Defended Cloud Data Storage and Sharing in Using Revocable-Storage Identity-Based Encryption

[1]Suresh Patnaik, [2]Dakineni Durga Prasad
[1,2]Dept. of CSE, Baba Institute of Technology and Sciences, Visakhapatnam, AP, India

Abstract
Cloud computing would be one of technologies which is going to play a vital role in the next generation of computer engineering field. The increased scalability and flexibility provided by the cloud computing has reduced the costs to a greater extent and therefore the technology has gained wide acceptance. The facility of Data outsourcing in the clouds enables the owner of the data to upload the data and other users can access the same. But, the data stored should be secure in the cloud servers. The data owner has lot of concern about security aspects present with the cloud computing. The data owners hesitate to adopt cloud computing services because of privacy protection issues of data and security of data. It is necessary to place cryptographically enhanced access control on the shared data. Identity-based encryption is a promising crypto graphical primitive to build a practical data sharing system. Revocable-storage identity-based encryption (RSIBE), which can provide the forward/backward security of cipher-text by introducing the functionalities of user revocation and cipher-text update simultaneously. Furthermore, we present a concrete construction of RS-IBE, and prove its security in the defined security model. The performance comparisons indicate that the proposed RS-IBE scheme has advantages in terms of functionality and efficiency, and thus is feasible for a practical and cost-effective data-sharing system.

Keywords
Identity based Encryption, Revocation, Outsourcing, Cloud Computing.

I. Introduction

Cloud Storage signifies "the limit of data online in the cloud," where the data is secured in and available from different spread and related resources that deal a cloud. In any case, the conveyed stockpiling is not completely trusted. Whether the data set up away on cloud are or not changes into a gigantic stress of the clients. So to secure data and client Identity ; Identity Based Encryption (IBE) is a captivating decision, which is proposed to streamline key association in an approval, in light of Public Key Infrastructure (PKI) by utilizing human sensible Identities (e.g., uncommon name, email address, IP address, and whatnot) as open keys. In this way, sender utilizing IBE does not have to look upward open key and affirmation, however especially scrambles message with recipient's Identities. As necessities be, beneficiary getting the private key related with the taking a gander at Identity from Private Key Generator (PKG) can unscramble such figure content. In, Boneh and Franklin endorsed that clients overhaul their private keys unpredictably and senders utilize the beneficiaries'. Characters connected with current time. In any case, this framework would understand an overhead load at PKG. In another word, every one of the clients paying little respect to whether their keys have been denied or not, need to contact with PKG spasmodically to show their Identities and overhaul new private keys. It requires that PKG is on the web and the shielded channel must be kept up for all exchanges, which will end up being a bottleneck for IBE structure as the measure of clients makes of systems. In this paper, we bring outsourcing computation into IBE repudiation, and formalize the security criticalness of outsourced revocable IBE oddly to the best of our understanding. To overcome the issue, Identity Based Encryption replaces this technique. In which the user's id (name, email address, ip address, port number, etc.) is used to generate the keys which are used to encrypt the data. This does not provide security to data shared in cloud because the data is stored for a longer period by then the data is accessible to the third party very easily. To avoid this Identity Based Encryption With Efficient Revocation was introduced. In this approach the data provider can provide the life time of the key provided to the user. Toward the finish of the life time the client can repudiate the key with the assistance of focal expert called Private Key Generator (PKG). After this Revocable Storage Identity Based Encryption is proposed, this gives both forward and in reverse security which is truant in past system. This system enables the information supplier to determine the life time of the information shared and in addition the private key gave to the information client. When this time lapses the private key generator (pkg) is in charge of disavowing the figure and private key of every client. This component of giving security in both the closures is called as forward and in reverse security.

II. Related Work

"Decentralizing Attribute-Based Encryption" Allison Lewko, Brent Waters. We propose a Multi-Authority Attribute-Based Encryption (ABE) system. In our system, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters. A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that react their attributes. A user can encrypt data in terms of any Boolean formula over attributes issued from any chosen set of authorities. Finally, our system does not require any central authority. In constructing our system, our largest technical hurdle is to make it collusion resistant. Prior Attribute-Based Encryption systems achieved collusion resistance when the ABE system authority \tied" together different components (representing different attributes) of a user's private key by randomizing the key. However, in our system each component will come from a potentially different authority, where we assume no coordination between such authorities. We create new techniques to tie key components together and prevent collusion attacks between users with different global identifiers. "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization". Brent Waters. We present a new methodology for realizing Ciphertext-Policy Attribute Encryption (CP-ABE) under concrete and non-interactive cryptographic assumptions in the standard model. Our solutions allow any encryptor to specify access control in terms of any access formula over the attributes in the system. In our most efficient system, ciphertext size, encryption, and decryption time scales linearly with the complexity of the access formula. The only previous work to achieve these parameters was limited to a proof in the generic group model. We present three constructions within our framework. Our rest system is proven selectively

secure under an assumption that we call the decisional Parallel Bilinear Diffie-Hellman Exponent (PBDHE) assumption which can be viewed as a generalization of the BDHE assumption. Our next two constructions provide performance trade of to achieve provable security respectively under the (weaker) decisional Bilinear-Diffie-Hellman Exponent and decisional Bilinear Diffie-Hellman assumptions. "Attribute Based Data Sharing with Attribute Revocation" Shucheng Yu, Cong Wang, Kui Ren. Cipher text-Policy Attribute Based Encryption (CP-ABE) is a promising cryptographic primitive for fine-grained access control of shared data. In CP-ABE, each user is associated with a set of attributes and data are encrypted with access structures on attributes. A user is able to decrypt a ciphertext if and only if his attributes satisfy the ciphertext access structure. Beside this basic property, practical applications usually have other requirements. In this paper we focus on an important issue of attribute revocation which is cumbersome for CP-ABE schemes. In particular, we re- solve this challenging issue by considering more practical scenarios in which semi-trustable on-line proxy servers are available. As compared to existing schemes, our proposed solution enables the authority to revoke user attributes with minimal effort. We achieve this by uniquely integrating the technique of proxy re-encryption with CP-ABE, and enable the authority to delegate most of laborious tasks to proxy servers. Formal analysis shows that our proposed scheme is provably secure against chosen ciphertext attacks. In addition, we show that our technique can also be applicable to the Key-Policy Attribute Based Encryption (KP-ABE) counterpart. "Identity-Based Encryption from the Weil PairingDan Bone, Matthew Franklin". We propose a fully functional identity-based encryption scheme (IBE). The scheme has chosen Cipher text security in the random oracle model assuming a variant of the computational Di e-Hellman problem. Our system is based on bilinear maps between groups. The Weil pairing on elliptic curves is an example of such a map. We give precise definitions for secure identity based encryption schemes and give several applications for such systems.

III. Problem Definition

A. Problem Definition

Cloud computing relies on sharing computing resources rather than having local servers or personal devices to handle applications and used as a metaphor for the internet so the phrase cloud computing means a type of internet based computing. To apply traditional supercomputing or high performance computing normally used by military and research to perform such as financial portfolios to deliver personalized information to provide storage or to power large uses networks of large groups of servers.

Cloud computing provides clients with a virtual computing infrastructure on which they can store data and run applications, introducing new security challenges since cloud operators are expected to manipulate client data without necessarily being fully trusted.

Cloud computing provides cryptography even in order to realize scalable flexible and fine grained access control of outsourced data, we analyze encryption methods and priority hierarchical structure of users.

Encryption is the conversion of data into a form called a cipher text that cannot be easily understood by unknown persons and decryption is the process of converting encrypted data return into its original form. Use of encryption/decryption is art of communication cipher often incorrectly called a code can be employed to keep the enemy from obtaining the contents of transmissions. In order to easily recover the contents of an encrypted signal the correct decryption key is required alternatively a computer can be used in an attempt to break the cipher. Fact that encryption might be accidently utilized on something that was not meant to be encrypted and the person who was meant to obtain the message may not be able to read the message sent to them, may not be strong enough and therefore others may be able to easily interpret information. Hierarchical structure of system users to achieve scalable flexible and fine grained access control low initial capital investment and maintenance.

B. Analysis for the above Problem

Cloud server is either proportional to the number of system attributes or linear to the size of the user access structure tree achieved. Our construction also protects user access privilege information again cloud server. Method for Hierarchical Attribute Solution:

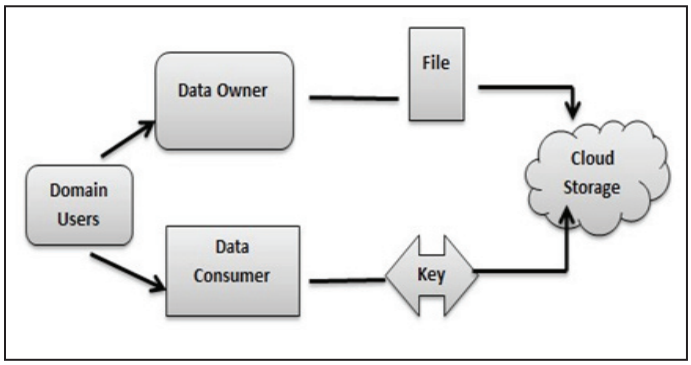

Fig. 1: Proposed Model

Data owner uploads the data in the cloud server for the security purpose, owner encrypts the file and store in the cloud and owner as rights to change the policy over data files by updating the expiration time.

Data Consumer user can only access the data files with the encrypted key if the user has the privilege to access the file. For all user level all the privilege is given by the domain authority and the data users are controlled by the domain authority.

Cloud service provider manages a cloud to provide data storage service, data owners encrypt their data files and store them in the cloud for sharing with data consumers. To access the shared data files data consumer download encrypted data files.

Authority person is responsible for generating and distributing system parameters and root master keys as well as authorizing the high level domain authorities. Domain authority is responsible for delegating keys to subordinate domain authorities at the next level or users in its domain.

IV. Encryption Methods in Cloud Computing

To achieve security and quality of data, it is very important to provide encryption and signature based scheme.

A. Identity Based Encryption

Identity based encryption cryptography is a third party server uses a simple identifier as an email address to generate key that can be used for encrypting and decrypting electronic data. Typical public key cryptography greatly reduces the complexity of the encryption process for users. Identity based encryption depends on the third party identity based encryption server that generates private keys, information stores permanently is a secret master key

a large random number that is exclusive to the security domain. The server uses this key to create a common set of public key parameters that are given to each user, the persons who are installed the identity based encryption software setup. When an outsourcing sender creates an encrypted message the identity based software on his system uses three parameters to generate the public key for the message.

B. Linear Search Algorithm

A symmetric encryption algorithm is used to encrypt the plain text for the cipher text of each keyword under symmetric encryption scheme a pseudo random sequence is generated with a length less that of the cipher text. At the same time check sequence is generated based on the pseudo random sequence and the cipher text. The sum of the lengths of the pseudo random sequence and the check sequence equals the length of the cipher text, the sum of the lengths pseudo random sequence equals the length of the cipher text.

C. Identity Based Signature

Identity based signature scheme is deterministic if the signature on a data by the same user is same, setup generates a private key provides the security parameter as the input to this algorithm generates the systems parameters and master private key. User extract his identity to private key generates as input and obtains the private key D and send to user through a secure channel. For generating a signature on a message m the users provides his identity private key D parameters and the message as input, the algorithm generates a valid signature on message by the user.

D. Attribute Based Encryption

The attribute and policies associated with the message and the user decides which user can decrypt a cipher text; the authority will create secret keys for the users based on attribute for each user. Users in the system have attributes receives a key from an authority for its set of attributes. Cipher text contains a policy predicate over the attribute space.

E. Homomorphic Encryption

Homomorphic encryption is cryptography which promises to make cloud computing perfectly secure a web user would send encrypted data to a server in the cloud, without decrypting it and send back a still encrypted result data. Sometimes however the server needs to know something about the data its handling otherwise some computational tasks become prohibitively time consuming if not outright impossible. Suppose for instance the task we outsourced to the cloud is to search a huge encrypted database for the handful of records that match an encrypted search tem. Homomorphic encryption ensures that the server has no idea what the search term or which records matches it. As a consequence however it has no choice record in the database. The user's computer can decrypt that information to see which records matched and which did not match then assuming much of the computational burden that was trying to offload to the cloud in the first.

V. Proposed System

In order to achieve efficient revocation, we introduce the idea of "partial private key update" into the proposed construction, which operates on two sides: (1) Utilized "hybrid private key" for each user in our system, which employs an AND gate connecting two sub-components namely the identity component (IK) and the time component respectively (TK). IK is generated by PKG in key-issuing but is updated by the newly introduced KU-CSP in key update; (2) In encryption, we take as input users identity as well as the time period T to restrict decryption, more precisely, a user is allowed to perform successful decryption if and only if the identity and time period embedded in his/her private key are identical to that associated with the ciphertext. Using such skill, we are able to revoke users decryption through updating the time component for private key by KU-CSP. Moreover, we remark that it cannot trivially utilize an identical updated time component for all users because revoked user is able to re-construct his/her ability through colluding with unrevoked users. To eliminate such collusion, randomly generated an outsourcing key for each identity, which essentially decides a "matching relationship" for the two sub-components. KU-CSP maintain a list UL to record user"s identity and its corresponding outsourcing key. In key-update, we can use OKID to update the time component TK[ID] T for identity ID . Suppose a user with identity ID is revoked at Ti . Even if he/she is able to obtain TK[ID`]Ti+1for identity ID` , the revoked user still cannot decrypt ciphertext encrypted under Ti+1.

The following fig. 2 shows the proposed system architecture.

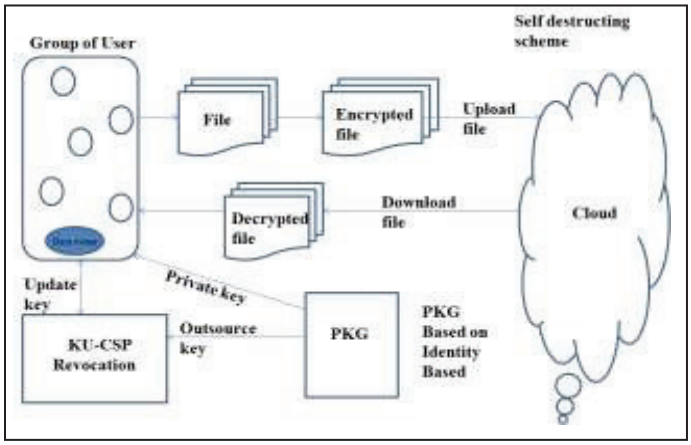

Fig. 2: Proposed System

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 users 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 KUCSP.
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 selfdestruction 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.

VI. Conclusion

Cloud computing brings great convenience for people. Particularly, it perfectly matches the increased need of sharing data over the Internet. In this paper, to build a cost-effective and secure data sharing system in cloud computing, we proposed a notion called RS-IBE, which supports identity revocation and ciphertext update simultaneously such that a revoked user is prevented from accessing previously shared data, as well as subsequently shared data. Furthermore, a concrete construction of RS-IBE is presented. The proposed RS-IBE scheme is proved adaptive-secure in the standard model, under the decisional ℓ-DBHE assumption. The comparison results demonstrate that our scheme has advantages in terms of efficiency and functionality, and thus is more feasible for practical applications.

References

[1] Jianghong Wei; Wenfen Liu; Xuexian Hu,"Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption", IEEE Transactions on Cloud Computing.

[2] Wei, Jianghong, Wenfen Liu, Xuexian Hu.,"Secure Data Sharing in Cloud Computing Using Revocable-Storage IdentityBased Encryption".
[3] Huang, Jyun-Yao, I-En Liao, Chen-Kang Chiang. "Efficient identity-based key management for configurable hierarchical cloud computing environment", Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on. IEEE, 2011.
[4] Qiu, Shuo, et al.,"Identity-Based Private Matching over Outsourced Encrypted Datasets".
[5] Tseng, Yuh-Min, et al.,"Identity-Based Encryption with Cloud Revocation Authority and Its Applications".
[6] Wang, Cong, et al.,"Secure ranked keyword search over encrypted cloud data", Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on. IEEE, 2010.
[7] Wang, Cong, et al.,"Privacy-assured outsourcing of image reconstruction service in cloud", Emerging Topics in Computing, IEEE Transactions on 1.1 (2013): pp. 166-177.
[8] Li, Jingwei, et al.,"Outsourcing encryption of attribute-based encryption with mapreduce", Information and Communications Security. Springer Berlin Heidelberg, pp. 191-201, 2012.
[9] Green, Matthew, Susan Hohenberger, Brent Waters. "Outsourcing the Decryption of ABE Ciphertexts", USENIX Security Symposium. Vol. 2011. No. 3. 2011.
[10] Agme, Varsha S., Archana C. Lomte,"Security Enhancement of Outsourced Data on Cloud Using Identity Based Encryption", 2014.

Suresh Patnaik Holds a B.Tech certificate in Computer Science Engineering from the University of JNTU Kakinada. He presently Pursuing M.Tech (CSE) department of computer science engineering from Baba Institute of Technology and Sciences, Visakhapatnam, AP, India.

Dakineni Durga Prasad, B.TECH, M.TECH is working as an Assistant Professor in the Department of Computer Science and Engineering in Baba Institute of Technology and Sciences, Visakhapatnam, A.P, India.