Achieving Flexible and Self-Contained Data Protection in Cloud Computing

K.Srinivasulu[1], A.Mallikarjuna[2], S.Ramakrishna[3]

[1]*Pg Student, Department Of Computer Science, S.V University, Tirupati – India*

[2]*Teaching Assistant, Department Of Computer Science, S.V University, Tirupati -India*

[3]*Professor, Department of Computer Science, S.V University, Tirupati- India*

Abstract- **For enterprise systems running on public clouds in which the servers are outside the control domain of the enterprise, access control that was traditionally executed by reference monitors deployed on the system servers can no longer be trusted. Hence, a self-contained security scheme is regarded as an effective way for protecting outsourced data. However, building such a scheme that can implement the access control policy of the enterprise has become an important challenge. In this paper, we propose a self-contained data protection mechanism called RBAC-CPABE by integrating role-based access control (RBAC), which is widely employed in enterprise systems, with the ciphertext-policy attribute-based encryption (CP-ABE). First, we present a data-centric RBAC (DC-RBAC) model that supports the specification of fine-grained access policy for each data object to enhance RBAC's access control capabilities. Then, we fuse DC-RBAC and CP-ABE by expressing DC-RBAC policies with the CP-ABE access tree and encrypt data using CP-ABE. Because CP-ABE enforces both access control and decryption, access authorization can be achieved by the data itself. A security analysis and experimental results indicate that RBAC-CPABE maintains the security and efficiency properties of the CP-ABE scheme on which it is based, but substantially improves the access control capability. Finally, we present an implemented framework for RBAC-CPABE to protect privacy and enforce access control for data stored in the cloud.**

Index Terms- **Role-based access control, ciphertext-policy attribute-based encryption, self-contained data protection, cloud computing**

I. INTRODUCTION

IN cloud computing, an increasing number of enterprises and organizations use cloud servers as their system platform. Today, role-based access control (RBAC) model is the most popular model used in enterprise systems; however, this model has severe security problems when applied to cloud systems. A classic RBAC model uses reference monitors running on data servers to implement authorization. However, the servers in the cloud are out of the control of enterprise domains and, therefore, must be considered untrusted by default. Hence, building an effective data protection mechanism for cloud-based enterprise systems has become a major challenge. Currently, encryption is the primary mechanism used in clouds to ensure data security. The Cloud Security Alliance (CSA) [1] suggests that an excellent method of increasing data security is to keep data encrypted both in transit and when stored within the cloud. Although classic encryption schemes such as public-key encryption and identity based encryption (IBE) [2] can ensure data confidentiality, they cannot enforce effective access control. However, if the encrypted data were to feature an internalized access policy and was able to authorize or deny users based on the access policy, then confidentiality and access control could be achieved by the data itself rather than having to rely on the untrusted cloud servers. This type of protection model, which is referred to as self-contained data protection in this paper, not only minimizes the reliance on the cloud servers but also prevents unauthorized data access and tampering during transmission. Therefore, self-contained data protection essentially gives data the ability to ensure its own security, and it is an effective mechanism to protect data in cloud. However, neither RBAC alone or classic public encryption—or even the combination of both techniques [3]–[5] can satisfy the requirements of self-contained data protection.

II. RELATEDWORK

Integrating RBAC with cryptography

The RBAC model was first proposed by Ferraiolo and Kuhn in 1992 [12] and was widely studied in the mid-1990s. The RBAC model introduced roles between users and permissions. Permissions are assigned to roles rather than users; users must be assigned to a role to gain the permissions assigned to that role. The RBAC model greatly simplified permission management; consequently, it has become themost widely used access control modelin the past few years. By developing different policies, RBAC can achieve the requirements of both discretionary access controls (DAC) and mandatory access controls (MAC). Some studies have focused on combining RBAC with various encryption schemes to protect data. Crampton [13] introduced a new characterization of RBAC policies, namely, using the partial order relation to describe the policies. This approach transforms RBAC policies into information flow policies; then, it uses cryptographic enforcements of the policies to construct a cryptographic RBAC mechanism. Zhuetal.[3]–[5]proposed role-keyhierarchymodel(RKH) consisting of a cryptographic RBAC model that can support role hierarchies. In RKH, each role corresponds to a unique role-key, and users are assigned an exclusive user-key associated with each role to which they belong. However, because users must maintain a private key corresponding to each role, this method increases the burden of key management for users especially when a user is assigned many roles.

ABE:

ABE is an extension of public-key encryption that allows users to encrypt and decrypt data based on attributes. The greatest advantage of ABE is that its encryption key and decryption key are not in a one-to-one relationship; an encryption key can correspond to multiple decryption keys. The underlying basis of ABE is a fuzzy identity-based encryption (FIBE) proposed by Sahai and Waters [6]. Goyal et al. [7] further developed FIBE and introduced the idea of KP-ABE, in which the ciphertext is associated with a set of attributes and the private key is associated with an access tree. Later, Be then court et al. [8] proposed the first CP-ABE scheme called the BSW scheme. CP-ABE reversed the idea in KP-ABE; in CP-ABE, the ciphertext is associated with an access tree while the private key is associated with a set of attributes. The originalABE schemes were proposed based on a tree structure that is relatively expressive and can support AND, OR and threshold operators (an (m,n)-threshold means a solution must satisfy at least m constraints among total n constraints; henceforth, we refer to an (m,n)-threshold as “threshold” for short). Subsequently, some approaches [16], [17] based on the Linear Secret Share Scheme (LSSS) were proposed. The expressive ability of LSSS nearly equals that of a tree structure except that each attribute can be used only once in a LSSS structure. There are also some schemes [18]–[20] that support only the threshold operator were proposed. In fact, the AND operator is an (n,n)-threshold; therefore, those schemes also can support AND operator. In addition to AND, OR and threshold operators, there are some more complex operators such as NOT and comparison operators (i.e., >,≥, < and≤) that are particularly useful in practice, but cannot be directly expressed.

III. PRELIMINARIES

CP-ABE Scheme:

InCP-ABE, the ciphertext is associated with an access policy, and the private key is associated with a set of attributes. If and only if the attributes in a user’s private key satisfy the access policy is the user able to decrypt the ciphertext successfully. The CP-ABE scheme consists of 4 algorithms: Setup, Keygen, Encrypt and Decrypt [8].

ECP-ABE:

Scheme ECP-ABE was proposed to improve the expressive ability of CP-ABE [10], [11]. By introducing extended leaf nodes into the access policy tree, ECP-ABE can support access policies involving complex operators including NOT, >,≥,< and≤ in addition to AND, OR and threshold. More specially, in the access policy tree of ECP-ABE, the original leaf node used in classic CP-ABE is replaced by an extended leaf node that has an operator node with at least two children. One of the children is referred to as an attribute name node; the others are referred to as attribute value nodes, as shown in Fig. 2 (a). The attribute name node and the attribute value node denote the attribute name and attribute value, respectively, that are associated with the operator.

The attribute described by an extended leaf node is called an extended attribute. Meanwhile, the range of the threshold value k of the extended leaf node is changed to less than 0 from the original value 1. Different values of k (k < 0) denote specific operators. The ECP-ABE scheme offers three operator types: • Comparison operators: >,≥, <,≤. • Interval operators: [ ], ( ), ( ], [ ). • Logical operator: NOT.

Security Model :

In the CP-ABE scheme, security under CPA is modeled by a game between a challenger and an adversary [7]. It includes the following six phases: • Init. The adversary sends the challenger an access policy tree T that he wants to challenge. • Setup. The challenger initializes the system to generate public parameters pk and master keys mk. Then he sends pk to the adversary. • Phase 1. The adversary is allowed to make private key requests for any attribute set w = {ai|ai∈W,ai / ∈ T} where W is the attribute universe in the system. Then, the challenger returns skw to the adversary. • Challenge. The adversary sends two equal length messages m0 and m1 to the challenger. The challenger chooses a random θ ∈ {0,1} and encrypts mθ with the access policy tree T. Then the ciphertext CT is returned to the adversary. • Phase 2. The same as Phase 1. • Guess. The adversary outputs a guess θ′ ∈{0,1}.

IV.DATA-CENTRIC RBAC MODEL

Main Idea:

The RBAC model simplifies the management of user permissions in a system. However, as mentioned in Section 1, in the context of self-contained data protection, the RBAC model needs to be able to describe fine-grained access policies that are appropriate to specific data and support arbitrary constraints. In other words, data owners should not only be able to specify access policies for data objects at the role-level but also define other necessary constraints. To meet these requirements, a data-centric RBAC (DCRBAC) model is needed. The DC-RBAC model should support role assignments, inheritance and constraints. It may appear that DC-RBAC is quite similar to RBAC3 which is a consolidation of RBAC1 and RBAC2. However, constraints in DC-RBAC and RBAC3 are quite different.The constraints in RBAC3 roughly include 4 cases: (1) mutually exclusive roles (i.e. separation of duties); (2) cardinality constraints (i.e. limiting the number of users assigned to a role and the number of roles assigned to a permission); (3) prerequisite constraints (i.e., a user can be assigned to a role A only if that user is already assigned to role B, and permission p can be assigned to a role A only if role A already possesses permission q); and (4) constraints associated with sessions, such as the number of sessions that a user can have active at the same time. Clearly, RBAC3 defines its policies at the system level to manage user’s privileges for multiple dataobjects.Itsgoalistoprotectthesecurityofthewholesystem. In DC-RBAC, the situation is different—the security objective of the system is achieved by protecting each data object. Therefore, the security requirement of each data object becomes the basis of a DC-RBAC policy.

STRUCTURE OF DC-RBAC :

The DC-RBAC model consists of five sets of entities called data (D), users (U), roles (R), user attribute constraints (Ac) and environment constraints (Ec), as shown in Fig. 3. data represents a data object that needs to be protected. users are human beings who want to access the protected data. roles, user attribute constraints and environment constraints together constitute the access policy of the data.

There are also two parts called user intrinsic attributes (Att(U)), which indicates a user’s intrinsic attribute information, and user environment information (Env(U)), which indicates the contextual information of the user’s environment, that correspond to the user attribute constraints (Ac) and the environment constraints (Ec), respectively, as indicated by the dashed arrows from Ac to Att(U) and Ec to Env(U).

V.CONSTRUCTION OF RBAC-CPABE

Motivation :

The self-contained data protection mechanism requires that data carry its own access policy and be capable of implementing authorization according to that policy. DC-RABCis an access control model that can enforce data-centric, flexible and fine-grained role-based access control. However, the model can not give data the ability to authorize users entirely by itself; access policy verification may still require the help of other parties. Hence, it is necessary to build a mechanism that can eliminate the dependence on third-party servers. At present, encryption is the primary mechanism to achieve data self-protection, and CP-ABE provides the possibility for integrating encryption and access control. By fusing DC-RBAC into CP-ABE, data can be encrypted with the access policy of DC-RBAC and the policy can be verified during decryption. Only those users whose attributes satisfy the DC-RBAC access policy will be able to decrypt the ciphertext. Therefore, we integrate DC-RBAC with CP-ABE and construct the RBAC-CPABE scheme, which provides a feasible way to achieve self-contained data protection. A CP-ABE scheme that successfully supports DC-RBAC must meet the following requirements: (1) It must support role inheritance (e.g. a senior role can inherit permissions from its successor roles). A role inheritance tree will be defined in advance to indicate the hierarchy relationships. (2) It must support policies containing AND, OR, threshold, NOT, comparison operators and so forth because the constraints of DC-RBAC policy may contain such complex operators. The ECP-ABE scheme proposed by Lang et al. [10], [11] can handle any type of complex operator and can be extended to support role inheritance easily. Therefore, we choose to integrate ECP-ABE with DC-RBAC to construct the self-contained data protection scheme RBAC-CPABE.

Expressing DC-RBAC policy with ECP-ABE:

To construct RBAC-CPABE, two problems must be solved. The first problem involves how to support role assignment in ECP-ABE. Because role assignment includes role inheritance, it should be expressed as an extended attribute. Although negative assignment (i.e. role! = R) can be expressed by reusing the NOT operator, there is no suitable extendedle a node that can express positive as signment (i.e. role = R). The second problem involves how to express a DC-RBAC access policy (as described in Section 4.2) using the extended tree of ECP-ABE. This is necessary because DC-RBAC and ECP-ABE have different policy models. To solve these problems, we first define a new threshold value for the operator node in ECP-ABE so it can support role assignment. Then, we present a policy mapping model totransform a DC-RBAC policy in to an equivalent extended tree form.

VI. IMPLEMENTATION OF RBAC-CPABE

To investigate the application of RBAC-CPABE, we present an implemented framework for this scheme. The framework is based on the model of the RBAC-CPABE scheme (see Fig. 7), which contains three parts: PKG, the encryption party and the decryption party. To reduce the computational burden and avoid PKG becoming an efficiency bottleneck, we introduce the Attribute Authority (AA), which assumes part of the work of a traditional PKG. To ensure secure communication, the sender should sign a message and the receiver should verify the sender's signature before responding to the request. In this framework, we use the IBE [2] scheme to sign and verify the identity. IBE does not require complex distribution and management of private keys, and the public parameters and private keys can be generated by PKG. Computations on the tree structure and pairing operations in CP-ABE cause its efficiency to be lower than that of symmetrical encryption schemes. To improve the efficiency, we use a hybrid encryption method that includes the advanced encryption standard(AES) and RBAC-CPABE. The implemented framework of RBAC-CPABE is illustrated in Fig. 9. The framework can be divided into three parts: the cloud server space, which is used to store the protected data; the user space, which contains encryption and decryption users of the community; and the trust center space, which contains trusted servers that are responsible for managing users' attributes and generating private keys.

VII. CONCLUSIONS:

To address the data protection problem in cloud computing, we propose and implement a role-based self-contained data protection scheme called RBAC-CPABE .Based on the classic RBAC model, we first propose a data-centric access control model, DC-RBAC, which allows the data owner to specify individualized RBAC policies for each data object. Besides role-level constraints, DC-RBAC also contains user attribute constraints and environment constraints, which correspond to information about the authorized users and contextual information about the environment, respectively. Hence, DC-RBAC achieves more flexible and fine-grained access control. Next, to construct the self-contained data protection mechanism, we fuse the DC-RBAC into ECP-ABE by extending ECP-ABE and defining a policy mapping model. By using RBAC-CPABE, information contained in the data itself determines whether users are authorized to perform decryption instead of relying on other parties. Besides ECPABE, RBAC-CPABE also can be constructed based on other tree-based ABE scheme to achieve the specific functionality of the ABE scheme. A security analysis and experiment results indicate that RBAC-CPABE does not add any security risk or computational overhead compared to the CP-ABE scheme on which it is based, but it substantially improves the access control capability. Hence, RBAC-CPABE can be used in clouds to achieve efficient protection for outsourced data.

REFERENCES

[1] C. S. Alliance. (2011) Security guidance for critical areas of focus in cloud computing v3.0.[Online]. Available: https://downloads. cloudsecurityalliance.org/initiatives/guidance/csaguide.v3.0.pdf

[2] 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.

[3] Y.Zhu, G.-J.Ahn,H.Hu, and H.Wang, “Crypto graphicrole-based security mechanisms based on role-key hierarchy,” in Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security. Beijing, China: ACM, 13-16 April 2010, pp. 314–319.

[4] Y. Zhu, H.-X.Hu, G.-J.Ahn, H.-X.Wang, and S.-B. Wang, “Provably secure role-based encryption with revocation mechanism,” Journal of Computer Science and Technology, vol. 26, no. 4, pp. 697– 710, 2011.

[5] Y. Zhu, G. J. Ahn, H. Hu, D. Ma, and S. Wang, “Role-based crypto system: A new crypto graphicrbac system based on role-key hierarchy,” IEEE Transactions on Information Forensics and Security, vol. 8, no. 12, pp. 2138–2153, 2013.

[6] 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.

[7] 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 communications security. Alexandria, Virginia, USA: ACM, 30 October-3 November 2006, pp. 89–98.

[8] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute-based encryption,” in IEEE Symposium on Security and Privacy. Berkeley, CA: IEEE, 20-23 May 2007, pp. 321–334.

[9] Y. Zhu,D.Huang,C.J.Hu ,and X.Wang, “From rbactoabac: Constructing flexible data access control for cloud storage services,” IEEE Transactions on Services Computing, vol. 8, no. 4, pp. 601–616, July 2015.

[10] B. Lang, R. Xu, and Y. Duan, “Extending the ciphertext-policy attribute based encryption scheme for supporting flexible access control,” in Proceedings of the 10th International Conference on Security and Cryptography. Reykjavik, Iceland: IEEE, 29-31 July 2013,

[11] “Self-contained data protection scheme based on cp-abe,” E-Business and Telecommunications, vol. 456, pp. 306–321, 2014.

[12] D. Ferraiolo and R. Kuhn, “Role-based access control,” in 15th National Computer Security Conference. Baltimore, Maryland: National Institute of Standards and Technology, 13-16 October 1992, p. 554´lC563.

[13] 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.

[14] L. Zhou, V. Varadharajan, and M. Hitchens, “Enforcing role-based access control for secure data storage in the cloud,” The Computer Journal, vol. 54, no. 10, pp. 1675–1687, 2011.

[15] C. Hong, Z. Lv, M. Zhang, and D. Feng, “A secure and efficient role-based access policy towards cryptographic cloud storage,” in 12th International Conference on Web–Age Information Management, vol. 6897. Wuhan, China: Springer Berlin Heidelberg, 14-16 September 2011, pp. 264–276.

[16] J.J.Pfeiffer,J.Neville,andP.N.Bennett,‘‘Overcomi ngrelationallearning biases to accurately predict preferences in large scale networks,’’ in Proc. 24th Int. Conf. World Wide Web, 2015, pp. 853–863.

[17] L. K. Mcdowell, ‘‘Relational active learning for link-based classification,’’ in Proc. IEEE Int. Conf. Data Sci. Adv. Anal., Oct. 2015, pp. 1–10.

[18] A. Kuwadekar and J. Neville, ‘‘Relational active learning for joint collective classification models,’’ in Proc. Int. Conf. Mach. Learn. (ICML), Bellevue, WA, USA, Jun./Jul. 2012, pp. 385–392.