Future Generation Computer Systems 62 (2016) 148–160

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Enabled/disabled predicate encryption in clouds

Shi-Yuan Huang [a], Chun-I Fan [b,*], Yi-Fan Tseng [b]

[a] CyberTrust Technology Institute, Institute for Information Industry, Taipei, Taiwan

[b] Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, 80424, Taiwan

HIGHLIGHTS

- We model timed-release and data self-destruction functionalities for predicate encryption in clouds.
- A sender can set the readable/unreadable time of the files to be sent to the receiver.
- It also supports long message encryption and undecryptable search.
- The cloud server can obtain only the matched ciphertexts after the search.

ARTICLE INFO

Article history:
Received 1 May 2015
Received in revised form
7 November 2015
Accepted 10 December 2015
Available online 18 December 2015

Keywords:
Predicate encryption
Timed-release
Data self-destruction
Search
Cloud computing

ABSTRACT

Predicate encryption is a cryptographic primitive that provides fine-grained control over access to encrypted data. It is often used for encrypted data search in a cloud storage environment. In this paper, we propose an enabled/disabled predicate encryption scheme, which is the first work that provides timed-release services and data self-destruction (they correspond to the terms "enabled" and "disabled," respectively). Owing to these properties, the sender can set the readable/unreadable time of the files to be sent to the receiver. The receiver can read the sent file only after the readable time. After the unreadable time, the structure of the file will be destroyed and the file will become unreadable. Furthermore, for practical usage purposes, the extended scheme, which is based on the proposed scheme, provides not only timed-release services and data self-destruction but also long message encryption and undecryptable search. In the extended scheme, the length of encrypted messages does not depend on the order of the group. Moreover, the cloud server can obtain only the matched ciphertexts after the search.

1. Introduction

Cloud computing has garnered increasing importance owing to its ability to deliver computing resources anytime and anywhere. Hence, it can be regarded as a service over the Internet. It can also be said that cloud computing entrusts data such as user data, storage, computation, and software to remote servers. However, there exist many security weaknesses or flaws in the cloud computing environment. The establishment of secure cloud computing services is a critical research topic. At present, an increasing number of cryptology methods have been proposed, which can provide different security solutions for cloud computing, such as cloud data integrity checking [1,2], functional encryption [3,4] (it includes predicate encryption [5–8] and attribute-based encryption [9–11]), proxy re-encryption [12,13], fully homomorphic encryption [14,15], and data deduplication [16,17].

Among them, cloud data integrity checking allows data integrity to be checked without a complete download of the cloud data. Functional encryption provides more fine-grained control over access to encrypted data. In proxy re-encryption, the proxy can translate a ciphertext of the delegator to another ciphertext of the same plaintext for the delegatee; however, the proxy cannot know the plaintext. Fully homomorphic encryption allows the cloud server to receive many ciphertexts and perform arbitrarily complex dynamically chosen computations on that data while it remains encrypted despite not having the secret decryption key. Data deduplication is a specialized data compression technique that can eliminate redundant cloud data. Here we focus on functional encryption for private search in the cloud environment.

In functional encryption (FE), a relation (or functionality) $R(x', y')$ has been defined. This relation determines what a private key with a parameter x' can decrypt a ciphertext which has been

* Corresponding author. Tel.: +886 7 5252000x4346; fax: +886 7 5254301.
E-mail addresses: shiyuan.huang@gmail.com (S.-Y. Huang), cifan@faculty.nsysu.edu.tw (C.-I. Fan), yftseng1989@gmail.com (Y.-F. Tseng).

http://dx.doi.org/10.1016/j.future.2015.12.008
0167-739X/

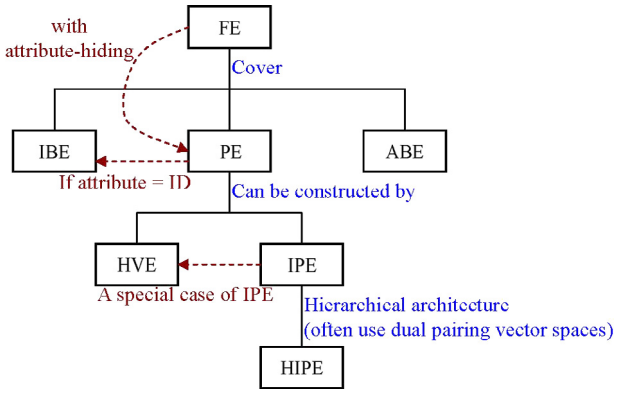

Fig. 1. Relation of functional encryptions.

encrypted under a parameter y'. FE is a generalized type of public key encryption (PKE), which often covers multiple encryptions, such as identity-based encryption (IBE) [18], predicate encryption (PE), and attribute-based encryption (ABE). Furthermore, predicate encryption can be constructed using hidden-vector encryption (HVE) [19–23] or inner-product encryption (IPE) [5–7,24–29].

In 2004, PKE with keyword search (PEKS) was proposed [30]. It is a special case of functional encryption. In PEKS, the relation $R(x', y')$ holds if the keywords x' are equal to the keywords y'. In order to fulfil the requirement of security, Boneh and Waters [19] (2007), and Katz et al. [5] (2008) proposed HVE and PE with "attribute-hiding", respectively. Attribute-hiding is a security notion that is stronger than payload-hiding (i.e., semantic security). It not only requires that the ciphertext hides the message, but also requires that the ciphertext hides all information about the parameter x' (e.g., the attributes). In general, attribute-hiding FE is known as PE. If attributes are assigned equal identities, we can say that PE is IBE, and attribute hiding PE is anonymous IBE.

PE for inner-product predicates (i.e., IPE) can support the disjunctions or conjunctions of equality tests, CNF or DNF formulas [5–7,24–27]. In IPE, the relation $R(x', y')$ holds if the inner product of attributes x' and y' equals zero (i.e., $\langle x', y' \rangle = 0$). HVE is a special case of attribute-hiding IPEs that provides simpler tests than IPE. Okamoto and Takashima used a key technique, called dual pairing vector spaces (DPVS), to construct the hierarchical IPE (HIPE) [24,27]. DPVS is a very useful technique to design the complex framework of IPE, such as HIPE. Further, some related works extended IPE. For example, Wei and Ye [31] extended the construction of Katz et al.'s IPE scheme and applied this extension to anonymous authentication. Fig. 1. shows the relation of functional encryptions.

Our results. This paper proposes a PE scheme for inner-product predicates that provides timed-release services and data self-destruction. The extended scheme, which is based on the proposed scheme, has the aforementioned properties and provides long message encryption and undecryptable search as well.

Encryption with timed-release services allows a sender to encrypt a message, and only the intended recipient can decrypt and read it only after a specified time [32]. Thus far, no timed-release predicate encryption scheme has been published. Geambasu et al. proposed a "Vanish" system, which ensures that all copies of data become unreadable after a specified time, without any specific actions by the user, and even if an attacker obtains a copy of the data, the user's private keys, and passwords [33]. In this paper, we integrated timed-release and data self-destruction properties with a PE scheme that is based on Park's IPE [25].

In our proposed scheme, we can ensure that the cloud data becomes readable after a specified time (it corresponds to the term "enabled"), and unreadable after another time (it corresponds to the term "disabled"). It is to be noted that our scheme has security stronger than the "Vanish" system. The proposed scheme can prevent the attacker from obtaining the entire ciphertext and performing cryptanalysis using any form of attacks (including the brute-force attack) on the ciphertext after the disabled time.

Furthermore, for practical usage purpose, the extended scheme supports long message encryption and undecryptable search. This extension differs from other PE schemes; it can encrypt messages of any arbitrary length. The length of messages does not depend on the order of the group. The extended scheme provides the test and the decryption procedures simultaneously. The cloud server can only obtain the matched ciphertexts; it cannot decrypt the ciphertexts.

Organization. The rest of this paper is organized as follows: Section 2 introduces preliminaries. Section 3 presents the proposed scheme and the extended scheme. Section 4 provides the properties discussion, the computation cost and the security proof. In Section 5, we will conclude the paper and describe the future works.

2. Preliminaries

In this section, we show the setting of bilinear pairing and hard problem assumptions, and discuss the security requirement of the proposed scheme.

2.1. Bilinear pairing setting

Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a non-degenerate bilinear pairing function, with the bilinearity property that $e(x^a, y^b) = e(x, y)^{ab}$ for all $x \in \mathbb{G}_1, y \in \mathbb{G}_2, x \neq 1, y \neq 1, a, b \in \mathbb{Z}_p$, where $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ are cyclic multiplicative groups of order p. It is asymmetric if $\mathbb{G}_1 \neq \mathbb{G}_2$. ψ is a computable isomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$ with $\psi(g_2) = g_1$, where g_1 is a generator of $\mathbb{G}_1$ and g_2 is a generator of $\mathbb{G}_2$. We denote an asymmetric bilinear pairing instance by a 7-tuple $I = [p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e]$. Assume that there exists an efficient generation procedure $\mathcal{G}$ that, on inputting a security parameter 1^k, outputs an instance I where $\log_2(p) = \Theta(k)$.

Decisional BDH problem. The Decisional BDH problem in $(\mathbb{G}_1, \mathbb{G}_2)$ is defined as follows: given a 8-tuple $[g_1, g_1^{z_1}, g_1^{z_2}, g_1^{z_3}, g_2, g_2^{z_1}, g_2^{z_2}, Z] \in \mathbb{G}_1^4 \times \mathbb{G}_2^3 \times \mathbb{G}_T$ for random $[z_1, z_2, z_3] \in \mathbb{Z}_p^3$ as input, output 1 if $Z = e(g_1, g_2)^{z_1 z_2 z_3}$ or 0 otherwise [21,25].

The advantage of an algorithm $\mathcal{A}$ in deciding the Decisional BDH problem in $(\mathbb{G}_1, \mathbb{G}_2)$ is defined below.

$$AdvDBDH_{\mathcal{A}} = \left| \begin{array}{c} Pr[\mathcal{A}(g_1, g_1^{z_1}, g_1^{z_2}, g_1^{z_3}, g_2, g_2^{z_1}, g_2^{z_2}, e(g_1, g_2)^{z_1 z_2 z_3}) = 1 \\ : g_1 \leftarrow \mathbb{G}_1, g_2 \leftarrow \mathbb{G}_2, z_1, z_2, z_3 \leftarrow \mathbb{Z}_p] \\ -Pr[\mathcal{A}(g_1, g_1^{z_1}, g_1^{z_2}, g_1^{z_3}, g_2, g_2^{z_1}, g_2^{z_2}, Z) = 1 \\ : g_1 \leftarrow \mathbb{G}_1, g_2 \leftarrow \mathbb{G}_2, Z \leftarrow \mathbb{G}_T, z_1, z_2, z_3 \leftarrow \mathbb{Z}_p] \end{array} \right|.$$

The probability is over the uniform random choice of the parameters to $\mathcal{A}$, and over the coin tosses of $\mathcal{A}$. We say that an algorithm $\mathcal{A}(t, \epsilon)$-solves the decisional BDH problem in $(\mathbb{G}_1, \mathbb{G}_2)$ if $\mathcal{A}$ runs in time at most t and $AdvDBDH_{\mathcal{A}}$ is at least ε.

Decisional Linear problem. The Decisional Linear problem in $(\mathbb{G}_1, \mathbb{G}_2)$ is defined as follows: given a 9-tuple $[g_1, g_1^{z_1}, g_1^{z_2}, g_1^{z_1 z_3}, g_1^{z_4}, g_2, g_2^{z_1}, g_2^{z_2}, Z] \in \mathbb{G}_1^5 \times \mathbb{G}_2^3 \times \mathbb{G}_1$ for random $[z_1, z_2, z_3, z_4] \in \mathbb{Z}_p^4$ as input, output 1 if $Z = g_1^{z_2(z_3+z_4)}$ or 0 otherwise [25]. The similar variants of the Decisional Linear problem were proposed by [21,20]. The assumption in [20] is a symmetric form, and the assumption in [21] is an asymmetric form.