Unconditionally Secure Searchable Encryption

Takahiro Yoshizawa*, Yohei Watanabe†, and Junji Shikata*‡
*Graduate School of Environment and Information Sciences, Yokohama National University, Yokohama, Japan
†Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofu, Japan
‡Institute of Advanced Sciences, Yokohama National University, Yokohama, Japan
E-mail: yoshizawa-takahiro-my@ynu.jp, watanabe@uec.ac.jp, shikata@ynu.ac.jp

***Abstract*—Searchable symmetric encryption (SSE) enables us to search encrypted data with an arbitrarily chosen keyword without leaking information on the data and keyword. SSE is expected to be used in, for example, cloud computing and genome analyses. In particular, privacy of genome data must be guaranteed for long periods, and therefore unconditionally secure cryptographic protocols, rather than computationally secure ones, should be used for protecting genome data. For this reason, we propose new constructions of unconditionally secure SSE schemes in this paper. Specifically, we define a model and security of unconditionally secure SSE, and we show a lower bound on secret-key sizes. We propose two kinds of constructions of unconditionally secure SSE schemes: One is *asymptotically optimal* in the sense of the secret-key size with some restriction on the security definition; and the other achieves full security at the sacrifice of the secret-key size.**

I. Introduction

A. Background

Searchable encryption enables a user (or a client) to encrypt data so that the data can be searched without leaking any useful information on the data and search keywords. Searchable encryption has been studied since the 2000s and it is expected to be used in, for example, cloud computing. With recent advances of cloud technologies, many searchable encryption schemes have been proposed in the asymmetric-key (public-key) setting (e.g., [1], [2], [3], [4]) and in the symmetric-key (secret-key) setting (e.g., [5], [6], [7], [8]), respectively. The former is called public-key encryption with keyword search (PEKS), and the latter is called searchable symmetric encryption (SSE). In SSE, which is the main focus in this paper, a common key is used for document encryption and tag generation for search. A user generates a tag corresponding to a keyword using a key and a server can search the corresponding document using the tag. On the other hand, SSE has security that it does not leak information on generated keyword and stored (encrypted) documents to the server.

While cryptography with computational security such as public-key cryptography is highly convenient, it cannot guarantee long-term security. According to NIST [9], a 2048-bit RSA cryptosystem, which is considered sufficiently secure in most applications, shall not be used after 2030. From the viewpoint of symmetric-key cryptography, although advanced encryption standard (AES) has not been known to be insecure after 2030 so far, it seems to be difficult to provide long-term security (e.g., over 50 years) as the lifetime of data encryption standard (DES) was less than 30 years. Therefore, computationally secure cryptographic protocols are insufficient for encrypting sensitive information such as genome data. However, all existing searchable encryption schemes except for [10] are *computationally secure* (i.e., assuming an adversary is a probabilistic polynomial-time algorithm).

B. Our Contribution

In this paper, we first propose *unconditionally secure SSE*, which can provide long-term security. Namely, our SSE scheme is secure against any computationally-unbounded adversary. Specifically, we formalize a model and security notion of SSE based on computationally secure SSE schemes [5], [6]. In our SSE scheme, n documents are encrypted with corresponding keywords, and a user can search at most τ different keywords without leaking any information on the documents and keywords. We derive a lower bound on the secret-key size required for secure SSE schemes. We then construct two unconditionally secure SSE schemes from scratch (i.e., by using algebraic structures). The first construction is *asymptotically optimal* in the sense that it asymptotically attains the lower bound with equality. However, we prove the security only on the restricted relation between documents and keywords. Specifically, the security can be proved only when each document contains the same number of keywords and the number of documents associated with each keyword is also the same. The second construction gets rid of such a restriction by allowing large secret-key sizes, and therefore achieves the security for any situations.

C. Related Work

Goldreich and Ostrovsky [11], [12] showed how to search encrypted data by using oblivious RAMs. Their approach can address strong security, however it requires a number of interactions and a high overhead. Song et al. [13] first considered the concept of SSE explicitly, and later, Curtmola et al. [5], [6] gave formal security definitions that are considered standard nowadays. Although their security definition is weaker than that in [11], [12], they realized more efficient schemes than Goldreich and Ostrovsky's solution. Recently, several *dynamic* SSE schemes, where stored data can be dynamically and efficiently added and/or removed, have been proposed [7], [8].

On the other hand, Boneh et al. [1] first realized searchable encryption in the public-key setting (i.e., PEKS), and later Abdalla et al. [2], [3] improved Boneh et al.'s result.

978-1-5090-4780-2/17/$31.00 ©2017 IEEE

There is only one important related work in terms of searchable encryption with information-theoretic security. Sedghi et al. [10] proposed a model for analyzing existing (computationally secure) searchable encryption schemes in the unconditional security sense. Specifically, they analyzed how existing schemes [1], [13], [14] should be modified so that the schemes could achieve unconditional security. However, unlike our results, their model is limited to only one document to be stored in a server, and they derived no lower bounds on the secret-key sizes.

II. Model and Security Definition

A. Model

In this section, we define a model of unconditionally secure SSE based on existing (computationally secure) schemes [5], [6]. Roughly speaking, unconditionally secure SSE is executed as follows. A user first chooses n documents to be stored in a server, and he/she chooses one or more keywords for each document. Then, the user encrypts the documents (and corresponding keywords), and then the user sends the encrypted documents to the server.

A user who wants to search a keyword creates a tag (or a trapdoor) of the keyword and sends it to the server. The server searches (encrypted) documents associated with the keyword by using the tag and returns the corresponding encrypted documents (more precisely, identifiers of documents), while the server cannot learn any information on the underlying keyword and documents.

Notation. For any natural number $\alpha \in \mathbb{N}$, let $[\alpha] := \{1, 2, \ldots, \alpha\}$. Let $\mathcal{M}$ be a set of possible documents with a probability distribution P_M, and $\mathcal{W} := \{w_1, \ldots, w_d\}$ (i.e., $|\mathcal{W}| = d$) be a set of possible keywords in lexicographic order with a probability distribution P_W. Let $\mathcal{K}$, $\mathcal{C}$, and $\mathcal{T}$ be sets of secret keys, ciphertexts, and tags, respectively. For any $\mathcal{D} = \{m_1, \ldots, m_n\} \subset \mathcal{M}$, $\mathcal{I} := \{id(m_1), \ldots, id(m_n)\}$ denotes a set of corresponding identifiers. Each $id(m_i)$ $(1 \leq i \leq n)$ denotes an identifier of m_i and is independent of m_i (the identifier does not include any information on the document at all). To represent relations between each document and keywords, we consider *a relation function* $f \in \mathcal{F}$, which is $f : \mathcal{M} \to 2^{\mathcal{W}} \setminus \{\emptyset\}$, where $\mathcal{F}$ is the set of all possible f (family of relation functions). Let $\mathcal{W}(m) := f(m)$ be the set of keywords associated with m. Let $\mathcal{W}(\mathcal{D}) := \bigcup_{m \in \mathcal{D}} f(m)$ for any $\mathcal{D} \subset \mathcal{M}$. For any $\mathcal{D} = \{m_1, \ldots, m_n\} \subset \mathcal{M}$ and $w \in \mathcal{W}$, let $\mathcal{D}(w) = \{id(m) \in \mathcal{I} \mid w \in \mathcal{W}(m \in \mathcal{D})\}$ be a set of identifiers of documents associated with w.

Formally, we define (n, τ)-SSE Π below. Let M, W, C and T be random variables taking values on finite sets $\mathcal{M}, \mathcal{W}, \mathcal{C}$ and $\mathcal{T}$, respectively. In this model, we assume that elements of $\mathcal{D}$ are distinct, and that all keywords $(w_1, \ldots, w_\tau)$ used for search are also different. Furthermore, the documents $\mathcal{D}$ and the keywords $(w_1, \ldots, w_\tau)$ for search are independently chosen since the both are chosen by different users. Namely, $(M_1, M_2, \ldots, M_n)$ and $(W_1, W_2, \ldots, W_\tau)$ are independent of each other.

Definition 1 ((n, τ)-SSE)**.** *An (n, τ)-SSE scheme Π consists of the following five-tuple algorithms (Gen, Enc, Tag, Search, Dec) and six finite sets $\mathcal{M}$, $\mathcal{K}$, $\mathcal{W}$, $\mathcal{C}$ $\mathcal{T}$, and $\mathcal{I}$.*

1) *$k \leftarrow Gen(n, \tau)$: A probabilistic algorithm for key generation. It takes the number of documents which the user can encrypt n and the maximum number of (different) keywords which the user can search τ as input, and outputs a secret key $k \in \mathcal{K}$.*
2) *$(\mathcal{I}, \mathcal{ED}) \leftarrow Enc(k, \mathcal{D})$: A deterministic algorithm for encryption. It takes the secret key k, n documents $\mathcal{D} := \{m_1, \ldots, m_n\} \subset \mathcal{M}$ as input, and outputs a set of identifiers $\mathcal{I} := \{id(m_1), \ldots, id(m_n)\}$ and a set of ciphertexts $\mathcal{ED} := \{c_1, \ldots, c_n\} \subset \mathcal{C}$. Note that c_i is a ciphertext of m_i.*
3) *$t \leftarrow Tag(k, w)$: A deterministic algorithm for tag generation. It takes the secret key k, a keyword $w \in \mathcal{W}$ as input, and outputs a tag $t \in \mathcal{T}$. We sometimes write this as $t \leftarrow Tag_k(w)$.*
4) *$\mathcal{X} \leftarrow Search(\mathcal{I}, \mathcal{ED}, t)$: A deterministic algorithm for keyword search. It takes the identifier set $\mathcal{I}$, the ciphertext set $\mathcal{ED}$, and a tag t as input, and generates a subset $\mathcal{X} \subset \mathcal{I}$ of the identifier set.*
5) *$m_i \leftarrow Dec(k, c_i)$: A deterministic algorithm for decryption. It takes the secret key k and a ciphertext c_i as input, and outputs a document m_i.*

We assume that Π satisfies the following requirements:

Search correctness: For all $n \leq |\mathcal{M}|$, all $\tau \leq d$, all $k \leftarrow Gen(n, \tau)$, all $\mathcal{D} := \{m_1, \ldots, m_n\} \subset \mathcal{M}$, and all $w \in \mathcal{W}$, it holds

$$\Pr\left[\mathcal{D}(w) = Search(Enc(k, \mathcal{D}), Tag(k, w))\right] = 1.$$

Namely, *Search* always outputs the correct search result.

Decryption correctness: For all $n \leq |\mathcal{M}|$, all $\tau \leq d$, all $k \leftarrow Gen(n, \tau)$, all $\mathcal{D} := \{m_1, \ldots, m_n\} \subset \mathcal{M}$, all $(\mathcal{I}, \mathcal{ED} := \{c_1, \ldots, c_n\}) \leftarrow Enc(k, \mathcal{D})$, and all $i \in \{1, \ldots, n\}$, it holds

$$\Pr\left[m_i \leftarrow Dec(k, c_i)\right] = 1.$$

B. Security Definition

We define perfect secrecy against the server. As in the previous works [5], [6], we assume the following *honest-but-curious model*: The server never modifies stored (and encrypted) documents, correctly executes the *Search* algorithm, and returns appropriate encrypted documents; however, the server attempts to get some information on documents and keywords. Namely, we consider a security notion that ciphertexts $\mathcal{ED}$ and tags $(t_1, t_2, \ldots, t_\tau)$ used for keyword search leak no information on the underlying documents $\mathcal{D}$ and keywords $(w_1, w_2, \ldots, w_\tau)$. In this paper, we consider the above perfect privacy dependent on some subfamily of relation functions $\mathcal{F}_0 \subset \mathcal{F}$. Of course, SSE schemes for all relation functions $\mathcal{F}$ is the most practical. Indeed, all the previous works on (computationally secure) SSE treated all of the relation functions $\mathcal{F}$. However, there might be applications of SSE schemes for some restricted class $\mathcal{F}_0$ of relation functions. For

example, suppose some application in which each document on personal data contains exact three keywords: gender, age, and job. Namely, this application only requires an SSE scheme for a restricted class $\tilde{\mathcal{F}}$ such that for every $f \in \tilde{\mathcal{F}}$, it holds $|f(m)| = 3$ for every $m \in \mathcal{M}$. Therefore, we believe it is meaningful to consider constructions only secure for some restricted relations $\mathcal{F}_0$. Indeed, although our first construction is secure for some subfamily $\mathcal{F}_0 \subset \mathcal{F}$ (not $\mathcal{F}$ itself), it is much more efficient than our second construction, which is secure for all relation functions $\mathcal{F}$. It might be expected that a more efficient SSE scheme can be constructed by restricting a class of relation functions even in the computational security setting. We emphasize that this paper first explicitly describes relations between documents and keywords and considers a security notion dependent on the relations. In fact, a similar formalization (i.e., associating security with some function family) can be found in some contexts, e.g., key-dependent-message (KDM) security [15], [16].

We formally define security of (n,τ)-SSE as follows.

Definition 2 (Security of (n,τ)-SSE)**.** *For $\mathcal{F}_0 \subset \mathcal{F}$, an (n,τ)-SSE scheme Π is said to be $\mathcal{F}_0$-(n,τ)-secure if it satisfies the following equation for all $f \in \mathcal{F}_0$.*

$$H(W_1,\ldots,W_\tau,M_1,\ldots,M_n \mid T_1,\ldots,T_\tau,C_1,\ldots,C_n) = H(W_1,\ldots,W_\tau,M_1,\ldots,M_n).$$

In particular, when $\mathcal{F}_0 = \mathcal{F}$, it is simply said to be (n,τ)-secure.

This means that the honest-but-curious server cannot get any information on keywords and documents from tags and ciphertexts.

III. Lower Bound on Secret-Key Sizes

In this section, we show a lower bound on secret-key sizes required for an (n,τ)-secure SSE scheme. Let K be a random variable taking a value on a finite set $\mathcal{K}$.

Theorem 1. *Let Π be an (n,τ)-secure SSE scheme. Then, it holds*

$$H(K) \geq H(M_1,\ldots,M_n) + H(W_1,\ldots,W_\tau).$$

Proof. First of all, we show the following lemma.

Lemma 1. *For any $k \in \mathcal{K}$, the mapping $Tag_k : \mathcal{W} \to \mathcal{T}$ is injective.*

Proof. For any fixed $k \in \mathcal{K}$, suppose that there exist $w, w' \in \mathcal{W}$ such that $w \neq w'$ and $Tag_k(w) = Tag_k(w')$. Then, it is obvious that *Search* algorithm does not satisfy the search correctness, which implies contradiction. Therefore, for any fixed $k \in \mathcal{K}$, it holds $Tag_k(w) \neq Tag_k(w')$ if $w \neq w'$, and hence Tag_k is an injective mapping. □

From Lemma 1, $Tag_k : \mathcal{W} \to Tag_k(\mathcal{W})$ is a bijective mapping. Namely, there exists an inverse mapping of Tag_k. Therefore, we have the following lemma. In the following, let $Z := (T_1,\ldots,T_\tau,C_1,\ldots,C_n)$ for convenience.

Lemma 2. *For any (n,τ)-secure SSE scheme, it holds $H(W_1,\ldots,W_\tau,M_1,\ldots,M_n \mid Z,K) = 0$.*

Proof. We have

$$\begin{aligned} &H(W_1,\ldots,W_\tau,M_1,\ldots,M_n \mid Z,K) \\ &\quad = H(W_1,\ldots,W_\tau \mid Z,K) \\ &\qquad + H(M_1,\ldots,M_n \mid Z,K,W_1,\ldots,W_\tau) = 0, \end{aligned}$$

where the last equality follows from Lemma 1 and decryption correctness. □

Then, we have

$$\begin{aligned} H(K) &= H(K) \\ &\quad + H(W_1,\ldots,W_\tau,M_1,\ldots,M_n \mid Z,K) \qquad (1) \\ &\geq H(K \mid Z) + H(M_1,\ldots,M_n \mid Z,K) \\ &\quad + H(W_1,\ldots,W_\tau \mid Z,K,M_1,\ldots,M_n) \\ &= H(W_1,\ldots,W_\tau,M_1,\ldots,M_n,K \mid Z) \\ &\geq H(W_1,\ldots,W_\tau,M_1,\ldots,M_n \mid Z) \\ &= H(W_1,\ldots,W_\tau,M_1,\ldots,M_n) \qquad (2) \\ &= H(W_1,\ldots,W_\tau) + H(M_1,\ldots,M_n) \qquad (3) \end{aligned}$$

where (1) follows from Lemma 2, (2) follows from the security definition (Definition 2), and (3) follows from that W and M are independent. □

As we will see in the next section, the above lower bound is (asymptotically) tight since our construction will asymptotically meet the above bound with equality. Therefore, we define asymptotic optimality of (n,τ)-secure SSE schemes as follows.

Definition 3 (Asymptotic optimality)**.** *A construction of an (n,τ)-secure SSE scheme is said to be* asymptotically optimal *if it asymptotically attains the lower bound in Theorem 1 with equality.*

IV. Constructions

In this section, we show two constructions of (n,τ)-secure SSE schemes based on the one-time pad and random permutations. The first construction achieves the *asymptotically optimal* secret-key size, though it is (n,τ)-secure for some subset $\mathcal{F}_0 \subset \mathcal{F}$ of all relation functions $\mathcal{F}$. A concrete subset $\mathcal{F}_0$ will be described later. The second construction can be proved to be (n,τ)-secure for any relation functions $f \in \mathcal{F}$. However, the key size is quite large, and therefore the second construction is not optimal even in asymptotic sense.

In both constructions, we assume that not only secrets key but also documents and keywords are uniformly chosen (i.e., $P_{M_1,M_2,\ldots,M_n}$ and $P_{W_1,W_2,\ldots,W_\tau}$ are uniform).

A. Asymptotically Optimal Construction for Restricted Relation

Let $\mathbb{F}_{2^\lambda}$ be a finite field whose cardinality is 2^λ, where λ is a prime and $2^\lambda > n$. Let τ $(d = \tau)$ be a prime number, and $\mathbb{Z}_\tau := \{0,1,\ldots,\tau-1\}$. We suppose that each keyword

$w_i \in \mathcal{W}$ is encoded to $i-1 \in \mathbb{Z}_\tau$, and we write $w_i \in \mathbb{Z}_\tau$. Let $\Sigma = \{\sigma\}$ be a set of permutations over $\mathbb{Z}_\tau$. Namely,

$$\sigma = \begin{pmatrix} 0 & 1 & \cdots & \tau-1 \\ \sigma(0) & \sigma(1) & \cdots & \sigma(\tau-1) \end{pmatrix}.$$

We consider the following subfamily $\mathcal{F}_0 \subset \mathcal{F}$ such that every $f \in \mathcal{F}_0$ satisfies the following conditions:

(i) $|f(m)| = \alpha$ for any $m \in \mathcal{M}$, where $\alpha \in [d]$.
(ii) $|\{m \in \mathcal{M} \mid w \in f(m)\}| = \beta$ for any $w \in \mathcal{W}$, where $\beta \in [n]$.

Namely, for every $f \in \mathcal{F}_0$, each document $m \in \mathcal{M}$ contains exact α keywords, and each keyword $w \in \mathcal{W}$ is associated with exact β documents.

We show an asymptotically optimal $\mathcal{F}_0$-(n,τ)-secure SSE scheme with $\mathcal{M}$, $\mathcal{W}$, $\mathcal{C}$, and $\mathcal{T}$, where $\mathcal{W} = \mathcal{T} = \mathbb{Z}_\tau$ and $\mathcal{M} = \mathcal{C} = F_{2^\lambda}$, below.

1) $k \leftarrow Gen(n,\tau)$: The algorithm chooses $\sigma \in \Sigma$ uniformly at random, and sets $a_i := \sigma(w_i)$ $(1 \le i \le \tau)$. It also chooses $b_j \in \mathbb{F}_{2^\lambda}$ $(1 \le j \le n)$ uniformly at random. Finally, it outputs $k := (a_1, \ldots, a_\tau, b_1, \ldots, b_n)$.
2) $(\mathcal{I}, \mathcal{ED}) \leftarrow Enc(k, \mathcal{D})$: For each $j \in \{1, \ldots, n\}$, the algorithm executes the following procedure.
   1. It computes $ct_j := m_j + b_j$.
   2. It sets $\mathcal{Y}_j := \{a_i \mid w_i \in \mathcal{W}(m_j)\}$.[1]
   3. It sets $id(m_j) := j$ and $c_j := (j, ct_j, \mathcal{Y}_j)$, where $j \in \{0,1\}^{\lfloor \log n \rfloor + 1}$.

   It outputs $\mathcal{I} = \{id(m_1), \ldots, id(m_n)\}$ and $\mathcal{ED} = \{c_1, \ldots, c_n\}$.
3) $t \leftarrow Tag(k, w)$: Suppose that w is i-th element of $\mathcal{W}$ in the lexicographic order. It outputs $t := a_i$.
4) $\mathcal{X} \leftarrow Search(\mathcal{I}, \mathcal{ED}, t)$: It outputs $\mathcal{X} := \{id(m_j) \in \mathcal{I} \mid t \in \mathcal{Y}_j\}$.
5) $m_i \leftarrow Dec(k, c_i)$: It extracts ct_i from c_i. It outputs $m_i = ct_i - b_i$.

Theorem 2. *An (n,τ)-SSE scheme Π by the above construction is $\mathcal{F}_0$-(n,τ)-secure.*

Proof. Let $\mathcal{L}(w) := \{i \mid m_i \in \mathcal{D}, w \in f(m_i)\}$ for $\mathcal{D} = \{m_1, m_2, \ldots, m_n\}$ and $w \in \mathcal{W}$. For simple analysis, assume $\mathcal{D}$ such that for any $w \in f(m_n)$, it holds $|\{i \mid m_i \in \mathcal{D} \setminus \{m_n\}, w \in f(m_i)\}| \le \beta - 1$, where $\mathcal{D} = \{m_1, m_2, \ldots, m_n\}$. Namely, each keyword associated with m_n is also associated with at most $\beta - 1$ other keywords. Note that the following proof can be applied in the case of any $\mathcal{D}$. If $|\{\mathcal{L}(w) = \beta \mid w \in \mathcal{W}\}| = j$, then for any $\mathcal{D}$ that satisfies the above, it holds

$$\Pr[Y_1 = \mathcal{Y}_1, Y_2 = \mathcal{Y}_2, \ldots, Y_n = \mathcal{Y}_n] = \frac{1}{\binom{\tau}{\alpha}\binom{\tau}{\alpha}\cdots\binom{\tau-j}{\alpha}},$$

where Y_i is a random variable of $\mathcal{Y}_i$, since $P_{M_1,M_2,\ldots,M_n}$ is uniform.

We describe C_i as (CT_i, Y_i). In the construction, it clearly holds $H(M_1, \ldots, M_n | CT_1, \ldots, CT_n) = H(M_1, \ldots, M_n)$ since each document is encrypted by the one-time pad. Therefore, for any $\mathcal{Y}_1, \ldots, \mathcal{Y}_n, ct_1, \ldots, ct_n$ it holds

$$\begin{aligned}&\Pr[\mathcal{Y}_1, \ldots, \mathcal{Y}_n, ct_1, \ldots, ct_n, t_1, \ldots, t_\tau, w_1, \ldots, w_j] \\ &\quad = \Pr[\mathcal{Y}_1, \ldots, \mathcal{Y}_n, t_1, \ldots, t_\tau, w_1, \ldots, w_j] + \Pr[ct_1, \ldots, ct_n],\end{aligned}$$

since $ct_1, \ldots, ct_n$ leak no information on documents (and hence tags and keywords).

It clearly holds $\Pr[ct_1, \ldots, ct_n] = (1/2^\lambda)^n$ for any $(ct_1, \ldots, ct_n)$ due to security of the one-time pad.

For any $j \in [\tau]$, for all $\mathcal{Y}_1, \ldots, \mathcal{Y}_n, t_1, \ldots, t_\tau, w_1, \ldots, w_j$:

$$\begin{aligned}&\Pr[t_1, \ldots, t_\tau, w_1, \ldots, w_j \mid \mathcal{Y}_1, \ldots, \mathcal{Y}_n] \\ &\qquad = \Pr[t_1, \ldots, t_\tau, w_1, \ldots, w_j] \qquad (4) \\ &\qquad = \Pr[t_1, \ldots, t_\tau] + \Pr[w_1, \ldots, w_j], \qquad (5)\end{aligned}$$

where (4) follows from conditions (i) and (ii) of f, from that $P_{M,..,M}$ is uniform, and from that M_i and W_j are independent for arbitrary $i \in [n]$ and $j \in [\tau]$, and (5) follows from that σ is chosen randomly.

For any distinct $t_1, \ldots, t_\tau$, it holds $\Pr[t_1, \ldots, t_\tau] = 1/\tau!$, and for any distinct $w_1, \ldots, w_j$, it holds $\Pr[w_1, \ldots, w_j] = 1/\tau(\tau-1)\cdots(\tau-j+1)$. Therefore, for any $\mathcal{Y}_1, \ldots, \mathcal{Y}_n, ct_1, \ldots, ct_n, t_1, \ldots, t_\tau, w_1, \ldots, w_j$ it holds

$$\begin{aligned}&\Pr[\mathcal{Y}_1, \ldots, \mathcal{Y}_n, ct_1, \ldots, ct_n, t_1, \ldots, t_\tau, w_1, \ldots, w_j] \\ &\quad = \frac{1}{\binom{\tau}{\alpha}\binom{\tau}{\alpha}\cdots\binom{\tau-j}{\alpha}} \cdot \frac{1}{(2^\lambda)^n} \cdot \frac{1}{\tau!} \cdot \frac{1}{\tau(\tau-1)\cdots(\tau-j+1)}.\end{aligned}$$

Let $Z := (Y_1, \ldots, Y_n, CT_1, \ldots, CT_n, T_1, \ldots, T_\tau)$ and $\mathcal{Z} := (2^{\mathcal{W}_\alpha})^n \times \mathcal{C}^n \times \mathcal{T}^\tau$, where $2^{\mathcal{W}_\alpha} := \{\mathcal{Y} \in 2^{\mathcal{W}} \mid |\mathcal{Y}| = \alpha\}$. The left-hand side of Definition 2 can be described as follows.

$$\begin{aligned}&H(M_1, \ldots, M_n, W_1, \ldots, W_\tau \mid Z) \\ &\quad = \sum_{i=1}^{\tau} H(W_i \mid Z, W_1, \ldots, W_{i-1}) \\ &\qquad + \sum_{i=1}^{n} H(M_i \mid Z, W_1, \ldots, W_\tau, M_1, \ldots, M_{i-1}).\end{aligned}$$

We prove that $H(W_i \mid Z, W_1, \ldots, W_{i-1}) = H(W_i \mid W_1, \ldots, W_{i-1})$ for any $i \in [\tau]$. For $H(W_i | Z, W_1, \ldots W_{i-1})$, it holds

$$\begin{aligned}&H(W_i | Z, W_1, \ldots W_{i-1}) \\ &= -\sum_{w_1 \in \mathcal{W}} \cdots \sum_{w_i \in \mathcal{W} \setminus \{w_j\}_{j=1}^{i-1}} \sum_{z \in \mathcal{Z}} \\ &\quad \Pr[W_1 = w_1, \ldots, W_{i-1} = w_{i-1}, Z = z] \\ &\qquad \Pr[W_i = w_i \mid W_1 = w_1, \ldots, W_{i-1} = w_{i-1}, Z = z] \\ &\qquad\quad \log \Pr[W_i = w_i \mid W_1 = w_1, \ldots, W_{i-1} = w_{i-1}, Z = z] \\ &= -\sum_{w_1 \in \mathcal{W}} \cdots \sum_{w_i \in \mathcal{W} \setminus \{w_j\}_{j=1}^{i-1}} \sum_{z \in \mathcal{Z}} \\ &\qquad \frac{1}{\binom{\tau}{\alpha}\binom{\tau}{\alpha}\cdots\binom{\tau-j}{\alpha}} \cdot \frac{1}{(2^\lambda)^n} \cdot \frac{1}{\tau!} \\ &\qquad\quad \cdot \frac{1}{\tau(\tau-1)\cdots(\tau-i+2)} \cdot \frac{1}{\tau-i+1} \cdot \log \frac{1}{\tau-i+1}\end{aligned}$$

[1] Note that a user knows i-th keyword (in the lexicographic order) corresponds to a_i since he/she knows the sequence $(a_1, \ldots, a_\tau)$ corresponding to $\mathcal{W}$.

$= \log(\tau - i + 1)$.

Also, for $H(W_i \mid W_1, \ldots, W_{i-1})$ it holds

$$
\begin{aligned}
&H(W_i \mid W_1, \ldots, W_{i-1}) \\
&= - \sum_{w_1 \in \mathcal{W}} \cdots \sum_{w_i \in \mathcal{W} \setminus \{w_j\}_{j=1}^{i-1}} \\
&\quad \Pr[W_1 = w_1, \ldots, W_{i-1} = w_{i-1}] \\
&\quad\quad \Pr[W_i = w_i \mid W_1 = w_1, \ldots, W_{i-1} = w_{i-1}] \\
&\quad\quad\quad \log \Pr[W_i = w_i \mid W_1 = w_1, \ldots, W_{i-1} = w_{i-1}] \\
&= - \sum_{w_1 \in \mathcal{W}} \cdots \sum_{w_i \in \mathcal{W} \setminus \{w_j\}_{j=1}^{i-1}} \\
&\quad \frac{1}{\tau \cdots (\tau - i + 2)} \cdot \frac{1}{\tau - i + 1} \cdot \log \frac{1}{\tau - i + 1} \\
&= \log(\tau - i + 1).
\end{aligned}
$$

Therefore, for any $i \in [\tau]$, $H(W_i \mid Z, W_1, \ldots, W_{i-1}) = H(W_i \mid W_1, \ldots, W_{i-1})$.

Similarly, we can prove that $H(M_i \mid Z, W_1, \ldots, W_\tau, M_1, \ldots, M_{i-1}) = H(M_i \mid W_1, \ldots, W_\tau, M_1, \ldots, M_{i-1})$ for any $i \in [n]$.

Hence, the construction is $\mathcal{F}_0$-(n, τ)-secure since $H(M_1, \ldots, M_n, W_1, \ldots, W_\tau | I_1, \ldots, I_n, C_1, \ldots, C_n, T_1, \ldots, T_\tau) = H(M_1, \ldots, M_n, W_1, \ldots, W_\tau)$. □

B. Fully Secure Construction for Any Relation

The security of the first construction relies on the restricted class of relation functions $\mathcal{F}_0$. Specifically, the reasons why the number of tags attached to each ciphertext leaks no information on the underlying plaintexts in the first construction are as follows: (1) Each ciphertext contains the same number of tags; (2) the number of documents associated with each keyword is the same; (3) $P_{M_1, M_2, \ldots, M_n}$ is uniform. Therefore, our strategy of the second construction, which is secure for any relation function $f \in \mathcal{F}$, is explained as follows: (1) to make the number of keywords associated with each document equal by adding dummy elements to tags associated with each ciphertext, and (2) to prepare multiple tags for the same keyword to make all tags attached to ciphertexts different.

Let $\mathbb{F}_{2^\lambda}$ be a finite field whose cardinality is 2^λ $(> n)$. Let $\ell := \max_{m \in \mathcal{M}} |\mathcal{W}(m)|$ be the maximum number of keywords associated with each document, and let $s := \ell n$, which indicates the number of all the tags attached to ciphertexts after adding dummies. In other words, the total number of tags attached to ciphertexts $\mathcal{ED}$ is s for any $\mathcal{D} \subset \mathcal{M}$. For simplicity, we assume d, ℓ, n are powers of two, respectively, and hence s is also a power of two. Let $\Sigma = \{\sigma\}$ be a set of permutations over $\{0,1\}^{\log(d\ell n)}$. We show an (n, τ)-SSE scheme with $\mathcal{M}, \mathcal{W}, \mathcal{C}$, and $\mathcal{T}$, where $\mathcal{M} = \mathcal{C} = \mathbb{F}_{2^\lambda}, \mathcal{W} = \{0,1\}^{\log d}$ and $\mathcal{T} = \{0,1\}^{\log(d\ell n)}$ below.

1) $k \leftarrow Gen(n, \tau)$: It chooses $\sigma \in \Sigma$ uniformly at random, and sets $a_{h,i,j} = \sigma(w_h \| i \| j)$ $(1 \le h \le d,\ 0 \le i \le \ell - 1,\ 1 \le j \le n)$. Let $\mathcal{A} := \{\{a_{h,0,j}\}_{j=1}^n\}_{h=1}^d$ and $\bar{\mathcal{A}} := \{\{a_{j,i,j}\}_{i=1}^{\ell-1}\}_{j=1}^n$. The latter will be used for dummy elements. It also chooses $b_j \in \mathbb{F}_{2^\lambda}$ $(1 \le j \le n)$ uniformly at random. Finally, it outputs $k = (\mathcal{A}, \bar{\mathcal{A}}, b_1, b_2, \ldots, b_n)$.
2) $(\mathcal{I}, \mathcal{ED}) \leftarrow Enc(k, \mathcal{D})$: Let $s' := \sum_{m \in \mathcal{D}} |\mathcal{W}(m)|$. For each $j \in \{1, \ldots, n\}$, the algorithm executes the following procedure.
   1. It computes $ct_j := m_j + b_j$.
   2. It computes $\mathcal{Y}_j := \{a_{i,0,j} \in \mathcal{A} \mid w_i \in \mathcal{W}(m_j)\}$.
   3. If $s' < s$, it adds $\{a_{j,i,j}\}_{i=1}^{\ell - |\mathcal{W}(m_j)|} \subset \bar{\mathcal{A}}$ to $\mathcal{Y}_j$.
   4. It sets $id(m_j) := j$ and $c_j := (j, \mathcal{Y}_j, ct_j)$, where $j \in \{0,1\}^{\log n}$.

   It outputs $\mathcal{I} = \{id(m_1), \ldots, id(m_n)\}$ and $\mathcal{ED} = \{c_1, \ldots, c_n\}$.
3) $t \leftarrow Tag(k, w)$: Suppose that w is the i-th element of $\mathcal{W}$ in the lexicographic order. It outputs $t = (a_{i,0,1}, \ldots a_{i,0,n})$.
4) $\mathcal{X} \leftarrow Search(\mathcal{I}, t)$: For $t = (t_1, t_2, \ldots, t_n)$, it outputs $\mathcal{X} := \bigcup_{i=1}^n \{id(m_j) \in \mathcal{I} \mid t_i \in \mathcal{Y}_j\}$.
5) $m_i \leftarrow Dec(k, c_i)$: It extracts ct_i from c_i. It outputs $m_i = ct_i - b_i$.

Theorem 3. *An (n, τ)-SSE scheme Π by the above construction is (n, τ)-secure.*

Proof. In the above construction, tag values are all different regardless of whether the keywords attached to the document are the same or different, and the number of tags attached to all encrypted documents is equal. Therefore, from adversary's viewpoint, the situation can be regarded as that of the first construction for $\mathcal{F}_0$ such that $\alpha = \ell$ and $\beta = 1$. Hence, we can prove this theorem as in Theorem 2. □

V. Asymptotic (Non-)Optimality of Constructions

We evaluate the secret-key sizes of both the constructions.

Proposition 1. *The secret-key size in the $\mathcal{F}_0$-(n, τ)-secure SSE scheme proposed in Section IV-A is given by*

$$\log |\mathcal{K}| = \lambda n + \tau(\lfloor \log \tau \rfloor + 1).$$

Moreover, the construction in Section IV-A is asymptotically optimal.

Proof. We obviously have $\log |\mathcal{K}| = \lambda n + \tau \log(\lfloor \tau \rfloor + 1)$.

We evaluate the entropy of the secret key as follows. Let $A_1, \ldots, A_\tau, B_1, \ldots, B_n$ be random variables of $a_1, \ldots, a_\tau, b_1, \ldots, b_n$ in the construction IV-A. We have

$$
\begin{aligned}
&H(K) = H(A_1, \ldots, A_\tau, B_1, \ldots, B_n) \\
&= - \sum_{a_1 \in \mathbb{F}_\tau} \cdots \sum_{a_\tau \in \mathbb{F}_\tau \setminus \{a_j\}_{j=1}^{\tau-1}} \sum_{b_1 \in \mathbb{F}_{2^\lambda}} \cdots \sum_{b_n \in \mathbb{F}_{2^\lambda}} \\
&\quad \Pr[A_1 = a_1, \ldots, A_\tau = a_\tau, B_1 = b_1, \ldots, B_n = b_n] \\
&\quad\quad \log \Pr[A_1 = a_1, \ldots, A_\tau = a_\tau, B_1 = b_1, \ldots, B_n = b_n] \\
&= - \sum_{a_1 \in \mathbb{F}_\tau} \cdots \sum_{a_\tau \in \mathbb{F}_\tau \setminus \{a_j\}_{j=1}^{\tau-1}} \sum_{b_1 \in \mathbb{F}_{2^\lambda}} \cdots \sum_{b_n \in \mathbb{F}_{2^\lambda}} \\
&\quad\quad\quad \cdot \frac{1}{\tau! \cdot (2^\lambda)^n} \log \frac{1}{\tau! \cdot (2^\lambda)^n}
\end{aligned}
$$

$$= \log(\tau! \cdot 2^{\lambda n}).$$

On the other hand, we have

$$\begin{aligned}
&H(M_1, \dots M_n) + H(W_1, \dots, W_\tau) \\
&= - \sum_{m_1 \in \mathbb{F}_{2^\lambda}} \cdots \sum_{m_n \in \mathbb{F}_{2^\lambda} \setminus \{m_j\}_{j=1}^{n-1}} \\
&\qquad \Pr[M_1 = m_1, \dots, M_n = m_n] \\
&\qquad\qquad \cdot \log \Pr[M_1 = m_1, \dots, M_n = m_n] \\
&\quad - \sum_{w_1 \in \mathbb{F}_\tau} \cdots \sum_{w_\tau \in \mathbb{F}_\tau \setminus \{w_j\}_{j=1}^{\tau-1}} \\
&\qquad \Pr[W_1 = w_1, \dots, W_\tau = w_\tau] \\
&\qquad\qquad \cdot \log \Pr[W_1 = w_1, \dots, W_\tau = w_\tau] \\
&= - \sum_{m_1 \in \mathbb{F}_{2^\lambda}} \cdots \sum_{m_n \in \mathbb{F}_{2^\lambda} \setminus \{m_j\}_{j=1}^{n-1}} \\
&\qquad \frac{1}{(2^\lambda) \cdots ((2^\lambda) - n + 1)} \log \frac{1}{(2^\lambda) \cdots ((2^\lambda) - n + 1)} \\
&\qquad - \sum_{w_1 \in \mathbb{F}_\tau} \cdots \sum_{w_\tau \in \mathbb{F}_\tau \setminus \{w_j\}_{j=1}^{\tau-1}} \frac{1}{\tau!} \log \frac{1}{\tau!} \\
&= \log(\tau! \cdot 2^\lambda (2^\lambda - 1) \cdots (2^\lambda - n + 1)).
\end{aligned}$$

Hence, the construction in Section IV-A is asymptotically optimal in the sense of Definition 3. □

Proposition 2. *The secret-key size in the (n, τ)-secure SSE scheme proposed in Section IV-B is given by*

$$\log |\mathcal{K}| = \lambda n + n(d + \ell - 1) \log(d\ell n),$$

where $\ell := \max_{m \in \mathcal{M}} |\mathcal{W}(m)|$.

Proof. In the construction, the sizes of each $a_{h,i,j}$ and b_j are $\log(d\ell n)$ bits and λn bits, respectively. Since $|\mathcal{A}| = dn$ and $|\bar{\mathcal{A}}| = n(\ell - 1)$, we have $\log |\mathcal{K}| = \lambda n + n(d + \ell - 1) \log(d\ell n)$.

We can check that this construction is not asymptotically optimal by evaluating the entropy of the secret key, however, we omit it due to the space limitation. □

VI. Conclusion

In this paper, we first introduced unconditionally secure searchable symmetric encryption (SSE). In addition, we derived the lower bound on the secret-key sizes, and presented two construction. One achieves the *asymptotically optimal* secret-key size, though it is secure only when each document is associated with the same number of keywords and each keyword is associated with the same number of documents. The other construction can be proved to be (n, τ)-secure regardless of such a restriction. However, the key size is quite large, and therefore it is not optimal even in the asymptotic sense. It would be interesting to realize an (asymptotically) optimal (n, τ)-secure SSE scheme for any relation function.

Acknowledgment

This work (Junji Shikata) is in part supported by JSPS KAKENHI Grant Number JP15H02710, and it is partialy supported by the MEXT Program for Promoting the Reform of National Universities. This work is also in part supported by Research Grant Program of KDDI Foundation. This work (Yohei Watanabe) is also in part supported by JSPS KAKENHI Grant Number JP16J10532 and Yohei Watanabe is supported by JSPS Research Fellowships for Young Scientists.

References

[1] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Advances in Cryptology – EUROCRYPT 2004*, vol. 3027. Springer Berlin Heidelberg, 2004, pp. 506–522.

[2] M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi, "Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions," in *Advances in Cryptology – CRYPTO 2005*, vol. 3621. Springer Berlin Heidelberg, 2005, pp. 205–222.

[3] ——, "Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions," *Journal of Cryptology*, vol. 21, no. 3, pp. 350–391, 2007.

[4] Z. Lv, C. Hong, M. Zhang, and D. Feng, "Expressive and secure searchable encryption in the public key setting," in *Information Security*, vol. 8783. Springer International Publishing, 2014, pp. 364–376.

[5] R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, New York, NY, USA: ACM, 2006, pp. 79–88.

[6] ——, "Searchable symmetric encryption: Improved definitions and efficient constructions," *Journal of Computer Security*, vol. 19, no. 5, pp. 895–934, 2011.

[7] M. Naveed, M. Prabhakaran, and C. Gunter, "Dynamic searchable encryption via blind storage," in *IEEE Symposium on Security and Privacy*, May 2014, pp. 639–654.

[8] S. Kamara, C. Papamanthou, and T. Roeder, "Dynamic searchable symmetric encryption," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, New York, NY, USA: ACM, 2012, pp. 965–976.

[9] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, "Recommendation for key management – part 1: General (revision 3)," NIST Special Publication 800-57, July 2012, available at http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57_part1_rev3_general.pdf.

[10] S. Sedghi, J. Doumen, P. Hartel, and W. Jonker, "Towards an information theoretic analysis of searchable encryption," in *Information and Communications Security*, vol. 5308. Springer Berlin Heidelberg, 2008, pp. 345–360.

[11] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious RAMs," *Journal of the ACM*, vol. 43, no. 3, pp. 431–473, May 1996.

[12] R. Ostrovsky, "Efficient computation on oblivious RAMs," in *Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing*, New York, NY, USA: ACM, 1990, pp. 514–523.

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

[14] E.-J. Goh, "Secure indexes," Cryptology ePrint Archive, Report 2003/216, 2003, http://eprint.iacr.org/.

[15] J. Black, P. Rogaway, and T. Shrimpton, "Encryption-scheme security in the presence of key-dependent messages," in *Selected Areas in Cryptography*, Springer Berlin Heidelberg, 2003, pp. 62–75.

[16] J. Camenisch and A. Lysyanskaya, "An efficient system for non-transferable anonymous credentials with optional anonymity revocation," in *Advances in Cryptology — EUROCRYPT 2001*, Springer Berlin Heidelberg, 2001, pp. 93–118.