ISSN:0975-9646

Securing Multimedia on Hybrid Architecture with Extended RBAC

Gurpinder Kaur,
Research Student,
Department Of Information Technology,
Chandigarh University ,Gharaun, India.

Er. Monika Bharti,
Assistant Professor,
Department Of Computer Science Engineering,
Chandigarh University, Gharaun, India.

Abstract— Cloud computing has revolutionized the way computing and software services are delivered to the clients on demand. It offers users the ability to connect to computing resources and access IT managed services with a previously unknown level of ease. Due to this greater level of flexibility, the cloud has become the breeding ground of a new generation of products and services. However, the flexibility of cloud-based services comes with the risk of the security and privacy of users' data. Thus, security concerns among users of the cloud have become a major barrier to the widespread growth of cloud computing. In this paper the authors proposed and implemented the architecture for cloud where the cloud data storage is divided into two sections. In first part only the organization's sensitive structure information is to be stored and in the second part of cloud the actual data needs to be stored. This implemented architecture not only will dispel the organization's concerns about risks of leaking sensitive structure information, but will also take full advantage of public cloud's power to securely store large volume of data. Furthermore a cryptographic algorithm is to be employed with the use of two tier encryption. All data on public cloud is to be stored in encrypted form by employing cryptographic techniques which will save data from misuse and restrict data access to only those intended by the data owners.

Keywords- Cloud, Cryptography, algorithm, security, privacy, encryption.

I INTRODUCTION

The most cited definition of cloud computing is the one proposed by the US National Institute of Standards and Technology (NIST). NIST provides the following definition [1]: "Cloud computing is a model for enabling convenient, on demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." Cloud computing has become a successful and popular business model due to its charming features. In addition to the benefits at hand, the former features also result in serious cloud-specific security issues. The people whose concern is the cloud security continue to hesitate to transfer their business to cloud. Security issues have been the dominate barrier of the development and widespread use of cloud computing. The Cloud Security Alliance has summarized five essential characteristics [2] that illustrate the relation to, and differences from, traditional computing paradigm.

- On-demand self-service:

A cloud customer may unilaterally obtain computing capabilities, like the usage of various servers and network storage, as on demand, without interacting with the cloud provider.

- Broad Network Access:

Services are delivered across the Internet via a standard mechanism that allows customers to access the services through heterogeneous thin or thick client tools (e.g., PCs, mobile phones, and PDAs).

- Resource Pooling:

The cloud provider employs a multitenant model to serve multiple customers by pooling computing resources, which are different physical and virtual resources dynamically assigned or reassigned according to customer demand. Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.

- Rapid Elasticity:

Capabilities may be rapidly and elastically provisioned in order to quickly scale out or rapidly released to quickly scale in. From customers' point of view, the available capabilities should appear to be unlimited and have the ability to be purchased in any quantity at any time.

- Measured Service**:**

The service purchased by customers can be quantified and measured. For both the provider and customers, resource usage will be monitored, controlled, metered, and reported.

There are several main challenges for building a secure and trustworthy cloud system:

1 Outsourcing:

Outsourcing brings down both capital expenditure and operational expenditure for cloud customers. However, outsourcing also means that customers physically lose control on their data and tasks. The loss of control problem has become one of the root causes of cloud insecurity. To address outsourcing security issues, first, the cloud provider shall be trustworthy by providing trust and secure computing and data storage; second, outsourced data and computation shall be verifiable to customers in terms of confidentiality, integrity, and other security services. In addition, outsourcing will potentially incur privacy violations, due to the fact that sensitive/classified data is out of the owners' control.

2 Multi-Tenancy:

Multi-tenancy means that the cloud platform is shared and utilized by multiple customers. Moreover, in a virtualized environment, data belonging to different customers may be placed on the same physical machine by certain resource allocation policy. Adversaries who may also be legitimate cloud customers may exploit the co-residence issue. A series of security issues such as data breach [3], [4], [5], computation breach [3], flooding attack [6], etc., are incurred. Although Multi-tenancy is a definite choice of cloud venders due to its economic efficiency, it provides new vulnerabilities to the cloud platform. Without changing the multi-tenancy paradigm, it is imperative to design new security mechanisms to deal with the potential risks.

3 Massive data and intense computation:
Cloud computing is capable of handling mass data storage and intense computing tasks. Therefore, traditional security mechanisms may not suffice due to unbearable computation or communication overhead. For example, to verify the integrity of data that is remotely stored, it is impractical to hash the entire data set. To this end, new strategies and protocols are expected.
The remainder of the paper is organized as follows. Section II discusses the related work. In Section III, the authors describe assumptions on which this research article rely and tried to fill the gap in the later sections. Section IV presents the details and the flow of the major work to be covered in this architecture. The authors introduce the construction of the new cloud security model and architecture with two tier encryption. In Section V, the authors provide implementation of the proposed architecture. In Section VI the authors discuss results and comparison study of implemented architecture with other models. Finally, Section VII discusses future extensions and concludes the paper.

II Literature Review

In recent years, vendors have begun implementing role-based access control (RBAC) features in their database management system, security management, and network operating system products, without general agreement as to what constitutes an appropriate set of RBAC features. Several RBAC models have been proposed [7, 8, 9, 10, 11], without any attempt at standardizing salient RBAC features. To identify RBAC features that exhibit true enterprise value and are practical to implement, the National Institute of Standards and Technology has conducted and sponsored market analysis [12, 13], developed prototype implementations [14], and sponsored external research [15]. Several recent surveys [16], [17] show that 88% potential cloud consumers are worried about the privacy of their data, and security is often cited as the top obstacle for cloud adoption. Authorization and access control has always been a fundamental security technique in systems like cloud computing in which multiple users share access to common resources. Several access control models have been proposed, such as, discretionary and mandatory access control models (DAC and MAC), Clark-Wilson model, Lipner's Integrity model, Chinese wall model, Task based models, and Role Based Access Control models and RBAC has further been extended up to some level. Among these models Role-based access control (RBAC) models have been receiving attention as they provide systematic access control security through a proven and increasingly predominant technology for commercial organizations. One of the main advantages of the RBAC over other access control models is the ease of its security administrations. RBAC models are policy neutral [18]; they can support different authorization policies including mandatory and discretionary through the appropriate role configuration. In spite of the success of the RBAC, researchers have determined that there are still many application security requirements that are not addressed by the existing RBAC models [19]. Sandhu et al [20] proposed RBAC 96 which is a family of four constitutes models. In RBAC permissions are associated with roles (the intermediate concept of roles can be seen as collections of permissions), and users are made members of appropriate roles. The notion of role is an enterprise or organizational concept. The definition of role is quoted from Sandhu et al. [20]: A role is a job function or job title within the organization with some associated semantics regarding the authority and responsibility conferred on a member of the role. Permissions are not directly assigned to users; instead they are assigned to roles. RBAC comprise a family of four references models: RBAC0, RBAC1, RBAC2 and RBAC3. RBAC0 contains the core concepts of the Model. It is the minimum requirement for any system that exploits features of RBAC. Users (U), roles (R), and permissions (P) are three sets of entities and the relations between these entities are defined by User-Role Assignment and Permission-Role Assignment [20]. These sets and relations are the main concepts of the RBAC. A user can be member of many roles and each role can have many users. A user can invoke multiple sessions within a session a user can invoke set of roles but each session belongs to only one user. Permission can be assigned to many roles and a role can have many permissions. RBAC1 adds to RBAC0 a role hierarchy (RH). Role hierarchies are an important concept for structuring roles to represent organization users responsibly and degree of authority. RBAC2 introduces the concept of constraints. RBAC adds static (not related to sessions) and dynamic (related to sessions) constraints between core concepts [20]. These constraints are considered to be the principle motivation for RBAC because constraints are powerful mechanism to lay out higher-level organizational mechanism [20].Constraints can be applied to User-Role Assignment, Permission-Role Assignment and session. RBAC3 includes all aspects of RBAC0, RBAC1 and RBAC2 and it is called a unified model of RBAC. RBAC3 combine RBAC1 and RBAC2 to combine both role hierarchy and constraints. In this model constraints can be applied to the role hierarchy in addition to the constraints in RBAC2.
In the literature, there exist many hierarchy access control schemes [21, 22, 23] which have been constructed based on hierarchical key management (HKM) schemes, and approaches using HKM schemes to enforce RBAC policies for data storage are discussed in [24, 25, 26]. However, these solutions have several limitations. For instance, if there is a large number of data owners and users involved, the overhead involved in setting up the key infrastructure can be very high indeed. Furthermore, when a user’s access permission is revoked, all the keys known to this user as well as all the public values related to these keys need to changed, which makes these schemes impractical. An

alternative approach for the management of keys is Hierarchical ID-based Encryption (HIBE), such as [27], [28]. However, in a HIBE scheme, the length of the identity becomes longer with the growth in the depth of hierarchy. In addition, the identity of a node must be a subset of its ancestor node so that its ancestor node can derive this node's private key for decryption. Therefore, this node cannot be assigned as a descendant node of another node in the hierarchy tree unless the identity of the other role is also the super set of this node's identity. Recently we have seen the development of schemes built directly on RBAC policies.

III Problem Formulation

In a public cloud, as data can be stored in distributed data centers; there may not be a single central authority which controls all the data centers. Furthermore the administrators of the cloud provider themselves would be able to access the data if it is stored in plain format. Hence there is a need of enhancing data security by employing cryptographic techniques to encrypt data from misuse together with some hybrid cloud architecture by which virtue of which the privacy and security of private cloud can be achieved on one end and mass data storage feature of public clouds on other end.

This hybrid cloud architecture should turn to be a composite of private cloud and public cloud, where the private cloud should store only the organization's sensitive structure information such as the role hierarchy and user membership information, and the public cloud should store the actual data that is in the encrypted form and later all extended RBAC policies are to be employed on it to make more robust. In this hypothetical architecture, the users who will access the data only- interact with the public cloud; there is no access for public users to access the private cloud, which greatly reduces the attack surface for the private cloud. This architecture not only will dispel the organization's concerns about risks of leaking sensitive structure information, but will also takes full advantage of public cloud's power to securely store large volume of data.

IV Proposed Model:

To protect the privacy of the data, some measure needs to be designed by virtue of which data owners can employ cryptographic techniques to encrypt the data in such a way that only users who are allowed to access the data as specified by the access policies will be able to do so**.** The authorized users who satisfy the access policies will be able to decrypt the data using their private key, and no one else will be able to reveal the data content. For this to happen, the design of a secure RBAC based cloud storage system needs to be designed where the access control policies are enforced by a new role-based encryption that was mentioned earlier. This design should enforce RBAC policies on encrypted data stored in the cloud with an efficient user revocation using some broadcast encryption mechanism. In this proposed scheme, the owner of the data should encrypt 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 should grant 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. This scheme should deal with role hierarchies also, whereby roles inherit permissions from other roles. A user should be 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. Also user should be revoked at any time in which case, the revoked user should not have access to any future encrypted data for this role. Based on the proposed scheme, the authors need to develop a secure cloud data storage architecture using a hybrid cloud infrastructure. This hybrid cloud architecture should be composed of private cloud and public cloud, where the private cloud is used to store only the organization's sensitive structure information such as the role hierarchy and user membership information, and the public cloud is used to store the actual data that is in the encrypted form. In this architecture, the users who wish to share or access the data only interact with the public cloud; there is no access for public users to access the private cloud, which greatly reduces the attack surface for the private cloud. This architecture should not only dispel the organization's concerns about risks of leaking sensitive structure information, but also should take full advantage of public cloud's power to securely store large volume of data.

In this proposed model, the architecture is to be designed where administrator of the system can generate consoles for role manager and general clients. The role manager has to manage all the architectural aspects of the RBAC. All kinds of required roles and users creation for the system will be the main aspect to be covered by the role manager. Here all restrictions on users per role, add transaction limits for data usage, changing of permissions for roles are all covered by the role manager. In this architecture, the key management is added which will help the administrator to convert the data to be stored on the public cloud into the cipher text. This will result into privacy of this proposed hybrid architecture for which even Cloud Service Provider has to take membership from the administrator to access the data stored on the public cloud. Then users are to be generated per role and also access is provided to roles. Here restrictions are added for the generation of users per role. This is supported by adding restrictions on number of accesses per day over such roles. Once this designed architecture is developed and then used on windows Azure, after then the authors can make the authorization of any user in terms of login over such architecture. This will act as an access control for granting services for desired users. Next the challenge of unauthorized users is addressed in which a fake user will try to create a new id which is to be checked and denied as per our security policies

All these features are summed up in the following flow based figure 1 and 2:

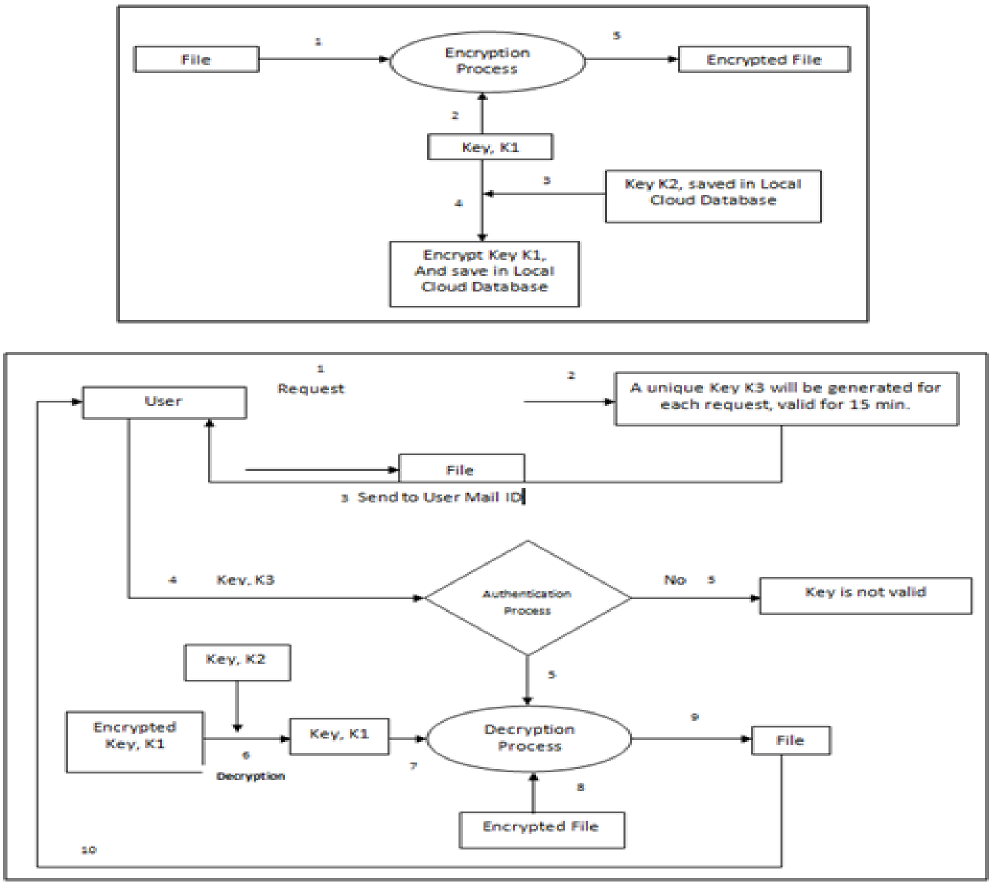

Figure 1: Two- way authentication algorithm for Encryption and Decryption

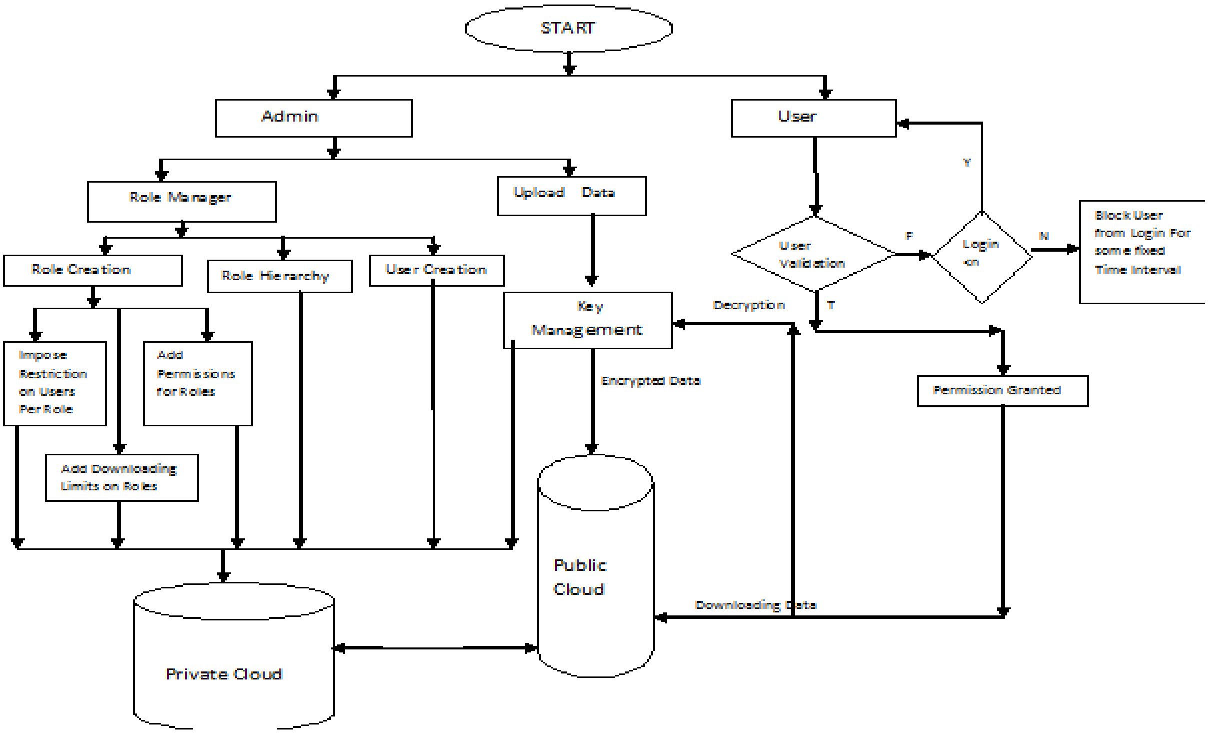

Figure 2: Flow Based diagram of Implemented Model

V IMPLEMENTATION

The authors have implemented the hybrid architecture for data storage with extended RBAC. The system is implemented in MVC Framework. Later this implemented web service is hosted on Microsoft's Azure cloud platform where the cloud use SQL database for its main storage of table contents which is to be stored on private cloud within an organization. Next the author uploaded the data in the form of cipher text on this third party Public Cloud via implemented interface for hybrid cloud architecture for whose storage the author created a storage account on Microsoft Azure Platform. All the mass storage data will be converted into cipher text and then uploaded on the cloud server. The alpha numeric keys are generated for encrypting the data files based on the GUID service of .Net framework. Next the Rijndael algorithm is used for encrypting the data files by using the generated keys. After then the data files are uploaded on the Public Cloud in cipher text form. Each of the uploaded file is encrypted by using separate keys. Once the file is encrypted by the generated alpha numeric key from the GUID service provided by the .NET platform, then the file is uploaded on the Public Cloud Server. However the key by which the author encrypted this data file need to be encrypted further by second generated key. After then the new second generated key is stored on private cloud server. Every time the user clicks on this file for downloading point of view, the second key is to be provided to him or her and not the original one. The time slice is applied on this mailed key and will remain active only for specific time interval. This will definitely save the architecture from the network attacks as anytime if any third party will try to hack this key and become successful, still they cannot afford to decrypt the required file.

The author has performed experiments on a machine with Intel (R) core 2 duo t5670 @ 1.80 GHz processor, 4 GB of RAM and Microsoft Windows 7 Home Premium 32 bit Operating System. Once the administrator of this hybrid cloud architecture makes successful login, then s/he will be redirected to the main interface of this hybrid architecture where s/he can now go for the creation of roles and then impose permissions on these created roles. Here at this time the administrator can decide how many users can be allowed to use this particular role in future and also the download limit is defined for these users. After then s/he can create users who can be made members for already created roles based on their accessing attributes. While creating the users, the administrator will decide that particular role to which this user is made a member and also the nature of accessing data is decided over here i.e. whether the permissions and downloading limit constraints are to be imposed on this user or not which will be decided on the premium membership to be purchased by this user. Here the administrator will generate the passwords for users as well by virtue of which users can make their login in future for accessing this architecture. Next the administrator will upload the data content which is to be stored in storage account already created on Windows Azure Cloud. Now authentic users can make login to this architecture to access the stored data on Microsoft Azure Cloud Storage account. Here if any unknown user will try to access this web interface and validation fails, then s/he can be warned in several attempts to go for authentic details and then in next attempt s/he can be blocked for some time specific time interval to be decided by the administrator. This is the added security mechanism in this architecture which will enhance the security mechanism for accessing the implemented architecture. Once making successful login the users can access and download the stored content on Cloud based on their permissions and restrictions imposed on them which were initially decided in admin interfaces of the architecture and moreover for downloading every data file on cloud, s/he needs to provide secret key for decrypting the data which will be already provided to that user in mailbox.

VI RESULTS:

The comparison study of this extended RBAC architecture is compared with other existing models to highlight the impact of this work in terms of following table structure:

Feature/Model	RBAC0	RBAC1	RBAC2	RBAC3	OURERBAC
Permissions on Roles	✓	✓	✓	✓	✓
Role Hierarchy	----	✓	✓	✓	✓
Role Constraints	----	---	✓	✓	✓
Limiting the users per Role	-----	---	---	✓	✓
Downloading Limit on Users	-----	-- -	---	---	✓
Varied Role Access based on Premium Membership	----	-----	------	-----	✓

Table I Results of Existing RBAC Models in Comparison to this model

The experimental study of this running system proved this architecture to be better in terms of constraints and the performance comparison of every activity to be shown as well in terms of the following Line chart:

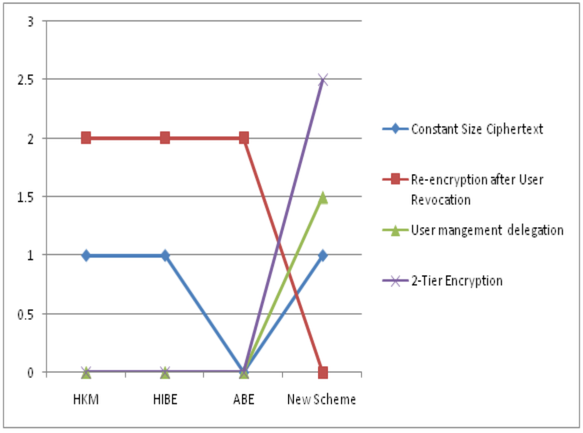

Figure 3: Performance comparisons in various models

VII Conclusion:

This implemented model has outlined a sketch for new RBAC which addresses the security features for any multi-centric application. Then the authors implemented a RBAC based hybrid cloud storage architecture which allowed an organization to store data securely in a public cloud, while maintaining the sensitive information related to the organization's structure in a private cloud. Then authors have implemented secure cloud storage system architecture and have shown that the system has several superior characteristics in terms of encryption and decryption key and later the authors implemented and applied the Extended RBAC for authentication on this hybrid architecture.

It is believed that the implemented system has the potential to be useful in commercial situations as it captures practical access policies based on roles in a flexible manner and provides secure data storage in the cloud enforcing these access policies.

References

[1] P.Mell and T. Grance, "The NIST Definition of Cloud Computing", 2011.

[2] Cloud Security Alliance (CSA). "Security Guidance for Critical Areas of Focus in Cloud Computing V2.1", Released December 17, [2009].

[3] Google Docs experienced data breach during March [2009].

[4] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds", Proc. 16th ACM conference on Computer and communications security, pp. 199-212. [2009]

[5] N. Santos, K.P. Gummadi, and R. Rodrigues, "Towards trusted cloud computing", Proc. 2009 conference on Hot topics in cloud computing, [2009]

[6] C. Dovrolis, P. Ramanathan, and D. Moore, "What do packet dispersion techniques measure?" In Proc. IEEE INFOCOM, pp. 905-914,[2001]

[7] D. Ferraiolo and R. Kuhn. "Role-Based Access Control." In Proc. of the NIST-NSA Nat. (USA) Comp. Security Conf., pp 554-563, [1992]

[8] M. Nyanchama and S. Osborn. "Access rights administration in role-based security systems". In J. Biskup, M. Morgenstern, and C. E. Landwehr, editors, Database Security, VIII: Status and Prospects, pages 37-56. North-Holland, [1994].

[9] D. Ferraiolo, J. Cugini, and R. Kuhn. "Role-based access control: Features and motivations". In Proc. of the Annual Computer Security Applications Conf., IEEE Press, [1995].

[10] L. Giuri and P. Iglio. "A formal model for role based access control with constraints". In proc. of the Computer Security Foundations Workshop, pp. 136-145. IEEE Press, [1996].

[11] R Sandhu, E. Coyne, H. Feinstein, and C. Youman. "Role-based access control models". IEEE Computer, 29(2), February 1996.

[12] D. Ferraiolo, D. Gilbert, and N. Lynch. "An examination of federal and commercial access control policy needs", In Proc. of the NIST-NSA Nat. (USA) Comp. Security Conf., pp 107-116, [1993]

[13] C. Smith, E. Coyne, C. Youman and S. Ganta,"A marketing survey of civil federal government organizations to determine the need for role-based access control security product", SETA Corp., July [1996].

[14] D. Ferraiolo, J. Barkley, and R. Kuhn. "A role-based access control model and reference implementation within a corporate internet." ACM Transactions on Information and System Security, 2(1), 1999.

[15] H. Feinstein. Final report: "NIST small business innovative research (SBIR) grant: role based access control: phase 2". SETA Corp., October [1996].

[16] F. R. Institute. (2010). Personal Data in the Cloud: A Global Survey of Consumer Attitudes [Online].

[17] (2010). *From* Hype to Future: KPMG's 2010 Cloud Computing Survey [Online].

[18] R. Sandhu. "Role hierarchies and constraints for latice-based access controls." In E. Bertino, H. Kurth, G. Martella, and E Monotolivo Eds. LNCS 1146, Proceedings of the European Symposium on Research in Computer Security 1996, Rome, Italy.

[19] E. Bertino, P. A. Bonati, and E. Ferrari, "TRBAC: A temporal role-based access control model", ACM Transactions on Information and System Security, 4(3):191-233, 2001.

[20] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-Based access control models", IEEE Computer, 29(2):38-47, 1996.

[21] S. G. Akl and P. D. Taylor, "Cryptographic solution to a problem of access control in a hierarchy", ACM Trans. Comput. Syst., vol. 1, no. 3, pp. 239–248, 1983.

[22] M. J. Atallah, K. B. Frikken, and M. Blanton, "Dynamic and efficient key management for access hierarchies", in Proc. ACM Conf. Comput. Commun. Sec., pp. 905-914.Nov. 2005.

[23] H. R. Hassen, A. Bouabdallah, H. Bettahar, and Y. Challal, "Key management for content access control in a hierarchy", Comput. Netw., vol. 51, no. 11, pp. 3197–3219, 2007.

[24] 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.

[25] 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, pp. 364–375, May 2009.

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

[27] C. Gentry and A. Silverberg, "Hierarchical ID-based cryptography," in ASIACRYPT, vol. 2501. New York, NY, USA: Springer-Verlag, , pp. 548–566 2002.

[28] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical identity based encryption with constant size ciphertext", in EUROCRYPT, vol. 3494. New York, NY, USA: Springer- Verlag, pp. 440–456 May 2005.