Review of Key-aggregate Searchable Encryption Technique for Group Data Shearing Literature

M.K.Kurhdkar, Prof.Mrs.V.M.Deshmukh

Abstract— The Ability of shearing encrypted data with group users via cloud network it concern over confidential data leak over network. Its challenge to developed encryption algorithm lies in efficient management of encrypted key.The desired flexibility of group of users demanding different keys for encryption for different documents. However, this also implies the necessity of distributing number of keys for decryption and search keys, and those users received the keys successfully send equal number of trapdoor keys to cloud to search documents over cloud . The implied the secularly sending group data to different users over network renders the approach impractical. It practicable by proposing literature on (KASE) Key Aggregation Searchable Encryption and instantiating the concept through a concrete KASE scheme, in data owner required only one key for encryption for group of documents and users required single trapdoor key for searching of different documents

Index Terms—Searchable encryption, data sharing, cloud storage, data privacy

1 INTRODUCTION

Cloud has large storage technology as a promising solution for providing ubiquitous, convenient, and on-demand easily accesses to large amounts of data shared over the Internet.Today, Users are shearing their personal data such as videos, audios and photos with their friends through social network websites developed on cloud storage on a daily basis. The peoples in Business are also being attracted towards the application developed by cloud storage due to its number of benefits, including minimum cost, efficiency, greater agility, and better resource utilization. However, while enjoying the convenience of sharing data via cloud storage, users are also increasingly concerned about inadvertent data leaks in the cloud. Such data leaks, caused by a malicious adversary or a misbehaving cloud operator, can usually lead to serious breaches of personal privacy or business secrets i.e Recently Celebrity photos leak over internet. To notify the confidential data leaks in clod storage, a common approach is for the data owner to encrypt the data before uploading them to the cloud, such that later the encrypted data may be retrieved and decrypted by those who have the decryption keys. Such a cloud storage is often called the cryptographic cloud storage [6]. However,The data encryption is very difficult job for users and also then retrieve the data containing keyword selectively. There are various way for data encryption one common solution is to apply a searchable encryption (SE) scheme in which the data owner is necessary to encrypt potential keyword and upload both the encrypted data and keyword for retrieving data matching data, such that, for retrieving data matching a keyword, The user send the trapdoor for performing search over the network the user will send the corresponding keyword trapdoor to the cloud for performing search over the encrypted data. It achieved the basic security after combining the searchable encryption with cryptography of a cloud storage, implementing such a system for large scale applications involving millions of users and billions of files may still be hindered by practical issues involving the efficient management of encryption keys, which, to the best of our knowledge, are largely ignored in the literature. First of all, the need for selectively sharing encrypted data with different users it means shearing of photos sharing a photo with certain friends in a social network application, or sharing a business document with certain colleagues on a cloud drive usually demands different with the people encryption keys to be used for different files. However number of key implies for encryption and trapdoor keys for searching, both for them to search over the encrypted files and to decrypt the files, will be proportional to the number of such files. Such a large number of keys must not only be distributed to users via secure channels, but also be securely stored and managed by the users in their devices. In addition, a large number of searching keys generated by users and submitted to the cloud in order to perform a keyword search over many files. The implied need for secure communication, storage, and computational complexity may render such a system inefficient and impractical. In this literature, we proposed to developed the new concept of key-aggregate searchable encryption (KASE), and instantiating the concept through a permanent developed KASE scheme. The proposed KASE scheme that supports the searchable group data sharing functionality and applies to any data storage , it means it selectively shearing selective documents with selective people, while allowing the latter to perform keyword search over the former. To search group data it required the key has two fold. First, it necessary to data owner has only single distributed keys (instead of a group of keys) to a user for sharing any number of files. Second, users need to submit only one aggregate trapdoor key instead of a group of trapdoors to the cloud for performing keyword search over any number of shared files. The KASE scheme proposed in this paper is the first known scheme that can satisfy both requirements (C). Contributions. More specifically, our main contributions are as follows.

1) KASE i.e Key Aggregation Searchable Encryption has compose of polynomial algorithms for security parameter setup, key generation, encryption, key extraction, trapdoor generation, trapdoor adjustment, and trapdoor testing. We describe both security analysis and functional requirements for designing KASE shceme KASE scheme.

2) We then instantiate the KASE framework by designing a concrete structure KASE scheme. After providing detailed development for the seven algorithms, we analyze the efficiency of the scheme, and establish its security through overall analysis.

3) We discuss various practical issues for proposed KASE scheme for group data shearing to different users with single aggregate key and different and evaluate its performance. The evaluation confirms our system can meet the performance requirements of practical applications. The rest of the paper is organized as follows. First, we review some background knowlege . We then define the general KASE framework in Section design a concrete KASE scheme and analyze its efficiency and security. We implement and evaluate a KASE-based group data sharing system in.

2 FRAMEWORK OF KEY- AGGREGATE SEARCHABLE ENCRYPTION (KASE)

In this section, we first describe the basic problem, and then define a genericframework for key-aggregate searchable encryption (KASE) and provide requirements for designing KASE scheme. First we introducing the problems are as follows

2.1 Problem Statement

Consider a scenario where two employees whose are working in company shearing confidential data over network using public cloud storage (e.g., dropbox or syncplicity). For instance, Adams wants to upload large financial documents to upload the cloud storage, which are meant for the heads of different departments to review. Suppose those documents contain highly confidential information that should only be accessed by authorized users, and Bill is one of the directors and is to download this authorized different documents. Due to concerns about potential data leakage in the cloud, Adams encrypts these documents with different encryption keys, and generates keyword cipher texts based on department names, before uploading to the cloud storage. Adams uploads and distributes those documents with the directors using the sharing functionality of the cloud storage. In order for Bill to see the documents related to his department, Adams must delegate to Bill the authorization both for keyword search over those documents i.e encryption key and trapdoor keys, and for decryption of documents related to Bill's department.With privious approach, Adams must securely send all the (SE) searchable encryption keys to Bill. After receiving these keys, Bill must store them securely, and then he must generate all the trapdoors keywords using these keys in order to perform document search. Adams is assumed to have a personal document setcign j=1, and for each document docj, a searchable encryption key kj is used. Without loss of generality, we suppose Alice wants to share m documents fdocigm j=1 with Bob. In this case, Adams must send all the searchable encryption keys fkigm j=1 to Bill. Then, when Bill wants to retrive documents containing a keys, he must generate keyword trapdoor Tri for each document doci with key ki and submit all the trapdoors fTrigm i=1 to the cloud server. When n is sufficiently large, the key distribution and storage as well as the trapdoor generation may become too expensive for Bills client-side device, which basically defies the purpose of using cloud storage.
In this paper, we propose the novel approach of (KASE) key-aggregate searchable encryption (KASE) as a better solution on problem, in KASE, Adams only necessary to distribute a single aggregate key, instead of fkigm i=1 for sharing m documents with Bob, and Bob only needs to submit a single aggregate trapdoor, instead of fTrigm i=1, to the cloud server. The cloud3. We describe related work in Section 4. Wedesign a concrete KASE scheme and analyze its efficiency and security in Section 5. We implement and evaluate a KASE-based group data sharing system in Section 6. Finally, we conclude the paper in Section 7.

3 PRLIMINARIES

His section which is from KASE this section, we review some cryptology and assumptions which will be important for this paper. In the rest of our discussions, let G and G1 be two cyclic groups of prime order p, and g be a generator of G. Moreover, let doc be the document to be encrypted, k the searchable encryption key, and Tr the trapdoor for keyword search.

3.1 Complexity Assumption through algorithm

Bilinear Map A bilinear map is a map e : G _ G ! G1 with the following properties: 1. Bilinearity: for all u; v 2 G and a; b 2 Z_ p , we have e(ua; vb) = e(u; v)ab. 2. Non-degeneracy: e(g; g) 6= 1.3. Computability: there is an efficient algorithm to compute e(u; v) for any u; v 2 G.2.1.2 Bilinear Diffie-Hellman Exponent Assumption The bilinear Diffie-Hellman exponent (BDHE) assumption has been widely used to prove the security of some broadcast encryption (BE) schemes (e.g., [27]).The l-BDHE problem in G is stated as follows. Given a vector of 2l + 1 elements (h; g; g_; g(_2); _ _ _; g(_l); g(_l+2); _ _ _; g(_2l)) 2 (G_)2l+1 as input, output e(g; h)(_l+1) 2 G1. For convenience, we use gi to denote gi = g(_i) 2 G.Note that the input vector is missing the term gl+1 (i.e., g(_l+1)) such that the bilinear map seems to be of little help in computing the required e(g; h)(_l+1). An algorithm A has advantage " in solving l-BDHE in G if **Pr**[A(h; g; g1; ___; gl; gl+2; ___; g2l) = e(gl+1; h)] _ ", where the probability is over the random choice of generators g; h in G, the random choice of _ in Zp, and the random bits used by A.

Definition 1. The (l; ")□BDHE assumption holds in G if no algorithm has advantage more than " in solving the l□BDHE problem in G.

3.2 Broadcast Encryption

In broadcast encryption scheme(BE) it encrypts a message for some users who are listening on broadcast A encryption scheme (BE), ieved by encryption key and users uses private key to decrypt the broadcast.A BE encryption key can be describe in the polynomial algorithm ie Setup, Encrypt, Decrypt as follows:I have taken this algorithm from KASE We illustrate This algorithm

**Setup**(1, n):this fully dependant on the input scheme this algorithm is run by the system to set up the scheme. It takes as input a security **server** can use this a one trapdoor (aggregate) and some information in the form of documents to perform search using keyword and return the result to Bill. Therefore,in Key aggregate searchable encryption i.e, provide keyword search right achieved by shearing single aggregate key. We note that the delegation of decryption rights can be achieved using the key-aggregate encryption approach recently proposed in [4], but it remains an open problem to delegate the keyword search rights together with the decryption rights, which is the subject topic of this paper. To describe, the problem of developing a KASE scheme can be stated as:

"To design aggregate searchable encryption key scheme under which any subset of the keyword cipher texts (developed by the Searchable Encryption algorithm t) from searchable any set of documents with a constant-size trapdoor (produced by SE.Trpdr algorithm) generated by a constant size aggregate key."

4 THE KASE FRAMEWORK

The KASE framework is composed of seven algorithms. Specifically, to set up the scheme, the cloud server generates local parameters of the system through the

Setup algorithm, and parameter can be reused by different data owner to shear different file. For each data owner, he/she should produce a public/master-secret key pair through the

Keygen algorithm. Encrypt the keyword for each document via the encryption algorithm

Encrypt algorithm with the searchable encryption key. Then, the data owner can use key for generate aggregate key for different groups of documents.

Extract algorithm. The aggregate key can be distributed securely to permitted users who need to access those documents. an authorized user can produce a keyword trapdoor via the

Trapdoor algorithm using this aggregate key, and submit the trapdoor keyword for search to the cloud. After receiving the trapdoor, to perform the keyword search over the specified set of related documents, the cloud server will run

……
Cloud
Bob
Tr
Tr1
Adjust
Trm
……
…
Test
Test
Alice
Aggregate Key
for {*k1*, *k2*, *k3*, ……, *km*}
k1 km kn

the **Adjust** algorithm to generate the exact trapdoor for document, and then run the

Test algorithm to test whether the document contains the keyword. This framework is summarized in the following given algorithm.

_ **Setup**(1_, n): this algorithm is run by the cloud service provider to set up the scheme. On input of asecurity parameter 1_ and the maximum possible number n of documents which belongs to a data owner, it outputs the public system parameter params.

_ **Keygen**: this algorithm is run by the data owner to generate a random key pair (pk,msk).

_ **Encrypt**(pk, i): this algorithm is run by the data owner to encrypt the i-th document and generate its keywords' ciphertexts. For each document, this algorithm will create a delta _i for its searchable encryption key ki. On input of the owner's public key pk and the file index i, this algorithm outputs data ciphertext and keyword ciphertexts Ci.

_ **Extract**(msk, S): this algorithm is run by the data owner to generate an aggregate searchable encryption key for delegating the keyword search right for a certain set of documents to other users It takes as input the owner's master-secret key msk and a set S which contains the indices of documents, then out-

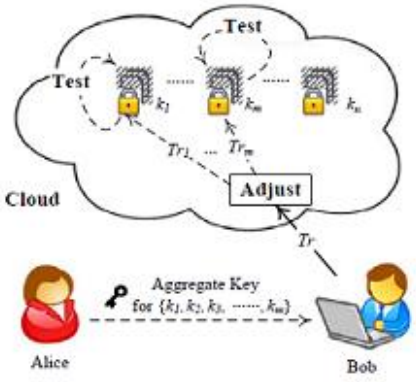

Fig. 2. Framework of key-aggregate searchable encryption.

5 REVIEW LITERATURE

Before we introduce our KASE scheme, this section
First reviews several categories of existing solutions
And explain their relationships to our work.

5.1 MUSE-Multi-user Searchable Encryption

There is a literature on searchable encryption, consisting SSE schemes and PEKS schemes .Its more common scenario In contrast to those existing work,in the context of cloud storage,

keyword search under the multi-tenancy setting . In such a scenario, the data owner would like to share a document only with a group of authorized users, and each user who has the access right can provide a trapdoor to perform trapdoor key search over cloud, the "multi-user searchable encryption" (MUSE) scenario.To study of [6], [13]–[15], [19] focus to such a MUSE technology, although they all adopt single-key combined with access control to achieve the goal. In[6], [19], MUSE schemes are build by sharing the document's searchable encryption key with all users who can access it, and broadcast encryption is used to achieve coarse-grained access control. In [13]–[18], attribute based encryption (ABE) is applied to achieve fine-grained access control aware keyword search. As a result, in MUSE, the main problem is how to control which users can access which documents, whereas how to reduce the number of shared keys and trapdoors is not considered. Key aggregate searchable encryption can provide the solution for the latter, and it can make MUSE more efficient and practical.

5.2 MKSE(Multi-Key Searchable Encryption)

In multiuser application consider the number of trapdoor keys for number of documents If user provides trapdoor keyword under each key with matching encrypted documents (if the user provides to the server a keyword trapdoor under each key with which a matching document might be encrypted), [28] first introduces the concept of MKSE multi-key searchable encryption and forward the first feasible scheme in 2013. MKSE multi-key searchable encryption authorized a user to provide a single keyword trapdoor to the server, but still allows the server to search for that trapdoor's keyword in documents encrypted with d keys. but these are in fact two completely different concepts. The goal of KASE is to delegate the keyword search right to any user by distributing the aggregate key to him/her in a group data sharing system, whereas the goal of MKSE multi-key searchable encryption is ensured that to identify using one trapdoor key to search documents correctly owing to a user.More specifically, denote by uki the key of user i. Suppose a user, say Bob (with key ukB), has m encrypted documents on the cloud server, and each is encrypted under a key kj for j = f1; _ _ _;mg. To allow the cloud server to adjust the trapdoor for each document with index j, Bob stores on the cloud server a public information called delta (denoted as _ukB;kj) which is relevant to both ukB and kj . when Bill wants to search for a word w over all the documents, he will use ukB to compute a trapdoor for the word w and submit it to the cloud server. The cloud server can use _ukB;kj to convert a keyword trapdoor under key ukB to a keyword trapdoor under kj ; this process is called adjust. In such a way, the cloud server can obtain trapdoors for word w under k1; ___; km while only receiving one trapdoor from Bob,and then perform a traditional single-key search with the new trapdoors.This approach of MKSE nspires us to focus on the problem of keyword search over a group of shared documents from the same user in the multiuser applications, and the adjust process in MKSE also provides a general approach to perform keyword search over a group of documents with only one trapdoor. However, the adjust process of MKSE needs a delta generated from both user's key and SE key of the document, so it does not directly apply to the design of a concrete KASE scheme.

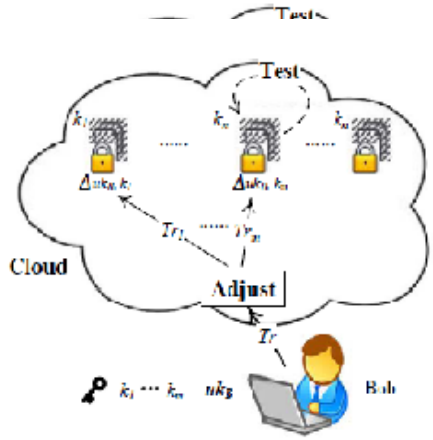

Fig. 3. Multi-Key Searchable Encryption.

5.3 Key-aggregate Encryption for Data Sharing

Data sharing systems based on cloud storage have attracted much attention recently [1]–[4]. In particular, Chu et al. [4] consider how to reduce the number of distributed data encryption keys. To share several documents with different encryption keys with the same user, the data owner will need to distribute all such keys to him/her in a traditional approach which is usually impractical. Aiming at this challenge, a keyaggregate Encryption (KAE) scheme for data sharing is proposed to generate an aggregate key for the user to decrypt all the documents. To allow a set of documents encrypted by different keys to be decrypted with a single aggregate key, user could encrypt a message not only under a public-key, but also under the identifier of each document. The construction is inspired by the broadcast encryption scheme [27]. In this construction, the data owner can be regarded as the broadcaster, who has public key pk and master-secret key msk; each document with identifier i can be regarded as a receiver listening to the broadcast channel, and a public information used in decryption is designed to be relevant to both the owner's msk and the encryption key; the message encryption process is similar to data encryption using
symmetric encryption in BE, but the key aggregation and data decryption can be simply regarded as the further mathematical transformation of BE.**Encrypt** algorithm and BE.**Decrypt** algorithm respectively.
The scheme [4] allows efficiently delegating the decryption rights to other users, and is the main inspiration of our study, but it does not support any search over the encrypted data. In the cloud environment,to achieve the goal of privacy-preserving
data sharing, keyword search is a necessary requirement. Fortunately, the KAE provides insights to the design of a KASE

scheme, although our scheme will require a more complex mathematical transformation to support keyword ciphertext encryption, trapdoor

5.4 Efficiency

In terms of efficiency, our team clearly achieved all the criteria's required for trapdoor ,encryption key and aggregate key:

1) The set S, which includes the indices of shared documents, has a linear size in the number of documents associated with the aggregate key. However, this does not affect the usefulness of the data sharing system, because the content of S can be safely stored in the cloud server (more details will be provided in section (5.5), such that there is no need to submit them to the cloud server when submitting the trapdoor.
2) The public system parameters PubK is O(n) in size, which is linear in the maximum possible number of documents belonging to a data owner, but not dependent on the number of documents stored in the cloud server, and hence this will not affect the system's practicality.

5.5 Security Analysis

To analyze the security of our scheme, and in particular show that the scheme satisfies the security requirements given in Section 3.3, we assume that the public cloud is "honest-but-curious". That is, the cloud server will only provide legitimate services according to pre-defined schemes, although it may try to recover secret information based on its knowledge. We also assume that the authorized users may try to access data either within or out of the scopes of their privileges. Moreover, communication channels involving the public cloud are assumed to be insecure.Based on the above considerations, we will prove the security of our scheme in terms of controlled searching and query privacy.

6 CONCLUSION

Thus we are considering the problems of data shearing over the cloud storage by analysis of developed system with proposed system the proposed system i.e. KASE Key Aggregate Searchable Encryption are very efficient to send the group data over network with different users.Their only single aggregate key send by data Owner and users will have only single trapdoor to search the documents over network .The concrete KASE framework analysis by different systems which found it very efficient and convenient to used and shearing confidential data over the network to the different users.

REFERENCES

[1]Baojiang Cui,Zeli Liu* and Lingyu Wang,"Key-Aggregete Searchable Encryption (KASE) For Group Data Shearing Via Cloud Storage",IEEE TRANCSACTION ON CMPUTERSVOL.6,NO.1,JANUARY 2014

[2]Reference from Internet Britanica Incyclopidia

[3] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure,Scalable, and Fine-Grained Data Access Control in Cloud Computing", Proc. IEEE INFOCOM, pp. 534-542, 2010.

[4] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance:The Essential of Bread and Butter of Data Forensics in Cloud Computing", Proc. ACM Symp. Information, Computer and Comm.Security, pp. 282-292, 2010.

[5] X. Liu, Y. Zhang, B. Wang, and J. Yan. "Mona: secure multiowner data sharing for dynamic groups in the cloud", IEEE Transactions on Parallel and Distributed Systems, 2013, 24(6): 1182-1191.

[6] C. Chu, S. Chow,W. Tzeng, et al. "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477.

[7] X. Song, D.Wagner, A. Perrig. "Practical techniques for searcheson encrypted data", IEEE Symposium on Security and Privacy, IEEE Press, pp. 44C55, 2000.

[8] R. Curtmola, J. Garay, S. Kamara, R. Ostrovsky. "Searchable symmetric encryption: improved definitions and efficient constructions", In: Proceedings of the 13th ACM conference on Computer and Communications Security, ACM Press, pp. 79-88, 2006.

[9] P. Van,S. Sedghi, JM. Doumen. "Computationally efficient searchable symmetric encryption", Secure Data Management, pp. 87-100, 2010.

[10] S. Kamara, C. Papamanthou, T. Roeder. "Dynamic searchablesymmetric encryption", Proceedings of the 2012 ACM conference on Computer and communications security (CCS), ACM, pp. 965-976, 2012.

[11] D. Boneh, C. G, R. Ostrovsky, G. Persiano. "Public Key Encryption with Keyword Search", EUROCRYPT 2004, pp. 506C522, 2004.

[12] Y. Hwang, P. Lee. "Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-user System", In: Pairing-Based Cryptography C Pairing 2007, LNCS, pp. 2-22, 2007.

[13] J. Li, Q. Wang, C. Wang. "Fuzzy keyword search over encrypted data in cloud computing", Proc. IEEE INFOCOM, pp. 1-5, 2010.

[13] Bosch, R. Brinkma, P. Hartel. "Conjunctive wildcard search over encrypted data", Secure Data Management. LNCS, pp. 114-127, 2011.

[14] C. Don G. Russello, N. Dulay. "Shared and searchable encrypted data for untrusted servers", Journal of Computer Security, pp. 367-397, 2011.

[15] F. Zhao, T. Nishde, K. Sakurai. Multi-User Keyword Search Scheme for Secure Data Sharing with Fine-Grained Access

Control. Information Security and Cryptology, LNCS, pp. 406-418,2012.

[16] J. W. Li, J. Li, X. F. Chen, et al. "Efficient Keyword Search over Encrypted Data with Fine-Grained Access Control in Hybrid Cloud", In: Network and System Security 2012, LNCS, pp. 490-502, 2012.

[17] J. Li, K. Kim. "Hidden attribute-based signatures without anonymity revocation", Information Sciences, 180(9): 1681-1689, Elsevier, 2010.

[18] X.F. Chen,. Li, X.Y. Huang, J.W. Li, Y. Xiang."Secure Outsourced Attribute-based Signatures",IEEE Trans. on Parallel and Distributed Systems,DOI.ieeecomputersociety.org/10.1109/TPDS.2013.180, 2013.

[19] J.Li, X.F. Chen, M.Q. Li, J.W. Li, P. Lee, Wenjing Lou. "Secure Deduplication with Efficient and Reliable Convergent Key Management",IEEE Transactions on Parallel and Distributed Systems,25(6): 1615-1625, 2014.

[20] Z. Liu, Z. Wang, X. Cheng, et al. "Multi-user Searchable Encryption with Coarser-Grained Access Control in Hybrid Cloud", Fourth International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), IEEE, pp. 249-255, 2013.

[21] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing", Proc. IEEE INFOCOM, pp. 525-533, 2010.

[22] B. Wang, B. Li, and H. Li, "Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud", Proc. 10th Int'l Conf. Applied Cryptography and Network Security, pp. 507-525, 2012.

[23] D. Boneh, C. Gentry, B. Waters." Collusion resistant broadcast encryption with short ciphertexts and private keys", Advances in CryptologyCCRYPTO 2005, pp. 258-275, 2005.

[24] D. H. Phan, D. Pointcheval, S. F. Shahandashti, et al. "Adaptive CCA broadcast encryption with constant-size secret keys and ciphertexts", International journal of information security, 12(4): 251-265, 2013.

[25] D. Boneh, B. Lynn, H. Shacham. "Short signatures from the Weil pairing", Advances in Cryptology ASIACRYPT 2001, pp. 514-532, 2001.

[26] L. B. Oliveira, D. F. Aranha, E. Morais, et al. "Tinytate: Computing the tate pairing in resource-constrained sensor nodes", IEEE Sixth IEEE International Symposium on Network Computing and Applications, pp. 318-323, 2007.

[27] M. Li, W. Lou, K. Ren. "Data security and privacy in wireless body area networks", Wireless Communications, IEEE, 17(1): 51-58, 2010.

[28] D. Boneh, C. Gentry and B. Waters. "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys", CRYPTO'05, pp. 258C275, 2005.

[29] R. A. Popa ,N. Zeldovich. "Multi-key searchable