International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Searching Keyword Using Public-Key Ciphertexts with Hidden Structures

K. Maheshwari[1], **S. Nithya**[2]

[1]PG student Sir Isaac Newton College of Engineering Technology, Pappakovil, Nagapattinam-611001, India

[2]Head of CSE Department Sir Isaac Newton College of Engineering Technology, Pappakovil, Nagapattinam-611001, India

Abstract: ***Searching keyword as fast as possible without sacrificing semantic security of the encrypted keywords using Searchable Public-key Ciphertexts with Hidden Structures (SPCHS). In this all keyword searchable ciphertexts are structured by hidden relations, and with trapdoor corresponding to a keyword, search algorithm provide guidance to find all matching ciphertexts efficiently. Construct SPCHS from scratch in which ciphertexts have a hidden star – like structure. It is semantically secure in the Random Oracle (RO) Model. Search complexity is depends on the actual number of ciphertexts containing queried keyword rather than the number of all ciphertexts. Finally we propose, Generic SPCHS is constructed from Identity – Based Encryption (IBE) and collision-free full-identity malleable Identity – Based Key Encapsulation Mechanism (IBKEM). In this collision-free full-identity malleable IBKEM instances are semantically secure and anonymous in the RO and standard models.***

Keywords: Public – key searchable encryption, Semantic security, Identity – Based Encapsulation Mechanism (IBKEM), Identity – Based Encryption (IBE)

1. Introduction

PUBLIC-KEY encryption with keyword search (PEKS), introduced by Boneh et al. in [1], has the advantage that anyone who knows the receiver's public key can upload keyword-searchable ciphertexts to a server. The receiver can delegate the keyword search to the server. More specifically, each sender separately encrypts a file and its extracted keywords and sends the resulting ciphertexts to a server; when the receiver wants to retrieve the files containing a specific keyword, he delegates a keyword search trapdoor to the server; the server finds the encrypted files containing the queried keyword without knowing the original files or the keyword itself, and returns the corresponding encrypted files to the receiver; finally, the receiver decrypts these encrypted files1. PEKS [1] also presented semantic security against chosen keyword attacks (SSCKA) in the sense that the server cannot distinguish the ciphertexts of the keywords of its choice before observing the corresponding keyword search trapdoors. It seems an appropriate security notion, especially if the keyword space has no high min-entropy. Existing semantically secure PEKS schemes take search time linear with the total number of all ciphertexts. This makes retrieval from large-scale databases prohibitive. accelerate the search over encrypted keywords in the public-key setting is deterministic encryption introduced by Bellare et al. in [2]. It focus on enabling search over encrypted keywords to be as efficient as the search for unencrypted keywords, such that a ciphertext containing a given keyword can be retrieved in time complexity logarithmic in the total number of all ciphertexts. This is reasonable because the encrypted keywords can form a tree-like structure when stored according to their binary values. Deterministic encryption has two inherent limitations. First, keyword privacy can be guaranteed only for keywords that are a priori hard-to-guess by the adversary; second, certain information of a message leaks inevitably via the ciphertext of the keywords since the encryption is deterministic.

Basic Idea is to improve search performance in PEKS without sacrificing semantic security if one can organize the ciphertexts with elegantly designed but hidden relations. Intuitively, if the keyword searchable ciphertexts have a hidden star-like structure, as shown in Figure 1, then search over ciphertexts containing a specific keywords may be accelerated. Specifically, suppose all ciphertexts of the same keyword form a chain by the correlated hidden relations, and also a hidden relation exists from a public Head to the first ciphertext of each chain. With a keyword search trapdoor and the Head, the server seeks out the first matching ciphertext via the corresponding relation from the Head. Then another relation can be disclosed via the found ciphertext and guides the searcher to seek out the next matching ciphertext. By carrying on in this way, all matching ciphertexts can be found. Clearly, the search time depends on the actual number of the ciphertexts containing the queried keyword, rather than on the total number of all ciphertexts.

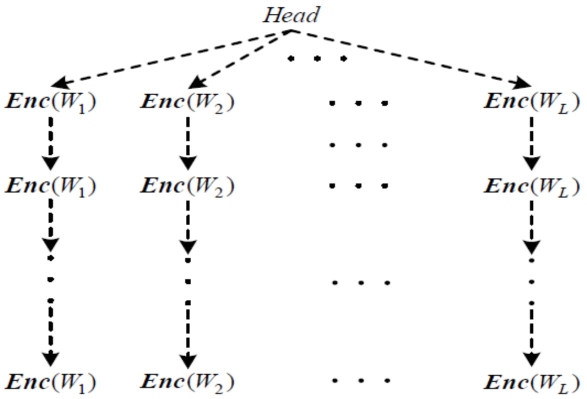


Figure 1: Hidden star-like structure formed by keyword searchable ciphertexts. (The dashed arrows denote the hidden relations. $Enc(W_i)$ denotes the searchable ciphertext of keyword W_i.)

Semantic security is preserved 1) if no keyword search trapdoor is known, all ciphertexts are indistinguishable, and no information is leaked about the structure, and 2) given a keyword search trapdoor, only the corresponding relations can be disclosed, and the matching ciphertexts leak no information about the rest of ciphertexts, except the fact that the rest do not contain the queried keyword.

Construct a simple SPCHS from scratch in the random oracle (RO) model. The scheme generates keyword-searchable ciphertexts with a hidden star-like structure. The search performance mainly depends on the actual number of the ciphertexts containing the queried keyword. For security, the scheme is proven semantically secure based on the Decisional Bilinear Diffie- Hellman (DBDH) assumption [3] in the RO model.

Also providing a generic SPCHS construction to generate keyword-searchable ciphertexts with a hidden star-like structure. Our generic SPCHS is inspired by several interesting observations on Identity-Based Key Encapsulation Mechanism (IBKEM). In IBKEM, a sender encapsulates a key K to an intended receiver ID. Of course, receiver ID can decapsulate and obtain K, and the sender knows that receiver ID will obtain K. However, a non-intended receiver ID^1 may also try to decapsulate and obtain K_0. (1) it is usually the case that K and K^1 are independent of each other from the view of the receivers, and (2) in some IBKEM the sender may also know K^1 obtained by receiver ID^1. An IBKEM scheme is said to be collision-free full-identity malleable if it possesses both properties.

In 2013, Abdalla et al. proposed several IBKEM schemes to construct Verifiable Random Functions$_2$ (VRF) [8]. In [9], Freire et al. utilized the “approximation” of multilinear maps [10] to construct a standard-model version of Boneh-and-Franklin (BF) IBE scheme [11].We transform this IBE scheme into a collision-free full-identity malleable IBKEM scheme with semantic security and anonymity in the standard model.

2. Modeling SPCHS

Hidden structure formed by ciphertexts as (C,Pri,Pub), where C denotes the set of all ciphertexts, Pri denotes the hidden relations among C, and Pub denotes the public parts. In case there is more than one hidden structure formed by ciphertexts, the description of multiple hidden structures formed by ciphertexts can be (C, (Pri_1,Pub_1),…, (Pri_N,Pub_N)), where
N **€ N**. Moreover, given (C,Pub_1,…,Pub_N) and (Pri_1,…, Pri_N) except (Pri_i, Pri_j) (where i ≠ j), one can neither learn anything about (Pri_i, Pri_j) nor decide whether a ciphertext is associated with Pub_i or Pub_j .

In SPCHS, the encryption algorithm has two functionalities. One is to encrypt a keyword, and the other is to generate a hidden relation, which can associate the generated ciphertext to the hidden structure. Let (Pri;Pub) be the hidden structure. The encryption algorithm must take Pri as input, otherwise the hidden relation cannot be generated since Pub does not contain anything about the hidden relations. In addition, SPCHS needs an algorithm to initialize (Pri;Pub) by taking the master public key as input, and this algorithm will be run before the first time to generate a ciphertext. With a keyword search trapdoor, the search algorithm of SPCHS can disclose partial relations to guide the discovery of the ciphertexts containing the queried keyword with the hidden structure.

SPCHS consists of five algorithms:

1) **System Setup**(1^k;W): Take as input a security parameter 1^k and a keyword space W, and probabilistically output a pair of master public-and-secret keys (PK,SK), where PK includes the keyword space Wand the ciphertext space C.

2) **Structure Initialization (PK):** Take as input PK, and probabilistically initialize a hidden structure by outputting its private and public parts (Pri,Pub).

3) **Structured Encryption (PK, W;, Pri):** Take as inputs PK, a keyword W € W and a hidden structure’s private part Pri, and probabilistically output a keyword searchable ciphertext C of keyword W with the hidden structure, and update Pri.

4) **Trapdoor (SK, W):** Take as inputs SK and a keyword W € W, and output a keyword search trapdoor T_W of W.

5) **Structured Search (PK, Pub, C, T_W):** Take as inputs PK, a hidden structure’s public part Pub, all keyword-searchable ciphertexts **C** and a keyword search trapdoor T_W of keyword W, disclose partial relations to guide finding out the ciphertexts containing keyword W with the hidden structure.

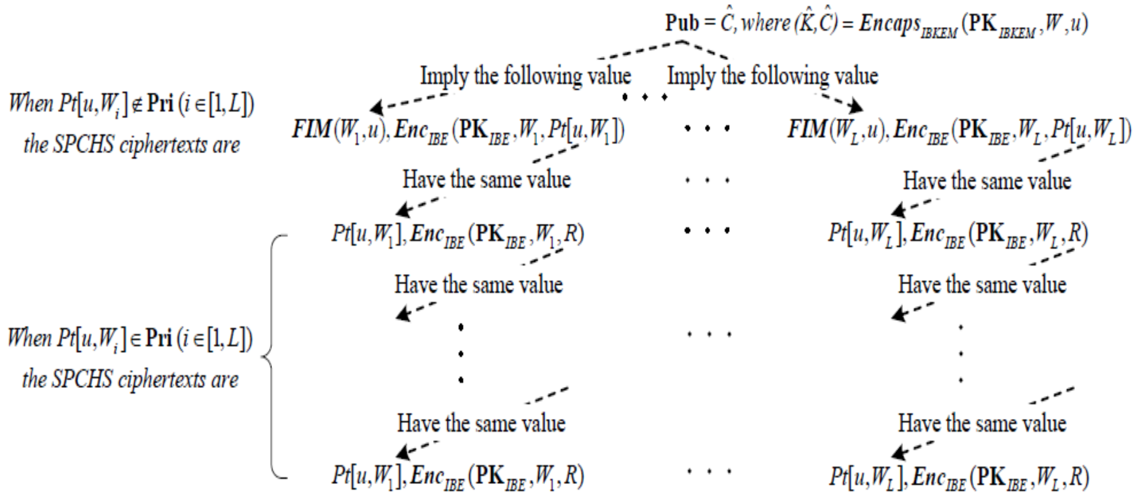
An SPCHS scheme must be consistent in the sense that given any keyword search trapdoor T_W and any hidden structure’s public part Pub, algorithm Structured Search (PK,Pub,**C**,T_W) finds out all ciphertexts of keyword W with the hidden structure Pub.

Search algorithm has two functionalities.
1. Algorithm Trapdoor allows the receiver to delegate a keyword search trapdoor to the server.
2. Structured Encryption

Advantage of an algorithm is that anyone who knows the receiver’s public key can upload keyword-searchable ciphertexts to a server. A wins in the SSCKSA game of the above SPCHS instance with advantage AdvSS-CKSA SPCHS;A, in which A makes at most qt queries to oracle QTrap()

3. Generic SPCHS Construction:

A generic SPCHS can be constructed from IBKEM and IBE.


$\mathbf{Pub} = \hat{C}, where\ (\hat{K}, \hat{C}) = Encaps_{IBKEM}(\mathbf{PK}_{IBKEM}, W, u)$
Imply the following value
Imply the following value
When $Pt[u, W_i] \notin \mathbf{Pri}\ (i \in [1, L])$
the SPCHS ciphertexts are
$FIM(W_1, u), Enc_{IBE}(\mathbf{PK}_{IBE}, W_1, Pt[u, W_1])$
$FIM(W_L, u), Enc_{IBE}(\mathbf{PK}_{IBE}, W_L, Pt[u, W_L])$
Have the same value
Have the same value
$Pt[u, W_1], Enc_{IBE}(\mathbf{PK}_{IBE}, W_1, R)$
$Pt[u, W_L], Enc_{IBE}(\mathbf{PK}_{IBE}, W_L, R)$
Have the same value
Have the same value
When $Pt[u, W_i] \in \mathbf{Pri}\ (i \in [1, L])$
the SPCHS ciphertexts are
Have the same value
Have the same value
$Pt[u, W_1], Enc_{IBE}(\mathbf{PK}_{IBE}, W_1, R)$
$Pt[u, W_L], Enc_{IBE}(\mathbf{PK}_{IBE}, W_L, R)$

Note that in each ciphertext, the value R is randomly chosen. For $i \in [1, L]$, $Pt[u, W_i]$ is initialized with a random value when generating the first ciphertext of keyword W_i, and it will be updated into R after generating each subsequent ciphertext of keyword W_i.

Construct a special keyword balanced binary tree as the index, and propose a "Greedy Depth-first Search" algorithm to obtain better efficiency than linear search.

4. Conclusion

Many of the evaluations in this paper investigated as-fast-as-possible search in PEKS with semantic security. We proposed the concept of SPCHS as a variant of PEKS. The new concept allows keyword-searchable ciphertexts to be generated with a hidden structure. Given a keyword search trapdoor, the search algorithm of SPCHS can disclose part of this hidden structure for guidance on finding out the ciphertexts of the queried keyword. Semantic security of SPCHS captures the privacy of the keywords and the invisibility of the hidden structures. We proposed an SPCHS scheme from scratch with semantic security in the RO model. The scheme generates keyword-searchable ciphertexts with a hidden star-like structure. It has search complexity mainly linear with the exact number of the ciphertexts containing the queried keyword. It outperforms existing PEKS schemes with semantic security, whose search complexity is linear with the number of all ciphertexts. We identified several interesting properties, i.e., collision-freeness and full-identity malleability in some IBKEM instances, and formalized these properties to build a generic SPCHS construction. We illustrated two collision-free full identity malleable IBKEM instances, which are respectively secure in the RO and standard models.

5. Acknowledgement

It is a great pleasure to acknowledge those who extended their support and contributed time and psychic energy for the completion of this project work. At the outset, I would like to thank my project guide Prof. S.Nithya, Head of CSE Dept, she served as sounding board for both contents and programming work. Her valuable and skillful guidance, assessment and suggestions from time to time improved the quality of work in all respects. I would like to take this opportunity to express my deep sense of gratitude towards her, for her invaluable contribution in this project work.

References

[1] Boneh D., Crescenzo G. D., Ostrovsky R., Persiano G.: Public Key Encryption with Keyword Search. In: Cachin C., Camenisch J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 506-522. Springer, Heidelberg (2004)

[2] Bellare M., Boldyreva A., O'Neill A.: Deterministic and Efficiently Searchable Encryption. In: Menezes A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 535-552. Springer, Heidelberg (2007)

[3] Boneh D., Boyen X.: Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In: Cachin C., Camenisch J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223-238. Springer, Heidelberg (2004)

[4] Boyen X., Waters B. R.: Anonymous Hierarchical Identity-Based Encryption (Without Random Oracles). In: Dwork C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 290-307. Springer, Heidelberg (2006)

[5] Gentry C.: Practical Identity-Based Encyrption Without Random Oracles. In: Vaudenay S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp.445-464. Springer, Heidelberg (2006)

[6] Ateniese G., Gasti P.: Universally Anonymous IBE Based on the Quadratic Residuosity Assumption. In: Fischlin M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 32-47. Springer, Heidelberg (2009)

[7] Ducas L.: Anonymity from Asymmetry: New Constructions for Anonymous HIBE. In: Pieprzyk J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 148-164. Springer, Heidelberg (2010)

[8] Abdalla M., Catalano D., Fiore D.: Verifiable Random Functions: Relations to Identity-Based Key Encapsulation and New Constructions. Journal of Cryptology, 27(3), pp. 544-593 (2013)

[9] Freire E.S.V., Hofheinz D., Paterson K.G., Striecks C.: Programmable Hash Functions in the Multilinear Setting. In: Canetti R., Garay J.A. (eds.) Advances in Cryptology - CRYPTO 2013. LNCS, vol. 8042, pp. 513-530. Springer, Heidelberg (2013)

[10] Garg S., Gentry C., Halevi S.: Candidate Multilinear Maps from Ideal Lattices. In: Johansson T., Nguyen P. (eds.) Advances in Cryptology - EUROCRYPT 2013. LNCS, vol. 7881, pp. 1-17. Springer, Heidelberg (2013)

[11] Boneh D., Franklin M.: Identity-Based Encryption from the Weil Pairing. In: Kilian J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213- 239. Springer, Heidelberg (2001)

Author Profile

Mrs. K.Maheshwari has completed her B.E in Computer Science & Engineering from E.G.S. Pillay Engineering college, Nagapattinam and currently pursuing M.E in Computer Science & Engineering from Sir Issac Newton College of Engineering and Technology, Pappakovil, Nagapattinam.

Prof. S.Nithya has completed her B.E in Computer Science & Engineering from J.J. College of Engineering and Technology, Trichy and M.E. in Computer Science & Engineering from R.V.S. Engineering College, Sembodai.