A Reliable Dual Server Public-Key Encryption with Data Consistency in Cloud

SIRIGIRI IMMANUEL[1], A.EMMANUEL RAJU[2]
[1] PG Scholar, Dept of CSE, Dr. K. V. Subba Reddy Institute of Technology, Kurnool, AP, India
[2] Assistant Professor, Dept of CSE, Dr. K. V. Subba Reddy Institute of Technology, Kurnool, AP, India

ABSTRACT

Cloud computing allows the users to outsource their data using cloud storage servers in order to reduce the economic cost. Cloud computing and storage solutions provide users and enterprises to store and process their data in third-party data centers that may be located far from the user– ranging in distance from across a city to across the world. Encryption is a potential way to protect the confidentiality of the outsourced data, but it also introduces much difficulty to performing effective searches over encrypted information. Although traditional searchable encryption scheme, Public key Encryption with Keyword Search (PEKS), allow users to securely search over encrypted data through keywords, these techniques support only boolean search. Unfortunately, it is demonstrated that the customary PEKS framework suffers from an inalienable insecurity called inside Keyword Guessing Attack (KGA) launched by the malevolent server. To address this security vulnerability, it is proposed a new PEKS framework named Dual-Server PEKS

(DSPEKS) with data consistency through TPA in cloud.

I.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.

II.RELATED WORK

Cloud computing represents today’s most exciting computing pattern shift in information technology. but, security and privacy are perceived as primary obstacles to its large adoption. Here, outline several critical security challenges and motivate further investigation of security solutions for a trustworthy public cloud environment cloud computing is the latest concept for the long-dreamed vision of computing as a usefulness. It is necessary to store information on information storage servers such as mail servers and record servers in encoded frame to

improve security and protection dangers. In any case, this typically suggests one needs to relinquish usefulness for security. For instance, if a customer wishes to recover just reports containing certain words, it was not beforehand known how to let the information stockpilingserver play out the inquiry and answers the question without loss of information secrecy. The issue of seeking on information that is encoded utilizing a public open key framework consider client Bob who sends email to client Alice scrambled under Alice's open key. An email passage needs to test whether the email contains the watchword \urgent" with the goal that it could course the email as needs be. Alice, then again does not wish to give the door the capacity to unscramble every one of her messages. We done and develop an instrument that empowers Alice to give a key to the portal that empowers the door to test whether the word \urgent "is a watchword in the email without learning whatever else about the email. We allude to this system as Public Key Encryption with watchword Search. As another case, consider a mail server that stores different messages openly scrambled for Alice by others. Utilizing our instrument Alice can send the mail server a key that will empower the server to distinguish all messages containing some keyword which is we want to search. The decent property in this plan permits the server to scan for a catchphrase, given the trapdoor. Thus, the verifier can just utilize an untrusted server, which makes this idea extremely down to earth. Taking after Bonehet, al's work, there have been ensuing works that have been proposed to upgrade this idea. Two vital ideas incorporate the supposed catchphrase speculating assault and secure channel free, proposed by Byun et al. what's more, Baek etal., separately. The previous understands the way that by and by, the space of the catchphrases utilized is extremely constrained, while the last considers the evacuation of secure channel between the beneficiary and the server to make PEKS down to earth. Lamentably, the current development of PEKS secure against catchphrase speculating assault is just secure under the irregular prophet display, which does not mirror its security in this present reality. Moreover, there is no total definition that catches secure channel free PEKS plans that are secure against picked catchphrase assault, picked cipher text assault, and against watchword speculating assaults, despite the fact that these thoughts appear to be the most pragmatic use of PEKS primitives. Another system, called secure server-assignment open key encryption with catchphrase seek (SPEKS),was acquainted with enhance the security of dPEKS (which experiences the on-line catchphrase speculating assault)by characterizing another security demonstrate 'unique cipher text in distinguish ability'.

III.PROBLEM STATEMENT

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 searchable 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. Searchable encryption can be realized in either symmetric or asymmetric encryption setting. In proposed keyword search on cipher text, known as Searchable Symmetric Encryption (SSE) and afterwards several SSE schemes 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.

IV.IMPLEMENTATION

Searchable file encryption is of speeding up interest for shielding the information privacy in secure searchable cloud storage. In relation to trapdoor generation, as all of the existing schemes don't involve pairing computation, the computation price is reduced in comparison with PEKS generation [4]. During this paper, we investigate security in the well-known cryptographic primitive, namely, public key file encryption with keyword search that's very helpful in a number of applying cloud storage. A DS-PEKS plan mainly includes. To obtain more precise, the KeyGen formula generates the general public/personal key pairs from the back and front servers instead of this within the receiver. Within the traditional PEKS, since there's just one server, when the trapdoor generation formula is public, your server can launch a guessing attack against a keyword cipher text to extract the encrypted keyword. Another one of the conventional PEKS and our suggested DS-PEKS may be the test formula is separated into two algorithms, Front Make certain Back Test operated by two independent servers. This is often required for achieving security from the inside keyword guessing attack. Within the DS-PEKS system, upon acquiring a question inside the receiver, the important thing server pre-processes the trapdoor and PEKS cipher texts getting its private key, then transmits some internal testing-states for that back server while using the corresponding trapdoor and PEKS cipher texts hidden. A corner server will pick which documents are queried using the receiver getting its private key along with the received internal testing-states at the front server [5]. You have to understand that both front server along with the back server here needs to be “honest but curious” and won't collude with one another. More precisely, both servers perform testing strictly transporting out an agenda procedure but could be thinking about the specific keyword. We must understand that the next security models also imply the safety guarantees outside adversaries that have less capacity in comparison to servers. We introduce two games, namely semantic-securityagainst selected keyword attack and indistinguishability against keyword guessing attack1 to capture the safety of PEKS ciphers text and trapdoor, correspondingly. The PEKS cipher text doesn't reveal any specifics of the specific keyword for the foe. This security model captures the trapdoor reveals no specifics of the specific keyword for that adversarial front server. Adversarial Back Server: The safety types of SS - CKA and IND - KGA in relation to an adversarial back server become

individuals against an adversarial front server. Here the SS - CKA experiment against an adversarial back server is equivalent to the main one against an adversarial front server apart from the foe is supplied the non-public type in the rear server instead of this right in front server. We omit the facts for simplicity. We reference the adversarial back server A within the SS - CKA experiment just as one SS - CKA foe and define its advantage. Similarly, this security model aims to capture the trapdoor doesn't reveal any information for that back server and so is equivalent to that right in front server apart from the foe owns the non-public type in the rear server instead of this right in front server. Within our defined security considered IND-KGA-II, it's crucial the malicious back server cannot learn any specifics of the specific two keywords involved in the internal testing-condition. To begin with, we must understand that both keywords involved in the internal-testing condition plays exactly the same role no matter their initial source Therefore, the job within the foe should be to guess the 2 underlying keywords within the internal testing overuse injury in general, rather for each within the initial PEKS cipher text along with the initial trapdoor. Therefore, it's inadequate for the foe to submit number of challenge keywords and so we must hold the foe to submit three different keywords within the challenge stage and guess which two keywords are selected because of the challenge internal-testing condition. A principal component of our construction for dual-server public key file encryption with keyword search is smooth projective hash function (SPHF), an idea created by Cramer and Shoup. During this paper, we must have another critical property of smooth projective hash functions [6]. Precisely, we must hold the SPHF to obtain pseudo-random. During this paper, we introduce a totally new variant of smooth projective hash function. Our plan's considered because the efficient in relation to PEKS computation. Because our plan doesn't include pairing computation. Particularly, this program necessitates most computation cost because of 2 pairing computation per PEKS generation. In relation to trapdoor generation, as all of the existing schemes don't involve pairing computation, the computation price is reduced in comparison with PEKS generation [7]. You have to note thetrapdoor generation within our plans a little more than individuals of existing schemes because of the additional exponentiation computations. You have to understand that this extra pairing computation is carried out across the user side rather within the server. Therefore, it may be the computation burden for users who are able to make use of a simple device for searching data. Within our plan, although we have to have another stage for the testing, our computation price is really lower in comparison with any existing plan once we don't require any pairing computation and searching jobs are handled using the server. We implemented data consistency through TPA in cloud.

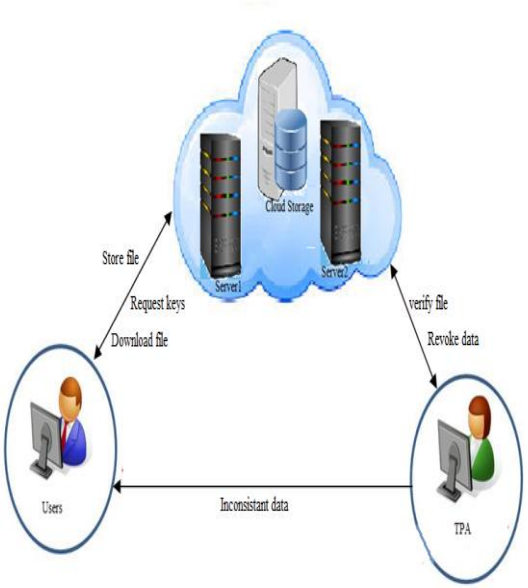

Fig: System Architecture

V.CONCLUSION

During this paper, we suggested a totally new framework, named Dual-Server Public Key File encryption with Keyword Search (DS-PEKS), that may steer obvious from the inside keyword guessing attack that's an natural vulnerability within the traditional PEKS framework. You have to understand that this extra pairing computation is carried out across the user side rather within the server. Therefore, it may be the computation burden for users who are able to make use of a simple device for searching data. We introduced a totally new Smooth Projective Hash Function (SPHF) and attempted round the extender to make a normal DS-PEKS plan. A dependable instantiation within the new SPHF while using Diffie-Hellman problem is also presented within the paper, which gives a dependable DS-PEKS plan without pairings. In relation to trapdoor generation, as all of the existing schemes don't involve pairing computation, the computation price is reduced in comparison with PEKS generation. We extended the system by adding data consistency through TPA in cloud

VI.REFERENCES

[1] Rongmao Chen, Yi Mu, Senior Member, IEEE, Guomin Yang, Member, IEEE, FuchunGuo, and Xiaofen Wang, "Dual-Server Public-Key Encryption With KeywordSearch for Secure Cloud Storage", ieee transactions on information forensics and security, vol. 11, no. 4, april 2016.

[2] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definition and efficient constructions," in Proc. 13th ACMConf. Comput. Commun. Secur. (CCS), 2006, pp. 79–88.

[3] C. Cocks, "An identity based encryption scheme based on quadratic residues," in Cryptography and Coding. Cirencester, U.K.: Springer, 2001, pp. 360–363.

[4] J. Baek, R. Safavi-Naini, and W. Susilo, "On the integration of public key data encryption and public key encryption with keyword search," in Proc. 9th Int. Conf. Inf. Secur. (ISC), 2006, pp. 217–232.

[5] D. Khader, "Public key encryption with keyword search based on K-resilient IBE," in Proc. Int. Conf. Comput. Sci. Appl. (ICCSA), 2006, pp. 298–308.

[6] K. Emura, A. Miyaji, M. S. Rahman, and K. Omote, "Generic constructions of secure-channel free searchable encryption with adaptive security," Secur. Commun. Netw., vol. 8, no. 8, pp. 1547–1560, 2015.

[7] 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, Jul. 2013.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.