International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

A Survey on Dual-Server Public-Key Encryption with Keyword Search for Secure Cloud Storage

Ghorpade Sneha[1], Dr. S. N. Kini[2]

[1]Department of Computer Engineering, Jaywantrao Sawant College of Engineering, Pune, Maharashtra, India

[2]Professor, Department of Computer Engineering, Jaywantrao Sawant College of Engineering, Pune, Maharashtra, India

Abstract: ***Searchable encryption is of expanding enthusiasm for ensuring the information protection in secure searchable distributed storage. In this work, we explore the security of an outstanding cryptographic primitive, in particular Public Key Encryption with Keyword Search (PEKS) which is exceptionally helpful in numerous utilizations of distributed storage. Shockingly, it has been demonstrated that the customary PEKS system experiences an inalienable instability called inside Keyword Guessing Attack (KGA) propelled by the malevolent server. To address this security defenselessness, we propose another PEKS structure named Dual-Server Public Key Encryption with Keyword Search (DS-PEKS). As another fundamental commitment, we characterize another variation of the Smooth Projective Hash Functions (SPHFs) alluded to as straight and homomorphism SPHF (LH-SPHF). We then demonstrate a bland development of secure DS-PEKS from LH-SPHF. To represent the possibility of our new structure, we give a proficient instantiation of the general system from a DDH-based LH-SPHF and demonstrate that it can accomplish the solid security against inside KGA.***

Keywords: Hash Functions, Keyword Guessing Attack, SPHFs, DS-PEKS

1. Introduction

Cloud storage outsourcing has turned into a well known application for ventures and associations to lessen the weight of keeping up enormous information lately. Nonetheless, as a general rule, end clients may not so much trust the cloud capacity servers and may like to encode their information some time recently transferring them to the cloud server keeping in mind the end goal to ensure the information security. This as a rule makes the information usage more troublesome than the customary stockpiling where information is kept in the nonattendance of encryption. One of the regular arrangements is the searchable encryption which permits the client to recover the encoded reports that contain the client determined watchwords, where given the catchphrase trapdoor, the server can discover the information required by the client without decoding.

Searchable encryption can be acknowledged in either symmetric then again deviated encryption setting. In [2], Song et al. proposed watchword look on cipher text, known as Searchable Symmetric Encryption (SSE) and subsequently a few SSE plans [3], [4] were intended for enhancements. Despite the fact that SSE plans appreciate high proficiency, they experience the ill effects of muddled mystery key appropriation. Correctly, clients need to safely share mystery keys which are utilized for information encryption. Else they are not ready to share the scrambled information outsourced to the cloud. To determine this issue, Boneh et al. [5] presented a more adaptable primitive, to be specific Public Key Encryption with Keyword Search (PEKS) that empowers a client to seek encoded information in the awry encryption setting. In a PEKS framework, utilizing the collector's open key, the sender joins some encoded watchwords (allowded to as PEKS cipher texts) with the encoded information. The beneficiary at that point sends the trapdoor of a to-be-sought catchphrase to the server for information seeking. Given the trapdoor and the PEKS cipher text, the server can test whether the catchphrase fundamental the PEKS ciphertxt is equivalent to the one chose by the recipient. Provided that this is true, the server sends the coordinating scrambled information to the recipient.

In spite of being free from mystery key circulation, PEKS plans experience the ill effects of an intrinsic instability with respect to the trapdoor catchphrase security, to be specific inside Keyword Guessing Assault (KGA). The reason prompting to such a security helplessness is that any individual who knows beneficiary's open key can create the PEKS cipher text of self-assertive watchword himself. In particular, given a trapdoor, the antagonistic server can pick a speculating catchphrase from the watchword space and after that utilization the catchphrase to produce a PEKS cipher text. The server then can test whether the speculating catchphrase is the one basic the trapdoor. This speculating then-testing strategy can be rehashed until the right catchphrase is found. Such a speculating assault has additionally been considered in numerous watchword based frameworks. Be that as it may, the assault can be propelled all the more productively against PEKS plans since the watchword space is generally the same as an ordinary word reference (e.g., all the important English words), which has a much littler size than a watchword lexicon (e.g., every one of the words Containing 6 alphanumeric characters). It is significant that in SSE plans, just mystery key holders can produce the watchword cipher text and henceforth the antagonistic server is not ready to dispatch within KGA. As the watchword dependably shows the protection of the client information, it is in this way of handy significance to beat this security danger for secure searchable encoded information outsourcing.

2. Traditional PEKS

Taking after Boneh et al's. Fundamental work, Abdalla et al. formalized unknown IBE (AIBE) and exhibited a nonexclusive development of searchable encryption from AIBE. They likewise demonstrated to exchange a progressive IBE (HIBE) conspire into an open key encryption with brief catchphrase seek (PETKS) where the trapdoor is as it were legitimate in a particular time interim. Waters demonstrated that the PEKS plans in light of bilinear guide could be connected to assemble scrambled and searchable reviewing logs. Keeping in mind the end goal to develop a PEKS secure in the standard model, Khader proposed a plan in view of the k-flexible IBE furthermore gave a development supporting different catchphrase look.
The main PEKS plot without pairings was presented by Di Crescenzo and Saraswat . The development is determined from Cock's IBE plot which is not extremely down to earth.

3. Secure Channel Free PEKS

The first PEKS plot requires a protected channel to transmit the trapdoors. To beat this confinement, Baek et al. Proposed another PEKS plot without requiring a protected channel, which is alluded to as a protected without channel PEKS (SCF-PEKS). The thought is to include the server's open/private key combine into a PEKS framework. The watchword cipher text and trapdoor are produced utilizing the server's open key and thus just the server (assigned analyzer) can play out the hunt. Rhee et al. later upgraded Baek et al's. security display for SCF-PEKS where the assailant is permitted to acquire the relationship between the non-challenge cipher texts and the trapdoor. They likewise displayed a SCF-PEKS conspire secure under the upgraded security show in the irregular prophet display. Another expansion on SCF-PEKS is by Emura et al. They upgraded the security show by presenting the adaptively secure SCF-PEKS, wherein a foe is permitted to issue test questions adaptively..

4. KGA (Keyword Guessing Attack)

Byun et al. presented the disconnected catchphrase speculating assault against PEKS as watchwords are looked over a much littler space than passwords what's more, clients as a rule utilize understood catchphrases for looking archives. They likewise called attention to that the plan proposed in Boneh et al. was defenseless to watchword speculating assault. Enlivened by the work of Byun et al. Yau et al. exhibited that outside foes that catch the trapdoors sent in an open channel can uncover the encoded watchwords through disconnected Key guessing attack and they likewise flaunted line watchword speculating assaults against the (SCF-)PEKS conspires .The principal PEKS conspire secure against outside Key guessing attack was proposed by Rhee et al. . In , the idea of trapdoor vagary was proposed and the creators appeared that trapdoor vagary is an adequate condition for avoiding outside watchword speculating assaults. Tooth et al. proposed a solid SCF-PEKS plot with (outside) KGA strength. Like the work in [15], they likewise considered the versatile test prophet in their proposed security definition.

By the by, every one of the plans said above are observed to be defenseless against Key guessing attack from a noxious server (i.e., inside KGA). Jeong et al. [22] demonstrated a negative outcome that the consistency/ accuracy of PEKS suggest instability to inside KGA in PEKS. Their outcome shows that developing secure and steady PEKS plans against inside KGA is unimaginable under the first system. A potential arrangement is to propose another structure of PEKS. In [10], Peng et al. proposed the thought of Public-key Encryption with Fuzzy Keyword Search (PEFKS) where every catchphrase relates to a correct trapdoor and a fluffy trapdoor. The server is as it were furnished with the fluffy trapdoor and subsequently can no more take in the correct watchword since at least two catchphrases share the same fluffy watchword trapdoor. In any case, their plan experiences a few constraints with respect to the security and effectiveness. On one hand, in spite of the fact that the server can't precisely figure the watchword, it is still ready to know which little set the basic catchphrase has a place with and in this manner the watchword security is not all around safeguarded from the server. On the other hand, their plan is illogical as the recipient needs to locally locate the coordinating cipher text by utilizing the correct trapdoor to sift through the non-coordinating ones from the set come back from the server.

5. DS-PEKS Framework

Compared to [1], we have reconsidered and advanced the work considerably in the accompanying perspectives. To begin with, in the preparatory work [1] where our non specific DS-PEKS development was exhibited, we indicated neither a solid development of the straight what's more, homomorphism SPHF nor a reasonable instantiation of the DS-PEKS structure. To fill this crevice and outline the plausibility of the system, in this paper (Section 6), we to begin with demonstrate that a direct and homomorphism dialect LDH can be gotten from the Diffie-Hellman supposition and at that point build a solid direct and homomorphism SPHF, alluded to as SPHFDH, from LDH. We give a formal verification that SPHFDH is right, smooth and pseudo-irregular development. We then present a solid DS-PEKS plot from SPHFDH. To investigate its execution, we first give a correlation between existing plans and our plan and after that assess its execution in trials. We too reconsidered the preparatory adaptation [1] to upgrade the presentation what's more, meaningfulness. In the related work part, analyzed to the preparatory rendition, we include more written works and give a clearer characterization of the current plans in light of their security. We exhibit the security models of DS-PESK as tests to make them more lucid. Besides, to make the ideas of SPHF and our recently characterized vari clearer, we include Fig. 4 and Fig. 5 to highlight their key properties.

A DS-PEKS plot primarily comprises of (KeyGen, DS-PEKS, DS-Trapdoor; FrontTest; BackTest). To be more exact, the KeyGen calculation creates general society/private key sets of the front and back servers rather than that of the collector. Besides, the trapdoor era calculation DS-Trapdoor characterized here is open while in the customary PEKS definition [5], [13], the calculation Trapdoor takes as info

the collector's private key. Such a distinction is expected to the diverse structures utilized by the two frameworks. In the customary PEKS, since there is just a single server, if the trapdoor era calculation is open, then the server can dispatch a speculating assault against a catchphrase ciphertext to recoup the scrambled catchphrase. Subsequently, it is difficult to accomplish the semantic security as characterized in [5], [13]. Be that as it may, as we will appear later, under the DS-PEKS system, we can in any case accomplish semantic security when the trapdoor era calculation is open. Another distinction between the customary PEKS and our DS-PEKS is that the test calculation is isolated into two calculations, FrontTest and BackTest keep running by two free servers. This is basic for accomplishing security against within watchword speculating assault.

In the DS-PEKS framework, after getting a question from the collector, the front server pre-forms the trapdoor what not the PEKS cipher texts utilizing its private key, and afterward sends some inside testing-states to the back server with the comparing trapdoor and PEKS cipher texts covered up. The back server can then choose which reports are questioned by the collector utilizing its private key and the got inside testing-states from the front server.

To begin with, Setup is performed and the framework parameters are created. Based on the created framework parameters, alternate systems are executed. It is portrayed underneath:

1) Setup(1_). Takes as input the security parameter _,generates the system parameters P;
2) KeyGen(P): Takes as input the systems parameters P,outputs the public/secret key pairs (pkFS; skFS), and(pkBS; skBS) for the front server, and the back server respectively;
3) DS-PEKS(P; pkFS; pkBS; kw1): Takes as input P, the front server's public key pkFS, the back server's public key pkBS and the keyword kw1, outputs the PEKS ciphertext CTkw1 of kw1;
4) DS-Trapdoor(P; pkFS; pkBS; kw2): Takes as input P, the front server's public key pkFS, the back server's public key pkBS and the keyword kw2, outputs the trapdoor Tkw2 ;
5) FrontTest(P; skFS;CTkw1 ; Tkw2): Takes as input P, the front server's secret key skFS, the PEKS ciphertext CTkw1 and the trapdoor Tkw2 , outputs the internal testing-state CITS;
6) BackTest(P; skBS;CITS): Takes as input P, the back server's secret key skBS and the internal testing-state CITS, outputs the testing result 0 or 1

6. Smooth Projective Hash Functions

Smooth Projective Hash Functions (SPHFs) were presented by Cramer and Shoup [CS02] with a specific end goal to accomplish IND-CCA security from IND-CPA encryption plans, which prompted to the principal effective IND-CCA encryption conspire provably secure in the standard model under the DDH presumption [CS98]. They can be viewed as a sort of certain assigned verifier confirmations of enrollment [ACP09, BPV12]. Essentially, SPHFs are groups of sets of capacities (Hash, ProjHash) characterized on a dialect L. These capacities are listed by a couple of related keys (hk, hp), where hk, the hashing key, can be viewed as the private key and hp, the projection key, as people in general key. On a word W ∈ L, both capacities ought to prompt to a similar outcome: Hash(hk,L, W) with the hashing key and ProjHash(hp,L, W, w) with the projection key just additionally a witness w that W ∈ L. Obviously, if W 6∈ L, such a witness does not exist, and the smoothness property expresses that Hash(hk,L, W) is free of hp. As a result, notwithstanding knowing hp, one can't figure Hash(hk,L, W).

7. Conclusions

In this paper, we proposed another structure, named Dual-Server Public Key Encryption with Keyword Search (DSPEKS) that can keep within catchphrase speculating assault which is an intrinsic helplessness of the conventional PEKS structure. We additionally presented another Smooth Projective Hash Function (SPHF) and utilized it to build a bland DSPEKS plot. An effective instantiation of the new SPHF in light of the Diffie-Hellman issue is additionally exhibited in the paper, which gives an effective DS-PEKS plot without pairings.

References

[1] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "A new general framework for secure public key encryption with keyword search," in Information Security and Privacy - 20th Australasian Conference, ACISP, 2015, pp. 59–76.
[2] 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.
[3] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order-preserving encryption for numeric data," in Proceedings of the ACM SIGMOD International Conference on Management of Data, 2004, pp. 563–574.
[4] 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, CCS 2006, 2006, pp. 79–88.
[5] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in EUROCRYPT, 2004, pp. 506–522.
[6] R. Gennaro and Y. Lindell, "A framework for password-based authenticated key exchange," in EUROCRYPT, 2003, pp. 524–543.
[7] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in NDSS, 2004.
[8] 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 CRYPTO, 2005, pp. 205–222.
[9] D. Khader, "Public key encryption with keyword search based on k-resilient IBE," in Computational Science and Its Applications - ICCSA, 2006, pp. 298–308.
[10] P. Xu, H. Jin, Q. Wu, and W. Wang, "Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack," IEEE

Trans. Computers, vol. 62, no. 11, pp. 2266– 2277, 2013.
[11] G. D. Crescenzo and V. Saraswat, "Public key encryption with searchable keywords based on jacobi symbols," in INDOCRYPT, 2007, pp. 282–296.
[12] C. Cocks, "An identity based encryption scheme based on quadratic residues," in Cryptography and Coding, 2001, pp. 360–363.
[13] J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," in Computational Science and Its Applications - ICCSA, 2008, pp. 1249–1259.
[14] H. S. Rhee, J. H. Park,W. Susilo, and D. H. Lee, "Improved searchable public key encryption with designated tester," in ASIACCS, 2009, pp. 376–379.
[15] K. Emura, A. Miyaji, M. S. Rahman, and K. Omote, "Generic constructions of secure-channel free searchable encryption with adaptive security," Security and Communication Networks, vol. 8, no. 8, pp. 1547–1560, 2015.
[16] J. W. Byun, H. S. Rhee, H. Park, and D. H. Lee, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in Secure Data Management, Third VLDBWorkshop, SDM, 2006, pp. 75–83.
[17] W. Yau, S. Heng, and B. Goi, "Off-line keyword guessing attacks on recent public key encryption with keyword search schemes," in ATC, 2008, pp. 100–105.
[18] J. Baek, R. Safavi-Naini, andW. Susilo, "On the integration of public key data encryption and public key encryption with keyword search," in Information Security ISC, 2006, pp. 217–232.
[19] H. S. Rhee, W. Susilo, and H. Kim, "Secure searchable public key encryption scheme against keyword guessing attacks," IEICE Electronic Express, vol. 6, no. 5, pp. 237–243, 2009.
[20] H. S. Rhee, J. H. Park,W. Susilo, and D. H. Lee, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," Journal of Systems and Software, vol. 83, no. 5, pp. 763–771, 2010.
[21] L. Fang,W. Susilo, C. Ge, and J.Wang, "Public key encryption with keyword search secure against keyword guessing attacks without random oracle," Inf. Sci., vol. 238, pp. 221–241, 2013.
[22] I. R. Jeong, J. O. Kwon, D. Hong, and D. H. Lee, "Constructing PEKS schemes secure against keyword guessing attacks is possible?" Computer Communications, vol. 32, no. 2, pp. 394–396, 2009.
[23] R. Cramer and V. Shoup, "Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption," in EUROCRYPT, 2002, pp. 45–64.