Improved One-Round Phrase Search Schemes Over Symmetrically Encrypted Data in Storage Outsourcing System

Ling Shen[1,2(✉)] and Jie Wang[3]

[1] Department of Computer Science,
Wuhan Donghu University, Wuhan 430212, China
aleenapple@163.com
[2] Wuhan University, Wuhan 430072, China
[3] Department of Computer Science,
University of Massachusetts Lowell, Lowell, MA 01854, USA

Abstract. Phrase search schemes over encrypted data are efficient methods for protecting users' data privacy in storage outsourcing systems. We analyze and improve two one-round phrase search schemes over encrypted data. One drawback of F. Kong's phrase search scheme is that it cannot support single-keyword search. We propose a modification and the improved scheme supports both single-keyword and phrase search. In M. Li's phrase search scheme, a simple unkeyed hash function is used instead of a keyed pseudo-random function. We present a security analysis and show that the usage of a keyed pseudo-random function can resist the leakage of some useful information. These phrase search schemes can be applied in cloud storage.

Keywords: Cloud storage · Information security · Searchable symmetric encryption · Phrase search

1 Introduction

The Internet of Things (IoT) and mobile technology are integrating various access devices such as radio-frequency identification (RFID) tags, sensors, smart phones, tablets, and wearable devices into a global network infrastructure [1–3]. The sharp increase of IoT, mobile devices, and social-networking services produces large amounts of data in either structured or unstructured format. While cloud computing and storage services [4] provides powerful, reliable, and on-demand computing and storage resources, users can outsource complex computation and their data to the public or private cloud services. For example, Microsoft's Azure storage service and Amazon's Simple Storage Service (S3) Storage services are public cloud storage services.

However, despite its benefits and popularity, cloud services face many security and privacy threats [5–9]. When users store their data into the remote cloud, they suspect that the cloud service providers may obtain their data. To protect their data privacy, users encrypt the data using secret cryptographic keys and send the cipher-text data to the cloud. The fundamental problem is how users can retrieve only files containing certain

F. Chen and Y. Luo (Eds.): Industrial IoT 2017, LNICST 202, pp. 205–213, 2017.
DOI: 10.1007/978-3-319-60753-5_22

keywords from the remote cloud. As the data has been encrypted by users, the cloud is required to perform search operations over encrypted data. The Fully-Homomorphic Encryption Schemes (FHE) [10, 11], which can support any computation on encrypted data, maybe one of the most amazing solutions. However, the inefficiency of FHE schemes makes them hard to use in practice.

Searchable Symmetric Encryption (SSE) proposed by D. Song, D. Wagner, and A. Perrig [12] is a very efficient scheme for secure storage outsourcing. In SSE scheme, the remote untrusted server can perform searches on encrypted data and no plain-text data is leaked. In [13], R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky gave two formal security definitions for Searchable Symmetric Encryption (SSE) schemes, called non-adaptive indistinguishability security and adaptive indistinguishability security.

Multi-keyword search over encrypted data [14, 15] is a natural expansion to single-keyword search. In multi-keyword search scheme, also called conjunctive keyword search, the documents containing all of the query keywords are returned. The difference between phrase search and multi-keyword search is that phrase search requires the query keywords occur consecutively in the document. In 2012, S. Zittrower and C.C. Zou [16] introduced a phrase search scheme on encrypted data by storing keyword-location value for each keyword. Y. Tang, D. Gu, N. Ding, and H. Lu [17] proposed a two-phase phrase search scheme, which was provably secure in the non-adaptive setting. Z.A. Kissel and J. Wang [18, 19] presented an efficient verifiable phrase search scheme and a single-round phrase search scheme. In [20], F. Kong and J. Wang gave an analysis and improvement to Z. A. Kissel's scheme. M. Li et al. [21] proposed the LPSSE phrase search scheme by combing R. Curtmola's secure linked list structure and keywords' relative position information. H.T. Poon and A. Miri [22, 23] adopted Bloom filters to reduce the storage amount in their phrase search schemes.

In this paper, we propose further improvements of two one-round phrase search schemes, known as F. Kong's scheme and M. Li's scheme. In F. Kong's phrase search scheme, users cannot perform single-keyword search. We propose an improvement to support both single-keyword and phrase search by combing R. Curtmola's method. In M. Li's phrase search scheme, a simple unkeyed hash function $h()$ is used instead of a keyed pseudo-random function $h_s()$ in Y. Tang's scheme. We give a security analysis and show that some information may be leaked when unkeyed hash function $h()$ is adopted. Thus it is better to use the keyed pseudo-random function $h_s()$.

The rest of the paper is organized as follows. In Sect. 2, we review searchable symmetric encryption schemes. In Sect. 3, we analyze F. Kong's phrase search scheme and propose an improvement to support single-keyword search. In Sect. 4, we give the security analysis of M. Li's phrase search scheme and present the improvement method. Finally, we give the conclusion and future research work in Sect. 5.

2 Searchable Symmetric Encryption Schemes

Storage-as-a-Service (STaaS) is known as a kind of cloud service providing users with storage outsourcing. When users want to retrieve some encrypted documents from the cloud, they should send the query to the cloud, who can perform searches on encrypted

data and return the result to the users. Searchable symmetric encryption schemes [12–23] are practical solutions for secure storage outsourcing.

2.1 Searchable Encryption Schemes and Security Definitions

D. Song, D. Wagner, and A. Perrig [12] presented practical methods for searches on encrypted data by scanning the cipher-text documents.

R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky [13] constructed two index-based Searchable Symmetric Encryption (SSE) schemes by using linked lists and look-up tables. They also gave two kinds of security definitions of searchable symmetric encryption schemes, known as non-adaptive indistinguishability and adaptive indistinguishability.

R. Curtmola et al. gave the conception of *History*, *Access Pattern*, and *Search Pattern* [13]. The trace of a history H_q denotes the information leaked about the history. The trace consists of the access and search patterns.

It was noted [13] that the accurate security definition for SSE is that nothing is leaked except the access pattern and the search pattern. The detailed descriptions of non-adaptive indistinguishability and adaptive indistinguishability are seen in [13].

2.2 Phrase Search Schemes Over Encrypted Data

In conjunctive keyword or multi-keyword search schemes on encrypted data [14, 15], the cloud tries to retrieve those documents containing all the keywords according to the query sent by users. In phrase search schemes on encrypted data [16–23], it is required that the keywords occur consecutively in the documents. Thus, we should preserve the location information of each keyword in the documents to judge whether the keywords occur as a phrase.

There are two kinds of methods of recording the location information of each keyword in the documents. One method [16] is to record the sequence number of each word in the documents. Another method [17] is to construct a relative position table of all the words in the documents.

S. Zittrower and C.C. Zou's phrase search scheme on encrypted data [16] stores the encrypted keyword-location information for each keyword in the documents. To resist statistical cryptanalysis, they truncated the encrypted words to a predefined number of bits.

In 2012, Y. Tang, D. Gu, N. Ding, and H. Lu [17] proposed a two-round phrase search protocol on encrypted data by using a binary matrix and a look-up table. In the first round, it is required to get a set of candidate documents containing each word in the phrase. In the second round, the cloud will find the documents containing the phrase by testing whether the keywords occur as a phrase.

In [19], Z.A. Kissel proposed a single-round phrase search scheme by using the encrypted next-word lists, which is an improved inverted index structure. In Z.A. Kissel's scheme, there are a set of postings list containing the document identifiers, number of occurrences, and the locations of each pair of words.

The idea of F. Kong's improved scheme [20] is to construct the encrypted next-word index for each pair of words (w_s, w_t) directly instead of using the next-word list N. The data structure of F. Kong's scheme is (A, P), where A is the encrypted list of all pairs of words (w_s, w_t) and P is the postings list of the corresponding locations of each pair of words in the documents.

M. Li et al.'s LPSSE phrase search scheme [21] is constructed based on R. Curtmola's scheme by adding the relative location information of words to the inverted index.

3 Improved F. Kong's Phrase Search Scheme Over Encrypted Data

In this section, we modify F. Kong's phrase search scheme to support one-keyword search over encrypted data.

3.1 The Idea of Our Modification

In R. Curtmola's non-adaptively secure SSE scheme [13], the encrypted index consists of an array of words and a look-up table. Each word w_i has a linked list which contains all the identifiers of the documents in $D(w_i)$. When the user wants to get the documents containing keyword w_i, he/she sends the query to the remote server. The server will find the linked list L_i of keyword w_i, and obtain all the document identifiers in the list L_i.

In [19], Z.A. Kissel's one-round phrase search scheme constructs the encrypted next-word lists to preserve the locations of each pair of words. In a next-word index, each word w_i is followed by a list of succeeding words $w_{i,1} \ldots w_{i,j} \ldots$ and the locations where a pair of adjacent words (w_i, $w_{i,j}$) occur. To perform one keyword search, the server must scan all the nodes in the next-word list of keyword w_i and form a set of all the document identifiers.

In F. Kong's phrase search scheme [20], the next-word list is built for each pair of encrypted words (w_i, w_j) instead of each word. For example, the server cannot know whether a pair of encrypted words (w_i, w_j) contain a keyword w_i or w_j. Therefore, the server cannot find the documents where a keyword w_i occurs.

The idea of our modification is adding the next-word list of each word to the index of pairs of words. Thus no matter one-keyword search or phrase search, the server can find the corresponding documents. We note that only document identifiers are recorded in the next-word list of each word and no one-word location information is leaked.

3.2 Our Improved Phrase Search Scheme

We use the similar notations and definitions in [19, 20]. Let $Pair_i$ denote a pair of words (w_s, w_t). Similar to F. Kong's scheme [20], our improved scheme also constructs an encrypted index (A, P), which consists of two arrays A and P. The difference is that the information of each word is added to the arrays A and P in our scheme.

Let $D = \{D_1, D_2, \ldots, D_n\}$ be the document collection and $\Delta = \{w_1, w_2, \ldots, w_d\}$ be the set of d words. Let $D(w_i)$ be the set of documents identifiers that contain keyword w_i. Let $p = (w_1, w_2, \ldots, w_n)$ be a phrase consisting of n words and $D(p_i)$ be the set of all the documents containing the phrase p_i. Let (G, E, D) denote a secure symmetric encryption algorithm. Three cryptographic pseudo-random functions φ, f, and ζ are described as follows:

φ: $\{0, 1\}^k \times \{0, 1\}^{\lg(m|\Delta|)} \rightarrow \{0, 1\}^{\lg(m|\Delta|)}$,
f: $\{0, 1\}^k \times \{0, 1\}^p \rightarrow \{0, 1\}^{k+\log(m|\Delta|)}$,
ζ: $\{0, 1\}^k \times \{0, 1\}^p \rightarrow \{0, 1\}^{\lg(|\Delta|)}$.

The improved one-round phrase search scheme on encrypted data consists of four probabilistic polynomial-time algorithms, similar as the schemes in [19, 20]:

Step 1 – Key generation: The client generates three random cryptographic keys x, y, and ω from the space $\{0, 1\}^k$.

Step 2 – Building the Index: The client builds the encrypted next-word index (A, P) as follows. Then the client sends the (A, P) and all the encrypted documents to the remote cloud.

(i) The head list A:

We create the element nodes in the array A for each pair of words and single word respectively.

For a pair of words $Pair_i$, the element in $A[\zeta_x(Pair_i)]$ is $(k_{i,0}\|\varphi_\omega(s_i)) \oplus f_y(Pair_i)$. It is noted that $\varphi_\omega(s_i)$ is the address of the head node of the pair of words $Pair_i$ and $k_{i,0}$ is the cryptographic key for encrypting the corresponding posting list P.

Similarly, for each word w_i, the element in $A[\zeta_x(w_i)]$ is $(k_{i,0}\|\varphi_\omega(s_i)) \oplus f_y(w_i)$, where $\varphi_\omega(s_i)$ is the address of the head node of the word w_i and $k_{i,0}$ is the key for encrypting the corresponding posting list P.

(ii) The posting list P:

For a single word w_i, we store only the document identifiers containing w_i in the array P. For a pair of words $Pair_i$, we store not only the document identifiers containing w_i but also the location information in the array P.

The element in $P[\varphi_\omega(c)]$ is $E(n_{i,j})$ encrypted with the cryptographic key $k_{i,j-1}$. For a pair of words, we have

$n_{i,j} = \mathrm{id}(d)\| l \| k_{i,j}\|\varphi_\omega(c + 1)$.
For a word w_i, we have
$n_{i,j} = \mathrm{id}(d)\| k_{i,j} \|\varphi_\omega(c + 1)$.

The counter c is initialized to 1. The identifier $\mathrm{id}(d)$ is the identifier for the document $d \in D(Pair_i)$ or $D(w_i)$. For a phrase, l is the location of the two-word phrase $Pair_i$ in the document d. The cryptographic key $k_{i,j}$ is the key for encrypting the next node of the posting list.

Step 3 – Generating the Trapdoor: The client computes the search trapdoor T_p for the phrase $p = (w_1, w_2, \ldots, w_n)$ and sends the trapdoor T_p to the remote cloud:

$$T_p = \left\{ \left(\zeta_x(Pair_1), f_y(Pair_1)\right), \left(\zeta_x(Pair_2), f_y(Pair_2)\right), \ldots, \left(\zeta_x(Pair_{n-1}), f_y(Pair_{n-1})\right) \right\},$$

where $Pair_i$ is the pair of words (w_i, w_{i+1}) with $1 \leq i < n$.

For a single keyword w_i, the client computes the search trapdoor T_p for the keyword w_i and sends the trapdoor $T_{wi} = \{(\zeta_x(w_i), f_y(w_i))\}$to the remote cloud.

Step 4 – Performing the Search: Once the cloud receives the trapdoor T_{wi} or T_p, it performs the query and returns all the documents containing the phrase $p = (w_1, w_2, \ldots, w_n)$ or the keyword w_i to the client.

Now we explain that the improved scheme supports the one-word search. For one-word search, upon receiving the query $T_{wi} = \{(\zeta_x(w_i), f_y(w_i))\}$, the cloud can find the element $(k_{i,0} \| \varphi_\omega(s_i)) \oplus f_y(w_i)$ in $A[\zeta_x(w_i)]$. Then it computes $A[\zeta_x(w_i)] \oplus f_y(w_i)$ and obtains $k_{i,0} \| \varphi_\omega(s_i)$. It is noted that $P(\varphi_\omega(s_i))$ is the head node of the posting list of the word w_i and $k_{i,0}$ is the encryption key. By decrypting using the cryptographic key $k_{i,0}$, the cloud recovers $\mathrm{id}(d) \| k_{i,j} \| \varphi_\omega(c + 1)$, in which $\mathrm{id}(d)$ is the identifier for the document $d \in D(w_i)$ containing the keyword w_i. Then the cloud scans all the nodes in the posting list by using a similar method. Therefore, the cloud obtains all the document identifiers containing the keyword w_i and return them to the client.

For phrase search, the cloud receives the $T_p = \{(\zeta_x(Pair_1), f_y(Pair_1)), (\zeta_x(Pair_2), f_y(Pair_2)), \ldots, (\zeta_x(Pair_{n-1}), f_y(Pair_{n-1}))\}$ and can obtains the document identifiers containing the phrase $p = (w_1, w_2, \ldots, w_n)$. This procedure is no difference with the schemes in [19, 20]. Thus the improved scheme supports one-keyword and phrase search and it is a remedy for F. Kong's scheme.

In fact, in Z.A. Kissel's phrase search scheme [19], we can perform one-keyword search besides phrase search. It is required to scan all the posting lists of the keyword w_i, which is followed by many words of all the existing pairs. It is a little more complicated than our scheme.

For security, our modification cannot bring new security risks. The symmetric encryption algorithm for encrypting keyword w_i or a pair of words $Pair_i$ must be a semantically secure symmetric algorithm [24, 25]. Thus the ciphertexts of (w_i, w_s), (w_i, w_t), or the keyword w_i are indistinguishable. So the cloud or the attacker cannot get useful information to learn whether an encrypted pair $Pair_i$ includes an encrypted keyword w_i. Therefore, our improved phrase search scheme has as good security as these schemes [13, 19, 20].

4 Analysis and Improvement of LPSSE Scheme

M. Li et al.'s LPSSE phrase search scheme [21] is proposed based on R. Curtmola's non-adaptive searchable symmetric encryption scheme by adding the relative location information of words to the inverted index.

Y. Tang et al. [17] proposed a two-round phrase protocol, in which the first round is to get all the document candidates containing all the words in the phrase and the second

is to find the documents containing the phrase. M. Li et al.'s scheme can be finished in one round of communication at the expense of more search time.

M. Li et al.'s scheme [21] uses a similar construction of relative location information as Y. Tang et al.'s scheme [17]. However, they use a simple unkeyed hash function $h()$ instead of a keyed pseudo-random function $h_s()$.

Now we give a security analysis and show that some information may be leaked when unkeyed hash function $h()$ is adopted. For simplicity, we adopt the notations of Y. Tang et al.'s scheme.

The look-up table A for each document D [17] is built for recording the location of each word in document D. To avoid leaking the location information, they assign a unique random number r_i to represent the location of the i^{th} word in D. The first column is $\Psi_z(w[i]\|id(D))$ and the remaining elements are $h_s(r_{i-1})\| r_i$. The cryptographic secret key s is computed by $f_{2t}(w[i-1]\|w[i]\|id(D))$. To check whether two words are located adjacently, we can compute $h_s(r_{i-1})$ of the first word and test whether it is equal to the right v bits the element of the second word in the table.

We note that no one can compute $h_s(r_{i-1})$ without the secret key s, which is given by $f_{2t}(w[i-1]\|w[i]\|id(D))$ in the client's query. However, in M. Li et al.'s scheme [21], they use a unkeyed hash function $h()$, which can be computed without a secret key. Thus some information may be leaked. For example, the client sends the query of the phrase $(w[1]\|w[2])$ to the remote cloud. In Y. Tang et al.'s scheme, the cloud can only judge whether the phrase $(w[1]\|w[2])$ is contained in a document D. However, in M. Li et al.'s scheme, the cloud can test the phrase $(w[1]\|w[2])$ and $(w[2]\|w[1])$ because both $h(r_{i-1})$ and $h(r_i)$ can be calculated. Thus we suggest that the keyed pseudo-random function $h_s()$ should be applied to improve the security of M. Li et al.'s scheme.

5 Conclusions

Searchable symmetric encryption techniques maybe one of the most efficient methods for secure cloud storage before full Homomorphic encryption schemes overcome their efficiency problem. We have given improvements of two one-round phrase search schemes, known as F. Kong's scheme and M. Li's scheme.

R. Curtmola et al. had propose an adaptively secure one-keyword search scheme. Unfortunately, these phrase search schemes are still non-adaptively secure. Therefore, it is an amazing work to construct adaptively secure phrase search schemes over encrypted data.

References

1. Xu, L.D., He, W., Li, S.: Internet of Things in industries: a survey. IEEE Tran. Ind. Inform. **10**(4), 2233–2243 (2014)
2. Dabbagh, M., Ammar, R.: Internet of Things Security and Privacy. In: Rayes, A., Salam, S. (eds.) Internet of Things From Hype to Reality, pp. 195–223. Springer, Cham (2017)

3. Sadeghi, A.-R., Wachsmann, C., Waidner, M.: Security and privacy challenges in industrial Internet of Things. In: Proceedings of the 52nd Annual Design Automation Conference, vol. 54. ACM, New York (2015)
4. Hashem, I.A.T., Yaqoob, I., Anuar, N.B., Mokhtar, S., Gani, A., Khan, S.U.: The rise of "big data" on cloud computing: review and open research issues. Inf. Syst. **47**, 98–115 (2015)
5. Chhabra, S., Dixit, V.S.: Cloud computing: state of the art and security issues. ACM SIGSOFT Softw. Eng. Notes (SIGSOFT) **40**(2), 1–11 (2015)
6. Tari, Z., Yi, X., Premarathne, U.S., Bertók, P., Khalil, I.: Security and privacy in cloud computing: vision, trends, and challenges. IEEE Cloud Comput. (CLOUDCOMP) **2**(2), 30–38 (2015)
7. Padilha, R., Pedone, F.: Confidentiality in the cloud. IEEE Secur. Priv. (IEEE SP) **13**(1), 57–60 (2015)
8. Ali, M., Khan, S.U., Vasilakos, A.V.: Security in cloud computing: opportunities and challenges. Inf. Sci. **305**, 357–383 (2015)
9. Cloud Security Alliance (CSA): Security guidance for critical areas of focus in cloud computing v3.0. White Paper. https://cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf
10. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC 2009), pp. 169–178. ACM, New York (2009)
11. Gentry, C., Groth, J., Ishai, Y., Peikert, C., Sahai, A., Smith, A.D.: Using fully homomorphic hybrid encryption to minimize non-interactive zero-knowledge proofs. J. Cryptol. (JOC) **28**(4), 820–843 (2015)
12. Song, D., Wagner, D., Perrig, A.: Practical techniques for searches on encrypted data. In: Proceedings of IEEE Symposium on Security and Privacy, pp. 44–55. IEEE (2000)
13. Curtmola, R., Garay, J., Kamara, S., Ostrovsky, R.: Searchable symmetric encryption: improved definitions and efficient constructions. In: ACM Conference on Computer and Communications Security (CCS 2006), pp. 79–88. ACM (2006)
14. Golle, P., Staddon, J., Waters, B.: Secure conjunctive keyword search over encrypted data. In: Jakobsson, M., Yung, M., Zhou, J. (eds.) ACNS 2004. LNCS, vol. 3089, pp. 31–45. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24852-1_3
15. Cao, N., Wang, C., Li, M., et al.: Privacy-preserving multi-keyword ranked search over encrypted cloud data. IEEE Trans. Parallel Distrib. Syst. **25**(1), 222–233 (2014)
16. Zittrower, S., Zou, C.C.: Encrypted phrase searching in the cloud. In: Global Communications Conference (GLOBECOM), pp. 764–770. IEEE (2012)
17. Tang, Y., Gu, D., Ding, N., Lu, H.: Phrase search over encrypted data with symmetric encryption scheme. In: 32nd International Conference on Distributed Computing Systems Workshops, pp. 471–480. IEEE (2012)
18. Kissel, Z.A., Wang, J.: Verifiable phrase search over encrypted data secure against a semi-honest-but-curious adversary. In: IEEE International Conference on Distributed Computing Systems, pp. 126–131. IEEE (2013)
19. Kissel, Z.A.: Verifiable symmetric searchable encryption. Ph.D. Dissertation, University of Massachusetts Lowell, August 2013
20. Kong, F., Wang, J., Yu, J., Wang, X.: Analysis and improvement of a verifiable phrase search over encrypted data in a single phrase. In: 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), pp. 840–843. IEEE (2015)
21. Li, M., Jia, W., Guo, C., Sun, W., Tan, X.: LPSSE: lightweight phrase search with symmetric searchable encryption in cloud storage. In: 2015 12th International Conference on Information Technology-New Generations (ITNG), pp. 174–178. IEEE (2015)

22. Poon, H.T., Miri, A.: An efficient conjunctive keyword and phase search scheme for encrypted cloud storage systems. In: 2015 IEEE 8th International Conference on Cloud Computing, pp. 508–515. IEEE (2015)
23. Poon, H.T., Miri, A.: A low storage phase search scheme based on bloom filters for encrypted cloud services. In: 2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing (CSCloud), pp. 253–259. IEEE (2015)
24. Katz, J., Lindell, Y.: Introduction to Modern Cryptography. In: Chapman & Hall/CRC Cryptography and Network Security Series. Chapman & Hall/CRC, Boca Raton (2007)
25. Stinson, D.R.: Cryptography: Theory and Practice. CRC Press, Boca Raton (2005)