Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 4, April 2016, pg.370 – 374

Key- Aggregate Searchable Encryption (KASE) For Group Data Sharing Via Cloud Storage

K.ANUSHA[1], V.LALITHA[2], P.SIVA KUMAR[3], S.S.V.R KUMAR.A[4]

[1]Department of CSE, VITS College of Engineering, Sontyam, Visakhapatnam, India
[2]Department of CSE, VITS College of Engineering, Sontyam, Visakhapatnam, India
[3]Department of CSE, VITS College of Engineering, Sontyam, Visakhapatnam, India
[4]Department of CSE, VITS College of Engineering, Sontyam, Visakhapatnam, India
anushakondabattulamail.com@gmail.com [1]; chinne.varma8@gmail.com [2]; sivakumar2150@gmail.com [3]; ssvrkumar@gmail.com [4]

Abstract- The capability of selectively sharing encrypted data with different users via public cloud storage may greatly ease security concerns over inadvertent data leaks in the cloud. A key challenge to designing such encryption schemes lies in the efficient management of encryption keys. The desired flexibility of sharing any group of selected documents with any group of users demands different encryption keys to be used for different documents. However, this also implies the necessity of securely distributing to users a large number of keys for both encryption and search, and those users will have to securely store the received keys, and submit an equally large number of keyword trapdoors to the cloud in order to perform search over the shared data. The implied need for secure communication, storage, and complexity clearly renders the approach impractical. In this paper, we address this practical problem, which is largely neglected in the literature, by proposing the novel concept of key aggregate searchable encryption (KASE) and instantiating the concept through a concrete KASE scheme, in which a data owner only needs to distribute a single key to a user for sharing a large number of documents, and the user only needs to submit a single trapdoor to the cloud for querying the shared documents. The security analysis and performance evaluation both confirm that our proposed schemes are provably secure and practically efficient.

Keywords- Searchable encryption, data sharing, cloud storage, data privacy.

I. INTRODUCTION

With growing dependency on internet for globalization, cost for owning IT Infrastructure, resources have increased. Cloud computing is a new concept that usually is an on demand leasing service for internet applications and IT resources. According to NIST definition, "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction".

Cloud computing reduces huge upfront investments and recurring ongoing maintenance cost due to its principle of "pay for what you use". In cloud computing, the resources can be in someone else's premises or network commonly known as providers. The resources can be leased and are accessed remotely by cloud users or cloud service buyers via internet or network. All request received by the cloud servers are processed and the output is sent back as normal process.

The cloud computing gives three sensitive states of concern in operational context of cloud

- Sending of data to the cloud,
- Receiving of data from the cloud to client's computer,
- Storage of data in cloud server which client may or may not own.

Cloud computing has several advantages but at the same time it opens up risks on security issues. The remote access could lead to security threats for which Information System (IS) Audit can be helpful.

II. OBJECTIVE

First, a data owner only needs to distribute a single Aggregate key (instead of a group of keys) to a user for sharing any number of files. Second, the user only needs to submit a single aggregate trapdoor (instead of a group of Trapdoors) to the cloud for performing keyword search over any numbers of shared files.

III. MODULE DESCRIPTION

Data Owner-In this module we executed by the data owner to setup an account on an untrusted server. On input a security level parameter and the number of cipher text classes n (i.e., class index should be an integer bounded by 1 and n), it outputs the public system parameter param, which is omitted from the input of the other algorithms for brevity.

Network Storage (Drop box)-With our solution, Alice can simply send Bob a single aggregate key via a secure e-mail. Bob can download the encrypted photos from Alice's Drop box space and then use this aggregate key to decrypt these encrypted photos. In this Network Storage is untrusted third party server or drop box.

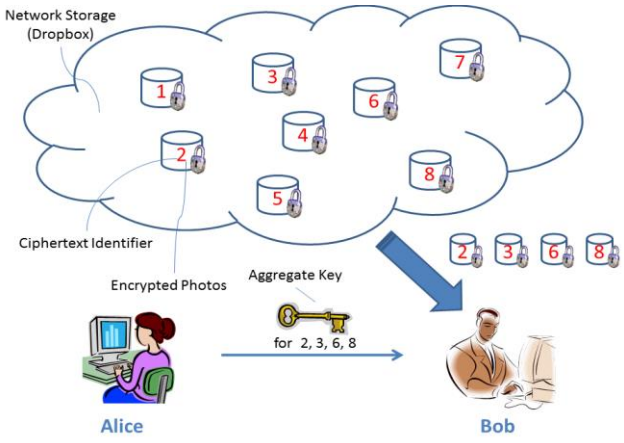

Fig 1: Network Storage (drop box)

Encrypted Aggregate Key and Searchable Encrypted key Transfer-The data owner establishes the public system parameter via Setup and generates a public/master-secret key pair via Key Gen. Messages can be encrypted via Encrypt by anyone who also decides what cipher text class is associated with the plaintext message to be encrypted. The data owner can use the master-secret to generate an aggregate decryption key for a set of cipher text classes via Extract. The generated keys can be passed to delegates securely (via secure e-mails or secure devices) finally; any user with an aggregate key can decrypt any cipher text provided that the cipher text's class is contained in the aggregate key via Decrypt.

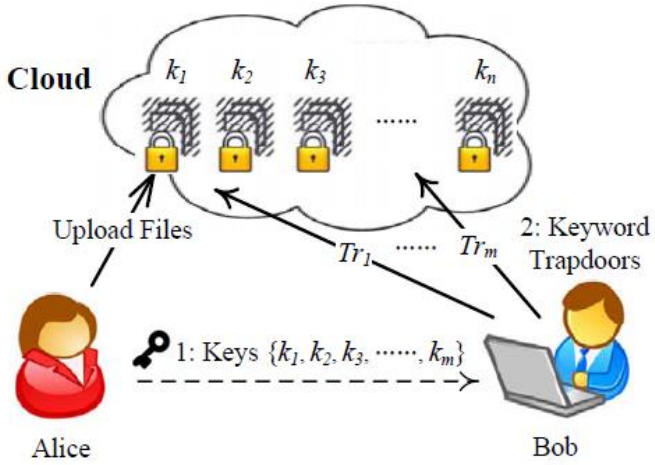

Fig 2: Encrypted Aggregate Key and Searchable Encrypted Key Transfer

Trapdoor generation-Trapdoor generation algorithm is run by the user who has the aggregate key to perform a search. It takes as input the aggregate searchable encryption key agg and a keyword w, then outputs only one trapdoor Tr.

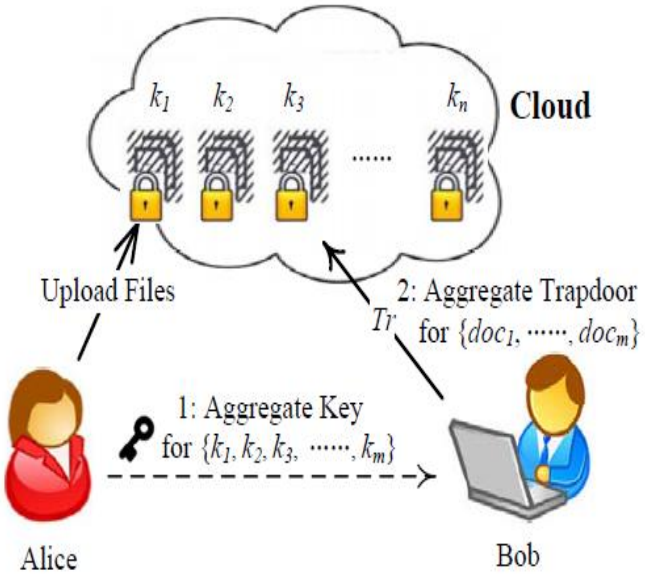

Fig 3: Trapdoor generation

File User-The generated keys can be passed to delegates securely (via secure e-mails or secure devices) finally; any user with the Trapdoor keyword generation process can decrypt any cipher text provided that the cipher text class is contained in the Encrypted aggregate key and Searchable Encrypted key via Decrypt.

We Propose A Concrete KASE Scheme As Follows:

1) **Setup(**1^{λ}**, n)**: the cloud server will use this algorithm to initialize system parameters as follows:

- Generate a bilinear map group system B=(p,G,G1, e(·,·)), where p is the order of G and $2^{\lambda} \leq p \leq 2^{\lambda+1}$.
- Set n as the maximum possible number of documents which belongs to a data owner.
- Pick a random generator g ε G and a random αεZp, and computes gi = g(α^{i}) ε G for i = {1,2,⋯⋯,n,n+2,⋯,2n}.
- Select a one-way hash function H: f0; 1g_ !G. Finally, cloud server publishes the system parameters params = (B, PubK, H), where PubK = (g; g1; _ _ _; gn; gn+2; :::; g2n) 2 G2n+1.

2) **Keygen**: data owner uses this algorithm to generate his/her key pair. It picks a random 2 Zp, and outputs:

pk = v = g; msk= γ.

3) **Encrypt**(pk, i): data owner uses this algorithm to encrypt data and generate its keyword cipher texts when uploading the i-th document. To generate the keyword ciphertexts, this algorithm takes as input the file index iε {1,⋯⋯,n}, and:

- randomly picks a t ε Zp as the searchable encryption key ki of this document.
- generates a delta Δ_i for k_i by computing:
  $c_1 = g^t$, $c_2 = (v \cdot gi)^t$.
- for a keyword w, outputs its ciphertext cw as:
  $c_w = e(g,,H(w))^t / e(g_1,g_n)^t$.

  Note that c1; c2 are public and can be stored in the cloud server.

4) **Extract**(msk, S): data owner uses this algorithm to generate an aggregate searchable encryption key. For any subset S which contains the indices of documents, this algorithm takes as input the owner's master-secret key msk and outputs the aggregate key kagg by computing:

$$kagg = \Pi_{j \epsilon S} g^{\gamma}{}_{n+1-j} .$$

To delegate the keyword search right to a user, data owner will send kagg and the set S to the user.

5) **Trapdoor**(kagg, w): the user uses this algorithm to generate the trapdoor to perform keyword search. For all documents which are relevant to the aggregate key kagg, this algorithm generates the only one trapdoor Tr for the keyword ω by computing.

IV. CONCLUSION

Considering the practical problem of privacy preserving data sharing system based on public cloud storage which requires a data owner to distribute a large number of keys to users to enable them to access his/her documents, we for the first time propose the concept of key-aggregate searchable encryption (KASE) and construct a concrete KASE scheme. Both analysis and evaluation results confirm that our work can provide an effective solution to building practical data sharing system based on public cloud storage. In a KASE scheme, the owner only needs to distribute a single key to a user when sharing lots of documents with the user and the user only needs to submit a single trapdoor when he queries over all documents shared by the same owner. However, if a user wants to query over documents shared by multiple owners, he must generate multiple trapdoors to the cloud. How to reduce the number of trapdoors under multi-owners setting is a future work. Moreover, federated clouds have attracted a lot of attention nowadays, but our KASE cannot be applied in this case directly. It is also a future work to provide the solution for KASE in the case of federated clouds.

REFERENCES

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

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

[3] 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.

[4] C. Chu, S. Chow. 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.

[5] X. Song, Dowager, A. Per rig. "Practical techniques for searches on encrypted data", IEEE Symposium on Security and Privacy, IEEE Press, pp. 44C55, 2000.

[6] R. Carmela, J. Gray, S. Tamara, 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.

[7] P. Van's. Sergei, JM. Doormen. "Computationally efficient searchable symmetric encryption", Secure Data Management, pp. 87-100, 2010.

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

[9] D. Bone, C. G, R. Ostrovsky, G. Persian. "Public Key Encryption with Keyword Search", EUROCRYPT 2004, pp. 506C522, 2004.

[10] 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.

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

[12] C. Bosch, R. Brinkman, P. Harte. "Conjunctive wildcard search over encrypted data", Secure Data Management. LNCS, pp. 114- 127, 2011.

[13] C. Dong, G. Russell, N. Duly. "Shared and searchable encrypted data for untrusted servers", Journal of Computer Security, pp. 367-397, 2011.

[14] F. Zhao, T. Nishide, 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.

[15] 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.

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

[17] 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.

