An Approach of Attribute Based Cryptosystem in Cloud Storage

Ranjini R[1], Girish[2], Phaneendra H D[3], Usha Rani J [4]
M.Tech (CNE) PG Student, Associate Professor, Professor, Assistant Professor
National Institute of Engineering, GSSSIETW, Mysuru

Abstract: In this world of the internet, people are increasingly opting for cloud storage for saving their data to off-site storage systems that are maintained by 3rdparties. Risks of losing important files and information are easily avoided by storing data on cloud storage. In this paper, we show how sensitive and confidential information can be shared securely, efficiently, and flexibly with others in cloud storage without unauthorized access. Here we show Cryptosystem scheme for data storage in cloud. One of the important techniques is ABE (Attribute Based Encryption). ABE is a public-key based one to many encryption techniques which allows users to encrypt and decrypt data based on user attributes. This technique effectively secures the data and also provides the correctness of the retrieved data along with the recovery mechanism for the transmitted data in case of malicious attack. The implementation of this scheme will show the correctness of the secure data storage and also advantages over other techniques supporting secure data storage in cloud.

Keywords: Cryptosystem, Key aggregation techniques, ABE (Attribute based encryption).

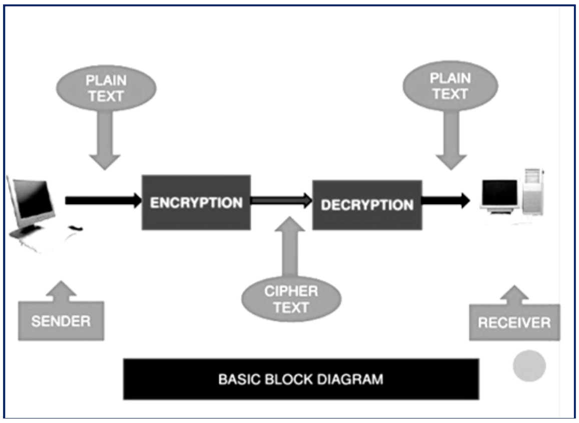
I INTRODUCTION

The use of cloud computing has enhanced speedily in several organizations. Cloud-based services include Software-as-a- service (SaaS) and Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Cloud computing offers various facilities for Information storage and information sharing. User generally deploys their information over cloud storage in terms of GB or TB. so cloud computing is advantageous in terms of low price and accessibility of knowledge. Making certain the safety of cloud storage could be a major think about the cloud computing atmosphere, as someday users store sensitive info with cloud storage. Whereas information sharing in cloud computing atmosphere, information from totally different users will be keep on separate virtual machines (VMs) however belongs to one physical machine. However information during a target VM can be taken by instantiating another VM on same physical machine. so considering ancient ways in which of knowledge privacy, some depends on the server to enforce the access management when authentication [3] or some permits a third-party auditor to ascertain the supply of files on behalf of the information owner while not unseaworthy the information [2]. However cloud user cannot totally depend upon cloud server for his or her information security and confidentiality purpose. so users square measure motivated to write in code their information with own keys so providing access to solely desired Receivers

Cryptosystem

A cryptosystem is an essential for keeping vital information safe on the Internet. Any time-sensitive personal information is passed through what is known as a secure server and the information is then encrypted, or put into Code. The person receiving the information must have the proper system for decoding, or decrypting, the information.

Fig 1: shows the sender sends plaintext and encryption process converts into cipher ext at the receiver side decryption process converts cipher text into plain text.



Mathematically, a cryptosystem or encryption scheme can be defined a tuple $(\mathcal{P},\mathcal{C},\mathcal{K},\mathcal{E},\mathcal{D})$ with the following properties.

$\mathcal{P}$ is a set called the "plaintext space". Its elements are called plaintexts.

$\mathcal{C}$ is a set called the "ciphertext space". Its elements are called ciphertexts.

$\mathcal{K}$ is a set called the "key space". Its elements are called keys.

$\mathcal{E}=\{E_k : k \in \mathcal{K}\}$ is a set of functions $E_k : \mathcal{P} \rightarrow \mathcal{C}$. Its elements are called "encryption functions".

$\mathcal{D}=\{D_k : k \in \mathcal{K}\}$ is a set of functions $D_k : \mathcal{C} \rightarrow \mathcal{P}$. Its elements are called "decryption functions".

For each $e \in \mathcal{K}$, there is $d \in \mathcal{K}$ such that $D_d(E_e(p)) = p$ for all $p \in \mathcal{P}$

II Literature Survey

One of the review paper describes a new type of Identity-Based Encryption (IBE) scheme that they call Fuzzy Identity-Based Encryption. In Fuzzy IBE an identity as set of descriptive attributes. A Fuzzy IBE scheme allows for a private key for an identity, ω, to decrypt a ciphertext encrypted with an identity, ω', if and only if the identities ω and ω' are close to each other as measured by the "set overlap" distance metric. A Fuzzy IBE scheme can be applied to enable encryption using biometric inputs as identities; the error-tolerance property of a Fuzzy IBE scheme is precisely what allows for the use of biometric identities, which inherently will have some noise each time they are sampled. Additionally, Fuzzy-IBE can be used for a type of application that we term "attribute-based encryption".

In our project two constructions of Fuzzy IBE schemes are implemented, our constructions can be viewed as an Identity-Based Encryption of a message under several attributes that compose a (fuzzy) identity. Here IBE schemes are both error-tolerant and secure against collusion attacks. Additionally, basic construction does not use random oracles. And proving the security of schemes under the Selective-ID security model. In this application a party will wish to encrypt a document to all users that have a certain set of attributes. For example, in a computer science department, the chairperson might want to encrypt a document to all of its systems faculty on a hiring committee. In this case it would encrypt to the identity{"hiring-committee","faculty","systems"}. Any user who has an identity that contains all of these attributes could decrypt the document

As more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encrypting data, is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). We develop a new cryptosystem for fine-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE). In our cryptosystem, cipher texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt.

In our paper demonstrating the applicability of construction to sharing of audit-log information and broadcast encryption. And construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE). In an ABE system, a user's keys and cipher texts are labeled with sets of descriptive attributes and a particular key can decrypt a particular cipher text only if there is a match between the attributes of the ciphertext and the user's key. In our system each ciphertext is labeled by the encryptor with a set of descriptive attributes. Each private key is associated with an access structure that specifies which type of ciphertexts the key can decrypt.

Attribute-based encryption provides good solutions to the problem of anonymous access control by specifying access policies among private keys or ciphertexts over encrypted data. In ciphertext-policy attribute-based encryption (CP-ABE), each user is associated with a set of attributes, and data is 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. CP-ABE is very appealing since the ciphertext and data access policies are integrated together in a natural and effective way. Most current CP-ABE schemes incur large cipher text size and computation costs in the encryption and decryption operations which depend at least linearly on the number of attributes involved in the access policy. In this paper, we present two new CP-ABE schemes, which have both constant-size cipher text and constant computation costs for a non-monotone AND gate access policy, under chosen plaintext and chosen ciphertext attacks. The security of first scheme can be proven CPA-secure in standard model under the decision n-BDHE assumption. And the security of second scheme can be proven CCA-secure in standard model under the decision n-BDHE assumption and the existence of collision-resistant hash functions. Our scheme can also be extended to the decentralizing multi-authority setting.

A new methodology for realizing Ciphertext-Policy Attribute Encryption (CP-ABE) under concrete and non interactive cryptographic assumptions in the standard model. Solutions allow any encrypt or to specify access control in terms of any access formula over the attributes in the system. In their system, ciphertext size, encryption, and decryption time scales linearly with the complexity of the access formula Here presenting three constructions within our framework. Our first, system is proven selectively secure under a 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 tradeoffs to achieve provable security respectively under the (weaker) decisional Bilinear-Diffie-Hellman Exponent and decisional Bilinear Diffie-Hellman assumptions.ABE has been applied in building a variety of secure systems

Two fully secure functional encryption schemes: a fully secure attribute-based encryption (ABE) scheme and a fully secure (attribute-hiding) predicate encryption (PE) scheme for inner-product predicates. In both cases, they were proven to be selectively secure. Both results use novel strategies to adapt the dual system encryption methodology introduced by Waters.In this paper constructing our ABE scheme in composite order bilinear groups, and prove its security from three static assumptions. Our ABE scheme supports arbitrary monotone access formulas. Our predicate encryption scheme is constructed via a new approach on bilinear pairings using the notation of dual pairing vector spaces proposed by Okamoto and Takashima. Attribute-Based Encryption as a new concept of encryption algorithms that allow the encrypter toset a policy describing who should be able to read the data. In an attribute-based encryption system, private keys distributed by an authority are associated withsets of attributes and ciphertexts are associated with formulas over attributes' user should be able to decrypt a ciphertext if and only if their private key attributes satisfy the formula.

III SYSTEM DESIGN

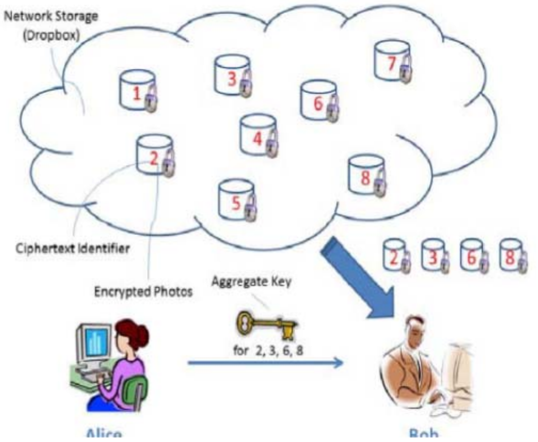


Fig: Example showing implementation KAC concept

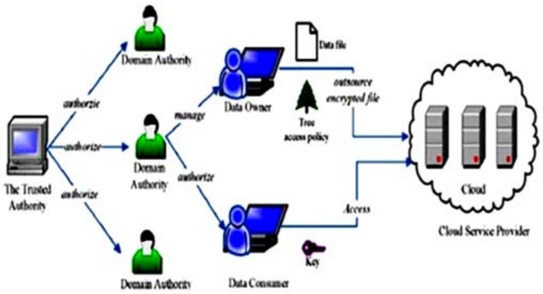
Cryptography is an effective way of protecting sensitive information as it is stored on media or transmitted through network communication paths. Although the ultimate goal of cryptography, and the mechanisms that make it up, is that to hide information from unauthorized individuals, most algorithms can be broken and the information can be revealed if the attacker has enough time, desire, and resources. So a more realistic goal of cryptography is to make obtaining the information too work-intensive to be worth it to the attacker.

Key Aggregate Cryptosystem (KAC) in which one can aggregate any set of secret keys and make them as compact as a single key, but it encompass the power of all the keys being aggregated.

As show in figure In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. That means the cipher texts are further categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes

IV SYSTEM IMPLEMENTATION

Data Flow Diagram



Here as show in the diagram the trusted authority will have authority to authorize the domain authority then the domain authority will manage the data owner(the person who is uploading the data/file to share in the cloud. likewise the data consumer(the person who is requesting a particular file/data in the cloud server for download)once the domain authority notifies that the data consumer is registered and he is verified that he is not the hacker he will get the key to download the encrypted file of the file which he is asked to the cloud server then he is going to download the file with the key.

V MODULES

For the system to be successfully implemented design of modules are necessary, so there are 5 modules as listed below with their description.

- System SETUP Model
- New User Grant Module
- File Upload Module
- File Access Module
- User Revocation Module

System Setup Model:

In this module, we develop the system module, which consist of Data owner. And this is controlled by domain authority, next data consumer and the cloud server provider.

Data Owner (Alice): In this module we executed by the data owner to setup an account on an untrusted server. On input a security level parameter 1^{λ} and the number of ciphertext 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.

Network Storage: With our solution, Alice can simply send Bob a single aggregate key via a secure e-mail. Bob can download the encrypted photos from Alice's Drop box space and then use this aggregate key to decrypt these encrypted photos. In this Network Storage is untrusted third party server.

New User Grant Module:

When a new user wants to join the system, with the aid issues an attribute private key to him/her based on his/her attributes. Based on the system model provided we attempt to define an underlying primitive namely OABE with outsourced key-issuing and decryption for realizing our access control system.

File Upload Module

In this module, we develop the file upload module process, where, When a data owner wants to outsource and share a file with some users, he/she encrypts the file to be uploaded under a specified attribute set (resp. access policy). In this module the Data Owner uploads only the text files.

File Access Module:

In this module, we create the file access module, When a user wants to access an outsourced file, he/she downloads cipher text from S-CSP and decrypts it with the help of D-CSP. Only when the user receives the secret key through the secure email.

- User Revocation Module

When there is a user to be revoked, updates \affected" users' private keys with the help of CSP, while the affected" cipher texts having been stored on S-CSP will be updated as well. User revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users.

Key Generation:

More important concept of this project is that the key generation scheme how is it generated the process is shown below:

1) Select File attribute1 – say File name
2) Convert the file name to Binary Codes
3) Select File attribute 2 – say file size
4) Convert the file size to Binary Codes
5) Perform AND Operation of File Attribute 1 and 2
6) Perform OR Operation of File Attribute 1 and 2
7 Result of AND Operation Stored as Secret Key
8) Result of OR Operation Stored as Public Key

- OR- ENCRYPT the data and AND- SECRET KEY generation

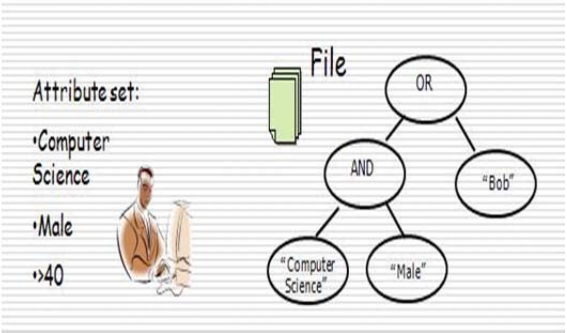


Fig: Representing example of key generation using attribute.

Here computer science and male are taken as attribute 1 and attribute 2, they converted to binary codes and then AND operation is performed and the result is stored as secret key, and same attributes are consider for OR operation and the result is being stored as public key and this public is sent to bob.

Aggregate Key Transfer:

A key-aggregate encryption scheme consists of five polynomial-time algorithms as follows. The data owner establishes the public system parameter via Setup and generates a public/master-secret key pair via Key Generation.

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 ciphertext classes via Extract. The generated keys can be passed to delegates securely (via secure e-mails or secure devices) finally; any user with an aggregate key can decrypt any ciphertext provided that the ciphertext's class is contained in the aggregate key via Decrypt

SEQUENCE DIAGRAM:

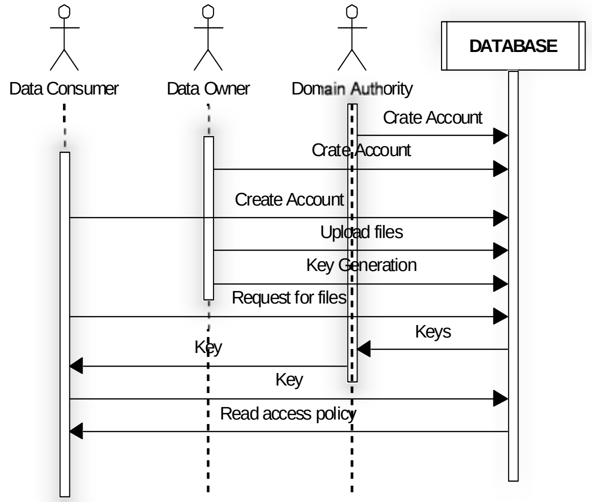


Fig: Sequence diagram showing the relationship between Data Consumer, Data Owner, and Domain Authority.

Figure Shows the detailed description of process of events taking place between the data owner, data authority, data consumer and how the information of all three parties are stored in database and the control and accessibility is only given to domain authority. When the consumer wants to access a file he has to request the secret key from the authority. Once he receives the secret key then he can download the file for his usage as shown in the figure.

As discussed above we have tried to implement the ABE(attribute based encryption) concept in the project so that based on the attribute of the data consumer and owner, the key will be generated to download the requested file for a consumer so that it overcomes the problem of hacker, hacking the files or data of any data owner who is uploaded for the use of many consumers in the cloud.

More important usage and highlight of ABE is that Attribute-based encryption(ABE) can be used for log encryption. Instead of encrypting each part of a log with the keys of all recipients, it is possible to encrypt the log only with attributes which match recipients attributes. This primitive can also be used for broadcast encryption in order to decrease the number of keys used.

CONCLUSION AND FUTURE ENHANCEMENT

Data privacy is a central concern of cloud storage. Storage. Cryptographic schemes are getting more and more versatile with the help of mathematical tools. Single application involves multiple keys. In this project, we consider how to "compress" or "aggregate" secret keys in public-key cryptosystems. This supports assignment of secret keys to different cipher text classes in cloud storage. The delegate to whom aggregate key is handed over always gets an aggregate key of constant size. In cloud storage the number of cipher texts grows rapidly. It will be better if maximum number of cipher text classes is unlimited. IDS should be more efficient so that when mobile users carry the delegated key the cryptosystem is leakage resilient. Key delegation can be more flexible.

The Future work of this can be continued for the image since this is applied to text only using some advanced key generation techniques this same can be applied to image encryption and decryption with the image processing techniques

REFERENCES

[1] Cheng-Kang Chu Dept. of Cryptography & Security,Singapore, Chow,S.S.M. ; Wen-Guey Tzeng ; Jianying Zhou ; Deng, R.H. Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage, Parallel and Distributed Systems, IEEE Transactions on (Volume:25 , Issue: 2, 11 April 2013)

[2] Data sharing in cloud storage with key-aggregate cryptosystem. Mrs. Komal Kate¹, Prof. S. D. Potdukhe² ¹PG Scholar, Department of Computer Engineering, ZES COER, pune, Maharash, International Journal of Engineering Research and General Science Volume 2, Issue 6, October-November, 2014 ISSN 2091-2730

[3] A Review on Data Sharing Across Cloud Storage Using Key Aggregate Cryptosystem Rashmi Khawale1 , Omprakash Tembhurne2,International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Impact Factor (2012): 3.358

[4] A LITERATURE SURVEY ON KEY AGGREGATION SYSTEM FOR SECURE SHARING OF CLOUD DATA, International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 3, Issue 12, December 2014

[5] Resilient Identity Based Encryption for Cloud Storage by using Aggregate Keys L.MohamedIrfan ,S.Muthurangasamy, T.Yogananth, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 3, March 2014

[6] Development of Enhanced Key-Aggregate Cryptosystem for Secure Cloud Storage, Goldie Lee Joe, Dr. N. K. Sakthivel, Vol. 4 - Issue 01 (January - 2015)e-ISSN: 2278-0181[IJARCET]

[7] Public Key Cryptosystem for Scalable Data Sharing In Cloud Storage R.S.Bhalerao 1, S.M.Rokade 2 , / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1) , 2015, 801-804,

[8] Decentralised Access Control with Aggregate Key Encyryption For Data Stored In Cloud Mr. Ashwin Chandra C, Ms. Dharani S International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014,