Multi owner Secure Data Storage and Sharing in Cloud Computing

P Swathi
Student, Department of Computer Science and Engineering
Baba Institute of Technology and Sciences Vizag
S Chanti
Asst.Prof, Department of Computer Science and Engineering
Baba Institute of Technology and Sciences Vizag

Abstract— In cloud, there is an imperative efficacy called Data sharing. Data sharing is additionally worried about the security, effectiveness and adaptability for the data to partake in the cloud storage. The limit of specifically sharing encoded data with various clients by means of open cloud server may extraordinary straightforwardness security worried over scatterbrained data spills in the cloud. The key testing of structuring encryption plot is lies between the proficient administrations of encryption keys. Here location the reasonable issue which is to a great extent ignored in the writing. By proposing the novel idea of Key-total accessible encryption and furthermore instantiating the idea through strengthen KASE scheme where the data proprietor need to disperse a solitary key to data client for sharing huge number of reports, and the client just needs to present a solitary trapdoor to cloud for questioning the common archives. Another open key cryptosystem is acquainted with deliver a consistent size figure content called KASE. In light of the content approach as a key the key is utilized to share the data securely. When the sharing is finished the key total contrasts from key total cryptosystem and this procedure gives effective arrangement than the current framework.

Keywords: Cloud Computing, Key-total encryption, Attribute based Encryption, Aggregate keys.

I. INTRODUCTION

Nowadays the storage in the cloud has showed up as an acceptable reaction for appropriate and on intrigue gets to tremendous proportions of data shared over the Web. Business customers are being engaged by cloud storage as a result of its few favorable circumstances, including lower cost, better deftness, and upgraded resource use. Customary customers are in like manner sharing private data, for instance, photos, and accounts, with their partners through casual association applications in perspective of the cloud. Of course, while benefitting from the comfort of sharing data through cloud storage, customers are in like manner gnawing by bit worried over spontaneous data reveal by the cloud. Such data revealing, will be performed by a noxious enemy or a malevolent cloud executive, can as often as possible direct to the genuine encroachment of private data or mystery data as for business. To discuss customer's uneasiness over possible data reveal in cloud storage, a general procedure is for the data proprietor to encode every one of the data previously moving them into the cloud, with the end goal that in a matter of seconds the scrambled data may get back and unscrambled by individuals who contains the interpreting keys. Such cloud storage is as often as possible called the cryptographic cloud storage [6].Though; the encryption of data

fabricates it asking for customers to look and after that best recoup only the data including the given catchphrases. A run of the mill course of action is to use an accessible encryption (SE) plan in which the data proprietor is required to scramble potential catchphrases and exchange them to the cloud together with encoded data, to such an extent that, for recouping data organizing a watchword, the customer will send the planning catchphrase to the cloud to react for the chase over the scrambled data. Regardless of the way that consolidating an accessible encryption Plan with cryptographic cloud storage can satisfy the key security needs of a cloud storage, executing such a system for generous scale application relating colossal number of customers and tremendous number of reports. May even now put off by functional issues relating the plain much dealt with organization of encryption keys, which, to the best of our understanding. Basically, the requirement for explicitly sharing encoded data with assorted customers as a general rule demands unmistakable encryption keys to be used for various records. On the other hand, this incorporates the amount of keys that ought to be spread to customers, both for them to look for the scrambled records and to unscramble the reports, will be in regard to the amount of such archives. Such a generous number of keys must not simply be spread to customers by methods for secure channels, moreover, be securely secured and dealt with by the customers in their contraptions. The undeniable essential for secure correspondence, storage, and computational inconvenience may realize structure deficiency.

II. RELATED WORK

In front of we start our all-encompassing KAC framework, this section principal audit various classification of available arrangement and in addition elucidate their relationship to our activity. Predominantly this framework produces key for symmetric-key cryptosystems, still anyway the info deduction may require measured science like use out in the open key cryptosystems, which are commonly more costly than "symmetric-key activity, for example, pseudorandom reason. In [4], accessible an adaptable makes utilization of cloud storage got ready for client require as it is crease get to data adjacent anyway to in participation at far off surface. It is critical to check the data put on the cloud. Therefore it is critical toward approve an open confirm for the benefit of earnestness re-appropriated data all over outsider evaluator (TPA). TPA be in addition costly for cloud service provider. Which check the precision of the redistributed data, aside from constraint be gauge clearness. In [5] displayed a security go between (SEM) move that allow a client toward protect the security. Client will transfer every data SEM in this manner won't capable toward distinguish the data still anyway it's leave-taking to produce affirmation occurring data. Since the client be sign by SEM must not know the uniqueness of data proprietor, trouble will limit the expectation situated on states of data protection and uniqueness security. In [6] displayed the multi accumulation input organization achieve the progressive appropriate to utilize oversee through apply a joined info diagram also dealing with the gathering key for unique client among various ideal to utilize foundation. Focal key association game plan use pecking order association to lessen data preparing, message and in addition storage overhead. Which keep up thing associated toward key alongside too refresh. It achieves a fused information outline for every client. A further methodology for distribution scrambled data is Attribute-Based Encryption (ABE) [7], conceivable to encode the data among property. A client contribution and additionally the characteristic match that have the capacity to decode the demanding figure content. While exhibiting k characteristic be cover between figure messages and in addition

private key the decoding be affirmed. A correspondence, estimation, with storage expense be limit like focal methodology. Chu et al. [8] think about how to diminish the measure of scattered data encryption key. While in transit to partition various papers among divergent encryption key among comparative client, data proprietor resolve require to assign each one such key to him/her in a regular advance toward which is normally not down to earth. Go for this test, key total Encryption (KAE) framework for data assignment is anticipated to create total key for the client to decode the whole papers. Popa [9] right off the bat presents the idea of multi-key accessible encryption (MKSE) and advances the primary achievable plan in 2013. MKSE enables a client to give a solitary watchword trapdoor to the server, yet at the same time enables the server to look for that trapdoor's catchphrase in archives scrambled with various keys. This may sound fundamentally the same as the point of KASE, however these are in certainty two altogether unique idea. The point of KASE is to hand over the watchword explore precise to some client through appropriate the aggregate contribution to him/her in a bunch data conveyance conspire, while the point of MKSE is to ensure the cloud server have the capacity to execute catchphrase research through single trapdoor more than unique papers inferable from a client.

III. DATA SHARING

KAC in meant for the data sharing. The data owner can share the data in desired amount with confidentiality. KCA is easy and secure way to transfer the delegation authority. The aim of KCA is illustrated in Figure 3.

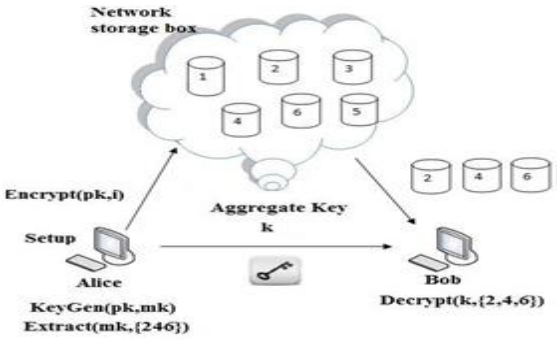

Fig Use of KAC for data sharing.

For sharing selected data on the server Alice first performs the Setup. Later the public/master key pair (pk, mk) is generated by executing the KeyGen. The msk master key is kept secret and the public key pk and param are made public. Anyone can encrypt the data m and this data is uploaded on server. With the decrypting authority the other users can access those data. If Alice is wants to share a set S of her data with a friend Bob then she can perform the aggregate key KS for Bob by executing Extract (mk, S). As kS is a constant size key and the key can be shared through secure e-mail. When the aggregate key has got Bob can download the data and access it.

IV. SECURITY OF CLOUD DATA STORAGE

Many cloud service providers provide storage as a form of service. They take the data from the users and store them on large data centers, hence providing users a means of storage. Although these cloud service providers say that the

data stored in the cloud is utmost safe but there have been cases when the data stored in these clouds have been modified or lost may be due to some security breach or some human error. Various cloud service providers adopt different technologies to safeguard the data stored in their cloud. But the question is: Whether the data stored in these clouds is secure enough against any sort of security breach? The virtualized nature of cloud storage makes the traditional mechanisms unsuitable for handling the security issues. These service providers use different encryption techniques like public key encryption and private key encryption to secure the data resting in the cloud. Another major issue that is mostly neglected is of Data-Remanence. It refers to the data left out in case of data removal. It causes minimal security threats in private cloud computing offerings, however severe security issues may emerge out in case of public cloud offerings as a result of data remanence. Various cases of cloud security breach came into light in the last few years. Cloud based email marketing services company, Epsilon suffered the data breach, due to which a large section of its customers including JP Morgan Chase, Citibank, Barclays Bank, hotel chains such as Marriott and Hilton, and big retailers such as Best Buy and Walgreens were affected heavily and huge chunk of customer data was exposed to the hackers which includes customer email ids and bank account details. Another similar incident happened with Amazon causing the disruption of its EC2 service. The damage caused had proved to be quite costly for both the users and the system administrators. The above mentioned events depict the vulnerability of the cloud services. Another important aspect is that the known and popular domains have been used to launch malicious software or hack into the companies' secured database. It is proved that Amazon is prone to side-channel attacks, and a malicious virtual machine, occupying the same server as the target, can easily gain access to confidential data [10]. The question is: whether any such security policy should be in place for these trusted users as well? An incident relating to the data loss occurred last year with the online storage service provider "Media max" also known as "The Linkup" when due to system administration error, active customer data was deleted, leading to the data loss. SLA's with the Cloud Service providers should contain all the points that may cause data loss either due to some human or system generated error. Virtualization in general increases the security of a cloud environment. With virtualization, a single machine can be divided into many virtual machines, thus providing better data isolation and safety against denial of service attacks [10]. The VMs provide a security test-bed for execution of untested code from un-trusted users.

V. DATA PRIVACY IN CLOUD COMPUTING ENVIRONMENT

Considering data privacy in cloud computing environment, a traditional way to ensure data privacy is to rely on the server to enforce the access control after authentication, which means any unexpected privilege increase will expose all data. In a shared-lease cloud computing environment, things become even bad. Data from different users can be hosted on separate virtual machines (VMs) but reside on a single physical machine. Data in a target Virtual Machine could be stolen by instantiating another Virtual Machine co occupant with the target one.

VI. SYSTEM ARCHITECTURE

A key-aggregate encryption scheme consists of five polynomial-time algorithms [1] as shown in Figure. The data owner establishes the public system parameter via Setup and generates a public/master-secret3 key pair via KeyGen.

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 receivers 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 Decrypt4.

A. Setup (1;n):

Executed by the data owner to setup an account on an untrusted server. On input a security level parameter 1 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.

B. KeyGen():

Executed by the data owner and randomly generates a master-secret key (msk).

C. Encrypt(pk; i;m):

Executed by data owner to encrypt data. On input msk, an index i denoting the cipher text class, and a message m, it outputs a cipher text C..

D. Extract(msk; S):

Executed by the data owner for delegating the decrypting power for a certain set of cipher text classes to the receiver. On input the master secret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by KS.

E. Decrypt(KS; S; i; C):

Executed by a receiver who received an aggregate key KS generated by Extract. On input KS, the set S, an index i denoting the cipher text class the cipher text C belongs to, and C, it outputs the decrypted result m if i in S.

VII. PROPOSED WORK

In this paper we propose a technique to make data sharing secure and leak resilient. The purpose of this article is to provide a way for secure data sharing on cloud using key aggregate encryption and Intrusion Detection (KAEID). In KAEID Decryption key is made more and more powerful so that it can decrypt multiple cipher texts. At the same time Intrusion detection system (IDS) monitors data exchange between two hosts and ensures if these are trusted hosts [2]. Specifically, the problem statement is "To generate a constant size aggregate decryption key by data owner which can decrypt multiple cipher text. The decryption key is aggregate key which encompasses the power of all secret keys. This data sharing system also supports intrusion detection to find out the suspicious activities of hosts. If hosts involved in communication are trusted hosts data sharing will take place else rejected." In KAEID

user encrypts message under public key cryptosystem. Messages are encrypted by one who decides public key as well as cipher text category. Cipher text is categorized under different “classes”. Plain messages which are subset of cipher text class possess few common features. Here all the hosts set up an account on the cloud server. Hosts can login to the cloud server; they can perform their task and logout of the server. The data owner generates public key/ master key pair. Public key is used for encryption while master key is kept secret. Master key is used for aggregating all the decryption keys. The aggregate key is extracted out of master key and corresponding cipher text class identifier. This aggregate key is delegated to data recipient. The data recipient compares the set of cipher text classes and decrypts the message. Hence, it also prevents the downloading of unwanted data. Each host in the data sharing system works as IDS. An IDS collects IP address of all hosts in its sub network, and keep eyes on suspicious activities in the network. If any suspicious host is found it is blacklisted. Data sharing with suspicious host is rejected. As shown in Fig-1. Two hosts data owner and data recipient are accessing the cloud network. Data owner encrypts the data and uploads data on cloud server. Aggregate key is delegated to Data recipient for decryption of requested messages. Hosts involved in communication are also working as IDS. IDS collects and lists IP addresses of corresponding sub network. Monitors the suspicious activities and reject data sharing with the hosts found blacklisted.

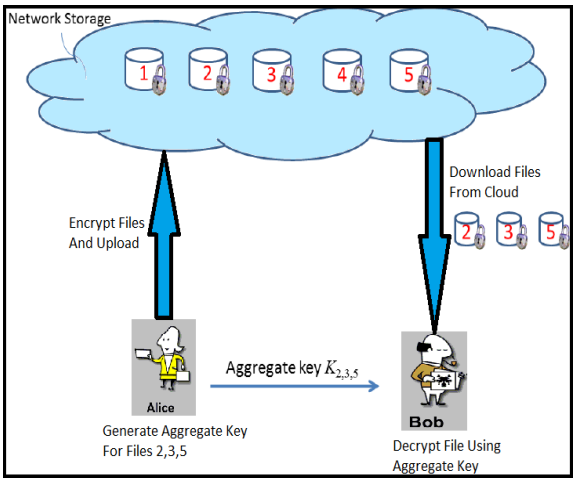

Fig. Proposed Architecture diagram

VIII. CONCLUSION

As we all know data security is a major concern for cloud users. This paper comes with a technique, which helps to achieve a secured and leak proof system. Here modern cryptographic algorithms and intrusion detection algorithms are used in order to achieve a secured way of data sharing. In this system data owner uses distinct encryption keys and encrypts messages before uploading it on cloud and sends a single decryption key to other host. This single decryption key decrypts multiple cipher text at a time thereby saving the time as well as storage space. Unwanted data will not be downloaded at data recipient’s side. Intrusion detection systems monitor the security breakdown in the network. Data sharing is stopped if any un-trusted party comes in the network. Obtaining an ideal system without data any leakage is practically is not possible, but this research work helps to solve certain problems very

efficiently. It saves the storage space; it also saves time spent in key exchange. Key sizes remains constant and compact.

REFERENCES

[1] Baojiang Cui, Zheli Liu and Lingyu Wang : Key-Aggregate Searchable Encryption for Group Data Sharing via Cloud Storage, IEEE Transactions On Computers, Vol. 6, No. 1, January 2014

[2] 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.

[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,W. Tzeng, et al. "KeyAggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477.

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

[6] 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.

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

[8] S. Kamara, 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. Boneh, C. G, R. Ostrovsky, G. Persiano. "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: PairingBased Cryptography C Pairing 2007, LNCS, pp. 2- 22, 2007.