferred to use a SAFE.

Our Contribution. Our goal is to define and construct secure SAFE schemes under standard assumptions. More precisely, this paper contributes towards the above goal via the following steps.

- We provide a formal definition and security model for SAFE schemes. In particular, the security model can capture the property of attribute-hiding (i.e., privacy-preserving).

[*] Corresponding author.
E-mail addresses: tvxp750@uowmail.edu.au (T.V. Xuan Phuong), gyang@uow.edu.au (G. Yang), wsusilo@uow.edu.au (W. Susilo), fuchun@uow.edu.au (F. Guo), csqhuang-c@my.cityu.edu.hk (Q. Huang).

http://dx.doi.org/10.1016/j.jisa.2017.06.002
2214-2126/© 2017 Elsevier Ltd. All rights reserved.

- We first present a SAFE construction (named SAFE-1) for binary strings. Then we address the problem of building a SAFE scheme for general character strings. For both schemes, the decryption can be processed successfully if and only if the Hamming Distance of two bit/character strings is equal to a value k associated with a user's key. In order to allow the matching test in our SAFE-2 scheme, we also propose a novel technique for computing the Hamming distance between two DNA strings including four characters {*A, T, G, C*}. Extensively, we show that our scheme can be used the general characters (in this case we use English characters) for the computation of Hamming distance between two strings. We also prove the security of our SAFE-1 and SAFE-2 schemes under two standard assumptions – the Decision Linear assumption and Decision Bilinear Diffie-Hellman assumption.
- In addition, we extend our SAFE-2 scheme to a Sequence Aware Keyword Search (SAKS) scheme, which allows a fuzzy matching test between two (encrypted) DNA strings.

Related Work. A SAFE scheme may look very similar to the "Fuzzy Identity-Based Encryption" proposed by Sahai and Waters [19]. In a Fuzzy IBE, a ciphertext created under an ID ω' can be decrypted by a private key corresponding to another ID ω if the set-overlap between ω and ω' is above a threshold. Sahai and Waters also mentioned that a Fuzzy IBE based on the Hamming Distance can be built by following a similar technique. However, the Sahai and Waters Fuzzy IBE requires the encryption identity to be attached with the ciphertext, and therefore does not satisfy the property of attribute-hiding. Cheung et al. [9] later proposed another Fuzzy IBE scheme with the attribute-hiding property. However, their construction is under the composite-order group. According to the results in [10,12], pairing in composite-order group is much more expensive (between 10 times to 192 times slower) than that in prime-order group. Compared with [9], both of our SAFE schemes are under prime-order group, and hence are more efficient.

Another work that is closely related to ours is the Wildcarded IBE (or WIBE for short) proposed by Abdalla et al. in [1,2]. A wildcarded IBE allows wildcard symbols to appear in an identity used in the encryption process, and the wildcard positions will be ignored when measuring the equality of two identity strings. It is easy to see that a WIBE scheme can also be considered as a SAFE scheme where the similarity metric is defined via an equality test supporting wildcard symbols. However, different from the hamming distance, in WIBE, the non-wildcard symbols at the same position of the two sequences must be identical.

In a predicate encryption system [8,14,13,22] for a (polynomial-time) predicate P, two inputs (besides some public parameters) are required in the encryption process, one is the message M to be encrypted, and the other one is an index string i. A decryption key is generated based on a master secret and a key index k. The decryption key can successfully decrypt a valid encryption of (i, M) if and only if $P(k, i) = 1$. In this sense, a SAFE scheme can also be treated as a predicate encryption scheme where the predicate P is the similarity between i and k.

As one type of predicate encryption, Hidden Vector Encryption (HVE) schemes [15,18,20] can also be treated as SAFE schemes based on equality test supporting wildcard symbols. However, in HVE schemes, wildcard symbols will appear in the attribute string associated with the user secret key rather than that of the ciphertext. Recently, Waters [23] proposed functional encryption for regular languages. Using the terminology of predicate encryption, the key index k is a Deterministic Finite Automata (DFA), while the index key i is an input for the DFA. The message can be recovered if and only if i is accepted by k.

Organization of the paper: In the next section, we present the definition, security models, and some preliminaries that will be used in the rest of the paper. The SAFE-1 scheme for bit string is then presented in Section 3, followed by the SAFE-2 scheme for general character strings in Section 4. We then give the security proofs for both schemes in Section 5. The Sequence Aware Keyword Search (SAKS) scheme is presented in Section 6. We compare and discuss in Section 7. The paper is concluded in Section 8.

2. Preliminaries

2.1. SAFE Scheme

Let $\vec{v}, \vec{x}$ be two vectors over a finite alphabet Σ and have the same length n. We define a predicate $F = \{f_{\vec{v}} | \vec{v} \in \Sigma\}$ based on the Hamming Distance of two vectors such that $f_{\vec{v}}(\vec{x}) = 1$ iff Hamming Distance $(\vec{v}, \vec{x}) = k$ where $0 \leq k \leq n$ is a fixed integer.

Definition 1. A SAFE scheme based on Hamming Distance is a probabilistic polynomial-time algorithms which has four algorithms as follows:

- **Setup**(1^λ, n) on input a security parameter 1^λ and the vector length $n = poly(\lambda)$, the algorithm outputs a public key PK and a master secret key MSK.
- **Encrypt**($M, PK, \vec{v} = (v_1, v_2, \ldots, v_n)$): on input a message M, the public key PK, and a vector $\vec{v} \in \Sigma^n$, it outputs a ciphertext CT.
- **KeyGen**($MSK, \vec{x} = (x_1, x_2, \ldots, x_n), k$): on input the master secret key MSK, a vector $\vec{x} \in \Sigma$, and an integer $k \leq n$, the algorithm outputs a secret key SK.
- **Decrypt**(CT, SK): on input a secret key SK (w.r.t. a vector $\vec{x}$) and a ciphertext CT (w.r.t. a vector $\vec{v}$), if $f_{\vec{v}}(\vec{x}) = 1$ (i.e. Hamming Distance $(\vec{v}, \vec{x}) = k$), the algorithm outputs a message M; otherwise, it outputs $\perp$.

2.2. *Security model for a SAFE scheme*

Definition 2. A SAFE scheme based on Hamming Distance is selectively secure if for any PPT adversary $\mathcal{A}$, its advantage defined in the following interactive game with a challenger $\mathcal{B}$ is negligible in the security parameter λ.

1. **Init:** $\mathcal{A}$ outputs two vectors $\vec{v}, \vec{x} \in \Sigma$ and an integer k.
2. **Setup:** The challenger $\mathcal{B}$ runs setup algorithm to generate PK and MSK, then sends PK to $\mathcal{A}$.
3. **Query Phase 1**: $\mathcal{A}$ can adaptively request keys for any vector $\vec{y} \in \Sigma$ with the following constrain
   - Hamming Distance $(\vec{v}, \vec{y}) = k$ if and only if Hamming Distance $(\vec{x}, \vec{y}) = k$.

   $\mathcal{B}$ responds to $\mathcal{A}$ with $SK \leftarrow KeyGen(MSK, \vec{y}, k)$.
4. **Challenge**: $\mathcal{A}$ outputs two messages M_0, M_1 with equal length. If $M_0 \neq M_1$, then it is required that Hamming Distance $(\vec{v}, \vec{y}) \neq k \neq$ Hamming Distance $(\vec{x}, \vec{y})$ for any $\vec{y}$ appeared in Query Phase 1. $\mathcal{B}$ flips a random coin $b \in \{0, 1\}$. If $b = 0$, $\mathcal{B}$ returns $CT \leftarrow Encrypt(PK, \vec{v}), M_0$ to $\mathcal{A}$; otherwise, if $b = 1$, $\mathcal{B}$ returns $CT \leftarrow Encrypt(PK, \vec{x}), M_1$ to $\mathcal{A}$.
5. **Query Phase 2**: Phase 1 is repeatedly.
6. **Guess:**$\mathcal{A}$ outputs a guess bit b' and succeeds if $b' = b$.

The advantage of $\mathcal{A}$ in this game is defined as

$$\mathbf{Adv}_{\mathcal{A}}(\lambda) = \left| \Pr[b' = b] - \frac{1}{2} \right|.$$

Full Security: In the above selective security model, the adversary is required to commit the challenge vectors before seeing the system parameters. In the full security model [16], the adversary can choose the challenge vectors in the Challenge phase, which makes the model stronger.