IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Implementation to Provide Security for Data in Cloud using Huffman Compression Approach

B. Anil Babu[1], K. Venkateswara Rao[2]

M.Tech, Department of CSE, Andhra Loyola Engineering College, Vijayawada, AP, India[1]

Assistant Professor, Department of CSE, Andhra Loyola Engineering College, Vijayawada, AP, India[2]

Abstract: Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. To keep sensitive user data confidential against un-trusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users. These solutions inevitably introduce a heavy computation overhead on the data owner for key distribution and data management when fine grained data access control is desired. Open issue by defining and enforcing access policies based on data attributes and allowing the data owner to delegate most of the computation tasks involved in fine grained data access control to un-trusted cloud servers without disclosing the underlying data contents. Several schemes employing attribute-based encryption (ABE) with Huffman compression has been proposed for access control of outsourced data in cloud computing. As Cloud Computing requires additional security which is provided using HASBE and this can emerge as a new security feature for various organizational platforms. It is implemented using cipher text policy by encrypting and decrypting the data in the cloud so that the cloud system becomes more scalable and flexible by enforcing data owners to share their data with data consumers controlled by the domain authority.

Keywords: Un-trusted servers, data owner, key distribution, Cipher text, data consumer.

I. INTRODUCTION

Cloud computing is a delivery of computing as a service rather than a product and information are provided to computers and other devices as a utility over a network. It provides computation, data access, software application, and data management without the requiring cloud users to know the location and other details of the computing infrastructure. End users access cloud based applications through a web browser or a light weight desktop or mobile application while the business software and data are stored on servers at a remote location. It providers strive to give the same or better service and performance than if the software programs were installed locally on end-user computers. On the need of sharing confidential corporate data on cloud servers, it is imperative to adopt an efficient encryption scheme with a fine-grained access control to encrypt outsourced data.

The Hierarchial Attribute Based Encryption allows the encryption of data by specifying an access control policy over attributes as one of the most promising encryption systems in this field. Hierarchical Attribute Based Encryption security for data's based on public key and master key with the help of Domain Authority Check. The hierarchical Attribute Set-Based Encryption (HASBE) scheme is for accessing control in cloud computing and extended the cipher text policy attribute set based encryption. Cloud computing holds the promise of providing computing as the fifth utility after the other four utilities. The benefits of cloud computing include reduced costs and capital expenditures, increased operational efficiencies, flexibility, scalability, immediate time to market. Different service-oriented cloud computing models have been proposed:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

One of the prominent security concerns is data security and privacy in cloud computing due to its Internet-based data storage and management. In cloud computing users have to give up their data to the cloud service provider for storage and business operations. Data is an important asset in any system and disclosure of data to business competitors and users leads to serious consequences. Data represents an extremely important asset for any organization and enterprise users will face serious consequences if its confidential data is disclosed to their business competitors or the public. Data confidentiality is not the only security requirement. The flexible and fine-grained access control is also strongly desired in the service-oriented cloud computing model. In addition to security, flexibility and fine grained access control is strongly desired in the service oriented cloud computing model.

II. RELATED WORK

We review the notion of attribute-based encryption (ABE), we examine existing access control schemes based on ABE. Several efforts followed in the literature to try to solve the expressibility problem. Ciphertexts are not

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

encrypted to one particular user as in traditional public key cryptography. A user is able to decrypt a ciphertext only if there is a match between his decryption key and the ciphertext. ABE schemes are classified into key-policy attribute-based encryption (KP-ABE) and cipher text-policy attribute-based encryption (CP-ABE).
KP-ABE, the authority determines what combinations of attributes must be present in order for this user to decrypt and gives the user the corresponding private key.
CP-ABE in that it allows complex rules specifying which private keys can decrypt which cipher texts. The private keys are associated with sets of attributes or labels and we encrypt to an access policy which specifies which keys will be able to decrypt.

Cipher-Text Policy: The trusted authority calls the algorithm to create system public parameters and master key. The public parameters will be made public to other parties and Master Key will be kept secret. Attributes asso- ciated with the ciphertext satisfy the tree access structure, can the user decrypt the ciphertext.

Kp-Abe Policy: We utilize KP-ABE to escort data encryption keys of data Files. These construction helps us to immediately enjoy fine- grandness of access control. CP-ABE scheme decryption keys only support user attributes that are organized logically as a single unit. Users can only use all possible combinations of attributes in a single set issued in their keys to satisfy policies as shown in the fig.1.

$a \equiv g^k \pmod p$; gcd(k,p-1) = 1; else a=1?
Message *M* (digraph, triblock graphs)
Public Key (g, p, $y \equiv g^k \pmod p$)
$M \equiv (xa + xb) \bmod (p-1)$

Where *k= Random secret value*
x= Private Key
Digital Signature (a,b) sent with M
$Y^a a^b \equiv g^M \pmod p$
The Math:
$g^M = g^{(xa + xb)} \bmod p$
$(g^x)^a (g^k)^b = y^a a^b \pmod p$
If *M* is modified, congruence would be violated

Fig.1: Kp-Abe Policy

III. CLOUD ARCHITECTURE DESIGN

Cloud computing has computational and sociological implications. Computational terms cloud computing is described as a subset of grid computing concerned with the use of special shared computing resources. It is described as a hybrid model exploiting computer networks resources, enhancing the features of the client/server scheme. Sociological standpoint on the other hand by delocalizing hardware and software resources cloud computing changes the way the user works as he/she has to interact with the "clouds" on-line.

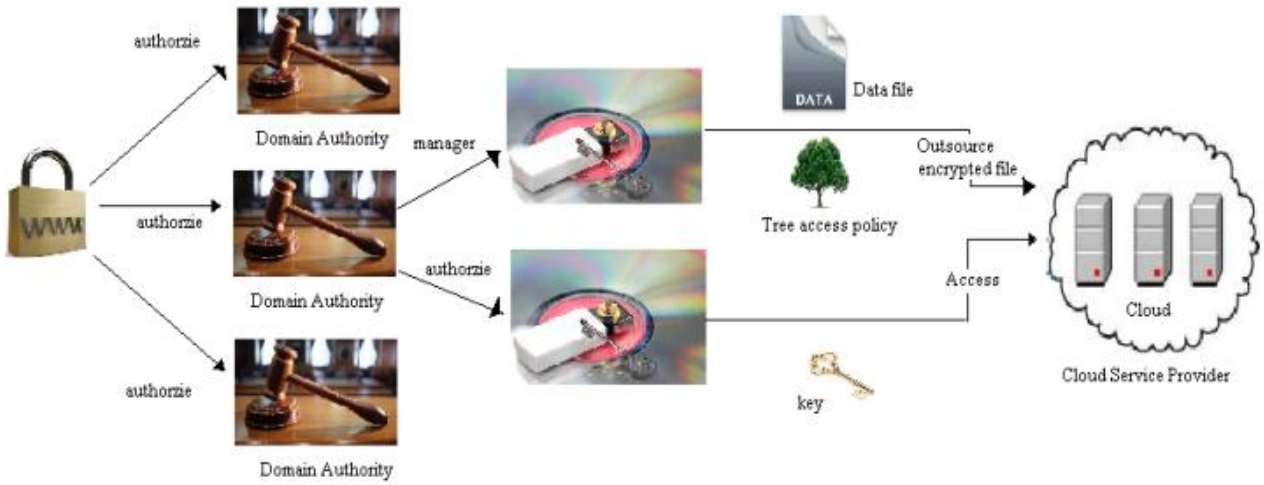

Fig.2. System Model

A hybrid cloud architecture opens the application to the "infinite" resources of the public cloud. Many factors play into the decision of selecting the appropriate infrastructure environment for the desired workload. Private cloud infrastructure typically provides a more controlled and optimized environment for deploying application workloads. Scalability can become an issue with a private cloud because resources are limited and finite.
Public clouds provide virtually infinite resources and provide an environment where applications can scale without bound. A common use case for the hybrid cloud is for applications that have stringent security requirements, which are best placed in the private cloud infrastructure.
All cloud environments also introduce management and provisioning challenges.

Facilitating the deployment of application components across multiple resource pools in a coordinated manner is complicated due to differing APIs.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

IV. HUFFMAN COMPRESSION PROCEDURE

Also known as Huffman encoding, a formula for the lossless pressure of information files based on the regularity of incident of an image in the data file that is being compacted. The Huffman criteria is based on statistical programming, which means that the probability of an image has a direct bearing on the duration of its reflection. The more probable the incident of an image is, the shorter will be its bit-size reflection. In any data file, certain figures are used more than others. Using binary reflection, the quantity of pieces required to signify each personality depends upon the quantity of figures that have to be showed. Using one bit we can signify two figures, i.e., 0 symbolizes the first personality and 1 symbolizes the second personality. Using two pieces we can signify four figures, and so on. Unlike ASCII rule, which is a fixed-length rule using seven pieces per personality, Huffman pressure is a variable-length programming system that assigns more compact requirements for more used figures and larger requirements for less often used figures in order to reduce the length of information files being compacted and transferred.

For example, in data with the following data:

XXXXXXYYYYZZ the regularity of "X" is 6, the regularity of "Y" is 4, and the regularity of "Z" is 2. If each personality is showed using a fixed-length rule of two pieces, then the quantity of pieces required to store this data file would be 24, i.e., (2 x 6) + (2x 4) + (2x 2) = 24.

If the above data were compacted using Huffman pressure, the more regularly happening numbers would be showed by more compact pieces, such as:

X by the rule 0 (1 bit), Y by the rule 10 (2 bits)

Z by the rule 11 (2 bits), therefore the length of the data file becomes 18, i.e., (1x 6) + (2 x 4) + (2 x 2) = 18.

In the above example, more regularly happening figures are assigned more compact requirements, resulting in a compact variety of pieces in the final compacted data file.

V. SYSTEM MODEL AND ASSUMPTION

As depicted in Fig. 2 the cloud computing system under consideration consists of five types of parties:

- cloud service provider

It manages a cloud to provide data storage service

- Data owners

It encrypt their data files and store them in the cloud for sharing with data consumers

- Data consumers

It download encrypted data files of their interest from the cloud and then decrypt them

- Domain authorities

Each data owner/consumer is administrated by a domain authority. It is managed by its parent domain authority or the trusted authority

- Trusted authority

It is the root authority and responsible for managing top-level domain authorities

As shown in the fig.2 data consumers, data owners, cloud service provider and the trust authority is organized in the hierarchial manner. Each top-level domain authority corresponds to a top-level organization, while each lower-level domain authority corresponds to a lower-level organization. Data owners/consumers may correspond to employees in an organization. Domain authority is responsible for managing the domain authorities at the next level or the data owners/consumers in its domain. The cloud is assumed to have abundant storage capacity and computation power. We assume that data consumers can access data files for reading only.

VI. SHARED RESOURCES AND TRUSTED AUTHORITY

The trusted authority acts as the root of trust and authorizes the top-level domain authorities. Domain authority is trusted by its subordinate domain authorities or users that it administrates, but may try to get the private keys of users outside its domain. The users may try to access data files either within or outside the scope of their access privileges. The trusted authority is responsible for generating and distributing system parameters and root master keys as well as authorizing the top-level domain authorities. Domain authority is responsible for delegating keys to subordinate domain authorities at the next level or users in its domain. Every user in the system is assigned a key structure which specifies the attributes associated with the user's decryption key.

VII. IMPLEMENTATION

The traditional method to protect sensitive data outsourced to third parties is to store encrypted data on servers. The decryption keys are disclosed to authorize users only. There are several drawbacks about this trivial solutions, such a solution requires an efficient key management mechanism to distribute decryption keys to authorized users. This approach lacks scalability and flexibility; as the number of authorized users becomes large. In case a previously legitimate user needs to be revoked, related data has to be re-encrypted and new keys must be distributed to existing legitimate users again. Data owners need to be online all the time so as to encrypt or re-encrypt data and distribute keys to authorize users. We have implemented a multilevel HASBE toolkit based on the CP-ABE toolkit developed for CP-ABE, which uses the Pairing-Based Cryptography library. Similar to the CP-ABE toolkit, our toolkit also provides a number of command line tools as follows:

- hasbe-setup (Generate Public key and Maste Key)
- hasbe-keygen (Generate Private key for Key structure)
- hasbe-keydel (Delegate some parts of Private keys)
- hasbe-keyup (Generate new private key that contain new attribute)
- hasbe-enc (Encryption of file)
- hasbe-dec (Decryption of file)
- hasbe-rec (Re-encryption of file)

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Our scheme can be extended to support any depth of key structure. Cost of this operation increases linearly with the key structure depth and the setup can be completed in constant time for a given depth. Top-Level Domain Authority Grant is performed with the command line tool "hasbe-keygen". Cost is determined by the number of subsets and attributes in the key structure. With the command hasbe-keydel a domain authority DA can perform New User/Domain Authority Grant for a new user or another domain authority in his domain. The cost grows linearly with the number of subsets to be delegated as shown in Fig. 3(a), when DA_i wants to delegate 45 of the attributes. As shown in the 3(b) the cost also increases linearly with the number of attributes in the subset.

As shown in the fig.3(c) the cost is linear with the number of the subsets, if the new attribute needs to be assigned to several subsets.

User Revocation operation consists of two steps:

- Key Update is implemented with the command hasbe-keyup. The root authority or domain authority can assign a new attribute to the user or domain authority.
- Data Re-encryption is performed with the command hasbe-rec. The data owner can re-encrypt the data file. When a user is revoked, the associated data file can be re-encrypted in this way and the new attributes can be assigned to valid user with command. Cost of operation Data Re-encryption depends on the number of attributes on the access tree.

Decryption should be done with the command hasbe-dec. The time of decryption is different depending on the access tree and key structure.

VIII. CONCLUSION

We introduced the HASBE scheme for realizing scalable, fine-grained access control, and flexible in cloud computing.The HASBE scheme seamlessly incorporates a hierarchical structure of system users by applying a delegation algorithm to ASBE. It not only supports compound attributes due to flexible attribute set combinations, but also achieves efficient user revocation because of multiple value assignments of attributes. The HASBE based on the security of CP-ABE and implemented the scheme and conducted comprehensive performance analysis and evaluation. We implement the proposed scheme, and conducted complete presentation analysis and estimate that showed its efficiency and advantages over existing schemes.

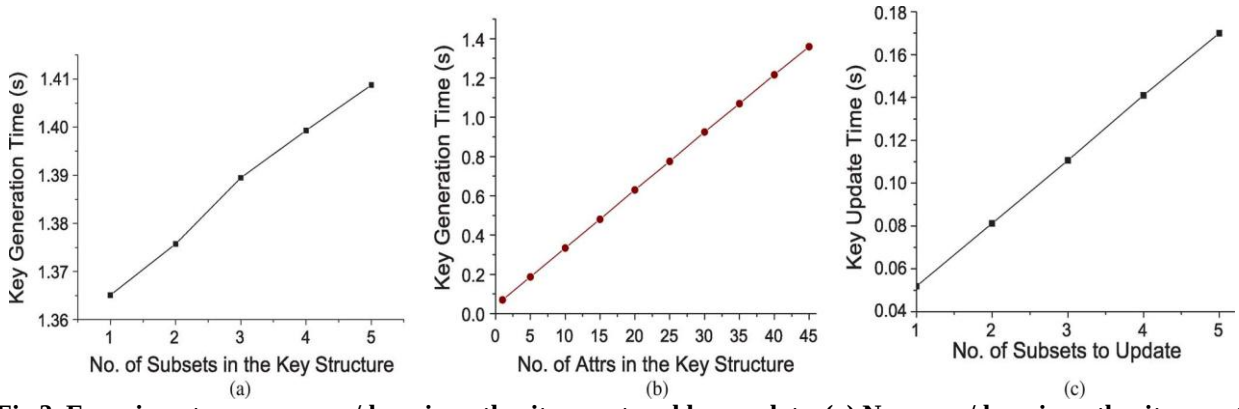

Fig.3. Experiments on new user/domain authority grant and key update. (a) New user/domain authority grant (b) new user/domain authority grant (c) key update

REFERENCES

[1] Zhiguo Wan, Jun'e Liu, and Robert H. Deng, "HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing," IEEE Transactions on INFORMATION Forensics and Security, VOL. 7, NO. 2, APRIL 2012.

[2] A.Vishnukumar, G.Muruga Boopathi, S.Sabareessh, " Scalable Access Control in Cloud Computing Using Hierarchical Attribute Set Based Encryption (HASBE)," International Journal of Emerging Science and Engineering (IJESE) ISSN: 2319–6378, Volume-1, Issue-4, February 2013.

[3] Chandana.V.R, Radhika Govankop,Rashmi N and R.Bharathi, "GASBE: A GRADED ATTRIBUTE-BASED SOLUTION FOR ACCESS CONTROL IN CLOUD COMPUTING," International Conference on Advances in Computer and Electrical Engineering (ICACEE'2012) Nov. 17-18, 2012.

[4] R. Martin, "IBM brings cloud computing to earth with massive new data centers," InformationWeek Aug. 2008 [Online]. Available: http://www.informationweek.com/news/hardware/data_centers/209901523

[5] Google App Engine [Online]. Available: http://code.google.com/appengine/

[6] K. Barlow and J. Lane, "Like technology from an advanced alien culture: Google apps for education at ASU," in Proc. ACM SIGUCCS User Services Conf., Orlando, FL, 2007.

[7] B. Barbara, "Salesforce.com: Raising the level of networking," Inf. Today, vol. 27, pp. 45–45, 2010.

[8] J. Bell, Hosting EnterpriseData in the Cloud—Part 9: InvestmentValue Zetta, Tech. Rep., 2010.

[9] R. Buyya, C. ShinYeo, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Comput. Syst., vol. 25, pp. 599–616, 2009.

[10] Amazon Elastic Compute Cloud (Amazon EC2) [Online]. Available: http://aws.amazon.com/ec2/

[11] Amazon Web Services (AWS) [Online]. Available: https://s3.amazonaws.com/