A Study of Conjunctive Keyword Searchable Schemes

Cheng-Chi Lee[1], Shih-Ting Hsu[2], and Min-Shiang Hwang[3]

(Corresponding author: Min-Shiang Hwang)

Department of Library and Information Science, Fu Jen Catholic University[1]
No. 510, Zhongzheng Rd., Xinzhuang Dist., New Taipei City 24205, Taiwan, R.O.C.
Department of Management Information Systems, National Chung Hsing University[2]
250 Kuo Kuang Road, Taichung 402, Taiwan, R.O.C.
Department of Computer Science and Information Engineering, Asia Universtiy[3]
500 Liufeng Road, Wufeng, Taichung 402, Taiwan, R.O.C.
(Email: mshwang@asia.edu.tw)

(Invited Paper)

Abstract

We study the development of conjunctive keyword searchable scheme which enables one to search encrypted documents by using more than one keyword. The notion of conjunctive keyword searching was presented by Golle *et al.* in 2004. However, their security model was constructed in a symmetric-key setting which is not applicable for the overall applications in the reality. So Park *et al.* extended Golle *et al.*'s security model into a public-key setting which calls the Public Key Encryption with Conjunctive Field Keyword Search (PECKS) scheme. In this paper, we examine six security models by concluding the secret-key setting and public-key setting, and sum up six security requirements that must satisfy to construct a secure conjunctive keyword searchable scheme. Then we compare and analyze the security and the performance of the security models. Finally, we list some issues that need to further discuss in the future.

Keywords: PECKS, conjunctive keyword searchable, off-line keyword-guessing attack, keyword field.

1 Introduction

Cloud computing has become the most popular issue in recent years. More and more cloud services have bloomed all around the world such as storage space outsourcing, computing resource and many kinds of software. Since variant cloud services have been used, millions of messages have been transferred in the public network and the relative security issues have arisen including privacy, security, resource abusing, and so on. Therefore, people usually use an extra safeguard before adopting cloud services. For example, when users wish to store the documents in the cloud storage space, they usually encrypt the documents before uploading them. After encrypting, the documents have become sequential characters which cannot be recognized. But how users can search the encrypted data that they want? In 2000, Song *et al.* [17] first gave the concept of searching the encrypted data with certain words. Although [17] requires very little communication between the user and the database, their model works in linear time in its size per query. Later, Boneh *et al.* [2] presented a security model searching on encrypted data by using a keyword in a public key system, which calls Public Key Encryption with Keyword Search (PEKS).

Suppose user Bob sends an encrypted emails $E_{A_{pub}}(M)$ to Alice using Alice's public key. In [2], Bob sends the encrypted data in the following form:

$$E_{A_{pub}}(M), PEKS(A_{pub}, w_1), \ldots, PEKS(A_{pub}, w_m)$$

where A_{pub} is Alice's public key, $(w_1, w_2, \ldots, w_m)$ denotes the keywords that Bob sets. When Alice wishes to read the emails that contain a specific keyword W, Alice sends a "trapdoor" to the mail server. The mail server should identify the corresponding encrypted emails without learning any information and route these emails to Alice. In a like manner, Alice wants to retrieve the encrypted emails which are associated to "urgent", "Monday, and "Marketing", the initial security model of PEKS cannot achieve this work since the user can only use one keyword to search the encrypted emails. However, searching on a large amount of data with more than one keyword can shrink the searching scope and improve the querying performance. Therefore, Golle, Staddon and Waters [8] (discuss in Section 2.2.1) proposed the notion of *secret key* encryption with conjunctive field keyword search scheme in 2004. They assume that there are n documents and m keyword fields associated with each document. They also make two assumptions that we will introduce in Section 2.

However, Golle *et al.*'s scheme, which is constructed in a symmetrical cryptosystem, is not applicable to a public key system. In the other hand, Boneh *et al.*'s scheme [2] is not suitable for the overall application in the reality neither. Since their schemes assume that there is a secure channel between the server and the receiver, building a secure channel is expensive. Therefore, Baek, Safavi-Naini and Susilo [1] presented a new security model that removes the secure channel assumption. Whenever the users wish to search the encrypted data, they can send the trapdoors via a *public channel*. But users can still search the encrypted documents with only one keyword in Baek *et al.*'s scheme. However, without the protection of a secure channel, Byun, Rhee, Park and Lee [4] pointed out that Baek *et al.*'s scheme might be attacked by the off-line keyword-guessing attacks. Since keywords are chosen from a much smaller space than passwords, users usually use well-known keywords for searching documents. Therefore, even if the keywords have been encrypted, the attackers still have chance to learn the embedded keywords by performing the off-line keyword-guessing attacks.

In 2005, Park, Kim and Lee [15] (discuss in Section 2.2.2) first presented a conjunctive keyword search scheme based on bilinear paring in the *public key* cryptosystem, named as Public Key Encryption with Conjunctive Field Keyword Search (PKCKS). In 2007, Ryu and Takagi [16] (discuss in Section 2.2.3) proposed an efficient construction for conjunctive keyword search scheme in a symmetric-key setting. Their scheme is based on bilinear map which has better performance than [8]. Hwang and Lee [10] also designed a PECKS based on bilinear map and extended their scheme to a multi-user system which is the first security model for multi-user public key encryption with conjunctive keyword search (mPECKS) scheme. Instead of equality tests [1, 2, 8, 10, 15, 16], Boneh and Waters [3] constructed a public-key system that supports comparison queries on encrypted data. If the mail satisfies a certain predicate P, the mail server routes the email to the receiver's mobile device, and place other emails in the receiver's another device, otherwise. However, the trapdoor size is linear to the number of searching keywords in Boneh and Water's scheme; that is, the efficiency reduces while the amount of keywords increasing. Later, Chen and Horng [6] (discuss in Section 2.2.4) presented a timestamp based conjunctive keyword searchable scheme in 2009. They adopted the timestamp to classify the encrypted data which can improve the efficiency of running time of Test step for the server. However, most of the existing conjunctive keyword searchable schemes adopted the assumptions that Golle *et al.* presented, which make conjunctive keywords regarded as one keyword since the keyword field limits the location of keywords [22]. Therefore, Zhang and Zhang [22] tended to eliminate these two assumptions. Instead of fixed keyword fields, they constructed a l-degree polynomial in PECKS algorithm which allows the users to list the keywords in any order. On the other hand, most of the existing conjunctive keyword searchable schemes using keyword fields produce long ciphertexts and trapdoors, Chen, Wu, Wang and Li [7] (discuss in Section 2.3.2) constructed two PECKS schemes in composite order groups model and in prime order groups model, which achieve constant ciphertext and short trapdoor. But Chen *et al.*'s scheme still have room for improving the size of ciphertext and computational load while the users upload and query the encrypted documents.

Except the assumptions in [8], most of the follow-up schemes used the fixed length keyword fields which are also the foundation in Golle *et al.*'s scheme. Comparing with variable length keyword fields, fixed length keyword fields has the advantages of security and convenience since it will reveal the least amount of information to the server [19]. The server which cannot learn the number of keywords per document has to be hidden. On the contrast, variable length keyword fields only needs less storage space. For the cloud service client, using variable length keyword fields can minimize the requirement of storage space that users need to buy from the cloud service provider (CSP). Both of fixed keyword fields and variable keyword fields have advantage and drawback. In this paper, we classify some existing conjunctive keyword searchable schemes into two categories: fixed keyword fields and variable keyword fields, and further discuss the performance of these schemes.

1.1 Requirements

To construct a secure conjunctive keyword searchable scheme, there are some security requirements needed to achieve as follows:

Unforgeable of the trapdoor [22]: Whenever the receiver wishes to search the encrypted data, he sends the trapdoor containing specific keywords to the server via a public network. This requirement means that others can get nothing from the trapdoor even if the trapdoors are captured by the adversaries.

Anonymous of the ciphertext [22]: Data senders encrypt the keywords by the authorized user's public key. Only the corresponding private key can decrypt the content. This requirement means that no one could get the embedded keywords from the ciphertext.

User authentication: After encrypting, no information can be derived from the ciphertexts and the trapdoors, but the server still has to recognize whether the users who send the trapdoor are the authorized users. This requirement means that the server must has the ability to authenticate the user's identities [11, 12, 18, 21].

Practicability: Whether the fixed keyword fields be adopted or not, the users should not have to memorize too much extra information as they search the encrypted data. For example, most of the schemes using the fixed keyword fields need to note all the keyword fields the users wish to search while the users generate the trapdoors. In other words, the users should learn all the keyword fields before they query the encrypted data. This requirement means that keywords should be listed in any order in the searching phase and the searching step should be convenience in users' view.

Efficiency: Most of the existing schemes need a large amount of computing time or produce long ciphertexts and trapdoors which are inefficient for users. This requirement means that the proposed scheme should be processed efficently in the reality.

Against off-line keyword-guessing attack: All the messages in the keyword searchable scheme are transferred via a public network and easy to eavesdrop. Not only the outside adversaries, the malicious servers are also regarded as the inside attackers. This requirement means that the proposed security model should stand against outside and inside off-line keyword-guessing attacks [13, 14].

1.2 Organization

This paper is organized as follows: In Section 2, we introduce the development of conjunctive keyword searchable schemes and analyze their advantages and shortcomings. We further evaluate whether the schemes in Section 3 conform the requirements mentioned above, and make a performance comparison. Finally, we discuss future researches in Section 4 and conclude in Section 5.

2 Security Model for Conjunctive Keyword Searchable Scheme

The notion of *secret key* encryption with conjunctive field keyword search scheme was proposed by Golle, Staddon and Waters [8] in 2004. They assumed that there are n documents and m keyword fields associated with each document. For example, if the documents are emails, there are the following 4 keyword fields: "From", "To", "Date", and "Subject". They also make two assumptions as follows:

- The same keyword never appears in two different keyword fields. It other words, all the keywords in one document are different from each other.
- Every keyword field is defined for every document. That is, every keyword field should assign a keyword even if one field is empty, we should assign "NULL" or some unmeaning symbols.

For each document, they identify with the vector of m keywords and denote the ith document by $D_i = (w_{i,1}, w_{i,2}, \ldots, w_{i,m})$. The above assumptions have been adopted to many conjunctive keyword searchable schemes. Since it identify a fixed number of keyword in each document, we classify some existing conjunctive keyword searchable scheme into two categories: fixed keyword fields and variable keyword fields, and introduce in this section.

2.1 Security Definitions

In order to prove the conjunctive keyword search scheme is secure, first we introduce three security games that Golle *et al.* defined in [8]. We say that a conjunctive keyword searchable scheme is secure in semantic-security (also called *indistinguishability*) with the following experiments:

2.1.1 Security Game ICC (Indistinguishability of Ciphertext from Ciphertext)

Let $\mathcal{A}$ be a polynomially bounded adversary (the server) and $\mathcal{B}$ be a challenger (the user). The goal of Game ICC is that $\mathcal{A}$ has to distinguish two encrypted documents, where D_0 and D_1 are chosen by $\mathcal{A}$. The conjunctive keyword search scheme is secure if $\mathcal{A}$ cannot distinguish between D_0 and D_1 successfully with non-negligible advantage.

1) An adversary $\mathcal{A}$ adaptively requests the encryption $\mathsf{Enc}(\rho, K, D)$ of documents D and search capabilities (trapdoors).

2) $\mathcal{A}$ chooses two documents D_0, D_1, and then sends them to the challenger $\mathcal{B}$.

3) $\mathcal{B}$ chooses b randomly from $\{0, 1\}$ and gives $\mathcal{A}$ an encryption of D_b.

4) $\mathcal{A}$ may again ask for encrypted documents and capabilities, with the restriction that $\mathcal{A}$ may not ask for a capability that is distinguishing for D_0 and D_1.

5) $\mathcal{A}$ outputs $b' \in \{0, 1\}$ and wins the game ICC if $b' = b$. We say the adversary $\mathcal{A}$ has an ϵ-advantage if the adversary's advantage is $Adv_A(1^k) = |Pr[b' = b] - 1/2| > \epsilon$.

2.1.2 Security Game ICR (Indistinguishability of Ciphertexts from Random)

Let $\mathcal{A}$ be a polynomially bounded adversary (the server) and $\mathcal{B}$ be a challenger (the user). The adversary chooses only on document D_0 and a keyword subset T of D_0. The goal of Game ICR is that $\mathcal{A}$ has to distinguish two encrypted documents, where D_0 is chosen by $\mathcal{A}$ and D_1 is produced by $\mathcal{B}$.

1) An adversary $\mathcal{A}$ adaptively requests the encryption $\mathsf{Enc}(\rho, K, D)$ of documents D and search capabilities (trapdoors).

2) $\mathcal{A}$ chooses a document D_0 and subset $T \subseteq \{1, \ldots, m\}$, then sends it to the challenger $\mathcal{B}$.

3) $\mathcal{B}$ creates a document D_1 =$Rand(D_0, T)$ and chooses a random bit $b \in \{0, 1\}$, then gives $\mathsf{Enc}(\rho, K, D_b)$ to $\mathcal{A}$.

4) $\mathcal{A}$ again asks for encrypted documents and capabilities, with the restriction that $\mathcal{A}$ may not ask for a capability that distinguishes D_0 from D_1.

5) $\mathcal{A}$ outputs $b' \in \{0, 1\}$ and wins the game ICR if $b' = b$. We say that the adversary $\mathcal{A}$ has an ϵ-advantage if the adversary's advantage is $Adv_A(1^K) = |Pr[b' = b] = 1/2| > \epsilon$.

2.1.3 Security Game ICLR (Indistinguishability of Ciphertexts from Limited Random)

Let $\mathcal{A}$ be a polynomially bounded adversary (the server) and $\mathcal{B}$ be a challenger (the user). The adversary chooses one document and a keyword subset T. Then, $\mathcal{B}$ generates two encrypted documents related to T. The goal of Game ICLR is that $\mathcal{A}$ has to distinguish two encrypted documents. This security game reflects a secure notion which guarantees that an adversary cannot gain the plaintext from the other documents [20].

1) An adversary $\mathcal{A}$ requests the encryption $\mathsf{Enc}(\rho, K, D)$ of any documents D and any search capabilities (trapdoors).

2) $\mathcal{A}$ chooses a documents D, and then sends it to the challenger $\mathcal{B}$.

3) $\mathcal{B}$ creates two documents D_0 =$Rand(D, T - \{t\})$ and D_1 =$Rand(D, T)$, where $T \subseteq 1, \ldots, m$ and a value $t \in T$, and chooses a random bit $b \in \{0, 1\}$, and then gives $\mathsf{Enc}(\rho, K, D_b)$ to $\mathcal{A}$.

4) $\mathcal{A}$ again asks for encrypted documents and capabilities, with the restriction that $\mathcal{A}$ may not ask for a capability that is distinguishing for D_0 and D_1.

5) $\mathcal{A}$ outputs $b' \in \{0, 1\}$ and wins the game ICLR if $b' = b$. We say that the adversary $\mathcal{A}$ has an ϵ-advantage if the adversary's advantage is $Adv_A(1^k) = |Pr[b' = b] - 1/2| > \epsilon$.

2.2 Fixed Keyword Fields Schemes

2.2.1 Golle *et al.*'s Scheme

Golle, Staddon and Waters [8] give the concept that if a user wishes to retrieve encrypted documents on an untrusted server, he gives the server a *capability* to identify the desired documents. Unlike the scheme searching encrypted document by using a simple keyword [2], Golle *et al.* first proposed a *conjunctive* keyword searchable scheme in 2004. Golle *et al.*'s scheme is constructed in *secret key* system and the security relies on the decision Diffie-Hellman (DDH) assumption. Their scheme consists of the following algorithms:

1) $\mathsf{Param}(1^k)$: Take a security parameters k and generate a group G with order q, a generator g of G, a keyed function $f : \{0, 1\}^k \times \{0, 1\}^* \rightarrow Z_q^*$ and a hash function h. Return system parameters $\rho = (G, g, f(\cdot, \cdot), h(\cdot))$.

2) $\mathsf{KeyGen}(\rho)$: Return a secret key $K \in \{0, 1\}^k$ for the function f, and we denote $f(K, \cdot)$ by $f_K(\cdot)$.

3) $\mathsf{Enc}(\rho, K, D_i)$: Set keywords of a document $D_i = (W_{i,1}, W_{i,2}, \ldots, W_{i,m})$ and compute $V_{i,j} = f_K(W_{i,j})$ for $j = 1, \ldots, m$. Select random values $a_i \in Z_q^*$ and output the keyword ciphertext $S = (g^{a_i}, g^{a_i V_{i,1}}, g^{a_i V_{i,2}}, \cdots, g^{a_i V_{i,m}})$.

4) $\mathsf{GenCap}(\rho, K, j_1, \cdots, j_t, W'_{j_1}, \cdots, W'_{j_t})$: Select a random number $s \in Z_q^*$ and compute $Q = (h(g^{a_1 s}), h(g^{a_2 s}), \cdots, h(g^{a_n s}))$. Then, we define the value $C = s + (\sum_{w=1}^{t} f_K(W'_{j_w}))$. Output $Cap = \{Q, C, j_1, j_2, \cdots, j_t\}$.

5) $\mathsf{Verification}(\rho, S, Cap)$: The server computes $R_i = g^{a_i C} \cdot g^{-a_i(\sum_{w=1}^{t} V_{i,j_w})}$ and returns "yes" if $h(R_i) = h(g^{a_i s})$ and "no" otherwise.

2.2.2 Park *et al.*'s Scheme

In Park, Kim and Lee's [15] opinion, Golle *et al.*'s scheme which is constructed in a symmetry system is not applicable to a public key system, so their scheme cannot be used in some applications. Therefore, Park *et al.* extend Golle *et al.*'s scheme into a public key cryptosystem. They adopted the assumptions in [8] and constructed two efficient schemes by using a bilinear map. Let G_1 and G_2 be two groups of order p. We use a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ between two groups. The map satisfies the following properties:

1) Bilinear: a map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ is bilinear if $\hat{e}(aU, bV) = \hat{e}(U, V)^{ab}$ for all $U, V \in G_1$ and all $a, b \in Z$.

2) Non-degenerate: if g is a generator of G_1 then $\hat{e}(g, g)$ is a generator of G_2.

3) Computable: there is an efficient algorithm to compute $\hat{e}(U, V)$ for any $U, V \in G_1$.

Park *et al.*'s scheme consists of the following algorithms:

1) $\mathsf{GlobalSetup}(1^k)$: The input security parameter k determines the size, p of the group G_1, G_2. Then construct a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$. Also, we need a hash function $H : \{0, 1\}^* \rightarrow G_1$. Finally, we output the global parameter $\mathcal{GP} = (p, G_1, G_2, \hat{e}, H)$.

2) $\mathsf{KeyGen}(\mathcal{GP})$: Choose two random values $s_1, s_2 \in Z_p$ and a generator $P \in G_1$. It outputs public key $A_{pub} = [P, Y_1 = s_1 P, Y_2 = s_2 P]$ and private key $A_{priv} = [s_1, s_2]$.

3) PECKS$(\mathcal{GP}, A_{pub}, D)$: Choose a random number $r \in Z_p$. It outputs the keyword ciphertext $S = [\hat{e}(rH(W_1), Y_1), \hat{e}(rH(W_2), Y_1), \cdots, \hat{e}(rH(W_m), Y_1), rY_2, rP]$.

4) Trapdoor$(\mathcal{GP}, A_{priv}, Q)$: Select a random value $u \in Z_P$ and make $T_Q = [T_1, T_2, I_1, I_2, \cdots, I_t]$ where $T_1 = (\frac{s_1}{s_2+u}\ mod\ p)(H(W_1') + H(W_2') + \cdots + H(W_t'))$, $T_2 = u$, and $I_1, I_2, \cdots, I_t$ are positions of the keywords in Q.

5) Test$(\mathcal{GP}, A_{pub}, S, T_Q)$: Let $S = [A_1, A_2, \cdots, A_m, B, C]$. Check if $A_{I_1} \times A_{I_2} \times \cdots \times A_{I_t} = \hat{e}(T_1, B + T_2C)$. If so, output "yes", and "no" otherwise.

2.2.3 Ryu and Takagi's Scheme

In 2007, Ryu and Takagi [16] also proposed *secret key* conjunctive keyword search scheme. Their scheme constructed in bilinear forms and adopted Golle *et al.*'s assumptions. Moreover, their scheme is more efficient than [8] and [5]. Ryu and Takagi's scheme consists of the following algorithms:

Let $H : \{0,1\}^* \rightarrow G_1$ be a hash function.

1) KeyGen(1^k): Given a security parameter k and determines two group $G_1 =< g_1 >$ and $G_2 =< g_2 >$ of a prime order p, where g_2 is not public. It returns a secret key $\alpha \in G_1$.

2) Enc(α, D_i): Let $h_{i,j} = H(W_{i,j})$ for $j = 1, 2, \cdots, m$. Select a random value $r_i \in Z_p^*$. It outputs the keyword ciphertext $C_i = (e(\alpha, g_2^{r_i}), g_2^{r_i}, (h_{i,1})^{r_i}, \ldots, (h_{i,m})^{r_i})$.

3) Trapdoor$(\alpha, \{j_1, \ldots, j_t\}, \{W'_{j_1} \ldots, W'_{j_t}\})$: Select a random value $s \in Z_p^*$. It returns the trapdoor $T_{w'} = (\alpha \prod_{w=1}^{t} (H(W'_{j_w}))^s, g_2^s)$.

4) Test$(T_{w'}, C_i)$: Let $T_{w'} = (T_1, T_2)$ and $C_i = (V_i, C_{i,0}, C_{i,1}, \ldots, C_{i,m})$. Check if

$$e(T_1, C_{i,0})/e(\textstyle\prod_{w=1}^{t} C_{i,j_w}, T_2) = V_i.$$

If so, output "yes", and "no" otherwise.

2.2.4 Chen and Hrong's Scheme

Chen and Horng [6] proposed a PECKS scheme based on timestamp in 2009. In order to improve the efficiency of running time that the server operates Test algorithm, Horng and Chen used the timestamp to classify the encrypted data. In terms of searching the corresponding encrypted documents, the server will create "the encrypted timing data" after receiving the keyword ciphertext from the data sender. Therefore, the server can use a part of trapdoor to find the corresponding encrypted documents in a short time which is faster than other schemes.

This scheme uses a hash function $H : \{0,1\}^* \rightarrow Z_p^*$, three groups G_1, G_2 and G_t of prime order p, and a bilinear map $\hat{e} : G_1 \times G_2 \rightarrow G_t$. The global parameter $\mathcal{GP} = (p, G_1, G_2, G_t, \hat{e}, H)$.

1) KeyGen$(\mathcal{GP})$: Pick a random value $\alpha \in Z_p^*$, a generator $P_1 \in G_1$ and a generator $P_2 \in G_2$. It returns public key $A_{pub} = [P_1, P_2, Y = \alpha P_1]$ and private key $A_{priv} = \alpha$.

2) PECKS$(\mathcal{GP}, A_{pub}, D)$: Choose a random value $r \in Z_p^*$ and compute $V_i = rH(W_i)Y$. It outputs the keyword ciphertext $S = [V_1, V_2, \ldots, V_m, rP_1]$.

3) Timestamp(S, k): When the server receives the ciphtertext S, it chooses a random value $s \in Z_{p-1}^*$. Then, it outputs "the encrypted timing data" $S_k = [V_1k, V_2k, \ldots, V_mk, rP_1, kP_2]$ and publishes the timestamp value kP_2.

4) Trapdoor$(\mathcal{GP}, A_{priv}, D, kP_2)$: Take a timestamp value kP_2 from the public information server and select a random value $s \in Z_p^*$. Compute $T_{w'} = [T_1, T_2, T_3, I_1, I_2, \ldots, I_t]$ where $I_1, I_2, \ldots, I_t$ are the keyword fields which the receiver wishes to search, and

$$T_1 = \textstyle\sum_{i=1}^{t} (H(W_i'))s\alpha(kP_2)$$
$$T_2 = sP_2$$
$$T_3 = kP_2$$

in which T_3 is a label for searching the corresponding groups of document for the server.

5) Test$(\mathcal{GP}, S_k, T_{w'})$: Let $S_k = [A_1, A_2, \ldots, A_m, B]$ and the server use T_3 to find the corresponding encrypted documents. Check if $\hat{e}(B, T_1) = \hat{e}(A_{l_1} + A_{l_2} + \ldots + A_{l_t}, T_2)$. If so, output "yes", and "no" otherwise.

Although the encrypted timing data shorten the searching time of Test algorithm, Chen and Hrong's scheme is still threatened by the outside off-line keyword-guessing attack.

2.3 Variable Keyword Fields Schemes

2.3.1 Zhang and Zhang's Scheme

Zhang and Zhang [22] pointed out that the assumptions in [8] make conjunctive keywords regarded as one keyword and limit the location of keywords since the keyword fields are fixed and inflexible for users. If a user wishes to search five keywords, he has to identify the exactly field that he wishes to query. It burdens the users with an extra information needed to remember. Therefore, Zhang and Zhang presented the following two concepts: (1) Keywords should be listed in any order. (2) The repetition of one keyword has nothing wrong about the performance. Zhang and Zhang assumed that each document has l keywords (l is fixed). Although users do not need to define m keyword fields for each document, they still have to build up a l-degree polynomial with l keywords. If the number of keywords is less than l, users can just add some useless keywords to realize the algorithm. Zhang and Zhang's scheme works as follows:

Assume that there are l keywords in the PECKS algorithm (l is fixed). This scheme uses three group G_1, G_2 and G_t of prime order p, two collision resistant hash functions: $H : \{0,1\}^* \rightarrow Z_p^*$ and $H' : G_t \rightarrow Z_p^*$. The bilinear group is $\mathcal{GP} = (p, G_1, G_2, G_t, \hat{e})$.

1) $\mathsf{Setup}(1^k, l)$: First choose $l+1$ parameters: $b_0, b_1, \ldots, b_l \in G_1$. Select two random generators $g_1, g_2 \in G_1$, a random generator $h \in G_2$ and a random value $\alpha \in Z_p^*$. Set $h_1 = h^\alpha$. Output the public key $pk = (g_1, g_2, h, h_1, b_0, b_1, \ldots, b_l)$ and the private key $sk = \alpha$.

2) $\mathsf{PECKS}(\mathcal{GP}, pk, w_1, w_2, \ldots, w_l)$: Choose random elements $a, k \in Z_p$, then construct a l-degree polynomial:

$$\begin{aligned} f(x) &= a \cdot (x - H(w_1))(x - H(w_2)) \cdots (x - H(w_l)) \\ &\quad +k, \\ &= a_l x^l + \cdots + a_1 x + a_0. \end{aligned}$$

Select a random element $r' \in Z_p$, then compute and output the keyword ciphertext S:

$$\begin{aligned} S &= (h^{r'k}, H'(e(g_2, h)^{(a_0 + a_1 + \ldots + a_l) \cdot r'}); \\ &\qquad h_1^{a_0 r'}, h_1^{a_1 r'}, \ldots, h_1^{a_l r'}; b_0^{a_0 r'}, b_1^{a_1 r'}, \ldots, b_l^{a_l r'}). \end{aligned}$$

3) $\mathsf{Trapdoor}(sk, w'_1, w'_2, \ldots, w'_s)$: Choose a random value $r \in Z_p$, then compute and output $T_{w'}$=$[g_2^{1/\alpha} \cdot (g_1^{H(w'_1)^0 + H(w'_2)^0 + \cdots + H(w'_s)^0 / \alpha \cdot s} \cdot b_0)^r = g_2^{1/\alpha} \cdot (b_0)^r$;

$$g_2^{1/\alpha} \cdot (g_1^{H(w'_1)^1 + H(w'_2)^1 + \cdots + H(w'_s)^1 / \alpha \cdot s} \cdot b_1)^r;$$

$$\vdots$$

$g_2^{1/\alpha} \cdot (g_1^{H(w'_1)^l + H(w'_2)^l + \cdots + H(w'_s)^l / \alpha \cdot s} \cdot b_l)^r; g_1^r; h_1^r]$.

4) $\mathsf{Test}(\mathcal{GP}, pk, S, T_{w'})$: Set

$$\begin{aligned} T_{w'} &= (T_0, T_1, \ldots, T_l; g_1^r, h_1^r), \\ S &= (C_0, C_1; H_0, H_1, \ldots, H_l; B_0, B_1, \ldots, B_l). \end{aligned}$$

Then compute the following parameters:

$$A_1 = \prod_{i=0}^{l} \hat{e}(T_i, H_i);$$
$$A_2 = \hat{e}(g_1^r, C_0) = \hat{e}(g_1^r, h^{r'l});$$
$$A_3 = \prod_{i=0}^{l} \hat{e}(B_i, h_1^r) = \prod_{i=0}^{l} \hat{e}(b_i^{a_i \cdot \alpha \cdot r'}, h^r);$$

Check if $H'(A_1/(A_2 \cdot A_3)) = C_1$. If so, output "yes", and "no" otherwise.

2.3.2 Chen *et al.*'s Scheme

Chen, Wu, Wang and Li [7] constructed two PECKS schemes of composite order groups model and in prime order groups model. The proposed scheme achieves a constant ciphertext and a short trapdoor. The admissible bilinear maps in composite order group is described as follows:

Let $G =< g >$ and G_t be two cyclic multiplicative groups of composite order $n = pqr$, and $\hat{e}$ be an admissible bilinear map from G^2 to G_t. Assume that it is a hard problem to factor p, q, r on n. Let G_p, G_q and G_r denote the subgroups of order p, q, r of G, and $G_{t,p}, G_{t,q}$ and $G_{t,r}$ denote as the subgroups of G_t, respectively; then $G = G_p \times G_q \times G_r$, and $G_t = G_{t,p} \times G_{t,q} \times G_{t,r}$.

In Chen *et al.*'s scheme, the subgroup G_q and G_r are used in the anonymity of encryption purpose and the correlation hiding between random values; and the subgroup G_p is used to prevent the vicious manipulation of the keyword ciphertext S or the trapdoor $T_{w'}$ from the adversaries, and then evaluate a query on the incorrect inputs. Chen *et al.*'s scheme consists of the following algorithms:

Let KS_w denotes the keyword space. The whole algorithm are as follows:

1) $\mathsf{Setup}(1^\lambda)$: First generate the bilinear group G of composite order $n = pqr$ where p, q and $r (p, q, r > m)$ are random primes of size $\theta(\lambda)$. Then pick $\alpha \in Z_p$ and $v, w_1, w_2, (u_1, h_1), \ldots, (u_l, h_l) \in G_p$ randomly. And also pick random elements $R_v, R_{w,1}, R_{w,2}, (R_{u,1}, R_{h,1}), \ldots, (R_{u,l}, R_{h,l}) \in Z_q$ and computes $V = vR_v, W_1 = w_1 R_{w,1}, W_2 = w_2 R_{w,2}$. For $i = 1, \ldots, l$, computes

$$U_i = u_i R_{u,i}, H_i = h_i R_{h,i}, E = \hat{e}(v, g)^\alpha,$$

Output the secret key $msk = (\alpha,\ v,\ w_1,\ w_2,\ (u_1,\ h_1),\ \ldots,\ (u_l,\ h_l))$ and the public parameters $\mathcal{GP} = [g_q, g_r, V, W_1, W_2, (U_1, H_1), \ldots, (U_l, H_l), E, KS_w]$.

2) $\mathsf{PECKS}(\mathcal{GP}, \vec{x})$: $\vec{x} = (x_1, \ldots, x_l) \in KW_w$ denotes the conjunctive keywords vector that the data sender sets. First pick $s \in Z_n$ and $Z_1, Z_2, Z_3, Z_4 \in G_q$, and outputs the ciphertext $S = [C_0, C_1, C_2, C_3, C_4]$ as follows:

$$\begin{aligned} C_0 &= E^s, C_1 = V^s Z_1, C_2 = W_1^s Z_2, \\ C_3 &= W_2^s Z_3, \\ C_4 &= (\prod_{i=1}^{l} H_i U_i^{x_i})^s Z_4. \end{aligned}$$

3) $\mathsf{Trapdoor}(\mathcal{GP}, msk, \vec{e})$: To generate a trapdoor for conjunctive keywords $\vec{e} = (e_1, \ldots, e_l) \in KS_w$, first randomly pick $r_1, r_2, r_3 \in Z_n$, and $Y_1, Y_2, Y_3, Y_4 \in G_r$. Then compute and output the trapdoor $T_{w_{\vec{e}}} = [K_1, K_2, K_3, K_4]$ as follows:

$$\begin{aligned} K_1 &= g^\alpha w_1^{r_1} w_2^{r_2} (\prod_{i=1}^{l} h_i u_i^{e_i})^{r_3} Y_1, \\ K_2 &= v^{r_1} Y_2, \\ K_3 &= v^{r_2} Y_3, \\ K_4 &= v^{r_3} Y_4. \end{aligned}$$

4) $\mathsf{Test}(\mathcal{GP}, S, T_{w'})$: Check if $\hat{e}(C_1, K_1) = C_0 \prod_{i=2}^{4} \hat{e}(C_i, K_i)$. If so, outputs "yes" and "no" otherwise.

Table 1: Security comparison

	GSW [8]	PKL1 [15]	RT [16]	CH [6]	ZZ [22]	CWWL [7]
Fixed Field	×	×	×	×	○	○
Unforg Trap	○	○	○	○	○	○
Anony Cipher	○	○	○	○	○	○
User Auth	○	○	○	○	○	○
Inside KG	×	×	×	×	×	×
Outside KG	○	×	○	×	○	○

Although the number of keyword ciphertext and trapdoor will not linearly expand as long as the number of the keywords increases, the size of ciphertext and trapdoor in Chen *et al.*'s scheme still larger than other schemes; also, it causes a long communication time for users to effect the quality of uploading and querying the documents.

3 Comparisons

3.1 Security Analysis

In this section, we analyze the security of the schemes that we have discussed in Section 2 and show the comparison in Table 1. We define some notion as follows and note each scheme with the abbreviation of author names: Fixed Field means whether the scheme uses the fixed field assumption or not. Unforg Trap is unforgeable of the trapdoor. Anony Cipher is anonymous of the ciphertext. User Auth is user authentication. Inside KG is against inside off-line keyword-guessing attack. Outside KG is against outside off-line keyword-guessing attack. Although Fixed Field is not one of the security requirements at all, we wish to note this characteristic in Table 1 to identify the assumption that each scheme has adopted. Furthermore, whether the scheme uses a fixed field or not will affect the practicability it has. The Efficiency requirement will be discussed in Section 3.2 by comparing the performance of all algorithms.

We can observe that all the schemes have satisfied the requirements of Unforg Trap, Anony Cipher and User Auth since these are the basic functions that a keyword searchable scheme must provide. When a user searches the encrypted documents with keywords, the whole query process should not reveal any information (keywords, user's identity, user's private key, etc.) to anyone. Under the above situation, all the schemes can make the server have the ability to recognize the authorized users and executes Test algorithm inerrably. GSW, RT, ZZ and CWWL can stand against the off-line keyword-guessing attack from an outside adversary successfully. Since GSW and RT are constructed in the symmetric-key setting, others cannot perform the off-line keyword-guessing attack without learning the secret information. Unfortunately, all the schemes cannot prevent the off-line keyword-guessing attack from the malicious server since it possesses enough information to perform Test algorithm. The more information the server has, the more easily the attack successes.

3.2 Performance Analysis

In this section, we analyze the performance of the schemes with the size of outputs and the computation load that each algorithm need, and display the comparison in Table 2. We define the notations as follows: num is the size in Z_p. $|p|$ is the size in G_1 (G_2 or G_t). m is the keyword field and n is the number of document (In general, the user searches s keywords where $s \leq m$. So, we assume the maximum computation requirement in Trapdoor). E is the operation of exponentiation. P is the operation of Maptopoint function which maps a keyword to an element in G_1 [9]. M is the operation of multiplication in G_1, G_2 or G_t. e is the operation of pairing. We ignore the hash operation since it only requiress little of computation load. On the contrast, a Maptopoint function produces a large amount of computation load which is inefficient.

Except GSW, all other schemes are constructed in bilinear form. In a key generation algorithm, CWWL produces the longest results in both public key and private key. In another word, the user should uses larger storage space to store the public/private key pairs. Although CWWL reduces the number of trapdoors and keyword ciphertexts, the computational load and the size of outputs are still high. Although it seems that PKL1 and CH are the alternatives for the users who use the lightweight devices since the lower computation are needed in both Encryption algorithm and Trapdoor algorithm, PKL1 and CH are not secure enough to prevent the outside and inside off-line keyword-guessing attack. Besides, other schemes still have a larger computational load in Encryption algorithm and Trapdoor algorithm for users. To sum up, the existing schemes still have a room for enhancing the security and the efficiency.

4 Future Research

Most of the existing conjunctive keyword searchable schemes cannot possess both security and efficiency at the same time. If a method focuses on enhancing the security that to stand against the inside and outside off-

Table 2: Performance comparison

	GSW [8]	PKL1 [15]	RT [16]	CH [6]	ZZ [22]	CWWL [7]
$\vert pk\vert$	-	$3\vert p\vert$	-	$3\vert p\vert$	$(m+5)\vert p\vert$	$(2m+5)\vert p\vert + num$
$\vert sk\vert$	num	$2num$	$\vert p\vert$	num	num	$(2m+5)\vert p\vert + num$
$\vert Encryption\vert$	$(m+1)\vert p\vert$	$2\vert p\vert + (m)num$	$(m+1)\vert p\vert + num$	$\vert p\vert + (m+1)num$	$(2m+1)\vert p\vert + num$	$4\vert p\vert + num + log\ m$
$\vert Trapdoor\vert$	$(n+1)\vert p\vert + log\ m$	$\vert p\vert + num$	$2\vert p\vert + log\ m$	$3\vert p\vert + log\ m$	$(m+2)\vert p\vert$	$7\vert p\vert + log\ m$
Encryption	$(m+1)E$	$(m)P + 2M + (m)e$	$(m+1)E+e$	$(m+1)M$	$(2m+4)E+e$	$(m+5)E+(m+3)M$
Trapdoor	$(n)E$	$(m)P+M$	$2E$	$2M$	$(2m+4)E+(m)M$	$(m+7)E+(m+7)M$
Test	$2E$	e	$2e$	$2e$	$(2m+3)e$	$4e$

line keyword-guessing attack successfully, the algorithm might consists of an exponentiation operation, Maptopoint function, and so on, which causes a huge amount of computational load for users. Therefore, how to reduce the computational load and improve the efficiency at the same time is one of the most important issues that needs to be solved.

Besides, in order to construct a secure enough conjunctive keyword searchable scheme, some schemes are constructed in symmetric-key cryptosystem. In contrast, it is securer in encrypting the keywords and searching the encrypted documents, but how to transit the secret key is another important question which need to further discuss. Hence, no matter what the cryptosystem is adopted, how to construct a thorough conjunctive keyword searchable scheme is another important issue that can be deeply researched in the future.

5 Conclusions

The first keyword searchable scheme was presented by Boneh *et al.* in 2004, enables a user to search the encrypted data by using a keyword. Later, Golle *et al.* extended Boneh *et al.*'s scheme into conjunctive keyword searchable scheme in a symmetric-key setting, and Park *et al.* further proposed the extended scheme in a public key cryptosystem, which named as Public Key Encryption with Conjunctive Field Keyword Search (PECKS) scheme. In this paper, we study the development of conjunctive keyword searchable scheme and classify these schemes into two categories to discuss; that is, fixed keyword fields and variable keyword fields. Moreover, we summarize six requirements from the existing literatures and analyze some existing schemes. Furthermore, we point out two important issues which can be enhanced to construct a more complete and secure conjunctive keyword searchable scheme in the future.

Acknowledgments

This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC101-2221-E-030-018.

References

[1] Joonsang Baek, Reihaneh Safavi-Naini, and Willy Susilo, "Public key encryption with keyword search revisited," in *ICCSA 2008*, vol. 5072 of *Lecture Notes in Computer Science*, pp. 1249–1259, Perugia, Italy, 2008.

[2] D. Boneh, G. D. Crescenzom, R. Ostrovsky, and G. Rersiano, "Public key encryption with keyword search," in *Advances in Cryptology - EUROCRYPT 2004, Lecture Notes in Computer Science*, vol. 3027, pp. 506–522, Interlaken, Switzerland, 2004.

[3] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *4th Theory of Cryptography Conference, TCC 2007*, vol. 4392 of *Lecture Notes in Computer Science*, pp. 535–554, 2007.

[4] 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 *Secure Data Management, Lecture Notes in Computer Science*, vol. 4165, pp. 75–83, Seoul, Korea, 2006.

[5] J. W. Byun, D. H. Lee, and J. Lim, "Efficient conjunctive keyword search on encrypted data storage system," in *proceedings of EuroPKI 2006*, vol. 4043 of *Lecture Notes in Computer Science*, pp. 184–196, 2006.

[6] Y. C. Chen and G. Horng, "Timestamped conjunctive keyword-searchable public key encryption," in *Innovation Computing Information and Control (ICICIC) 2009 Forth International Conference on*, pp. 729–732, 2009.

[7] Z. Chen, C. Wu, D. Wang, and S. Li, "Conjunctive keywords searchable encryption with efficient pairing, constant ciphertext and short trapdoor," in *Proceedings of PAISI 2012*, vol. 7299 of *Lecture Notes in Computer Science*, pp. 176–189, 2012.

[8] P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data," in *In Proceedings of Applied Cryptography and Network Security Conference*, vol. 3089 of *Lecture Notes in Computer Science*, pp. 31–45, 2004.

[9] C. Gu and Y. Zhu, "New efficient searchable encryption schemes from bilinear pairings," *International Journal of Network Security*, vol. 10, no. 1, pp. 25–31, 2010.

[10] 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 2007*, vol. 4575 of *Lecture Notes in Computer Science*, pp. 2–22, 2007.

[11] M. Kumar, "A new secure remote user authentication scheme with smart cards," *International Journal of Network Security*, vol. 11, no. 2, pp. 88–93, 2010.

[12] C. C. Lee, "On security of an efficient nonce-based authentication scheme for SIP," *International Journal of Network Security*, vol. 9, no. 3, pp. 201–203, 2009.

[13] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64–67, 2013.

[14] C. T. Li and Y. P. Chu, "Cryptanalysis of threshold password authentication against guessing attacks in ad hoc networks," *International Journal of Network Security*, vol. 8, no. 2, pp. 166–168, 2009.

[15] D. J. Park, K. Kim, and P. J. Lee, "Public key encryption with conjunctive field keyword search," in *Information Security Applications, 5th Interational Workshop, WISA 2004*, vol. 3325 of *Lecture Notes in Computer Science*, pp. 73–86, 2005.

[16] E. K. Ryu and T. Takagi, "Efficient conjunctive keyword-searchable encryption," in *Advanced Information Networking and Application Workshops, 2007, AINAW '07. 21st International Conference on*, vol. 1, pp. 409–414, 2007.

[17] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Security and Privacy, 2000. S P 2000. Proceedings. 2000 IEEE Symposium on*, pp. 44–55, 2010.

[18] C. S. Tsai, C. C. Lee, and M. S. Hwang, "Password authentication schemes: current status and key issues," *International Journal of Network Security*, vol. 3, no. 2, pp. 101–115, 2006.

[19] A.H.P. Van Vliet. *Secure Data Storage Outsourcing with Conjunctive Keyword Search.* Thesis, Delft University of Technology, 2009.

[20] P. Wang, H. Wang, and J. Pieprzyk, "Threshold privacy preserving keyword searches," in *SOFSEM 2008: Theory and Practice of Computer Science*, vol. 4910 of *Lecture Note in Computer Science*, pp. 646–658, 2008.

[21] C. Y. Yang, C. C. Lee, and S. Y. Hsiao, "Man-in-the-middle attack on the authentication of the user from the remote autonomous object," *International Journal of Network Security*, vol. 1, no. 2, pp. 81–83, 2005.

[22] Bo Zhang and F. Zhang, "An efficient public key encryption with conjunctive-subset keywords search," *Journal of Network and Computer Application*, vol. 34, no. 1, pp. 262–267, 2011.

Cheng-Chi Lee received the Ph.D. degree in Computer Science from National Chung Hsing University (NCHU), Taiwan, in 2007. He is currently an Associate Professor with the Department of Library and Information Science at Fu Jen Catholic University. Dr. Lee is currently as an editorial board member of International Journal of Network Security and Journal of Computer Science. He also served as a reviewer in many SCI-index journals, other journals, other conferences. His current research interests include data security, cryptography, network security, mobile communications and computing, wireless communications. Dr. Lee had published over 100+ articles on the above research fields in international journals.

Shih-Ting Hsu was born in Taoyang County, Taiwan, in 1988. She received her B.M. in Information Management from Yuan Ze University in 2011. She is currently pursuing the M.S. degree with the Department of Management Information Systems from National Chung Hsing University. Her research interests include cloud computing, information security, and cryptography.

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China (ROC), in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field "Electronic Engineer" in 1988. He also passed the National Telecommunication Special Examination in field "Information Engineering", qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999-2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002-2003. He is currently a professor of the department of Management Information System, National Chung

Hsing University, Taiwan, ROC. He obtained 1997, 1998, 1999, 2000, and 2001 Outstanding Research Awards of the National Science Council of the Republic of China. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang had published 170+ articles on the above research fields in international journals.