Volume 6, Issue 10, October 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Role Based Access Control Model for Cloud Computing Using RBE Scheme

Sneha Warang*
Student M.E. (Computer Engineering),
Terna Engineering College, Nerul, University of Mumbai,
Mumbai, Maharashtra, India

Tabassum Maktum
Asst. Professor (Computer Engineering)
Terna Engineering College, Nerul, University of Mumbai,
Mumbai, Maharashtra, India

Abstract— Cloud computing is a one type of computing that relies on sharing computer resources rather than having local servers or personal devices to handle applications. Nowadays, organizations use cloud services for data storage and its daily operations. Despite of various advantages security is the major concern for cloud computing. One of the security issues is how to control and prevent unauthorized access to data stored on the cloud. There are various techniques presented in literature to control unauthorized access to data. One such technique is RBAC (Role Based access Control) model. RBAC method controls the access to data based on roles given to individual users within an organization. RBAC model provides flexible control and management using two simple mappings first is User to their role in organization and second is Roles to accessible data to that Role. We propose RBE (Role Based Encryption) scheme which combines encryption technique with traditional RBAC model. In this system role hierarchy is used for efficient user management.

Keywords— Encryption, Cloud, RBAC, RBE Scheme, Security

I. INTRODUCTION

Nowadays, this Cloud computing provides users and enterprises various capabilities to store and process their data in third-party data centres that may be located far from the user ranging in distance from across a city to across the world. The public cloud is available to any user. The public cloud provides facility to user who want to use it can use in pay-as-you-go manner. In public cloud, the administrator of the cloud provider themselves would be able to access the data if it is stored in plain format. To protect the privacy of data, the data owners use cryptographic technique to encrypt the data. The users access the data according to their roles and their permissions.

In this research paper we have addressed the issue of storing the data on public cloud securely [1]. The public cloud is distributed geographically. It is formed by two or more data centres. User does not know that where the actual data is stored and there is a strong perception that user have lost control over the data after it is uploaded to the cloud. In order to provide the control to the user for their data which is stored in the public cloud some suitable access control and mechanism is required and policies must restrict data access to only those user intended by the owner of data.

In RBAC system data access is provided to the user according to their role. The roles are mapped to access permissions and users are mapped to appropriate roles. The Administrator assigned the roles to users based on their responsibilities and qualifications in their organization. Permissions are assigned to roles as per their qualifications instead of users. In RBAC, role hierarchy structure is used. The roles can inherit permissions from other roles. The RBAC system provides flexible control and management by having two mappings of user to role and roles to privileges on data objects.

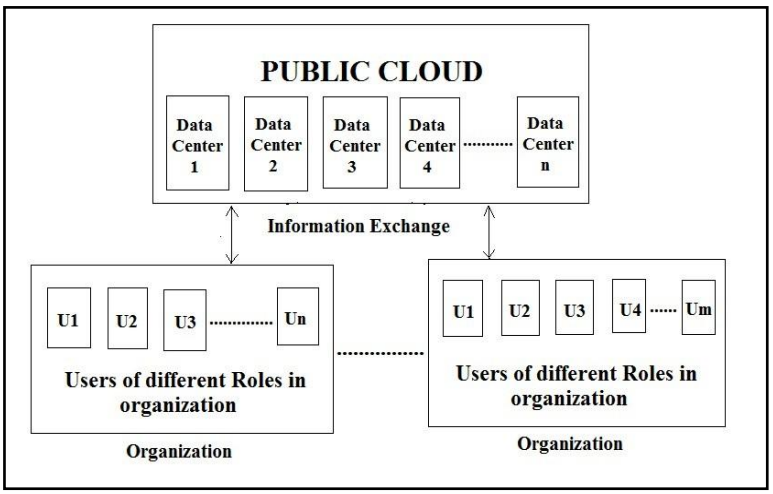

Fig 1: Public Cloud

This RBE scheme enforces RBAC policies on encrypted data stored in the cloud with an efficient user revocation using broadcast encryption mechanism described in [2]. In proposed RBE scheme, the owner of the data encrypts the data in such a way that only the users with appropriate roles as specified by a RBAC policy can decrypt and view the data. The role grants permissions to users who qualify the role and can also revoke the permissions from existing users of the role. The cloud provider (who stores the data) will not be able to see the content of the data if the provider is not given the appropriate role.

Proposed RBE [3] scheme is able to deal with role hierarchies, whereby roles inherit permissions form other roles. A user is able to join a role after the owner has encrypted the data for that role. The user will be able to access that data from then on, and the owner does not need to re-encrypt the data. A user can be revoked at any time in which case, the revoked user will not have access to any future encrypted data for this role. With our new RBE scheme [3], revocation of a user from a role does not affect other users and roles in the system.

In this architecture users who share and access the data only interact with public cloud. There is no access for the external users. We have developed a secure cloud storage system using new RBE scheme. The most frequently used operations are encryption and decryption of data.

II. LITERATURE SURVEY

Role based access control is invented in 1970's and only limited forms of access constraints based on the user's role within an organization. The role based system was relatively simple and application specific. This model is modified in 1992. Here a formal definition of role as a set of permissions, role hierarchies, subject role activation, subject-object mediation, as well as constraints on user/ role membership and role activation is presented [4]. In 1994, a role graph model for RBAC system is developed, analysing role relationship developed efficient algorithm [5]. After that in 1995, the RBAC model with formal definitions of static separation of duty and dynamic separation of duty has been presented [6]. In 1996, RBAC model is implemented with traditional multilevel security policies [7]. In 1998, a multilevel secure system is implemented in which the role hierarchy is a tree rather than a partial order [8]. This model provides a method of implementing DAC using RBAC. In 1999 and 2000, open source prototype RBAC for web servers is developed and RBAC standards are proposed.

Initially, the first access control problem is transformed into the key management problem. Here the target is achieved by using the Hierarchical key management (HKM) i. e. data storage in RBAC policies is achieved using HKM. The access control schemes are based on the hierarchical key management. The HKM technique applied to many hierarchies and used to achieve fastest key derivation. The limitation is if many users are involved then overhead is incurred in setting up the key infrastructure [9-11].

The approach for key management is Hierarchical ID-Based Encryption (HIBE). In HIBE, Private Key Generation (PKG) distributes the workload and identity authentication to lower level PKG. The advantage of HIBE is practically total collusion resistance and secure against chosen- cipher text attack. The drawback of this system is that length of the identity becomes longer with the growth in the department of hierarchy [3-12].

In [13-14] the Role Based Encryption (RBE) scheme is proposed. In RBE the user revocation is achieved by updating 9all the role related parameters. The drawback is user revocation may result in updation of all roles related parameters.

The alternate approach to RBE is Attribute Based Encryption (ABE). In ABE data encrypted in attribute form, which can be access by only those users who have the private key associated with it. The limitation of the ABE scheme is size of the user key is not constant and the revocation of a user results in key update of all the users of same role [15-16].

In present techniques there are some issues:

1) The user revocation is inefficient.
2) The key management is difficult.
3) The decryption is inefficient.

We propose a methodology to overcome the issues of present techniques. Thus proposed system we implement an efficient user revocation, imp rove the encryption/ decryption time and improve the efficiency of the existing system.

Motivation

Nowadays organizations are moving to clouds. In organizations confidentiality and data theft protection has become important aspects. They need to protect their private data in public cloud. There are dozens of access control models proposed in literature. The DAC and MAC achieved the success in practice. The DAC controls access based on the identity of subjects. In DAC, information may be accessed by unauthorized users because there is no control on copies of objects. The MAC makes access control decision based on the security level of subjects and objects. The MAC deals with information flow and solves this problem by attaching security levels on both users and objects. All users are required to obtain certain clearance to access objects. To overcome issues in DAC and MAC system RBAC is proposed. RBAC provides access to users according to their role. For role management the role hierarchy is used. Role based hierarchy provides only permitted data access to individual role respectively. Hence maintains data security within the organization itself. It helps to reduce the insider attack.

III. PROPOSED SYSTEM

To design this proposed system we use the new technology or new algorithms for RBAC policy with public cloud. A public cloud is one cloud computing model, in which a service provider makes resources, such as applications

and storage, available to the general public over the Internet. The services of Public cloud may be free or offered on a pay-per-usage model. Data centres of public cloud are located are at different location even the user who uses the public cloud service don't know where his data is actually stored.

In this research paper we address the security issues of the previous system where it had role based access of uploaded and downloaded files but there was no encryption and decryption of these files. Hence this led to privacy problem of accessing the files on the public cloud.

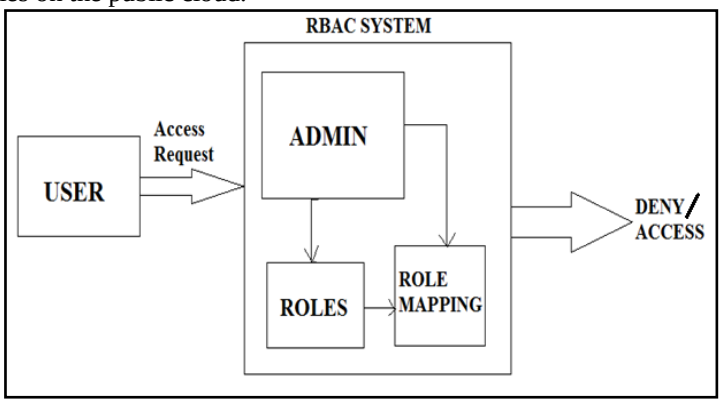

Fig 2: System Architecture

Public Cloud:

We know that the public cloud is un-trusted, because data centres of public cloud is located at different location we even don't know where our data is actually stored. Data stored in the public cloud could be accessed by unauthorized parties, such as employees of the cloud provider and users from other organizations who are also using services from the same cloud. An un-trusted public cloud may deny a user's request for accessing stored data in the cloud or provide users within correct data. Such behaviours will result in the users not being able to access the data stored in cloud, but will not cause violation of RBAC policies.

User:

Users are those who wish to access the data from the cloud which is uploaded by the owner or employee from the organization. The users are authenticated by the Admin. Which role have to provide to user and activate this user is depends on their eligibility criteria and system administrator. When user's authentication is successfully done, the user can access the data as per their permission.

Admin:

Admin will add different role and will generate id related to the role. Then Admin provides this role related information to cloud/ server. Only admin can access the role related information. Administrator will add the different user to different role which are generated by the System. He will be able to remove the particular user from the particular role. When administrator activate or deactivate the user will send the message to user via email.

Roles:

A role is a higher level representation of access control. Role is a mean for naming many-to-many relationship among individual users and permissions. Role includes a set of sessions where each session is a mapping between a user and an activated subset of roles that are assigned to users. In System, role hierarchy is used to assign the role to users.

Role Mapping:

The admin provides access of files to be downloaded or uploaded by the user depending on the roles of the user. It classifies the users according to their roles.

3.1 Algorithm:

In RBE System, following algorithms are used:

1) **Setup:** This algorithm takes public parameter as an input and generates master secrete key (mk) and public key (pk).
2) **ManageRole:** System Administrator (SA) executes this algorithm to manage Role with identity. Here role hierarchy is maintained. All the roles are stored in the system are defined as per their criteria. SA generates public parameter.
3) **AddUser:** SA executes this algorithm in which SA gives Role to user and also provides authentication. Role User List RUL_R is updated in cloud.
4) **RevokeUser:** This algorithm is executed by a SA. SA revokes the role membership from a user ID_U. The Role Public Parameter (pub_R) and Role User List (RUL_R) is updated in the cloud.
5) **Encrypt:** Encryption is done by the owner of the data. This algorithm takes role_id, public_key and point on the elliptic curve as an input and generates ciphertext of the message. The details of the encrypted data are stored on the cloud.

6) **Decrypt:** This algorithm executed by those users who possess access according to their role. This algorithm takes public_key, decryption_key, role_id and ciphertext of message and generates original message.

IV. IMPLEMENTATION OF SYSTEM

In this paper we address the issues of security of stored data. Here we use the public cloud to store the data. In our system RBAC policy is used. The roles are defined by the system according to their qualification and experience. Admin is the one authorized person who has the rights to activate users and assign the role as per their eligibility criteria. Users are the parties who want to store their data securely in the cloud. They want to access and decrypt the stored data in the cloud. Cloud is the place where data is stored and it provides interfaces so all the other entities can interact with it.

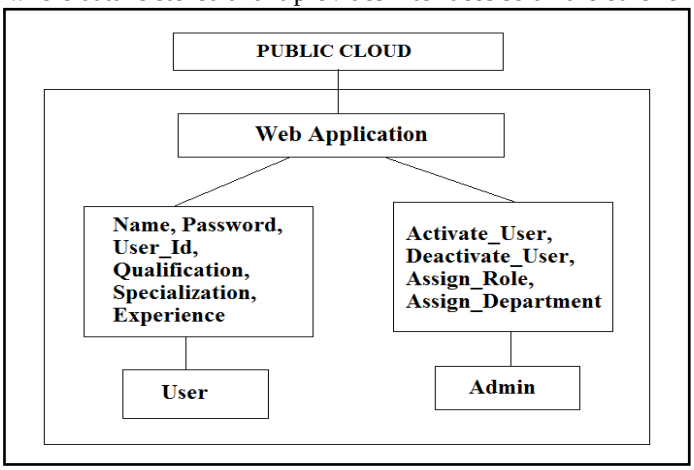

Fig 3: Access Application

Public Cloud Module:

In proposed system, user is the part of organization. System defines some role according to qualification, experience and specialization. Admin of organization will assign role, department and activate the user. Uploading and downloading of document is depends on the role of a user.

A. User

The owner initially enters all the parameters like owner name, mail_id, qualification, specialization, experience and password during registration.

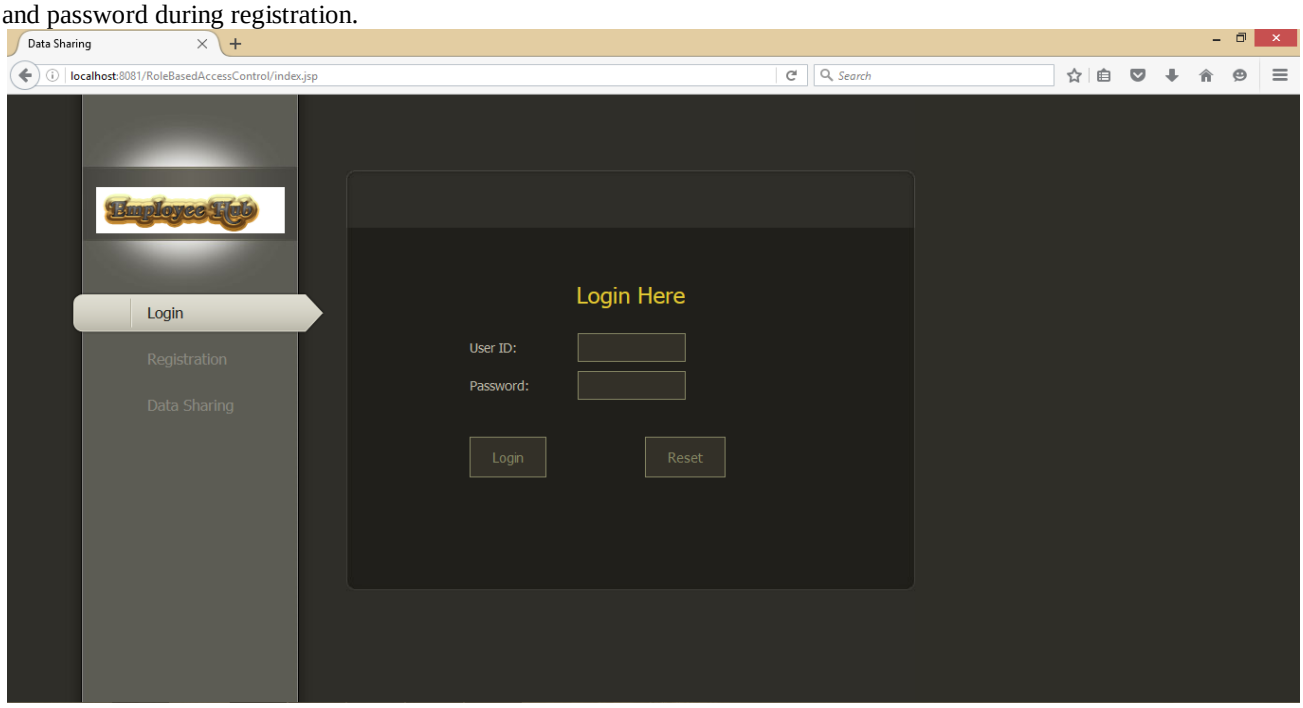

Fig 4: User Login

The upload and download facility is available to all users. But it works according to their role in the system. When the users upload the documents it is in the encrypted form. When owner/user uploads or downloads the file, he or she need not enter the parameters again since it is been automatically accessed and stored in the database by the application.

The upload and download facility is available to all users. But it works according to their role in the system. When the users upload the documents it is in the encrypted form. When owner/user uploads or downloads the file, he or she need not enter the parameters again since it is been automatically accessed and stored in the database by the application.

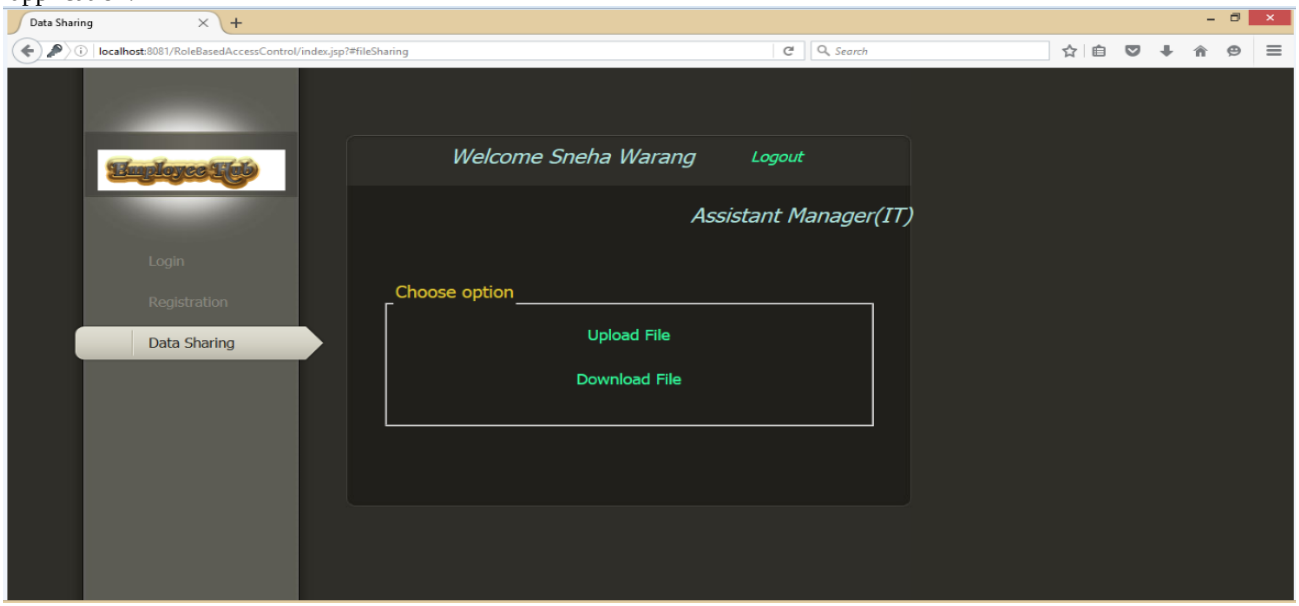

Fig 5: Data Sharing

All the information entered by the use will be verified at public cloud. If all the information which is passed or entered by user is true or verified then he will be given the decryption key of the file. If that user decryption key is not present then user will not be given the file decryption key.

B. Admin

The admin has a role of deciding whether to activate the registered user or not. The admin can activate and deactivate the users in the system. When activation and deactivation is done, the admin send the mail to user for as a confirmation. The updation regarding role of user and selection of department is done by the admin. The selection of role and department is decided as per their specialization, qualification and experience.

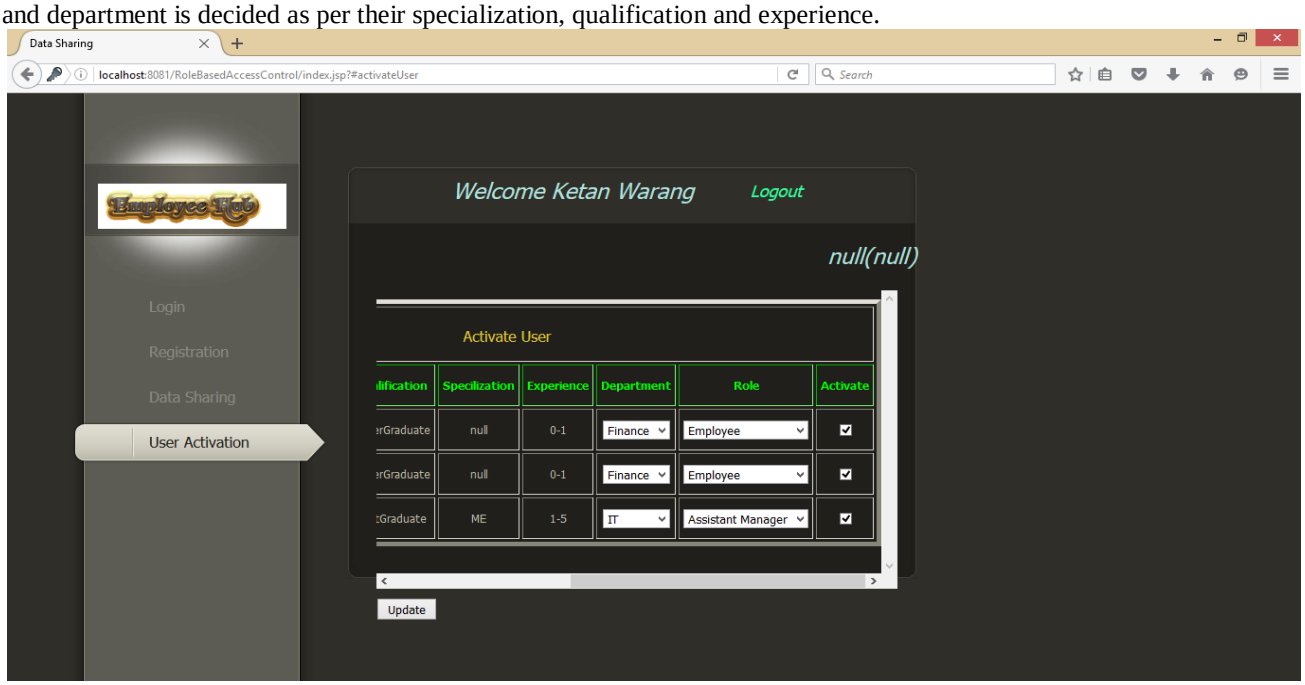

Fig 6: User Activation

V. APPLICATION

This policy can be implemented in any organization where role hierarchy plays an important role .The organization which wish to upload the document to the cloud with security .This policy provide the full security to the documents. This project can be used in colleges or company need to provide the access to the file to appropriate role and to user. As we know that there exists the different role and user in these organization and can be implemented easily.

VI. CONCLUSIONS

The cloud storage requires secure access control to preserve privacy of data. We propose a RBAC based model which allows an organization to store data securely in a public cloud. The proposed RBE model performs the user revocation and decryption operations efficiently. The proposed system combines RBE scheme with traditional RBAC model. The role hierarchy is used to improve efficiency of decryption and user revocation operations. Thus in this system we will provide the higher security than previous models.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. H. Katz, A.Konwinski, et Al., ―*A view of Cloud Computing* Common. ACM, vol. 53, no. 4, pp. 50-58 2010.

[2] Lan Zhou, Vijay Varadharajan, and Michael Hitchen ―"*Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage"*‖ IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 8, NO. 12, DECEMBER 2013.

[3] Y. Zhu, H.Hu, G. –J. Ahn, H. Wang and S.-B Wang, ―"*Provably secure role–based encryption with revocation Mechanism",*‖ Comput. JSci Techno vol 26, no. 4, pp. 697 -710, 2011.

[4] D.F. Ferraiolo and D.R. Kuhn (1992) *"Role Based Access Control"* 15th National Computer Security Conference, Baltimore, October 1992.

[5] M. Nyanchama and S.L. Osborn. *"Access rights administration in role-based security Systems"*, Proc. IFIP WG11.3 working conference on database security, 1994.

[6] D.F. Ferraiolo, J. Cugini, D.R. Kuhn, *"Role Based Access Control: Features and Motivations"*, Computer Security Applications Conference, (1995).

[7] Sandhu, R., *"Role Hierarchies and Constraints for Lattice Based Access Controls"*, Proc. Fourth European Symposium on Research in Computer Security, Rome, Italy, Sept. 25–27, 1996.

[8] D.R. Kuhn. *"Role Based Access Control on MLS Systems without Kernel Changes" Third ACM Workshop on Role Based Access Control*, October 22-23, 1998.

[9] S. D. C. Di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, *"Over- encryption: Management of access control evolution on outsourced data*", in Proc. VLDB, Sep. 2007, pp. 123–134.

[10] C. Blundo, S. Cimato, S. D. C. Di Vimercati, A. D. Santis, S. Foresti, S. Paraboschi, et al., *"Efficient key management for enforcing access control in outsourced scenarios*", in SEC (IFIP), vol. 297. New York, NY, USA: Springer-Verlag, May 2009, pp. 364–375.

[11] P. Samarati and S. D. C. di Vimercati, *"Data protection in outsourcing scenarios: Issues and directions"*, in Proc. ASIACCS, Apr. 2010, pp. 1–14.

[12] C. Gentry and A. Silverberg, *"Hierarchical ID-based cryptography*", in ASIACRYPT (Lecture Notes in Computer Science), vol. 2501. New York, NY, USA: Springer-Verlag, 2002, pp. 548–566.

[13] D. Boneh, X. Boyen, and E.-J. Goh, *"Hierarchical identity based encryption with constant size ciphertext*", in EUROCRYPT (Lecture Notes in Computer Science), vol. 3494. New York, NY, USA: SpringerVerlag, May 2005, pp. 440–456.

[14] L. Zhou, V. Varadharajan, and M. Hitchens, *"Enforcing role-based access control for secure data storage in the clouD,",* Comput. J., vol. 54, no. 13, pp. 1675–1687, Oct. 2011.

[15] 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. Sec., Oct./Nov. 2006, pp. 89–98.

[16] A. Sahai and B. Waters, "*Fuzzy identity-based encryption*", in Proc. EUROCRYPT, 2005, pp. 457–473.

