# FINE-GRAINED ACCESS CONTROL AND SELF-PROTECTED DATA IN CLOUD COMPUTING

**R.ROHIT SAGAR**
Mtech (Computer Science and Engineering)
Institute of Aeronautical Engineering,
Dundigal, Hyderabad
Rohit.Rekabu@Gmail.Com

**Dr. G.RAMU**
Prof. Dept of Computer Science and Engineering
Institute of Aeronautical Engineering,
Dundigal, Hyderabad
g.ramucse@gmail.com

## ABSTRACT

*For any organisation it is a tedious task to hold such humungous amount of data in their data base. Ensuing, the concept of clouds has been introduced, where an external service provider manages the space for storing the data in an environment away from the organisations vicinity. It is unreliable for the organisation to store the data in the alien environment which poses a threat of data being infringed upon. To provide a self-protection mechanism to such data, RBAC is introduced along with an encryption mechanism. CP-ABE provides with access control and private key generation for the data. Hence, in order to improve the efficiency by assigning policies and providing an encryption mechanism RBAC is held in cohesion with CPABE and the concept of RBAC-CPABE has been introduced. RBAC uses a data centric method for providing security to the data object and an extended version of CP-ABE which is ECP-ABE is used for mapping attributes to the extended tree nodes. Ultimately, the concept of RBAC-CPABE is introduced to provide with the self-contained data protection in the cloud and regulate the access of the required users.*

## INTRODUCTION

Cloud computing provides an opportunity of saving the data that has been outsourced in an environment away from the provenance. Various such space providers are available where the data can be outsourced for saving. The problem that any organisation generally faces is that of providing security to such data in the cloud, where the data will be at the behest of the service provider. It is thus essential for the organisation to provide security for data by itself rather than depending upon the external service providers. The information is at risk even in transit and when the data is stored in the cloud from external intruders. To provide security from such infringement on data a security mechanism is to be provided. Data if provided with the internal access control mechanism where it can provide confidentiality to itself by allowing users who are recognised depending on the role assigned to them, this mechanism is called self-contained data protection. Role based access control (RBAC) method provides with such security where the data in itself is provided with the access control capability rather than depending on the service providers which is not provided with the orthodox identity based encryption(IBE) [1] mechanism. Conventional RBAC mechanism provides with the holistic protection of the data rather than providing

protection to the data objects. While utilizing only RBAC have its own repercussions as

- In assigning a role to the user by the owner, if a role is assigned to the users for accessing the data, then all those users who have the required criteria will be eligible to access the data which prompts for the breach of confidentiality.
- In considering a situation where every given user is assigned with a distinct role then it results in myriad of roles to be assigned and indeed requires a very large repository. This vitiates the space and increases burden over the operator.

Ensuing, to provide security to such data over the network and self-contained data protection an encryption policy has been utilized which is attribute based encryption (ABE)[2]. In this mechanism the private key allotted to the user and the cipher-text which is encrypted are assigned to certain attributes. When both attributes are tantamount to each other the user can decrypt the data. With the use of ABE access control and security through encryption are provided to the data. ABE is of two types among which one is key-policy attribute based encryption (KP-ABE)[3] where the private key is coalesced with the access policy and the cipher-text is coalesced with the attributes. The other form of ABE is cipher-text policy attribute based encryption (CP-ABE)[4] where the mechanism is the inverse to KP-ABE i.e. cipher-text is coalesced with the access policy and private key is coalesced with that of attributes. We consider CP-ABE in this paper, as pertinent to the concept of providing access policy and security provision through encryption of data. In CP-ABE, where cipher-text holds access policy, it verifies the policy that has been assigned by the user for providing authenticity. When the authenticity of the user is recognised by policy matching, set of attributes are matched by the private key and when these attributes are in tandem with users, private key is generated. Though, the required access control is provided by ABE, it can't be used for role hierarchy which is very much pervasive in the concept of RBAC.

For addressing this problem, Zhu et al [5] propounded a system by using ABE scheme, where a role is mapped to various attributes and introduced a hierarchical structure of attributes to recognise the role assigned to the user by the competent authority. The criteria specified in the access policy that the owner provides has to be met by the user in order to confirm the authentication and get access to the data. Various operators are used in order to provide access control, it is an essentiality to utilise NOT operator for enhancing the capability of access policy. To improve the competence for utilisation of comparative operators (<,>,<=,>=) and NOT operator the extended version of CP-ABE has been opted which is called extended cipher-text policy attribute based encryption (ECP-ABE). Then we envisaged integrating RBAC with ECP-ABE as RBAC provides with the identification of roles and ECP-ABE provides an opportunity for mapping the roles to attributes, authenticating the user and recognising the access policy, through its competence of using any given operator.

The data that is in the control of service providers in cloud always have a chance of getting corrupt, for securing such data self-contained mechanism is provided where the encryption of the every data object is done through ABE and an access policy is assigned such that only those who have the authenticity can access the data. Though the data is away from the parent company, it is safe and secure in the cloud far from infringement. For providing self-contained protection to data, we took the cognisance of using data centric RBAC model (DC-RBAC), where the access policy is concerned with the data. Hence, in order to provide security, access control, and self-contained data protection mechanism we propounded for the cohesion of DCRBAC-CPABE. Here private key generator is used as an authenticator for producing the private keys for the end user to access the information. When data is requested by the end user the private key generator (PKG) utilises the signature of the user through IBE and sends him the requested key, when the user uses the key he can decrypt the cipher-text encrypted by ABE mechanism. In order to reduce the burden off the PKG, we use attribute authenticator (AA) which is encompassed by the private key generator as a separator module. The functioning of AA is authenticating the user after verifying the roles of the user as assigned by the owner.

## LITERATURE SURVEY

The concept of role based access control was emanated in 1992 by Farraiolo and Kuhn[6]. This concept gained popularity in the mid 90's where the organisations were expanding and work was to be disseminated. This concept has gained a lot of momentum through granting of permissions in the recent years. RBAC provided with the perfect accommodation where it identified the target where the data has to be delivered without being impinged upon by the external intruders. Various studies were made and found it viable to provide RBAC with encryption mechanism in order to provide security to data. As with the emergence of usage of RBAC, a concept was evolved where the data that is encrypted, has been united with RBAC by using various encryption mechanisms. Crampton [7] then introduced a mechanism for encrypting RBAC where it uses cryptography in order to produce RBAC cryptographic mechanism. Zhu et al[5] then provided with role key hierarchy model where every user has to hold a private key for the role. This method increased the burden where each user had to have an individual key holding for his role.

As RBAC was providing security only for the accessibility of the user it was essential to provide protection to the data. In order to provide data protection encryption of the data was essential while it was in transit or when it was stored in cloud. When the data is stored in cloud, it was on the service providers to provide security. Service providers cannot be relied upon for providing security and hence an encryption mechanism was introduced as attribute based encryption. It provides security by converting the plain text into cipher-text when such data is converted in cipher-text, a mechanism is provided for enhancing security to the data by integrating RBAC with ABE. As ABE in the present context

provided access control and encryption to the data it couldn't be used for role hierarchy. For supporting such concept, ECP-ABE was introduced where it could use NOT and other comparative operators as well for the enhancement of the access control policy.

While attribute based encryption has the concept, where for one given encryption key, there can be multiple decryption keys which was based on fuzzy identity based (FIBE) encryption that was proposed by Sahai and Waters [8]. Later Goyal et al[9] extended this FIBE concept and introduced with the concept of key-policy attribute based encryption where the cipher-text is related with attributes and private key is related with an access tree. This concept was used by bethencourt and others [10] in evolving cipher-text policy attribute based encryption (CP-ABE) which is an inverse of KP-ABE. Here, cipher-text is related with the access policy and private key is related with attributes. The user whose attributes matches with the stipulated specifications gets to view the data in the cloud and request for the access. The attribute authentication on verifying the access policy and on finding it to be same provides an opportunity for requesting for the key.

ABE had the potential of supporting access tree by the usage of various operators as AND, OR, THRESHOLD (m,n (where at least m number of constraints are to be satisfied from n number of constraints)). But ABE didn't have the potential of supporting other operators as NOT and comparison operators. There were some schemes that supported only THRESHOLD operator and hence not viable to be introduced in the access tree structure. NOT operator is essential as it can be used in the context where a given element is to be kept aloof from getting access to the data. Though many developments were evolving, there was no proper scheme which can use all the operators under a single ambit. For enhancing the capability of using all defined operators, Lang et al [7] proposed an extended version of CP-ABE which is called extended cipher text policy attribute based encryption (ECP-ABE). In this policy, the leaf nodes are extended for the attributes to use comparative operators and compare between the attribute name node and attribute value node by using various different comparative operators as <,>,<=,>=. ECP-ABE being an extended version carries forward the capabilities of CP-ABE where it supports all the previously defined operators as AND, OR, THRESHOLD operators.

## METHODOLOGIES

The procedure which is involved in this paper is dependent on identification of the role and then authenticating the user depending on the access policy through providing a private key. Ensuing, CP-ABE scheme has been adopted, for the encryption of data and providing authentication to the data which is traversing over the network and to provide security in the server space in cloud.

### Cipher-Text Policy Attribute Based Encryption (CP-ABE)

As mentioned, in this scheme cipher-text is combined with access tree, the private key is combined with attributes. When the user's private key matches with the specified attributes of the user, then the access

permission for the data is granted. In CP-ABE, there are three parties that are associated with it.

Private Key Generator (PKG): The main function of the PKG is to initialise the system, identifying the authenticated user and providing him with the access through private key generator using keygeneration algorithm. In order to reduce the overload from PKG, attribute authentication (AA) has been used as a different component for identifying the user. AA verifies the access policies specified by the owner and checks for user's attributes to provide authentication. Then the user gets permission to request for a private key from private key generator (PKG).

Encryption Party: For the information to be secure over the network it is required that it has to be in an encrypted format. By using an encryption algorithm, the owner of the message specifies the access policies and encrypts the data.

Decryption party: The user for getting the access to the data has to meet the specified criteria. The private key which is generated by the PKG is used by the user in gaining access to the data which is stored in the cloud. The decryption party after meeting the criteria specified by access policy and obtaining the private key can access the data from the cloud by decrypting the cipher-text.

**Extended Cipher-text Policy Attribute Based Encryption (ECP-ABE)**

In order to enhance the competence of CP-ABE, a new scheme has been introduced which is ECP-ABE which is an extended version of CP-ABE. In ECP-ABE extended leaf nodes are used for matching the attributes by using different operators in an access tree. The operators which were not supported earlier by any other schemes in its ambit are encompassed by ECP-ABE scheme. ECP-ABE scheme has the capability of utilizing NOT and comparative operators (<,>,<=,>=) which no other scheme was supporting along with AND, OR, and THRESHOLD operators. Earlier, negating conditions were not used and for restricting a single given user access permission was to be granted to all the users except one. This burden has been reduced drastically by the use of ECP-ABE scheme. With the use of ECP-ABE, the threshold value which had a lower limit of 0 has been changed to -1. By using a negative number the comparative operator can be used to verify the access permission which is to be granted by the attribute authenticator. The three kinds of operators that are provided by ECP-ABE scheme are

- Logical operator: NOT
- Comparative operators: <,>,<=,>=
- Interval operators: [],[),(],()

Figure one provides with an example of extended leaf node holding the attributes. A regular tree is known as a standard tree and the tree which has an extension of leaf nodes is known as extended access tree. For the purpose of encryption the extended tree is converted into a standard tree and then encryption is performed. During decryption the attribute verification algorithm identifies the attributes in the extended leaf and PKG then generates the key depending upon the attributes verification.

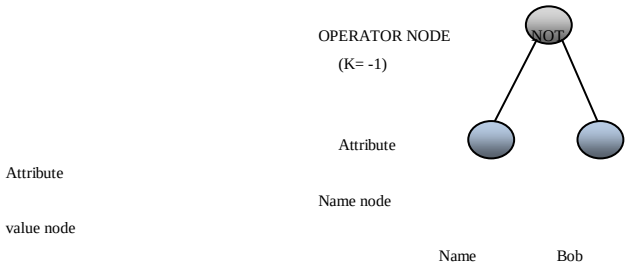

Fig. 1. Example

explaining extended leaf node.

**Role Based Access Control**

In order to specify conditions for granting permissions for the users, RBAC plays an important role as it requires fewer relationships to manage. RBAC also decreases the time complexity from O(mn) to O(m+n). There exist two conditions of granting access permissions which are discrete and mandatory access controls. Where it is depending upon the criteria access permission is granted in discrete access control and it is a compulsion on the authenticator to provide access in mandatory access control. While the development of RBAC initiated from RBAC 0 which involved users, roles, permissions and sessions. User is assigned with the permission of accessing the data in a session. The role the user holds grants him to access the data in a session for which role-permission has to be obtained by the user. Then RBAC 1 was introduced which is a combination of RBAC 0 and includes role hierarchies. For the extension of the concept a new model known as RBAC 2 was introduced which is a combination of RBAC1 and included constraints such as static mutual exclusion constraints and cardinality constraints. The reason for using constraints is to provide convenience when administration is centralised in a higher level organisation and acts as a tool to enforce high level policies when the administration is decentralised. The question that arises in RBAC is to whether multiple roles should be allowed or not? For addressing this problem RBAC96 is introduced which provides with an opportunity for using multiple roles to be assigned to the user.

**Data Centric Role Based Access Control Model (DC-RBAC)**

As it was mentioned earlier, RBAC has to provide a fine grained access control to the data and provide self- contained mechanism to protect itself by assigning arbitrary constraints. In order to specify such constraints for the data, a specific data centric role based access control mechanism (DC-RBAC) has to be utilised which should support role inheritance, assigning constraints and even role assignments. DC-RBAC appears to be similar to RBAC3 model which encompasses the conditions specified under RBAC1 and RBAC2. But there is a minor difference that exists in both the models. RBAC3 encompass four cases of constraints as

1. Constraints are associated with that of sessions, which says the number of sessions the user should be active at a particular time.
2. Cardinality constraints where the cap is fixed on the number of users who are assigned with a specified role and number of permission-role assignments.
3. Roles which are mutually exclusive.
4. Conditional constraints which specify that only whose users who are assigned to a given role are only allowed the permission to access the data.

While in RBAC3, the concept is provision of security to the whole data. But DC-RBAC has a different objective of providing security to every data object individually. The underpinning reason we consider to use DC-RBAC is, the third constraint is expressed by the use of NOT operator. Second and fourth constrains which are related with role assignments are to be kept with

DC-RBAC. As sessions are not required in DC-RBAC the previous parts requiring permissions for it are not addressed. The major difference between the usage of RBAC and DC-RBAC is the conventional method of RBAC uses only positive statements where as DC-RBAC uses both positive as well as negative statements. Consider a statement where role is not equal to the symbol r which can be represented by DC-RBAC model as (role !=r).

DC-RBAC model two different constraints as user attribute constraints, where the user related attributes such as name, experience, salary, etc are considered. The other constraint that is used is environmental constraints which include time of access, internet protocol address etc. Thus, DC-RBAC is more flexible for the usage when compared to conventional RBAC model.

**Performance Analysis**

In cloud security alliance (CSA), ECP-ABE is secure. But RBAC-CPABE had new security issues, first is every attribute that is used in the private key should not be encompassed in access tree. Here the intruder infringes upon the security parameters and attacks the access policy but not the extended tree. Second issue is the mapping which is visible to the intruder which is done before encryption. The security provided by RBAC-CPABE is can be analysed as

1. For addressing the first issue DC-RBAC holds an access policy P where the intruder has to meet the specified criteria which is 'a' does not belong to P. where a is the accessor.
2. The second issue is of mapping, the attacker tries to access the attributes of the user and though the mapping is visible he cannot retain data without the access permission.

Thus, when compared with the access policy of the ECP-ABE scheme, the security provided by RBAC-CPABE is the same.

The efficiency is the same of ECP-ABE when compared to that of RBAC-CPABE as it uses the same algorithm. Even the cost for calculation and the length of encrypted text are the same. There exist two differences between them, which are: the extended tree is to be mapped with the DC-RBAC access policy during encryption. During the key generator phase the extended attributes in the access tree are to be verified and extended. Thus, the efficiency is higher when compared to the BSW. The time that is required to map DC-RBAC to the extended tree is quite negligible and can is ignored. Even during the key application, there is no extra overload upon the system during role inheritance. Thus, ECP-ABE scheme enhances the access control capabilities and at the same time requires the same amount of time utilised by its previous version of CP-ABE. Maintaining the same computational overhead and not increasing any security risk RBAC and an extended version CP-ABE is used.

**CONCLUSION**

The cloud provided an opportunity for saving the data in a different server thus reducing the burden of maintaining huge chunks of data. Through this the data is provided with the self-contained data protection mechanism. By utilising the basic RBAC model and through making modifications on ABE scheme and integrating it with ECP-ABE scheme the access

control capability and the usage of private key provided with the authenticity of the user. Attribute authenticator (AA) has been used in order to reduce the burden off the PKG. DC-RBAC which provided with a fine-grained access control capability. By combining DC-RBAC with ECP-ABE a self-contained data protection mechanism has been achieved for the data in the cloud. It can be thus concluded that RBAC-CPABE mechanism is a viable solution for providing security to the data in the cloud.

## REFERENCES

[1] *D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Advances in Cryptology–CRYPTO. California, USA: Springer Berlin Heidelberg, 19-23 August 2001, pp. 213–229.*

[2] *V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and communi-cations security. Alexandria, Virginia, USA: ACM, 30 October-3 November 2006, pp. 89–98.*

[3] *N. Attrapadung, B. Libert, and E. Panafieu, "Expressive keypolicy attribute-based encryption with constant-size ciphertexts," in Proceedings of the 14th International Conference on Practice and Theory in Public Key Cryptography. Springer, 2011, pp. 90–108.*

[4] *Long Li, Tianlong Gu, Liang Chang "A Ciphertext-Policy Attribute-Based Encryption Based on an Ordered Binary Decision Diagram", IEEE access, vol.5, pp.1137-1145, Jan 2017.*

[5] *Y. Zhu, D. Huang, C. J. Hu, and X. Wang, "From rbac to abac: Con-structing flexible data access control for cloud storage services," IEEE Transactions on Services Computing, vol. 8, no. 4, pp. 601–616, July 2015.*

[6] *D. Ferraiolo and R. Kuhn, "Role-based access control," in 15th National Computer Security Conference. Baltimore, Maryland: Na-tional Institute of Standards and Technology, 13-16 October 1992,p. 554lC563.*

[7] *J. Crampton, "Cryptographic enforcement of role-based access control," in Formal Aspects of Security and Trust. Pisa, Italy: Springer Berlin Heidelberg, September 16-17 2011, pp. 191–205.*

[8] *A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Advances in Cryptology–EUROCRYPT 2005, vol. 3494. Aarhus, Denmark: Springer Berlin Heidelberg, 22-26 May 2005, pp. 457– 473.*

[9] *V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and communi-cations security. Alexandria, Virginia, USA: ACM, 30 October-3 November 2006, pp. 89–98.*

[10] *J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proceedings of IEEE Symposium on Security and Privacy. IEEE, 2007, pp. 321–334*