BVG Trust

J. Sci. Trans. Environ. Technov. 2017, 11(1) : 50-53

Scientific Transactions in Environment and Technovation

Multi Server Public Key Encryption with Keyword Search for Big Data based Cloud Servers

https://doi.org/10.56343/STET.116.011.001.009
http://stetjournals.com

S. Eswari* and S. Manikandan

Department of Computer Science, Naina Mohamed College of Arts & Science, Rajendrapuram.
Department of Computer Science & Engineering, Sriram Engineering College, Perumalpattu, Chennai - 602 024.

Abstract

A password guessing attacks has been happening everywhere in this cloud era. The big data based cloud architectures are threatened to be in the security breach by anonymous users. it has been shown that the conventional framework suffers from an insecurity called Keyword Guessing Attack (KGA) done by the hackers. To solve this security problem, we propose a new MSKE Multi-Server Key Encryption) framework named Multi-Server Key Encryption with Keyword Search. As another main contribution, we define a new variant of the Smooth Projective Hash Functions (SPHFs) referred to as linear and homomorphic SPHF (LH SPHF). We then show a generic construction of secure DS PEKS from LH-SPHF. To illustrate the feasibility of our new framework, we provide an efficient instantiating of the general framework from a DDH-based LH-SPHF and show that it can achieve the strong security against inside KGA.

Key words : Index Terms—Keyword search, secure cloud storage, encryption, inside keyword guessing attack, smooth projective hash function.

Received : July 2015 Revised and Accepted : July 2017

INTRODUCTION

Cloud storage outsourcing has become a popular application for enterprises and organizations to reduce the burden of maintaining big data in recent years. However, in reality, end users may not entirely trust the cloud storage servers and may prefer to encrypt their data before uploading them to the cloud server in order to protect the data privacy. This usually makes the data utilization more difficult than the traditional storage where data is kept in the absence of encryption. One of the typical solutions is the search able encryption which allows the user to retrieve the encrypted documents that contain the user-specified keywords, where given the keyword trapdoor, the server can find the data required by the user without decryption. Search able encryption can be realized in either symmetric or asymmetric encryption setting (Song *et al.,* 2000).Proposed keyword search on cipher text, known as Search able Symmetric Encryption (SSE) and afterwards several SSE schemes *(*Agrawal *et al.,* 2004*;*Curtmola,2006*)*.Were designed for improvements. Although SSE schemes enjoy high efficiency, they suffer from complicated secret key distribution. Precisely, users have to securely share secret keys which are used for data encryption.Otherwise they are not able to share the encrypted data outsourced to the cloud. To resolve this problem, Boneh *et al.* (2004). Introduced a more flexible primitive, namely Public Key Encryption with Keyword Search (PEKS) that enables a user to search encrypted data in the asymmetric encryption setting. In a PEKS system, using the receiver's public key, the sender attaches some encrypted keywords (referred to as PEKS cipher texts) with the encrypted data. The receiver then sends the trapdoor of a to-be-searched keyword to the server for data searching. Given the trapdoor and the PEKS cipher text, the server can test whether the keyword underlying the PEKS ciphertxt is equal to the one selected by the receiver. If so, the server sends the matching encrypted data to the receiver.

MOTIVATION OF THIS WORK

Despite of being free from secret key distribution, PEKS schemes suffer from an inherent insecurity regarding the trapdoor keyword privacy, namely inside Keyword Guessing Attack (KGA). The reason leading to such a security vulnerability is that anyone who knows receiver's public key can generate the PEKS cipher text of arbitrary keyword himself. Specifically, given a trapdoor, the adversarial server can choose a guessing keyword from the keyword space and then use the keyword to generate a PEKS cipher text. The server then can test whether the guessing keyword is the one underlying the trapdoor. This guessing-then-testing procedure can be repeated until the correct keyword is found. Such a guessing attack has also been considered in many password-based systems. However, the attack can be launched more efficiently against PEKS schemes since the keyword space is roughly the same as a normal dictionary (e.g., all the

*Corresponding Author :
email: eswari.mail2014@gmail.com

P - ISSN 0973 - 9157
E - ISSN 2393 - 9249

meaningful English words), which has a much smaller size than a password dictionary (e.g., all the words containing six alphanumeric characters). It is worth noting that in SSE schemes, only secret key holders can generate the keyword cipher text and hence the adversarial server is not able to launch the inside KGA. As the keyword always indicates the privacy of the user data, it is therefore of practical importance to overcome this security threat for secure search able encrypted data outsourcing.

OUR CONTRIBUTIONS

The contributions of this paper are four-fold.

- We formalize a new PEKS framework named Dual- Server Public Key Encryption with Keyword Search (DSPEKS) to address the security vulnerability of PEKS.
- A new variant of Smooth Projective Hash Function (SPHF), referred to as linear and homomorphic SPHF, is introduced for a generic construction of DS-PEKS.
- We show a generic construction of DS-PEKS using the proposed Lin-Hom SPHF.
- To illustrate the feasibility of our new framework, an efficient instantiating of our SPHF based on the Diffie-Hellman language is presented in this paper.

RELATED WORK

We describe a classification of PE schemes based on their security. Traditional PEKS. Following Boneh *et al*. (2004). Abdalla *et al*. (2005) formalized anonymous IBE (AIBE) and presented generi construction of search able encryption from AIBE. They also showed how to transfer a hierarchical IBE (HIBE) scheme into a public key encryption with temporary keyword search (PETKS) where the trapdoor is only valid in a specific time interval.Waters *et al.* (2004) showed that the PEKS schemes based on bilinear map could be applied to build encrypted and search able auditing logs. I order to construct a PEKS secure in the standard model, Khader, (2006) proposed a scheme based on the k-resilient IBE and also gave a construction supporting multiple-keyword search. The first PEKS scheme without pairings was introduced by Di Crescenzo *et al*. (2007). The construction is derived from Cock's IBE scheme Cock's, (2001) which is not very practical.

SECURE CHANNEL FREE PEKS

The original PEKS scheme Boneh et al., 2004 requires a secure channel to transmit the trapdoors. To overcome this limitation, Baek *et al*. (2008) proposed a new PEKS scheme without requiring a secure channel, which is referred to as a secure channel-free PEKS (SCF-PEKS).The idea is to add the server's public / private key pair into a PEKS system. The keyword cipher text and trapdoor are generated using the server's public key and hence only the server (designated tester) is able to perform the search. Rhee *et al.* (2009) later enhanced security model Baek *et al*. (2008) for SCF-PEKS where the attacker is allowed to obtain the relationship between the non-challenge cipher texts and the trapdoor. They also presented an SCF-PEKS scheme secure under the enhanced security model in the random oracle model. They enhanced the security model by introducing the adaptively secure SCF-PEKS, wherein an adversary is allowed to issue test queries adaptive.

AGAINST INSIDE KGA

Nevertheless, all the schemes mentioned above are found to be vulnerable to keyword guessing attacks from a malicious server (i.e., inside KGA) showed a negative result that the consistency / correctness of PEKS implies insecurity to inside KGA in PEKS. Their result indicates that constructing secure and consistent PEKS schemes against inside KGA is impossible under the original framework. A potential solution is to propose a new framework of PEKS.

Differences Between This Work and Its Preliminary Version[1]

Portions of the work presented in this paper have previously appeared as an extended abstract Chen *et al*. (2015). Compared to Chen *et al*. (2015) we have revised and enriched the work substantially in the following aspects. First, in the preliminary work Chen *et al*. (2015) where our generic DS-PEKS construction was presented, we showed neither a concrete construction of the linear and homomorphic SPHF nor a practical instantiating of the DS-PEKS framework. To fill this gap and illustrate the feasibility of the framework, we first show that a linear and homomorphic language LDH can be derived from the Diffie-Hellman assumption and then construct a concrete linear and homomorphic SPHF, referred to as SPHFDH, from LDH.

ORGANIZATION

We propose a new framework, namely DSPEKS, and present its formal definition and security models. We then define a new variant of smooth projective hash function (SPHF). A generic construction of DS-PEKS from LH-SPHF is shown in Section 5 with formal correctness analysis and security proofs. Finally, we present an efficient instantiating of DS-PEKS from SPHF based on a language defined by

Diffie-Hellman. We also analyze the performance of our scheme through comparisons with existing works and experimental evaluation.

A NEW FRAMEWORK FOR PEKS

In this section, we formally define the Dual-Server Public Key Encryption with Keyword Search (DS-PEKS) and its security model.

DEFINITION OF DS-PEKS

A DS-PEKS scheme mainly consists of (KeyGen, DS-PEKS, DS-Trapdoor; FrontTest; BackTest). To be more precise, the KeyGen algorithm generates the public / private key pairs of the front and back servers instead of that of the receiver. Moreover, the trapdoor generation algorithm DS-Trapdoor defined here is public while in the traditional PEKS definition (Boneh *et al*., 2004; Baek *et al*., 2008). The algorithm Trapdoor takes as input the receiver's private key. Such a difference is due to the different structures used by the two systems. In the traditional PEKS, since there is only one server, if the trapdoor generation algorithm is public, then the server can launch a guessing attack against a keyword cipher text to recover the encrypted keyword. As a result, it is impossible to achieve the semantic security as defined in (Boneh *et al.,* 2004; Baek *et al.,*2008). However, as we will show later, under the DS-PEKS framework, we can still achieve semantic security when the trapdoor generation algorithm is public. Another difference between the traditional PEKS and our proposed DS-PEKS is that the test algorithm is divided into two algorithms, FrontTest and BackTest run by two independent servers. This is essential for achieving security against the inside keyword guessing attack.In this DS-PEKS system, upon receiving a query from the receiver, the front server pre processes the trapdoor and all the PEKS cipher texts using its private key, and then sends some internal testing-states to the back server with the corresponding trapdoor and PEKS cipher texts hidden. The back server can then decide which documents are queried by the receiver using its private key and the received internal testing-states from the front server.

SECURITY MODELS

In this subsection, we formalise the following security models for a DS-PEKS scheme against the adversarial front and back servers, respectively. One should note that both the front server and the back server here are supposed to be "honest but curious" and will not collide with each other. More precisely, both the servers perform the testing strictly following the scheme procedures but may be curious about the underlying keyword. We should note that the following security models also imply the security guarantees against the outside adversaries which have less capability compared to the servers.

SMOOTH PROJECTIVE HASH FUNCTIONS

A central element of our construction for dual-server public key encryption with keyword search is smooth projective hash function (SPHF), a notion introduced by Cramer and Shoup. We start with the original definition of an SPHF.

Original Definition of SPHFs

In summary, an SPHF has the property that the projection key uniquely determines the hash value of any word in the language L but gives almost no information about the hash value for any point in X n L. In this paper, we require another important property of smooth projective hash functions that was introduced in Gennaro *et al*. (2003). Precisely, we require the SPHF to be pseudo-random. That is, if a word W 2 L, then without the corresponding witness w, the distribution of the hash output is computationally indistinguishable from a uniform distribution in the view of any polynomial-time adversary

GENERIC CONSTRUCTION OF DS-PEKS

Let SPHF = (SPHFSetup; HashKG; ProjKG; Hash; ProjHash) be a LH-SPHF over the language L onto the set Y. Let W be the witness space of the language L and KW be the keyword space. Our generic construction DS-PEKS works.

PERFORMANCE EVALUATION

we first give a comparison between existing schemes and our scheme in terms of computation, size and security. We then evaluate its performance in experiments. Computation Costs. As shown in Table 1, all the existing schemes (Boneh *et al.* (2004); Xu *et al.* (2013). Require the pairing computation during the generation of PEKS cipher text and testing and hence are less efficient than our scheme, which does not need any pairing computation. In our scheme, the computation cost of PEKS generation, trapdoor generation and testing are 4ExpG1 +1HashG1+ 2MulG1 , 4ExpG1 +1HashG1+2MulG1 , and 7ExpG1 +3MulG1 respectively, where ExpG1 denotes the computation of one exponentiation in G1, MulG1 denotes the costs of one multiplication in G1, MulG1 and HashG1 respectively denote the cost of one multiplication and one hashing operation in G1.

EXPERIMENT RESULTS

To evaluate the efficiency of schemes in experiments, we also implement the scheme utilizing the GNU

Multiple Precision Arithmetic (GMP) library and Pairing Based Cryptography (PBC) library. The following experiments are based on coding language C on Linux system (more precise, 2.6.35-22-generic version) with an Intel(R) Core(TM) 2 Duo CPU of 3.33 GHZ and 2.00-GB RAM. For the elliptic curve, we choose an MNT curve with a base filed size of 159 bits and p=160 bits and jqj=80 bits. Our scheme is the most efficient in terms of PEKS computation. It is because that our scheme does not include pairing computation. Particularly, the scheme Xu *et al*. (2013) requires the most computation cost due to 2 pairing computation per PEKS generation as all the existing schemes do not involve pairing computation, the computation cost is much lower than that of PEKS generation. It is worth noting that the trapdoor generation in our scheme is slightly higher than those of existing schemes due to the additional exponentiation computations. When the searching keyword number is 50, the total computation cost of our scheme is about 0.25 seconds scheme Xu *et a*l. (2013). Cost the most time due to an additional pairing computation in the exact testing stage. One should note that this additional pairing computation is done on the user side instead of the server. Therefore, it could be the computation burden for users who may use a light device for searching data. In our scheme, although we also require another stage for the testing, our computation cost is actually lower than that of any existing scheme as we do not require any pairing computation and all the searching work is handled by the server.

CONCLUSION & FUTURE WORK

We proposed a new framework, named Dual- Server Public Key Encryption with Keyword Search (DSPEKS), that can prevent the inside keyword guessing attack which is an inherent vulnerability of the traditional PEKS framework. We also introduced a new Smooth Projective Hash Function (SPHF) and used it to construct a generic DSPEKS scheme. An efficient instantiating of the new SPHF based on the Diffie-Hellman problem is also presented in the paper, which gives an efficient DS-PEKS scheme without pairings.

REFERENCES

Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P. and Shi, H. 2005. "Search able encryption revisited: Consistency properties, relation to anonymous ibe, and extensions," in CRYPTO, P. 205–222. https://doi.org/10.1007/11535218_13

Agrawal, R., Kiernan, J., Srikant, R., and Xu, Y. 2004. "Order-preserving encryption for numeric data," in Proceedings of the ACM SIGMOD International Conference on Management of Data, P.563–574. https://doi.org/10.1145/1007568.1007632

Baek, J., Safavi-Naini, R. and Susilo, W. 2008. "Public key encryption with keyword search revisited," in Computational Science and Its Applications - ICCSA, P.1249–1259. https://doi.org/10.1007/978-3-540-69839-5_96

Boneh, D., Crescenzo, G.D., Ostrovsky, R. and Persiano, G. 2004. "Public key encryption with keyword search," in EUROCRYPT, P. 506–522. https://doi.org/10.1007/978-3-540-24676-3_30

Chen, R., Mu, Y., Yang, G., Guo, F. and Wang, X. 2015. "A new general framework for secure public key encryption with keyword search," in Information Security and Privacy - 20th Australasian Conference, ACISP, P. 59–76. https://doi.org/10.1007/978-3-319-19962-7_4

Cocks, C. 2001. "An identity based encryption scheme based on quadratic residues," in Cryptography and Coding, P. 360–363. https://doi.org/10.1007/3-540-45325-3_32

Crescenzo, G.D. and Saraswat, V. 2007. "Public key encryption with search able keywords based on jacobi symbols," in INDOCRYPT, P. 282–296. https://doi.org/10.1007/978-3-540-77026-8_21

Curtmola, R.J., Garay, A., Kamara, S. and Ostrovsky, R. 2006. "Search able symmetric encryption: improved definitions and efficient constructions,"in Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS, P. 79–88. https://doi.org/10.1145/1180405.1180417

Gennaro, R., and Lindell, Y.2003. "A framework for password-based authenticated key exchange," in EUROCRYPT, P. 524–543. https://doi.org/10.1007/3-540-39200-9_33

Rhee, L.S., Park, J.H., Susilo, W. and Lee, D.H. 2009. "Improved search able public key encryption with designated tester," in ASIACCS, P. 376–379 PMid:19097734 https://doi.org/10.1145/1533057.1533108

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

Xu, P., Jin, H., Wu, Q. and Wang, W. 2013. "Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack," IEEE Trans. Computers, vol. 62, no. 11, P. 2266– 2277. https://doi.org/10.1109/TC.2012.215