Employing HASBE Scheme for Assisting Access Controls in Out-Sourced Data Clouds

[1]Darwin.V.Tomy, [2]Dhanalakshmi.S, [3]Karthik.S
[1]Dept. of CSE (PG), SNS College of Technology, Coimbatore, Tamilnadu, India
[2,3]Dept. of CSE, SNS College of Technology, Coimbatore, Tamilnadu, India

Abstract
Cloud computing has appeared as one of the most leading standards in the IT engineering in past decades. Since this innovative computing technology requires users to deliver their valuable data to cloud service providers, there have been growing security and privacy concerns on data from outside supplier. Several schemes employing Attribute-Based Encryption (ABE) have been suggested for access control of outsourced data in cloud computing; however, most of them suffer from rigidity in applying complex access control strategies. As the cloud uses virtualization in back end all the methods implemented in real platforms can also be implemented in cloud. In several scattered systems a user should only be able to access data if a user holds a certain set of passes or attributes. Recently, the only method for enforcing such policies is to employ a trusted server to store the data and intermediate access control. However, if safety of server storing the data is compromised, then the secrecy of the data will be exposed.

Keywords
Access Control, Cloud Computing, Attribute-Based, Hierarchy, Encryption, Data Security, Blowfish Algorithm

I. Introduction

Cloud Computing have been used very effectively in these days. It provides a better promising future for highly scalable, secure, high performance, cost and energy efficient computing. Recently the cloud is been used in private organizations schools and colleges in near future all the corporations which include it and non IT departments will start using the cloud but however many researches are carried out cloud still faces security threats in both the public and private clouds and there are many remedies have been suggested recently.

Of those the attribute-Based Access control is the most effective way for setting up security standards in Cloud Computing. Attribute-Based Access Control is not a new technology because it has been widely used in other firms too for example simple networks, wireless sensor networks, ad-hoc networks, data centers even in vehicular ad hoc networks. Access control appears in many aspects of our life, reaching from physical access to security gates at a building, to simulated access by a computer program to memory. Attribute-based encryption (ABE) is a cryptographic mechanism for enforcing fine-grained access control to plaintext, be it a password for opening a door, or a confidential document unfolding the strategic plan of an enterprise. The access control policy is fine-grained in the sense that several attributes can be involved and policies involving different attributes can be combined together to form a more complex policy. For example, one might consider the rank or the department of an employee as attributes, and encrypt a document describing the wages of the employees of a company, in such a manner that either the CEO or administration from the human-resource department can decrypt the file. As a formula, the access control policy is:
(Rank = `CEO' or (Rank = `Management' AND Dept = `HR').

In a world overrun with digital identities and passwords, network access rights are often old-fashioned and overly broad. A new technology so-called attribute-based encryption (ABE) promises to significantly advance the field of trustworthy computing by basing access on a person's job description or constellation of role-based characteristics rather than identity. This two-faceted project specifically addresses the research challenges associated with ABE and considers future directions for research. One side examines ABE encryption, comprising a multi-authority system the other considers the challenges of integrating ABE into trustworthy systems. The goal behind the attribute based encryption is to provide the flexible, scalable access control for decentralized, collaborative environments and open systems

In a system the authorization decision is based on attributes of resource requestor such as rights to access the specific resources or designation they occupy their home organizations experiences they might have degrees or other characteristics like their age or nationality. Credentials are used to establish someone's authorization online these credentials contain signed policy statements about attributes of specific individuals, principles identified in the credentials or they can also contain rules for deriving attributes.

The idea of attribute based encryption (ABE) was first presented by Sahai and Waters as a step towards developing encryption systems with high expressiveness. Goyal et al additionally developed this idea and introduced two variants of ABE namely cipher text-policy attribute based encryption (CP-ABE) and key-policy attribute based encryption (KP-ABE). In a CP-ABE scheme, a user's private key is associated with a set of attributes (describing the properties that the user has) and an encrypted cipher text will specify an access policy over attributes. A user will be allowed to decrypt if and only if his attributes satisfy the cipher text's policy.

Roughly, it can be viewed as a generalization of identity-based encryption. So as in identity-based encryption, there is a single public key, and there is a master private key that can be used to make more limited private keys. However, CP-ABE is much more flexible than plain identity-based encryption, in that it allows complex rules specifying which private keys can decrypt which cipher texts. Specifically, the private keys are associated with sets of attributes or tags, and when we encrypt, we encrypt to an access policy which specifies which keys will be able to decrypt. We may want to provide scalable access to our files to others using additional resources available elsewhere.We may want more reliability in case of failures. In this case we may want to replicate our files in different data centers or with different organizations. But we want security. We may have requirements on who can access which files. The interesting thing is, there is a tension between security and the other properties. The more we replicate our files, the more we introduce potential points of compromise and the more trust we require. It's this tension which makes this sort of problem interesting, and provides a context in which CP-ABE may be useful.

Apart from the above exemplifying scenarios, ABE, as an encryption scheme, is applicable whenever secrecy is required.

Examples include distributed network storage (or storage in the cloud), and social network (with different portions of your profile and different categories of posts encrypted such that only your selected circles of friend can decrypt). Since ABE is a simplification of Identity-Based Encryption (IBE), it naturally supports the applications of IBE such as lightweight secure email and cryptographic workflow.

Finally note that one can always use encryption as a mechanism for authentication (the ability to decrypt a cipher text implies the possession of a certain equivalent credential), and hence ABE technology also supports attribute-based authentication. It has already been 5 years since the development of ABE. Schemes with various features have been devised. We have key-policy ABE (the access control policy is associated with the decryption key and the attributes are used to label the cipher text), cipher text-policy ABE (the decryption key is defined by a set of attributes and the access control policy is specified in each encryption) and dual-policy ABE. Policies can be developed by logical formula involving OR, AND, NOT, and threshold (e.g., at least 4 out of 6 formula is satisfied). We also have multi-authority ABE in which different authorities are responsible for issuing key for different domains of attribute, which not only delivers superior flexibility, but also guarantees that no single authority can decrypt all the cipher texts of the system. Research on these issues is still actively continuing.

II. Related Work

In this section, we review the types of attribute-based encryption (ABE).In this paper we discuss about various types of ABE techniques and researches which are being carried out in this field.

A. Attribute Based Encryption

The concept of attribute based encryption was introduce by Amit Sahai and Brent Waters in 2004.It is a classification of public-key encryption in which the public key of a user and the ciphertext are dependent about attributes. In such system, The decryption of a ciphertext is possible only if the set of attributes of the user key matches the attributes of the ciphertext. Attribute based encryption can be used for log encryption. Instead of encrypting each part of a log with all the key of all the receivers, it is possible to encrypt the log only with attributes which match recipient's attributes. This scheme can also be used for broadcast encryption in order to decrease the number of key used.

B. Ciphertext-Policy Attribute-Based Encryption (CP-ABE)

Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is a type of attribute-based encryption scheme which can be used to enforce ABAC cryptographically and address some of the aforementioned requirements. In CP-ABE, the data owner encrypts the data according to an access control policy P defined over a set of attributes, and the receiving end can decrypt the encrypted data only if his secret key associated with a set of attributes satisfies P. For example, suppose Alice encrypts her data according to an access policy P= (a1 AND a2) OR a3. Bob can decrypt the encrypted data only if his secret key is associated with a set of attributes that satisfy the access policy. To satisfy P, Bob must have a secret key associated with at least one from the following attribute sets: (a1, a2), (a3) or (a1, a2, a3). In general, CP-ABE scheme consists of the following four algorithms

C. (CP-ABE) Setup Algorithm

Setup algorithm (MK, PK) ← Setup (1k): is run by the trusted authority or the security administrator. The setup algorithm takes as input a security parameter k and outputs a master secret key MK and a master public key PK.

Key Generation algorithm (SK) ← Key Gen (MK, ω): is run by the trusted authority, and takes x as input a set of attributes ω and MK. The algorithm outputs a user secret key SK associated with the attribute set ω.

Encryption algorithm (CT) ← Encrypt (m, PK, P): is run by the encryptor. The input of the algorithm is defined as message m, a master public key PK and an access control policy P, the output of the algorithm is a ciphertext CT encrypted under the access control policy P.

Decryption algorithm (m) ← Decrypt (CT, SK): is run by the decryptor. The input of the algorithm is a ciphertext CT to be decrypted and a user secret key SK. The output of the algorithm is a message m, if the attribute set of the secret key satisfies the access policy P under which the message was encrypted, or an error message if the attribute set of the secret key does not satisfies the access policy P under which the message was encrypted.

D. Key Policy Attribute-Based Encryption (KP-ABE)

KP-ABE is a public key cryptography primitive for one-to-many transportations. In KP-ABE, data are linked with attributes for each of which a public key component is defined. The encryptor links the set of attributes to the message by encrypting it with the corresponding public key components. Each user is allocated an access structure which is usually defined as an access tree over data attributes, i.e., interior nodes of the access tree are threshold gates and leaf nodes are associated with attributes. User secret key is defined to replicate the access structure so that the user is able to decrypt a cipher text if and only if the data attributes satisfy his access structure [23]. A KP-ABE scheme is composed of four algorithms which can be defined as follows

E. (KP-ABE) Setup Algorithm

Setup algorithm This algorithm is used to set attributes for users. This is a randomized algorithm that takes no input other than the implicit security parameter. It defines a bilinear group G1 of prime order p with a generator g, a bilinear map $e : G1 \times G1 \rightarrow G2$ which has the properties of computability, bilinearity, and non-degeneracy. From these attributes public key and master key for each user can be determined.

1. Key Generation Algorithm

This is a randomized algorithm that takes as input an access tree T, the master key MK, and the public key K. It outputs a generated user secret key SK as follows. First, it outlines a random polynomial pi(x) for each node i of T in the top-down manner starting from the root node r. For each non-root node j, pj(0) = p parent(j)(idx(j)) where parent(j) represents j's parent and idx(j) is j's unique index given by its parent. For the root node r, pr(0) = y. And it outputs SK as follows.

SK = {ski} $i \in L$

where L means the group of attributes attached to the leaf nodes of T and ski = gpi(0)/ti .

2. Encryption Algorithm

This is a randomized algorithm that recivies a message M, the public key PK, and a set of attributes I as input. It results the cipher text E with the resulting format:

E = (I, Ẽ, {Ei ∈ I) where Ẽ= MYs, Ei = Tis. and s is randomly chosen from Zp

3. Decrypt Algorithm

This algorithm takes as input the cipher text E encrypted under the attribute set I, the user's private key SK for access tree T, and the public key PK. It first calculates e(Ei, ski) = e(g, g)pi(0)s for leaf nodes. Then, it calculate these pairing results in the bottom-up manner using the polynomial interpolation method. Finally, it may recuperate the blind factor Ys=e(g,g)ys and output the respective message M if and only if I satisfies T.

III. Blowfish Algorithm

Blowfish is a symmetric block cipher that can be effectively used for encryption and protection of data. It takes a variable-length key, from 32 bits to 448 bits, creating it ideal for safeguarding data. Blowfish was developed in 1993 by Bruce Schneier as a fast, free alternative to current encryption algorithms. Blowfish Algorithm is unpatented and license-free, open source and is available free for all uses. Blowfish is a variable-length key, 64-bit block cipher. The algorithm comprises of two parts: a key-expansion part and a data- encryption part. Key expansion transforms a key of at most 448 bits into several subkey arrays totaling 4168 bytes. The data encryption happens via a 16-round Feistel network. Each round consists of a key dependent transformation, and a key- and data-dependent substitution. All procedures are XORs and additions on 32-bit words. The only further operations are four indexed array data lookups per round. Blowfish requires about 5KB of memory. A careful employment on a 32-bit processor can encrypt or decrypt a 64-bit message in approximately 12 clock cycles. (Not-so-careful employments, like Kocher, don't growth that time by much.) Lengthier messages increase computation time in a linear fashion; for example, a 128-bit message takes about (2 x 12) clocks. Blowfish compute with keys up to 448 bits in length.

A. Key-expansion Part

Break the original key into a set of subkeys. Exactly, a key of no more than 448 bits is separated into 4168 bytes. Blowfish uses a large number of subkeys. These keys must be identified before any data encryption or decryption.The P-array contains of 18 32-bit subkeys: P1, P2,..., P18.

There are four 32-bit S-boxes with 256 entries each:

S1,0, S1,1,..., S1,255;
S2,0, S2,1,..,, S2,255;
S3,0, S3,1,..., S3,255;
S4,0, S4,1,..,, S4,255.

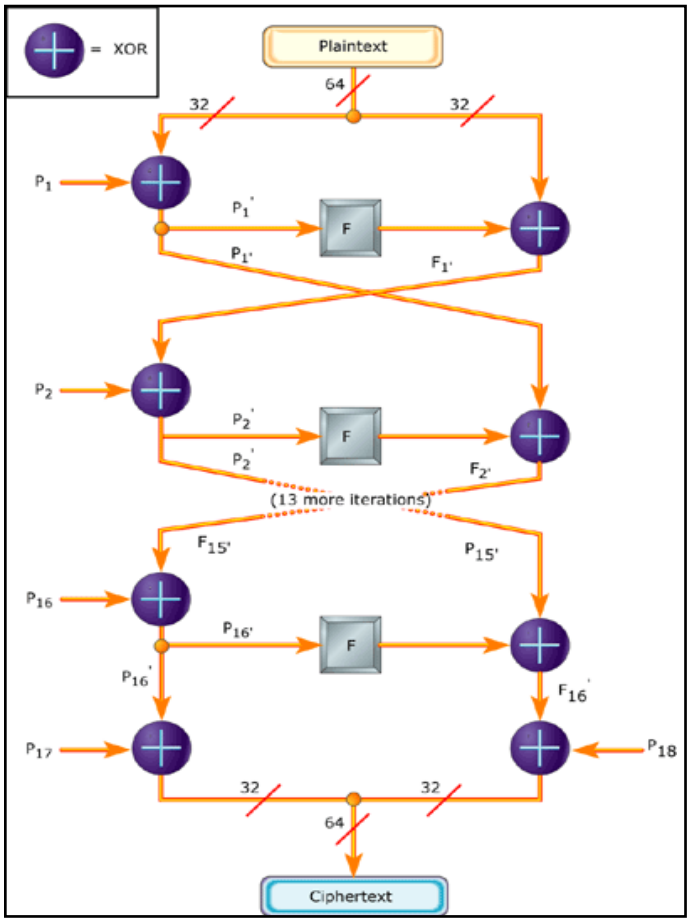

Fig. 1:

B. Data- encryption Part

Decryption is exactly the same as encryption, except that P1, P2,..., P18 are used in the reverse order.

IV. The HASBE Model

The HASBE scheme incorporates a hierarchical structure of system users by applying a delegation algorithm to ASBE. HASBE Scheme not only supports compound attributes due to flexible attribute set arrangements, but also achieves efficient user revocation because of multiple value assignments. In this project, we address this open issue and propose a secure and scalable fine-grained data access control scheme for cloud computing. The proposed scheme is partially based on our observation that, in real-world application scenarios each data file can be associated with a set of attributes which are meaningful in the context of interest.

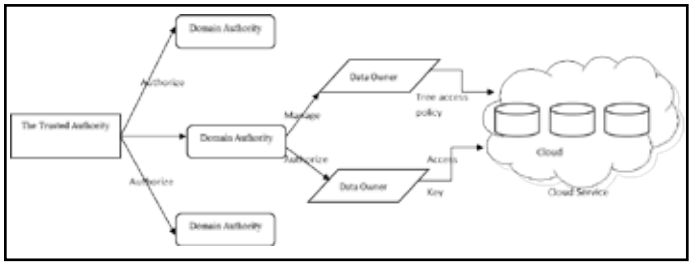

Fig. 2:

The access structure of each user can thus be defined as a unique logical expression over these attributes to reflect the scope of data files that the user is allowed to right to use. As the logical expression can epitomize any desired data file set, fine-grained of data access control is attained. To enforce these access structures,

we define a public key component for each attribute. Data files are encrypted via public key components corresponding to their attributes. User secret keys are to be defined to redirect their access structures so that a user is able to decrypt a cipher text if and only if the data file attributes satisfy his access structure.

V. Conclusion

We focus on improving the efficiency of ABE by leveraging a previously overlooked fact, i.e., the often-found hierarchy relationships among the Access control in that are inherent in many Cloud Computing Scenarios. As the first research effort along this direction, we coin the notion of hierarchical ABE (HABE), which can be viewed as the generalization of traditional ABE

References

[1] Allison Lewko,"New Proof Methods for Attribute Based Encryption: Achieving Full Security througt Selective technique", University of Texas at Austin alewko.

[2] John Bethencourt Carnegie Mellon University, Amit Sahai,"Ciphertext-Policy Attribute Based Encryption".

[3] Junbeom Hur, Dong Kun Noh,"Attribute- Based Access Control with Efficient Revocation in Data Outsourcing System".

[4] Mariana Raykova, Hang Zhao, Stevev M.Bellovi Coiumbia University, USA, Fmariana, Zhao,"Privacy Enhanced Access Control for Outsourced Data Sharing".

[5] Melissa Chase,"Multi-Authority Attribute Based Encryption", Computer Science Department Brown University providence, RI 02912.

[6] Melissa Chase Microsoft Reserch, Microsoft Way Redmond, WA 98052, USA and Sherman S.M Courant Institution of Mathematical Science New York University,"Improving Privacy and Security in Multi-Authority Attribute-Based Encryption".

[7] Ming li, Shucheng Yu, Yao Zheng, Kui Ren, Wenjing Lou,"Scalable and Secure Sharing of Personal Helth records in Cloud Computing using Attribute-Based Encryption".

[8] Sascha M uller, Stenfan Katzenbeisser, Claudia Eckert, "Distributed Attribute-Based Encryption", Technische University at Darmstadt Hochschulstr.

[9] Shucheng Yu, Kui Ren, Wenjing Lou,"FDAC: Toward Fine-Grained Distributed Data Access Control in Wirless Sensor Networks".

[10] Shucheng Yu, Cong Wang, Kui Ren, Wenjing Lou, "Achieving Secure, Scalable, and Fine –grained data Access Control in Cloud Computing", Dept. of ECE, Worcester Polytechnic Institute, Dept. of ECE, Illionis Institute of Technology.

Darwin V Tomy received his B.E. degree in Computer Science &Engineering from Vinayaka Missions Kirupanadha Variyiar Engineering College salem, Tamil Nadu India, affiliated to Vinakaya Missions University (VMU) in 2011, and currently pursuing ME (software engineering) in SNS college of technology, Coimbatore affiliated to Anna University Chennai. His research interests include Mobile Computing, Cloud computing and Network security.

Professor Dr.S.Karthik is presently Professor & Dean in the Department of Computer Science & Engineering, SNS College of Technology, affiliated to Anna University- Coimbatore, Tamilnadu, India. He received the M.E degree from the Anna University Chennai and Ph.D degree from Ann University of Technology, Coimbatore. His research interests include network security, web services and wireless systems. In particular, he is currently working in a research group developing new Internet security architectures and active defense systems against DDoS attacks. Dr.S.Karthik published more than 35 papers in refereed international journals and 25 papers in conferences and has been involved many international conferences as Technical Chair and tutorial presenter. He is an active member of IEEE, ISTE, IAENG, IACSIT and Indian Computer Society.