Identity Based Encryption and Data Self Destruction in Cloud Computing

Madhavi S. Langute
M.E (Computer) Department of Computer Engineering,
Jayawantrao Sawant College of Engineering,
Savitribai Phule Pune University,
Pune, Maharashtra, India -411007.
madhavilangute14@gmail.com

Prof. H.A.Hingoliwala
Head of Department of computer Engineering
Jayawantrao Sawant College of Engineering,
Savitribai Phule Pune University,
Pune, Maharashtra, India -411007
Ali_hyderi@yahoo.com

Abstract—When it comes to storing data, cloud storage is rapidly turning into the procedure for choice. Cloud storage is quickly becoming the strategy for decision. Putting away files remotely instead of by locally boasts an array of preferences for both home and professional clients. Cloud storage means “the storage of data online in the cloud”, however, the cloud storage is not completely trusted. Whether the data put away on cloud are in place or not turns into a significant concern of the clients also access control becomes a difficult job, particularly when we share data on cloud servers. To tackle this issue outsourcing Revocable IBE scheme for efficient key generation and key updating process is introduce. Also to improve the efficiency of cloud server in terms of storage new secure data self-destructing system in cloud computing is used. In this system, each cipher text (encrypted file) is labeled with a time interval. If the attributes associated with the cipher text satisfy the key’s access structure and both the time instant is in the allowed time interval then the cipher text is decrypted. After a user-specified end time the data at cloud server will be securely self-destructed

Keywords- *Cloud Computing, Self-Destruction, Identity Based Encryption (IBE), Revocation, Outsourcing.*

*****

I. INTRODUCTION

Cloud computing alludes to the usage of computing resources, those being programming or equipment that reside on a re-mote machine and are conveyed to the end client as an service over a system, with the most common example being the web. Cloud storage is gaining popularity and importance very rapidly. To share data securely the Identity-based encryption technique or use of combination of Identity’s is used [2]. The identity-based encryption (IBE) is a significant primitive of ID-based cryptography. As such it is a kind of public-key encryption in which the public key of a user is several unique information about the identity of the user (e.g. a user’s email address). This means that a sender who has access to the public parameters of the system can encrypt a message using e.g. the text-value of the receiver’s email address as a key. The receiver obtains its decryption key from a central authority, which needs to be trusted as it generates secret keys for every user. It lets any party to produce a public key from a recognized identity value. The corresponding private keys generated by a trusted third party, called the Private Key Generator (PKG). To function, the PKG primary publishes a master public key, and keeps the equivalent master private key. Any party can calculate a public key equivalent to the identity ID by unite the master public key with the identity value given the master public key. To get a matching private key, the party authorized to use the identity ID associates the PKG, which uses the master private key to make the private key for identity ID. When a user leaves the group or behave badly, this user must be revoked from the group for security reasons. As a result, this revoked user should no longer be able to access and modify shared data. For this revocable Identity Based Encryption technique is stated by A. Boldyreva, V. Goyal, and V. Kumar [3], but it as a drawback of computation overhead at single point i.e. admin or important person from the organization, to overcome the drawback an outsourcing computation into IBE revocation is introduced. System propose a scheme to offload all the key generation connected processes during key-issuing and key-update, leaving only a constant number of simple operations for PKG and entitled users to perform locally. Also a new collusion-resistant key issuing technique is proposed which utilizes a hybrid private key for each user, in which an AND gate is involved in key generation process, namely the identity component and the time component.

Also to improve the cloud storage space a secure data self-destructing system in cloud computing is proposed. In this system, while private key is connected with a time instant each ciphertext is labeled with a time inter-val. If both the time instant is in the allowed time interval and the identities associated with the ciphertext satisfy the key’s access structure then the ciphertext can be decrypted. In general, the owner has the right to specify that certain sensitive information is only valid for a limited period of time i.e. self-destructed after completion of time interval set by the owner, or should not be unconfined before an exacting time.

II. RELATED WORK

In this paper [4] the author suggests a fully functional identity-based encryption scheme (IBE). Assuming a variant of the computational Diffie Hellman problem the system has selected ciphertext security in the random oracle model. The system is based on bilinear maps between groups. The Weil pairing on elliptic curves is an example of such a map.

In this paper [3] the Identity-based encryption is proposed, as IBE eliminates the need for a Public Key Infrastructure (PKI), it is an exciting alternative to public-key encryption. Any setting, PKI- or identity-based, must give a means to revoke users from the system. Proficient revocation is a well-studied difficulty in the traditional PKI setting. However in the setting of IBE, there has been little work on studying the revocation mechanisms. When encrypting, the most practical solution need the senders to also use time periods and by contacting the trusted authority all the receivers to update their private keys regularly. But this solution does not scale well the work on key updates becomes a bottleneck, as the number of user's increases. We propose an IBE scheme that appreciably progresses key-update effectiveness on the side of the trusted party, while staying proficient for the users. Our system constructs on the ideas of the Fuzzy IBE primitive and binary tree data structure, and is provably secure.

In this paper [5] the author studied that the kind of Identity-Based Encryption (IBE) plan that call as Fuzzy Personality Based Encryption. In Fuzzy IBE a way of life as set of illustrative qualities are used. A Fluffy IBE plan takes into account a private key for a personality, !, to unscramble a cipher text scrambled with a personality, !0, if and just if the characters ! What's more, 0 are near one another as measured by the "set cover" separation metric. A Fuzzy IBE plan can be connected to empower encryption utilizing biometric inputs as personalities; the blunder resistance property of a Fuzzy IBE plan is correctly what takes into ac-count the utilization of biometric personalities, which inalienably will have some commotion every time they are inspected. Moreover, we demonstrate that Fuzzy-IBE can be utilized for a sort of application that we term "quality based encryption".

In this paper [6] the author addresses the issue of utilizing untrusted (possibly malevolent) cryptographic partners. A formal security definition to safely outsourcing calculations from a computationally constrained gadget to an untrusted partner is proposed. In this model, the will disposed environment composes the product for the partner, however then does not have direct correspondence with it once the gadget begins depending on it. Not with standing security, it likewise gives a structure to measuring the effectiveness also; check ability of an outsourcing usage. It also introduce two pragmatic outsource secure plans. In particular, it demonstrate to safely outsource measured exponentiation, which presents the computational bottleneck in most open key cryptography on computationally restricted gadgets. Without outsourcing, a gadget would require O(n) particular augmentations to complete particular exponentiation form bit types. The heap lessens to O (log2 n) for any exponentiation-based plan where the genuine gadget may utilize two untrusted exponentiation programs; they highlight the Cramer-Shoup cryptosystem and Schnor marks as samples. With a casual thought of security, we accomplish the same burden diminishment for another CCA2-secure encryption plan utilizing stand out untrusted Cramer-Shoup encryption program.

In this paper [7] the author demonstrated that the Trait based encryption (ABE) is a promising cryptographic apparatus for ne-grained access control. Be that as it may, the computational taken at online encryption ordinarily develops with them any-sided quality of access arrangement in existing ABE plans, which turns into a bottleneck constraining its application. In this paper, a novel worldview of outsourcing encryption of ABE to cloud administration supplier to calm neighborhood calculation trouble is proposed. It utilizes an enhanced development with MapReduce cloud which is secure under the suspicion that the expert hub and in addition at minimum one of the slave hubs is straightforward. In the wake of outsourcing, the computational taken a toll at client side amid encryption is decreased to inexact four exponentiations, which is steady. Another point of preference of the proposed development is that the client can assign encryption for any arrangement.

In this paper [8] the author proposed ABE scheme, the Attribute based encryption (ABE) is a promising cryptographic primitive, which has been widely applied to design fine-grained access control system recently. Though, ABE is being criticized for its high scheme over-head as the computational cost grows with the complexity of the access formula. Because they have constrained computing resources this disadvantage becomes more serious for mobile de-vices. Aiming at attempting the above confront, it presents a general and proficient solution to apply attribute-based access control system by establishes secure outsourcing methods into ABE. More exactly, two cloud service providers (CSPs), namely key generation-cloud service provider (KG-CSP) and decryption-cloud service provider (D-CSP) are establish to perform the outsourced key-issuing and decryption on behalf of attribute authority and users respectively.

In this paper [9] the author proposed the pro-totype of forward security for Cryptographic computations was introduced. Secret keys are updated at usual periods of time; contact of the secret key matching to a given time period does not allow an challenger to "break" the scheme for any previous time period in a forward-secure scheme. A number of constructions of forward-secure digital signature schemes, key-exchange protocols, and symmetric-key schemes are known. The main building attains security beside chosen-plaintext attacks under the decisional bilinear Diffie-Hellman supposition in the standard model. This system is practical, and with the total number of time periods all parameters grow at most logarithmically.

III. PROPOSED SYSTEM

The following Fig.1 shows the proposed system architecture.

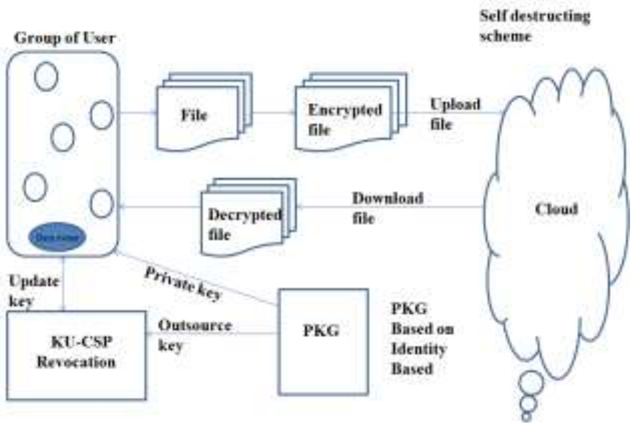

Fig 3.1: Proposed System

A. System Overview

The user registers himself at server and then login with valid username and password in to system. After login, user request for keys to KU-CSP [1]. The user / owner encrypt the files using the keys and uploaded these files at cloud server for specific time interval and become free from the burden. When any user leave the group ,the list of remaining user is send to KU-CSP, where the KU-CSP generate the new key or update the keys to maintain the security of the system and send the new keys to the key requested user. At cloud server if the specified time for the file is end then the file is destructed / delete from the server and it is no longer available for users. This increases the storage space at cloud server.

In previous work the system stores the data at cloud server and the user itself has delete the data stored at cloud if he no longer needed the data, it increases overhead of user and also uses more space at cloud server, to overcome the drawback of previous system, the system pro-poses data self-distractive scheme, In this user upload the data at cloud server for specific time duration (for example, 2/2/2016-2/2/2017,).at cloud server data is valid for only one year i.e. from start date to end date specified by user after completion of time period data is self-destructed from the cloud and it frees the space at cloud server.

B. Self-Destructing Scheme

A Self-Destructing Scheme called key-policy identity based encryption with time specified attributes scheme, which is based on inspection that, in sensible cloud application situation, every data item can be linked with a set of attributes and each attribute is linked with a specification of time interval, indicating that the encrypted data item can only be decrypted between on a specified date and it will not be recoverable that day. In which every user's key is associated with an access tree and each leaf node is associated with a time instant the data owner encrypts his/her data to share with users in the system. As the logical expressionof the access tree can signify any desired data set with any time interval, it can attain fine-grained access control. If the time instant is not in the specified time interval, the ciphertext cannot be decrypted, i.e., this ciphertext will be self-destructed and no one can decrypt it because of the expiration of the secure key. Therefore, secure data self-destruction with fine-grained access control is attained. In order to decrypt the ciphertext effectively, the valid attributes should gratify the access tree where the time instant of each leaf in the users key should belong to the in the matching attribute in the ciphertext.

C. Algorithm

1) Setup ( ): PKG run the setup algorithm. It chooses a random generator g 2R G as well as a random integer x 2R Zq and sets g1 = gx. Then, A random Element PKG picked by g2 2R G and two hash functions H1; H2: f0; 1g! GT. Finally, output the public key PK= (g; g1; g2; H1; H2) and the master key MK = x.

2) KeyGen (MK, ID, RL, TL, and PK): PKG firstly checks whether there quest identity ID exists in RL, for each user's private key request on identity ID, if so the key generation algorithm is terminated. Next, PKG randomly selects X1 2R Zq and sets x2 = x x1. It randomly chooses, and computes. Then, PKG reads the current time period Ti from TL. Accordingly, it randomly selects Ti 2R Zq and computes, where and finally, output SKID = (IK [ID]; TK [ID] Ti) and OKId = x2.

3) Encrypt (M, ID, Ti+, and PK): Assume a user needs to encrypt a message M under identity ID and time Ti period. He/She chooses a random value s 2R Zq and computes, C0 = Me (g1; g2) s; C1 = gs; EID = (H1 (ID)) s and Finally, publish the ciphertext as CT = (C0; C1; EID; ETi).

4) Decrypt (CT; SKID; PK): Assume that the ciphertext CT is encrypted under ID and Ti, and the user has a private key SKID = (IK[ID]; TK[ID]Ti), where IK[ID] = (d0; d1) and TK[ID]Ti = (dTi0; dTi1).

5) Revoke(RL; TL; {IDi1; Idi2; ::::Idik}) : If users with identities in the set {IDi1; Idi2; ::::Idik} are to be revoked at time period Ti, PKG updates the revocation list as RL0 = RL{IDi1; Idi2; ::::Idik} as well as the time list. Through connecting the recently created time period Ti+1 onto

original list TL. Finally send a copy for the updated revocation list as well as the new time period Ti+1 to KU-CSP.

6) Key Update (RL; ID; Ti+1; OKID): Upon receiving a key update request on ID , KU-CSP firstly checks whether ID exists in the revocation list RL , if so KU-CSP returns and key-update is terminated. Other-wise, KU-CSP gets the corresponding entry (ID; OKID = x2) in the user list UL. Then, it randomly selects Ti+1 2R Zq.

7) Data self-destruction after end: Previously the current time instant tx lags behind after the threshold value (expiration time) of the valid time interval tR; x, the user cannot obtain the true private key SK. Therefore, the ciphertext CT is not capable to be decrypted in polynomial time, ease the self-destruction of the shared data after end.

D. Complexity Analysis

Time Complexity of ECC is O (n).

E. Mathematical Model

System S is represented as S= {U, CS, KU-CSP}
1) User US = {R, L, Q, E, V}
Where,
R= Registration Process
L= Login Process
Q= Key Request Process
E= File Encryption Process
V= Revocation Process

2) KU-CSP={PK,SK}
Key Generation PK={pk1, pk2, pk3 ...pkn}
Where PK is the set of generate public keys.
SK= {sk1, sk2, sk3 ...skn}
Where SK is the set of generate private keys related to public key.

3) Cloud Server CS ={U, D}
Where,
U = Upload file
D= {T, F}
Where,
D = Self-Destructive Process
T=Time Interval
F=Number of files

F. Dataset

The System uses multiple files with various sizes from 1 KB to 100 MB as dataset.

G. Experimental Setup

The system used Netbeans (version 8.0) tool for development and Java framework (version jdk 1.8) on Windows platform as a front end. Any standard machine is capable of running the application. The system doesn't need any specific hardware to run.

IV. EXPECTED RESULT

A. Results

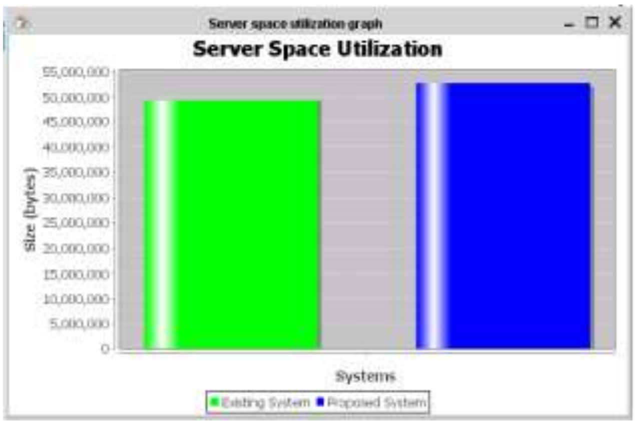

Fig 4.1 Server Space Utilization Graph

Figure 4.1 shows the server space utilization graph comparing the existing system and proposed system in which we can see that the proposed system use server space more efficiently.

B. Screen shots

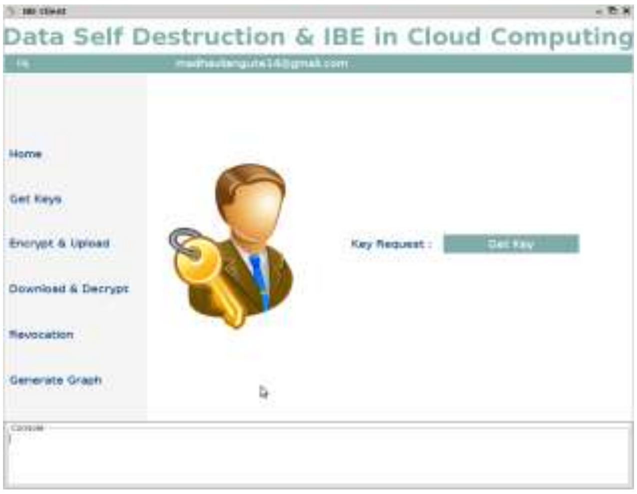

Fig. Key Request

Above figure shows the Key Generation request form where user / data owner will requests a key which is used for encryption and decryption of the file. On this request user will get a public as well as a private key.

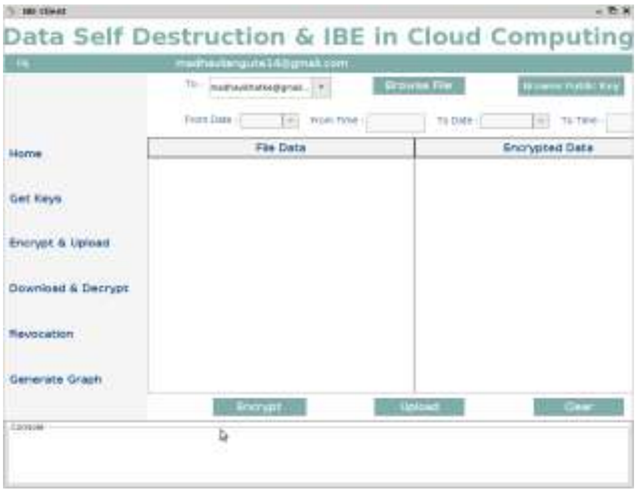

Fig. File Selection

Above for shows the file selection form where user / data owner will choose a file to be uploaded as well as for encryption. On the left window the original file content will be displayed and in right window the encrypted text will be shown.

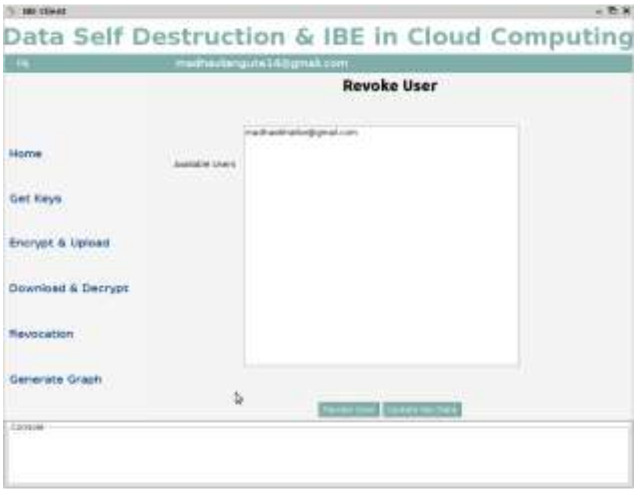

Fig. User Revocation

Above figure show the user revocation form. When a user leaves the organization all the keys associated with the user has to be changed/ deleted so that the user outside of the system cannot access the data.

V. ACKNOWLEDGMENT

The authors would like to thank the researchers as well as publishers for making their resources available and teachers for their guidance. We are also thankful to the reviewer for their valuable suggestions.

VI. CONCLUSION

Many recent challenges have appeared with the fast growth of adaptable cloud services. One of the most significant problems is how to securely delete the outsourced data stored in the cloud severs. In order to solve the problems by implementing flexible fine-grained access control during the authorization period and time-controllable self-destruction after expiration to the shared and outsourced data in cloud computing, this paper proposed a data self-destructing system which is able to attain the time specified ciphertext. Also a revocable outsourcing computation into IBE is introduced to overcome issue of identity revocation. There is No secure channel or user authentication is required during key-update between user and KU-CSP, also with the help of KU-CSP, the sys-tem has features such as; steady effectiveness for both computations at PKG and private key size at user.

REFERENCES

[1] Jin Li, Jingwei Li, Xiaofeng Chen, Chunfu Jia, and Wenjing Lou, "Identity-Based Encryption with Outsourced Revo-cation in Cloud Computing", in IEEE transactions on computers, vol. 64, no. 2, february 2015.

[2] W. Aiello, S. Lodha, and R. Ostrovsky, "Fast digital identity revocation," In Advances in Cryptology (CRYPTO98). New York, NY, USA:Springer, 1998, pp. 137-152.

[3] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based en-cryption with efficient revocation," in Proc. 15thACMConf. Comput. Commun.Security (CCS08), 2008, pp. 417-426.

[4] D. Boneh and M. Franklin, "Identity-based encryp-tion from the Weilpairing," in Advances in Cryptology (CRYPTO '01), J. Kilian, Ed.Berlin, Germany: Springer, 2001, vol. 2139, pp. 213-229.

[5] A. Sahai and B. Waters, "Fuzzy identity-based encryption,"in Advances in Cryptology (EUROCRYPT'05), R. Cramer, Ed. Berlin, Germany: Springer, 2005, vol. 3494, pp. 557-557.

[6] J. Li, C. Jia, J. Li, and X. Chen, "Outsourcing encryp-tion of attribute based encryption with mapreduce," in Information and Communications Security. Berlin, Heidel-berg:Springer, 2012, vol. 7618, pp. 191-201.

[7] B. Zhang, J. Wang, K. Ren, and C. Wang, "Privacy-assured Trans. Emerging Topics Comput., vol. 1, no. 1, p. 166-177, Jul. Dec. 2013 outsourcing of image reconstruction service in cloud," IEEE.

[8] J. Li, X. Chen, J. Li, C. Jia, J. Ma, and W. Lou, "Fine-grained access control system based on outsourced attribute-based encryption," in Proc. 18th Eur. Symp. Res. Comput. Secu-rity (ESORICS), 2013,pp. 592-609.

[9] R. Canetti, S. Halevi, and J. Katz, "A forward-secure public-key Encryption scheme," in Advances in Cryptology (EU-ROCRYPT'03), E. Biham, Ed. Berlin, Germany: Springer, 2003, vol. 2656,pp. 646-646.

[10] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," Nat. Inst. Stand. Technol., Tech. Rep. SP 800-145, 2011.

[11] C. Wang, K. Ren, and J. Wang, "Secure and practical outsourcing of linear programming in cloud computing," in Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM), 2011, pp. 820–828.

[12] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts," in Proc. 20th USENIX Conf. Security (SEC'11), 2011, pp. 34–34.