Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 13, Issue 1, January 2022: 386-395

An Authorized and Efficient Searching Mechanism based on Keywords in Cloud Computing

Dr. U. Mohan Srinivas #1, Dr. D. Bujji Babu #2, L. Sravani #3, Ch. Amrutha #4, K.V.Sudish Kumar #5, K. Sri Hari #6
#1 Associate Professor, #2 Professor & HOD, #3,4,5,6 MCA Scholars
Department of MCA, QIS College of Engineering and Technology (Autonomous), Ongole
Corresponding Mail: sriharikosanam12345@gmail.com

Abstract
Cloud hosting becomes the main way to store and share information for corporate customers. The ideal method to safeguard trade secrets is to encrypt information before it is downloaded into the cloud, but regular operations such as cloud searching are not feasible. Moreover, since workers have various architectural layers, a high-level employee should be able to see lower-level employee data to see whether such users are violating the law without alerting the employees. The PEKS is a well-known cryptography basis which enables keywords in cryptosystem configurations to be searched without decryption, making it appropriate for safe cloud storage. Unfortunately, no known PEKS scheme allows the monitoring feature without sender permission. We are proposing a hierarchy cryptographically search encoding (HPEKS), as well as PEKS variants, as well as a semi-generic architecture based on PKTree and PEKS plans to solve this problem. We have developed an excellent HPEKS technique called DHPEKS, which combines public and symmetrical key encryption with safe keyword search, to better fit business private data exchange. We show that our DHPEKS technology is safe using the random oracle safety criteria. It guarantees the security of external offline keywords and the transparency property, in order to transmission encrypted data to the organisation, the sender does not need to know the internal hierarchical structure of a company. Theoretical evaluations and detailed testing show that with existing PEKS systems, our DHPEKS system has comparable efficiency.
Keywords: Public key encryption, keyword search, attack devaluation, cloud storage, concealed data sharing. Keywords

I. **Introduction**

Company or corporate data increase quickly with the rapid growth of computer and communication technology. PCS is becoming a key concern for most businesses because of the low cost of data handling. Statistics show that most businesses have used at least one PCS[1]. It enables users to access the majority of the material over the internet using mobile devices with limited storage. One of the tasks the PCS provides is to store and manage messages and documents exchanged by users. However, the risk of a security breach also restricts certain

businesses' use of PCS[1]. According to data, security issues are the most important worry for respondents[2]. Encryption is an easy way to prevent this risk before uploading, but the application is restricted to data. One of the cloud computing applications is the Office Automation (OA) system. Every day, a business or the government produces thousands of messages. Real OA services are often outsourced to the cloud service provider, since they offer considerable advantages in data storage and management costs. Messages are often stored as plaintexts in the OA system which enable attackers to read and therefore lead Economic loss or unfairness to the public.

For example, a project headed by a business may be stopped by its commercial competitor if the related information leaks in advance. Another example is the leakage of certain government policies that may lead anybody ahead, which is unfair for the public. To avoid such a situation, an appropriate access control method should be developed that enables authorised users to see the corresponding messages and denies access to them to unregistered users or less privileged users. For example, the CEO supervises a number of other employees in a business, including the Chief Operating Officer (COO), the CFO, the Chief Information Officer (CIO), and the Chief Technology Officer (CTO). A number of employees are supervised by the COO, CFO, CIO and CTO. In order to allow the CEO to communicate from the COO to the CIO, the CEO effectively controls the business. The risk of information leakage may be decreased in order to limit CFO access to CTO-CEO communications. Every information escapes when the system breaks in although the OA system has an Access Control Strategy 1. The ciphertext access control technique is thus a practical and important way to preserve privacy. Crypt messages with distinct public keys for users is the basic conception of ciphertext-level access control method, which only displays ciphertexts even if the OA system is affected, exposing less information. The shell of the encryption system satisfies the following requirements.

1) Searchability The download and decryption of all ciphertexts is inefficient to find some specific documents like 'contract' documents A ciphertext search method without decryption should be used to rapidly find these documents without disclosing information about the content of the document.

2) Control of user priority access. Different individuals in the current world often have to convey and manage a message or work. Only the person with the equivalent or higher priority of access should decode the encrypted message. A tree-like access control method may be used to decode ciphertexts, taking into consideration the structure of receivers.

3) Operativity. A transmitter Alice does not need to know the receiving organization's internal structure. If Alice transmits an encrypted message to Bob, it's just necessary to know the valid public key of Bob. In other words, Alice does not need to grasp the importance of Bob in the business, or who has a higher priority in the company than Bob.

No primitive cryptography, to the best of our knowledge, complies with the above requirements simultaneously. Public search keyword encryption (PEKS) enables you to search for a keyword without decryption, but does not have an access control mechanism. Access control at ciphertext level is provided by attribute-based encryption and identity-based Hierarchicalws (HIBE). The searchable encryption system seems to be feasible to combine ABE or HIBE with PEKS but the sender must be acquainted with the structure behind the receiver in ABE and HIBE. Therefore, a new basic cryptography suitable for the aforementioned cloud data sharing scenario is needed. The contributions of our work are:

1. Public Key Tree: We have incorporated Public Key Tree (PKTree) in searchable encryption and created a new idea called hierarchical Public Key Search Encryption (HPEKS). In HPEKS users may use their public keys to search for encrypted ciphertexts. In particular, if you have a collection of users with a hierarchical structure, you may search the higher access allowance via ciphertexts sent to users with lower access allowances. For example, when Alice watches Bob and Carlos, Alice may perform search operations by utilising ciphertexts that are encoded using Bob's and Carlos' public keys.

2) HPEKS semi-generic PEKS construction: a semi-generic HPEKS design in combination with a bilinear current PEKS pairing approach in literature shows the feasibility of the proposed technique PKTree.

3) Enhanced HPEK SCREEM dDHPEKS: We provide an advanced HPEKs system called a decryptionable public hierarchy key search key encryption (dDHPEKS), in which users may only search for a specific server, to avoid offline keyword guessing attacks. Furthermore, our system incorporates PEKS and PKE, which not only searches but also decrypts keywords. Our dDHPEKS system has an interesting function and is transparent in order for the sender not to know the internal hierarchical structure of the organisation prior to encrypting the recipient's keyword. The sender must encrypt the keyword only under the chosen recipient's public key. HIBE or ABE in combination with PEKS does not provide this feature.

(4) Security and Effectiveness: we explicitly describe dDHPEKS security covering key secret security, keywords privacy and plaintext privacy and test the security of our dDHPEKS software in accordance with the relevant safety standards. In theory, Run ning overhead of dDHPEKS is evaluated using C language and PBC library[2]. The analytical and test results show that our dDHPEKS system is comparable to existing PEKS systems.

II. Related Works

In 2000 Song et al.[3] introduced the SSE concept and the very first privacy preserving technique. It enables a search key user to browse through encrypted content for a phrase without decryption. But Song et alsystem [4] and other methods [6] are often restricted to the condition that a query keyword cannot be communicated with anyone else, so that the data holder may only look for the cipher communications. Very first cryptographically SE method, Public Key Cryptography with Search Query (PEK SE), is proposed by Boneh et al.[7] in SE to resolve the exchange of data. In PEKS, the data owner (encrypter) uses the key of the recipient, but instead the recipient uses its secret key to build a trapdoor to browse through the cypher for relevant keywords. Scholars have also highlighted additional PEKS problems. Park et al. and Golle et al. developed public key search methods[8][9] that let recipients look for articles in the same look with all terms. A novel concept dubbed the Inter searching rankings was suggested for a rating google results list[10][11][12]. The suggested Multi-Receiver Cryption Settings were proposed by Bao et al. [13] and others [14]. The search query (ABEKS) has been suggested to cipher keywords[17][14] in order to offer PEKS fine-access control. Byun et al.[18] and Yau et al.[19] have stressed the importance of the new online attack on PEKS system and other current PEKS systems (KGA). They also pointed out that almost current PEKS are not able to manage the online devaluation of hostile assaults (IKGA), such as search servers. Fang et al. [20] suggested that safe free PEKS route witch may withstand KGA, but not IKGA. Huang and Li have suggested the concept of a Public Key Verified Keyword Search Security (PAEKS)[21][22] which challenges IKGA to encryption by rejecting keywords of the opponents. He et al.[23] and Li et al.[24] incorporated PAEKS in the public certificate codes and identifying settings. Chen et al.[25] and Chen et al.[26], who used two system model servers, have also solved the IKGA issue.

III. Proposed Method.

Here we show the HPEKS system (Figure 1). The system comprises of three entities which refer to senders, data recipients and datacenters.

(1) Send data: encrypts receiver data from the open variable of the receiver and sends a cypher text to the cloud server recipient.

(2) Data recipients: the data recipients have a hierarchical tree-based structure. Each receiver has a specified tree node. Instead, a data receptor may search for ciphertexts transmitted to itself and all its offspring.

(3) Cloud server: offers data receivers with storage and computation services. Public variables of the receiver group and transmitted encrypted messages are kept on the cloud server. It also allows recipients to search for encrypted message.

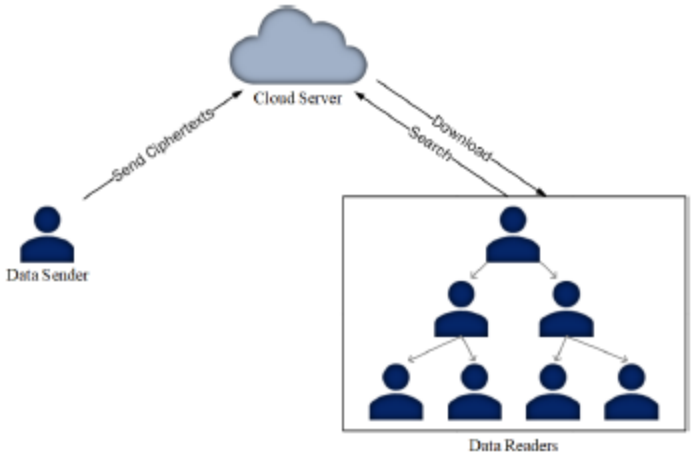

Fig. 1. System Model of Tree-based Hierarchical Multi-receiver Searchable Encryption

In this section we demonstrate how an HPEKS system based on the proposed PKTree may be built by providing a semi-generic HPEKS. For decryption of ciphertexts, we offer an improved HPEKS technique to retain coded keywords and plaintext integrated. This half-generic HPEKS solution allows users to search for a particular keyword by delegating the cloud server without decrypting the incoming chips. In particular, a recipient may look for the ciphertexts transmitted to its monitored recipients. However, the decryption function is not supported on the basis of current PEKS methods. While Baek and alPKE/PEKS[30] allow the description of ciphertexts, the duration of the plaintext is restricted. We have developed an advanced method called the decryptable public key search keyword (dDHPEKS) to encode changeable plaintext length and decrypt matching chip text to address this issue. In addition, only the tester may check whether the ciphertext contains the same word as the question trapdoor. Our approach is superior than HIBE (with search terms), since it is visible. In order to disseminate an encrypted record with many recipients, a sender on our system need not know its hierarchy. It just has to encrypt the record at the lowest level under the recipient's public key. In contrast, a sender in HIBE must know the hierarchy of identification.

IV. Discussion and Results

We compare it to prior searchable encryption systems [7, 30, 31] to assess how well the dDHPEKS system is operating. Table I indicates that the operational efficiency of our idea is similar to system comparison. Certain operating efficiencies may be affected to improve safety and functionality. Of the four methods, only DDHPEKS can withstand assaults outside keywords, including PKE and PEKS. Our approach also permits PKTree-based node monitoring, which is not possible with prior PEKS systems. We used the C and PBC libraries[2] at Ubuntu OS 19.04 for BDOP-PEKS[7], BSS-PKE/PEKS[30] and HL-PEKS[31], for comparative schemes with Intel Intel 2.3 GHz i5 CPUs with LPDDR3 RAM 8GB 213 3 MHz. A kind- A pairing was selected to initialise the 1024-bit RSA level encryption mechanism. The initial step

is to search for a particular data collection by extracting keywords. Each item should be linked to a keyword for each dataset. The keyword is encrypted using the suggested encryption algorithm and encrypted by a certain extra encryption technique, such as AES. The inverted index usually allows efficient encryption of data search. The inverted index basically contains a list of (encrypted) keywords with each keyword linked to a queue of keyword entries. Keywords may be put in a frequent dictionary, and differing keyword bits do not impact the application performance, because each keyword is hacked before encryption to a fixed string. We thus choose the Oxford dictionary as the keyword for implementation and use all its terms to enter the relevant scheme algorithms. Our dDHPEKS encoding approach, as shown in Figure 2, is slower than the other three. It is reasonable to encrypt approximately half the K session key that is not in BDOPPEKS or HL-PEKS.

TABLE I: COMPARISONS WITH PEKS SCHEMES IN LITERATURE

Scheme	Encrypt	Trapdoor	Test	KGA	INT	INT/L	HMSE
BDOP-PEKS[7]	$2Exp_{G_1}$ + 2Hash + 1Pairing	$1Exp_{G_1}$ + 1Hash	1Hash + 1Pairing	No	No	No	No
BSS-PKE/PEKS[30]	$3Exp_{G_1}$ + 4Hash + 1Pairing	$1Exp_{G_1}$ + 1Hash	1Hash + 1Pairing	No	Yes	No	No
HL-PEKS[31]	$2Exp_{G_1}$ + 2Hash + 3Pairing	$3Exp_{G_1}$ + 1Hash	$1Exp_{G_1}$ + 1Hash + 1Pairing	Yes	No	No	No
Our dDHPEKS	$4Exp_{G_1}$ +$2Exp_{G_2}$ + 5Hash + 1Pairing + 1AES	$1Exp_{G_1}$ + $3Exp_{G_2}$ + 3Hash	$1Exp_{G_2}$ + 2Pairing	Yes	Yes	Yes	Yes

KGA: The schemes can resist outside keyword guessing attacks;
INT: The schemes integrate the encrypted keyword and plaintext;
INT/L: The schemes are adaptable of any length of the integrated plaintexts;
HMSE: The schemes support hierarchical multi-user keyword search.

Note that the overall AES encryption method of our dDHPEKS system was not included in the findings of the experiment since the plaintext length is uncertain and impacted. Figure 3 demonstrates that our dDHPEKS method has an efficient trapdoor algorithm, comparable to and more effective than the BDOP-PEKS and BSS-PKE/PEKS systems. Figure 4 shows the overhead test methods of the four approaches. The mean overhead for BDOP-PEKS and PEKS searches is close to 0.75 seconds for a keyword across 1000 ciphertextes, the HL-PEKS scheme is 2.00 seconds and our dDHPEKS method is approximately 2.85 seconds. Although the two previous methods are quicker, they can't withstand assaults beyond the offline keyword. The search efficiency of our dDHPEKS method is similar to HL-PEKS at the same security level.

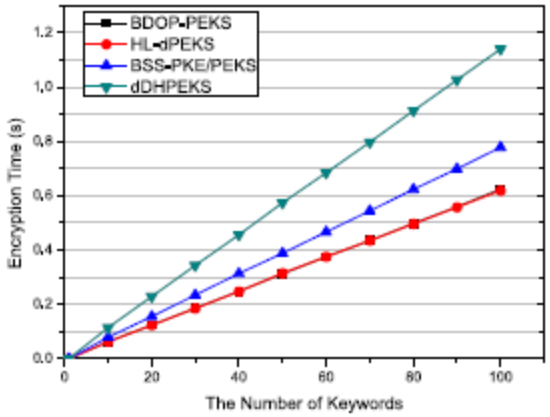

Fig. 2. Comparison of Encryption Algorithms

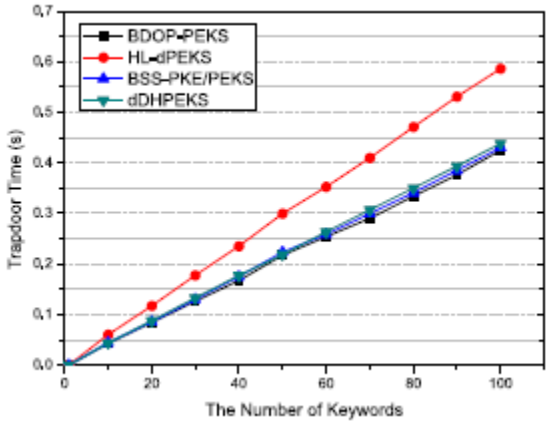

Fig. 3. Comparison of Trapdoor Algorithms

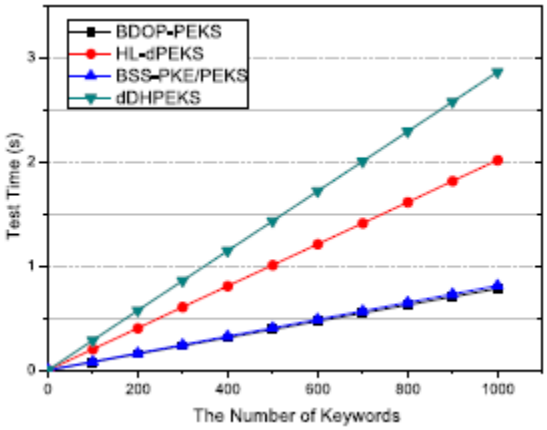

Fig. 4. Comparison of Test Algorithms

V Future Scope and Conclusion

In this article, we have suggested a new technique called HPEKS (hierarchical public key encryption) to monitor your kid users. We have created a PKTree structure and utilised it to

generate a semi-generic HPEKS build. We have suggested an up-to-date HPEKS dDHPEKS scheme that combines PEKS and PKE, and which can withstand assaults outside offline. Experiments indicate that our dDHPEKS system has an efficiency similar to that of current PEKS systems.

References

[1] W. Kim, "Cloud computing trends: 2018 state of the cloud survey," URL https://www.rightscale.com/blog/cloud-industryinsights/ cloud-computing-trends-2018-state-cloudsurvey, [Online], 2018.

[2] B. Lynn and et al, "Pairing-based cryptography library," https://crypto.stanford.edu/pbc/, 2013.

[3] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proceedings of IEEE Symposium on Security and Privacy. S&P 2000, 2000, pp. 44–55.

[4] Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in International Conference on Applied Cryptography and Network Security. Springer, 2005, pp. 442–455.

[5] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," Journal of Computer Security, vol. 19, no. 5, pp. 895–934, 2011.

[6] K. Kurosawa and Y. Ohtaki, "Uc-secure searchable symmetric encryption," in International Conference on Financial Cryptography and Data Security. Springer, 2012, pp. 285–298.

[7] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, Public Key Encryption with Keyword Search. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 506–522.

[8] D. J. Park, K. Kim, and P. J. Lee, "Public key encryption with conjunctive field keyword search," in International Workshop on Information Security Applications. Springer, 2004, pp. 73–86.

[9] P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data." in Applied Cryptography and Network Security, Second International Conference, ACNS 2004, Yellow Mountain, China, June 8-11, 2004, Proceedings, 2004, pp. 31–45.

[10] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacypreserving multi-keyword ranked search over encrypted cloud data," IEEE Transactions on parallel and distributed systems, vol. 25, no. 1, pp. 222–233, Jan 2014.

[11] Z. Fu, X. Sun, Z. Xia, L. Zhou, and J. Shu, "Multikeyword ranked search supporting synonym query over encrypted data in cloud computing," in Performance Computing and Communications Conference (IPCCC), 2013 IEEE 32nd International. IEEE, 2013, pp. 1–8.

[12] H. Li, D. Liu, Y. Dai, T. H. Luan, and X. S. Shen, "Enabling efficient multi-keyword ranked search over encrypted mobile cloud data through blind storage," IEEE Transactions on Emerging Topics in Computing, vol. 3, no. 1, pp. 127–138, 2015.

[13] F. Bao, R. H. Deng, X. Ding, and Y. Yang, "Private query on encrypted data in multi-user settings," in International Conference on Information Security Practice and Experience. Springer, 2008, pp. 71–85.

[14] F. Zhao, T. Nishide, and K. Sakurai, "Multi-user keyword search scheme for secure data sharing with finegrained access control," in International Conference on Information Security and Cryptology. Springer, 2011, pp. 406–418.

[15] Y. Yang, H. Lu, and J. Weng, "Multi-user private keyword search for cloud computing," in Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on. IEEE, 2011, pp. 264–271.

[16] C. Van Rompay, R. Molva, and M. O¨ nen, "Multi-user searchable encryption in the cloud," in International Information Security Conference. Springer, 2015, pp. 299–316.

[17] M. Hattori, T. Hirano, T. Ito, N. Matsuda, T. Mori, Y. Sakai, and K. Ohta, "Ciphertext-policy delegatable hidden vector encryption and its application to searchable encryption in multi-user setting," in IMA International Conference on Cryptography and Coding. Springer, 2011, pp. 190–209.

[18] J. W. Byun, H. S. Rhee, H.-A. Park, and D. H. Lee, "Offline keyword guessing attacks on recent keyword search schemes over encrypted data," in Workshop on Secure Data Management. Springer, 2006, pp. 75–83.

[19] W.-C. Yau, S.-H. Heng, and B.-M. Goi, "Off-line keyword guessing attacks on recent public key encryption with keyword search schemes," in International Conference on Autonomic and Trusted Computing. Springer, 2008, pp. 100–105.

[20] L. Fang, W. Susilo, C. Ge, and J. Wang, "A secure channel free public key encryption with keyword search scheme without random oracle," in Cryptology and Network Security, International Conference, CANS 2009, Kanazawa, Japan, December 12-14, 2009. Proceedings, 2009, pp. 248–258.

[21] Q. Huang and H. Li, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," Information Sciences, vol. 403-404, pp. 1 – 14, 2017.

[22] M. Ma, S. Zeadally, N. Kumar, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," Cryptology ePrint Archive: Report 2018/007, 2018.

[23] D. He, M. Ma, S. Zeadally, N. Kumar, and K. Liang, "Certificateless public key authenticated encryption with keyword search for industrial internet of things," IEEE Transactions on Industrial Informatics, vol. 14, no. 8, pp. 361 8–3627, 2017.

[24] H. Li, Q. Huang, J. Shen, G. Yang, and W. Susilo, "Designated-server identity-based authenticated encryption with keyword search for encrypted emails," Information Sciences, vol. 481, pp. 330 – 343, 2019.

[25] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "Dualserver public-key encryption with keyword search for secure cloud storage," IEEE Transactions on Information Forensics and Security, vol. 11, no. 4, pp. 789–798, April 2016.

[26] R. Chen, Y. Mu, G. Yang, F. Guo, X. Huang, X. Wang, and Y. Wang, "Server-aided public key encryption with keyword search," IEEE Transactions on Information Forensics and Security, vol. 11, no. 12, pp. 2833–2842, Dec 2016.

[27] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Annual International Cryptology Conference. Springer, 2001, pp. 213–229.

[28] J. Katz and Y. Lindell, Introduction to modern cryptography, Second Edition. CRC press, 2014.

[29] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in Annual International Cryptology Conference. Springer, 2004, pp. 41–55.

[30] J. Baek, R. Safavi-Naini, and W. Susilo, On the Integration of Public Key Data Encryption and Public Key Encryption with Keyword Search. Springer Berlin Heidelberg, 2006.

[31] C. Hu and P. Liu, "A secure searchable public key encryption scheme with a designated tester against keyword guessing attacks and its extension," in International Conference on Computer Science, Environment, Ecoinformatics, and Education. Springer, 2011, pp. 131–136.