



## Attribute-Based Access To Multimedia In Cloud-Assisted Online Content Sharing

Priya A. Kamble\*

PG Student, M. Tech CSE,  
AGPCE, RTMNU, India

Pragati Patil

Assistant Professor, Department of CSE  
AGPCE, RTMNU, India

---

**Abstract**— Cloud computing, is the significant computing paradigm which allows the users to store their data into the cloud. This paper presents an Attribute-Based access to the media in the cloud where it uses a multiple message cipher-text policy Attribute-Based Encryption (CP-ABE) technique to create an access control structure. By using the advance algorithms for implementing the access policy; the user attributes are used so as to generate a public key. Thus the sharable data is encrypted and a secret key consisting of user attributes to decrypt the data and is used as an access policy in order to restrict the access of the user. This requires flexible and accessible cryptographic key management to support difficult access policies. The policy that is to be used for assigning a key to each user is based on a novel Multi-message Ciphertext Policy Attribute-Based Encryption (MCP-ABE) technique. The scheme allows a content provider to specify an access policy and encrypt multiple messages within one Cipher ext such that only the users whose attributes satisfy the access policy can decrypt the Ciphertext. Moreover, the technique shows how to support resource-limited mobile devices by offloading computational intensive operations to cloud servers while without compromising data privacy.

**Keywords**— Access control, Attribute Based Encryption, Ciphertext policy, Scalable media content, CP-ABE

---

### I. INTRODUCTION

Cloud computing offers the abstract view to the users and developers as it hides many of the implementation details. It is mainly used in content sharing networks. Examples of these networks are social networking where they are dynamic in terms of storage required. However, due to the weak security issues the use of cloud is not very fast in content sharing networks. Access policy is a mechanism that provides security facilitates the data to user in a controlled manner. The traditional mechanism is that the data is encrypted with the user's public keys. The data owners encrypt the data using this users public key and then uploads the file to the cloud. The user whenever wanted to download the file should decrypt the file with his generated secret key. By doing this there are a few problems like the owner has to get the public key of the user and the same data is encrypted with different public keys this result in storage overhead. The access policy here is completely based on permission relationship where the relationship is among user attributes and resource attributes. The attributes may be any information about the user's profession, job roles that is provided and is used to grant the access. However, in order to design an access policy mechanism there are many challenges to overcome some of them are-

1. Any individual is able to freely produce any number and any kind of online media such as text, image, sound, video, and presentations.
2. Any individual is able to grant any access to his media to anyone, at any time;
3. An individual may reveal a large number of attributes (e.g., name, age, address, friendship, classmate, fans, hobby, personal interest and mobility) and some of them can be very dynamics.
4. Individuals may share contents using various devices and bandwidth and hence demand different access privileges for the same media.

This approach allows the user to implement the access control on their data directly in content sharing service rather than through a central administrator. In order to provide a complex access policy mechanism we need flexible and scalable cryptographic key management algorithms. For improving these disadvantages we are using attribute based encryption hence we employ CP-ABE (Ciphertext policy — attribute based encryption) technique as a remedy to the above mentioned problem. In CP-ABE the recipient can decrypt the data only when the user attribute satisfy the access policy and this can be seen as one-to-many public key encryption and the data owner provide access to many users. In this system the users private key is associated with the user attributes and on the other hand the party that is encrypting the data specifying an access policy.

### II. LITERATURE SURVEY

The traditional access control architectures usually assume the data owner and the servers storing the data are in the same trusted domain, where the servers are fully entrusted as an omniscient reference monitor responsible for defining and enforcing access control policies [1]. This assumption, however, no longer holds in cloud computing since

the data owner. Furthermore, we observe that there are also cases in which cloud users themselves are content providers. They publish data on cloud servers for sharing and need fine-grained data access control in terms of which user (data consumer) has the access privilege to which types of data. In content-based access control, individuals are explicitly authorized to access collections of records matching certain criteria. We refer to these as collections as content slices. This approach supports individuals who do not have precisely defined roles, such as contractors or medical researchers. The nature of a content slice can be highly specific. The semantics of an attribute indicate some purpose or characteristic and, when used within larger collections, enable efficient identification and classification of like objects. For example, individuals in enterprise systems are often segregated into groups of common interest or duty based on a given set of attributes [2], e.g., function, department, university. These attributes are then used to associate sets of permissions and tasks to the specified individuals.

Existing systems, principally rely on the assignment and subsequent enforcement of policies by trusted and often centralized servers. However, these servers are acutely ill-equipped to deal with disconnected and asynchronous clients. Reliance upon centralized servers further limits scalability and mandates a single point of trust. Attribute-based encryption (ABE) [3], a generalization of identity-based cryptosystems, incorporates attributes as inputs to its cryptographic primitives. Objects are encrypted using a set of attributes describing the intended receiver. A principal possessing this subset as part of their pool of attributes can recover the original plaintext. More flexible requirements are achievable through the use of a thresholding primitive, for which only k-of-n attributes are necessary to perform decryption. Furthermore, the decryption under both the standard and threshold approaches is collusion-resistant as multiple parties are unable to meaningfully pool attributes. Such cryptographic mechanisms allow encryption to inextricably bind expressive, enforceable access policy for objects. As a significant research area for system protection[4], data access control has been evolving in the past thirty years and various techniques [5] have been developed to effectively implement fine-grained access control, which allows flexibility in specifying different access rights of individual users.

Traditional access control architectures usually assume the data owner and the servers storing the data are in the same trusted domain, where the servers are fully entrusted as an omniscient reference monitor [9] responsible for defining and enforcing access control policies. Recently, Wu Y. et al. [6] proposed an ABE system with outsourced decryption that largely eliminates the decryption overhead for users. In such a system, a user provides an untrusted server, say a cloud service provider, with a transformation key that allows the cloud to translate any ABE Ciphertext satisfied with that user's attribute or access policy into a simple Ciphertext, and it only incurs a small computational overhead for the user to recover the plaintext from the transformed Ciphertext. Security of an ABE system with outsourced decryption ensures that an adversary (including a malicious cloud) will not be able to learn anything about the encrypted message; however, it does not guarantee the correctness of the transformation done by the cloud.

### III. METHODOLOGY

Cloud computing is an emerging computing paradigm that brings forth many new challenges for data security and access control when users outsource sensitive data for sharing on cloud servers, which are not within the same trusted domain as data owners. 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, and thus do not scale well. In content sharing applications like social networking media, the mapping between user identity and resource is dynamic, access control methods related to our work can increase the security level against user collusion attacks due to use of attribute-based encryption. So the fundamental policy is to implement a control model where we can provide attributes attached to both users and resources. In online content sharing applications, as a mapping between user identity and resource is dynamic, access control methods related to our work can be implemented using the following steps for more secured content sharing.

Components in an attribute -based access control scheme includes subjects each specified by a set of attributes, objects and access policies. Attribute Authority (AA), a trusted third party, sets up the system parameters of attribute-based encryption system (such as system-wide public key and master key), verifies every user's attribute (e.g., group membership, role, security clearance or authorization information) and issues personal secret key corresponding to the set of attributes of the user. In practice, there could be multiple AAs in a system. For example, a university or corporate may run an AA, and a user may act as an AA for his/her extended family members.

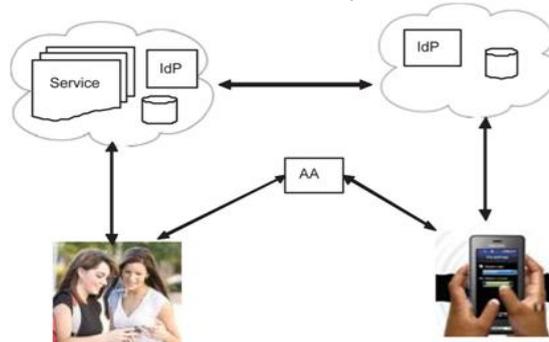


Fig. 1- System Architecture

To keep the presentation simple, we assume a single AA in the rest of the paper. The user may be a data owner, or a data consumer, or both. A data owner produces (protected or unprotected) media content (text, voice, video, etc.), and uploads the media content to cloud servers. To enforce access control to his data, the data owner assigns access privileges to data consumers whom the data owner may or may not know. A data consumer downloads, media content of her interest from cloud servers, and obtains the content based on her attributes and the access policy of the data owner.

To this end, the data consumer must obtain from AA a personal secret key bound to her set of attributes. In this data owner-consumer model, the backend servers pro-build the fundamental platform for storage, networking, etc; the foreground servers provide the interface for media generation, transmission, and computational assistance to users; while AA issues personal secret keys so that access control can be enforced flexibly based on user attributes and media scalability.

#### IV. PROPOSED SYSTEM

This section introduces the basic concepts of one-way hash function, bilinear map, access tree, and CP-ABE [3]. For simplicity, we assume that the data owner and a consumer has different genders. The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delays, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Preparing input validations and steps to follow when error occur.

##### A. One-Way Hash Function

This phase is to ensure the security and for data management purposes. The “one way” means that it’s nearly impossible to derive the original text from the string. A hash function takes a variable-length input string and converts it into a fixed-length output string, called a hash value. A one-way hash function, denoted as, works in one direction only: it is easy to compute a hash value from a pre-image; however, given an image, it is hard to find a pre-image such that.

A one-way hash function  $H$  operates on an arbitrary length input message  $M$ , returning  $h=H(M)$ . The important properties are:

Given  $M$ , easy to compute  $h=H(M)$

Given  $h$ , hard to compute  $M$  such that  $h=H(M)$  -- "one-way"

Given  $M$ , hard to find  $M'$  (different from  $M$ ) such that  $H(M)=H(M')$

(Not always satisfied) Hard to find  $M, M'$  such that  $H(M)=H(M')$  -- "collision resistant"

##### B. Access Tree

In any access control scheme, there is an access policy which defines the access conditions under which a subject can access an object. An access tree is a graph representation of the access policy. Such a tree includes non-leaf nodes and leaf nodes. Each leaf node is associated with a user attribute (e.g., age, gender, profession), while each non-leaf node has child nodes which may be leaf nodes, other non-leaf nodes or both.

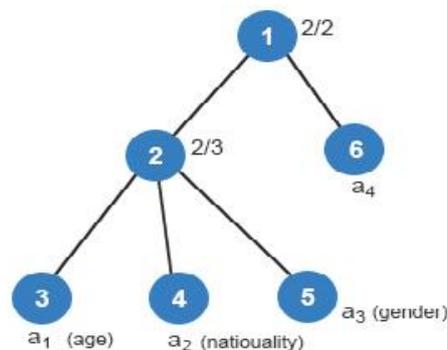


Fig. 2- Assigning Attributes To Nodes

##### C. Cipher-Text Policy Attribute-Based Encryption

As per the concept used in base paper, every user’s personal secret key is associated with a set of attributes while every Ciphertext is associated with an access policy. A user successfully decrypts a Ciphertext only if her set of attributes satisfies the access policy specified in the Ciphertext. We briefly describe the CP-ABE



Fig. 3- Proposed System Model

We will extend this CP-ABE scheme to MCP-ABE scheme and use the latter in our access control scheme.

1. **AB-Setup-** It is an initialization algorithm run by an Attribute Authority (AA). It takes as input a security and outputs a public key (PK) and a master secret key.
2. **AB-Keygen-** It is run by AA to issue a personal secret key to a user. It takes as input MK and the set of attributes A of the user, and outputs the personal secret key SK associated with specifically, for each user.
3. **AB-Encrypt-** Data owner to encrypt a message according to an accessible tree.
4. **AB-Decrypt-** Data consumer in possession of a set of attributes A and the secret key SK in order to decrypt the cipher-text CT with an access policy.

The system asks the user to do registration where user can set the attribute value. Here user can decide if he wants to share any content to the set of users based on their location or he/she wanted to access the media which has been shared already by the other registered users. The shared content is stored in the encrypted format. User can only access the sharable content if and only if his/her attributes matches the attributes specified in the access policy. There can be two conditions to determine. The first will be if the attribute of the user-2 matches with that of the all attributes (age, nationality, gender) specified in the access policy, the encrypted file would be decrypted in user-2 system with the decryption algorithm and a secret key would be generated. The key will be used to decrypt the shared data and user-2 can view the media content. The second condition is based on the contradictory method. If the attributes of user-2 did not match with the attributes specified in access policy of user-1 then the access would be restricted and an error message will be shown to the user. Here the encrypted text would remain the same and no decryption would be done.

## V. RESULT AND ANALYSIS

Components in an attribute-based access control scheme includes subjects each specified by a set of attributes, objects and access policies. Components in an attribute-based access control scheme includes subjects each specified by a set of attributes, objects and access policies.

### A. Experimental Results

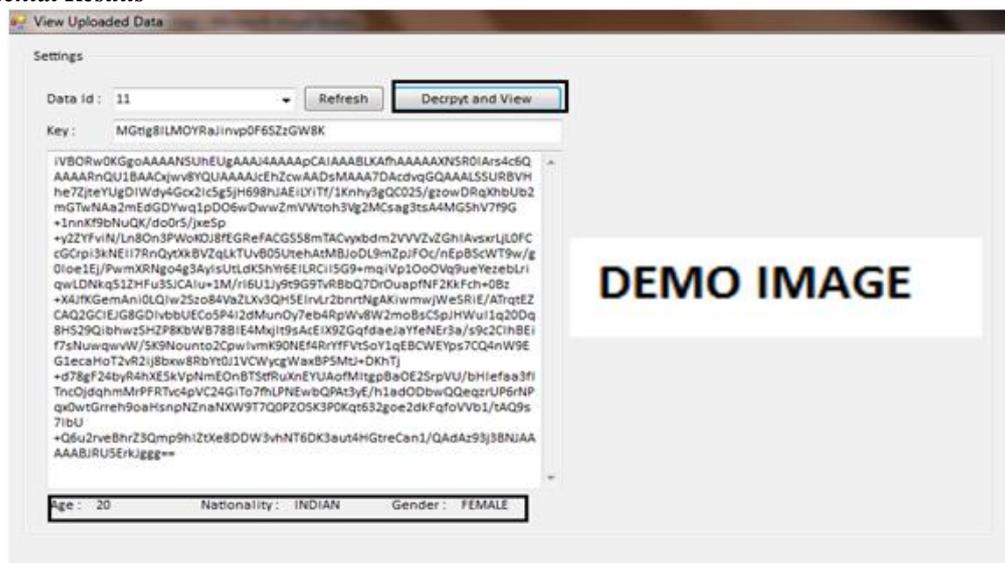


Fig. 3- Key Decryption Using Users Attribute (Case 1-Access Granted)

The shared media by user 1 will be accessible by other users only if the attributes entered by user 1 are matched with those attributes of the users who wants to access it.

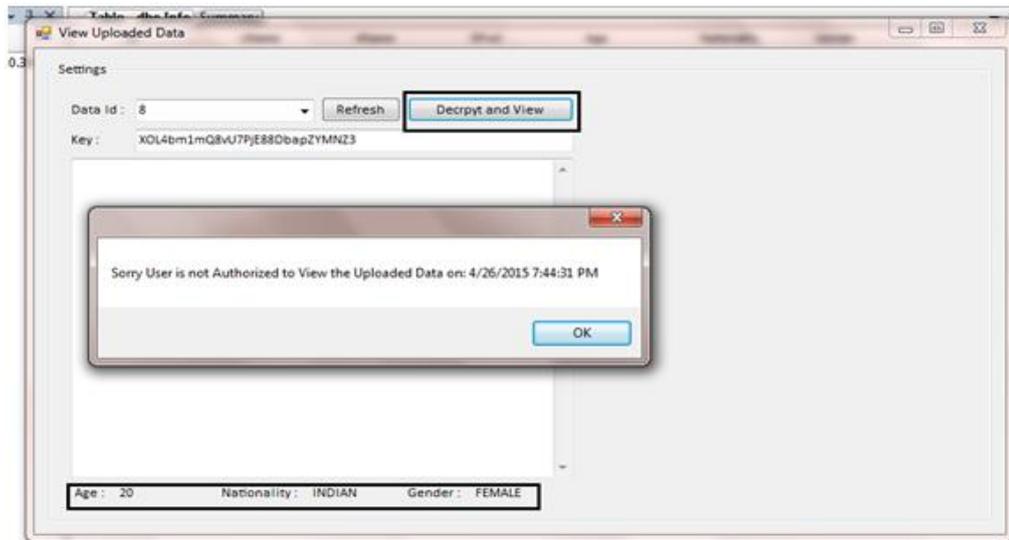


Fig. 4- Key Decryption Using Users Attribute (Case 1-Access Denied)

Fig. 4 shows the opposite case where user-3 gets an error message when his/her attributes did not match with the entered one.

**B. Comparative Analysis:**

The overall observation shows that the CP-ABE scheme forms a strong foundation for encryption and access control. Having ABE as a base, many ABE based schemes were proposed by enhancing, extending and modifying it. And next to the ABE scheme, the MCP-ABE proved as a prominent scheme by deciding who can decrypt the data stored on cloud servers. Following that there were other schemes which focused on scalability, fine-grained and multi authority based concepts. Some schemes focussed on reducing the decryption overhead by outsourcing the decryption process to the third party or to the cloud service providers.

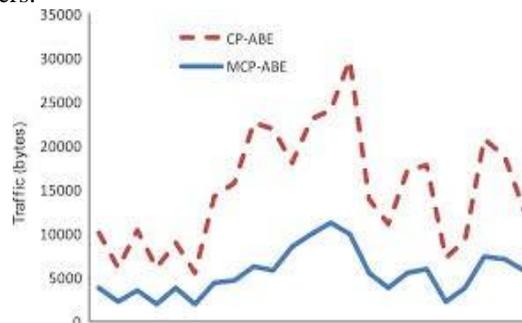


Fig 4. Comparative Graph In Terms Of Access Authentication

The existing systems used to support users with both positive and negative attributes, the later versions supported with different attributes based on the key policy.

TABLE 1 COMPARATIVE ANALYSIS OF OUTPUT PARAMETERS

Parameters	Existing System	Proposed System
Password Complexity	Less	More
Computation Overhead	Average	Above Average
Decryption And User Revocation Efficiency	Average	High
Collusion Resistant	Average	Above Average
Association of Attributes	With Ciphertext	With Key
Association of Access Policy	With Key	With Ciphertext

The scheme increases the efficiency of both attribute revocation and decryption process through their own token generation algorithm and considered as latest version of the ABE with high efficiency. The existing systems used to support users with both positive and negative attributes, the later versions supported with different attributes based on the key policy. However the proposed scheme not supports with different and multiple attributes, but even with their validation. Thus the scheme proved faster than most of the ABE based schemes.

## VI. CONCLUSION

In order to share media content in a controllable manner, a suitable access control mechanism should be deployed. As cloud computing is increasingly being adopted and mobile devices are becoming pervasive, the present access control scheme allows a mobile user to offload computationally intensive MCP-ABE operations to cloud servers while without compromising user's security. Below are the main points that can be concluded after the experimental sharing of image, audio and video to the group of limited users.

- The scheme has several benefits that ensure password complexity and security control over the sharable data.
- The present scheme is also secured against user collusion attacks due to use of attribute-based encryption.
- The experiments demonstrate that the present scheme is applicable on a smartphone, especially when a cloud platform is available.
- We present an access control scheme for scalable media. The scheme has several benefits which make it especially suitable for content delivery.

## ACKNOWLEDGMENT

I wish to place on record our sincere thanks & acknowledge indebtedness to Prof. Pragati Patil, HOD, Dept. of M. Tech. CSE for her valuable guidance, inspiration and affectionate encouragement to embark this project. I also acknowledge our overwhelming gratitude and immense respect to our Principal and other staff members who inspired us a lot to achieve the highest goal. In the ending lines, we cannot skip to express our thankfulness to our family without their moral support the work would not have been possible.

## REFERENCES

- [1] E. Messmer, "Are security issues delaying adoption of cloud computing?," Network World, Apr. 2009 and "Security of virtualization, cloud computing divides IT and security pros," Networkworld.com, Feb. 2010
- [2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute based encryption," in Proc. IEEE Symp. Security Privacy, 2007, pp. 321–334.
- [3] National Inst. Standards and Technol., Secure Hash Standard (SHS), FIPS Publication 180-1, 1995.
- [4] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST Special Publication 800-145, 2011.
- [5] M. D. Soete, "Attribute certificate," in *Encyclopedia of Cryptography and Security*, H. C. A. Van Tilborg and S. Jajodia, Eds., 2nd ed. Berlin, Germany: Springer, 2011, p. 51. 788 IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 15, NO. 4, JUNE 2013
- [6] B. Carunar, Y. Yu, W. Shi, M. Pearce, and V. Vasudevan, "Query privacy in wireless sensor networks," ACM Trans. Sensor Networks, vol. 6, no. 2, 2010, Art. ID 14.
- [7] X. Zhang, M. Nakae, M. J. Covington, and R. Sandhu, "Toward a usage-based security framework for collaborative computing systems," ACM Trans. Inf. Syst. Security, vol. 11, no. 1, pp. 1–36, 2008.
- [8] S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-based access control in social networks with efficient revocation," in Proc. ACM Symp. Inf. Computer Commun. Security, Mar. 2011, pp. 411–415.
- [9] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc. IEEE Int. Conf. Comput. Commun., 2010, pp. 1–9.
- [10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute based encryption for fine-grained access control of encrypted data," in Proc. ACM Conf. Comput. Commun. Security, 2006, pp. 89–98.
- [11] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, "Secure attribute based systems," in Proc. ACM Conf. Comput. Commun. Security, 2006, pp. 99–112.
- [12] J. A. Akinyele, C. U. Lehmann, M. D. Green, M. W. Pagano, Z. N. J. Peterson, and A. D. Rubin, "Self-protecting electronic medical records using attribute-based encryption," in Proc. ACM Conf. Comput. Commun. Security: Workshop on Security and Privacy in Smartphones and Mobile Devices, Oct. 2011, pp. 75–86.
- [13] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: An online social network with user defined privacy," in Proc. ACM SIGCOMM Conf. Data Commun., 2009, pp. 135–146.