I.J. Education and Management Engineering, **2017, 3, 15-24**
Published Online May 2017 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2017.03.02

Available online at http://www.mecs-press.net/ijeme

Analysis of Access Control Methods in Cloud Computing

Madhura Mulimani [a], Rashmi Rachh [b]

[b] *Visvesvaraya Technological University (VTU), Machhe, Belgaum – 590018, India*

Abstract

Cloud Computing is a promising and emerging technology that is rapidly being adopted by many IT companies due to a number of benefits that it provides, such as large storage space, low investment cost, virtualization, resource sharing, etc. Users are able to store a vast amount of data and information in the cloud and access it from anywhere, anytime on a pay-per-use basis. Many users are able to share the data and the resources stored in the cloud. Hence, there arises a need to provide access to the data to only those users who are authorized to access it. This can be done by enforcing access control schemes which allow only the authenticated and authorized users to access the data and deny access to unauthorized users. In this paper, a comprehensive review of all the existing access control schemes has been discussed along with the analysis of these schemes.

Index Terms: Role-based access control, attribute-based access control, attribute-based encryption**.**

1. Introduction

Cloud computing is an emerging technology whose growth is on a rise and is being widely adopted by various IT conglomerate companies such as Google, IBM, Salesforce.com. It combines many technologies such as utility computing, grid computing, virtualization, etc. and leverages the advantages of these technologies to provide many benefits that include low investment cost, large storage, faster computations, virtualization, etc. Users store and share their data and information on the cloud and are able to access it from anywhere, anytime. One of the main benefits of cloud computing is users / organizations can make use of the resources on the cloud such as storage, infrastructure, computing power, etc., on a pay-per-use basis. This is especially beneficial for those organizations which cannot invest an enormous amount of money for the infrastructure. In cloud computing, there are a number of users using the resources or uploading their data to the cloud. Since the cloud service provider uses the multitenancy model [1], the data stored in it is accessible to multiple users. Thus, there is a high threat to the security of outsourced data in the cloud. Also, the cloud service providers and the data owners are most likely to be in different domains, making it necessary to provide security against these untrusted service providers. Each of these technologies has its own security mechanisms to ensure security and privacy of the user's data. However, security mechanism of one technology cannot be applied to cloud computing as a whole. Protecting the data from the malicious users in the cloud is of utmost

importance. Data can be secured and protected by ensuring that only the authenticated and authorized users access it. One of the solutions for providing security and privacy to the data is through the use of access control mechanisms. In this paper, we will provide an analysis of the various access control schemes that have been used earlier.

This paper has been structured as follows: In section II, the related work is described. In section III, various access control methods, their advantages and disadvantages have been discussed. In section IV, an analysis of the methods discussed in section III and their applications has been provided. Finally, the conclusion has been presented in section V.

2. Related Work

This section lists the various access control methods proposed by many authors for providing security to the data along with the challenges, and also, the solution proposed to overcome the challenges.

Kuhn *et al.* [2] have proposed to provide information security by assigning roles to the users and each role is assigned a collection of permissions. However, it has the disadvantages of difficulty in initial role structure setting, inflexibility as domains cannot change rapidly and no support for dynamic attributes. If it does support dynamic attributes, it might lead to role expansion resulting in the creation of thousands of roles. In order to solve this, attributes and rules in attribute-based access control can be used that do not require separate roles for sets of subject attributes. Though it is easy to setup and specify access rules, it is difficult to determine a particular user's permissions as it needs a large set of rules to be executed in exactly the same order as does the system.

Jin Li *et al.* [3] have addressed the issue of illegal key sharing among the colluding users. To solve this issue, they have defined and enforced access policies based on data attributes, and in addition, have also implemented user accountability using traitor tracing mechanism.

Ruj *et al.* [4] have proposed an access control scheme to preserve the privacy of the data providing a trustee's token only to the authenticated users, who will then be able to perform operations on the file (read, write, execute) in the cloud. In this case, the authentication and access control scheme is decentralized and robust unlike the other access control schemes that were designed for clouds which are centralized.

According to Goyal *et al.* [5], encryption of sensitive data stored in the cloud provides just the coarse-grained level access, when, in fact, a fine-grained level access control is required. Decryption is possible either when the user acts as an intermediary and decrypts all the entries for the party or gives the party its private decryption key. But, problems arise when setting the audit logs. To resolve this issue, Sahai and Waters [5] have introduced the concept of Attribute-Based Encryption (ABE). Goyal *et al.* [5] have developed a new cryptosystem, called Key-Policy Attribute-Based Encryption (KP-ABE) to provide a fine-grained sharing of the encrypted data.

Ruj *et al.* [6] have emphasized that preservation of the security of data and privacy of users is of utmost importance. Maintaining an access control list of all the valid users in a dynamic cloud environment becomes difficult. Usage of encryption might lead to the data getting encrypted several times, which incurs huge storage costs. To resolve this, they have provided a solution of using Attribute-Based Encryption (ABE) to achieve access control in clouds. They have also proposed the new distributed access control mechanism, in which the owners decide which attributes users should possess in order to access the data and users are provided with the decryption keys to access the records for which they have authorization rights.

3. Access Control Methods

Access control is a mechanism in which users may be granted or denied access to the data for the purpose of providing security and privacy to the data and protecting it from unauthorized and malicious users. Access control is concerned with whether a subject (any entity that can manipulate information), can access an object, (entity through which information flows through the actions of a subject) and how, in particular, this access can

occur [7].

One of the major areas where access control is used extensively is the medical health care, in which access to the sensitive information about the patients is granted only to the medical professionals, hospital staff, researchers and policy makers. Access control is also gaining lot of importance in online social networking. Other areas where it is being used are governments, airlines, telecommunication carriers, etc.

Access Control List (ACL):

In this mechanism, the names of all the registered users along with their access privileges to a particular system object are maintained in a list [8]. These system objects may be a file or a directory. The access control list is generally created by the system administrator or the object owner. So, any time a user requests the use of data or the resource from the cloud, the list is checked to verify whether the user is registered or not. If it is on the list, the user is granted permission to access the data or the resource. Some of the operating systems that use access control lists are Windows NT/2000, UNIX-based systems, Novell's Netware. Each of these operating systems uses a different implementation for the access control list.

Advantages: An object's ACL provides an easy way to determine modes of access that a subject is currently authorized for that object. Replacing the existing ACL with an empty one makes it easier to revoke all accesses to an object [9].

Disadvantages: It is difficult to determine all accesses a subject possesses for various objects in the system. During an access review with respect to a subject, examination of ACL of every object is necessary [9]. It can be used only in a static environment with a limited number of users. The cloud computing environment, being a large distributed system, cannot use the access control list method for access control, mainly because of the huge number of dynamic users, who join and exit the environment in a dynamic manner, a large number of resources and also, the flexible constructions of the networks.

Mandatory Access Control (MAC):

It is a system-wide policy which defines who is allowed to have access to the resources in the system. This mechanism relies on the system to control the access. So, an individual user cannot alter the access. In this mechanism, there exists a classification of objects and subjects. Any user requesting some resource is the subject and the resource being requested is the object. Each of the subjects and objects is assigned a security level. The security level helps in identifying the access state of the object. For an object, it reflects the sensitivity of the information contained in the object, whereas for a subject, it reflects the user's trustworthiness not to disclose sensitive information to users not cleared to see it. An object can be accessed by a subject only if some relationship is satisfied between the security levels associated with the two [8].

Advantages: It focuses on controlling disclosure of information by assigning security levels to objects and subjects. It limits access across security levels and consolidation of all classification and access control into system [7].

Disadvantages: MAC is not flexible, because once a security level is assigned to any subject in the hierarchy, it cannot be modified. This results in user frustration, as they cannot dynamically change the underlying access policies. Also, it is difficult and expensive to implement [8].

Discretionary Access Control (DAC):

This method is based on the concept of users having control over the system resources. The access control of the objects (e.g., the files and resources) in the system is left to the discretion of the object's owner, who determines the object access privileges and thus, specifies what permissions users of the same group may have and also what permissions all other users may have[8].

Advantages: It enables fine-grained control over system objects using which, DAC can easily implement

least access privileges. Being intuitive in implementation and mostly invisible to users, it is regarded as the most cost-effective method for home and small business users [7].

Disadvantages: Since the users are allowed to control object access permissions, this mechanism makes the system susceptible to Trojan Horse and also, system maintenance and security principles verification are extremely difficult for the DAC systems.

Role-Based Access Control (RBAC):

In this method, security policies are defined by granting access rights to roles rather than to individual users. Thus, the roles determine the user's access to the system based on the job. Roles are assigned to all the users by the system based on the concept of least privileges, i.e., the role is assigned with the least amount of permissions required for the job to be done [10], and each role is assigned a set of access privileges. If the privileges for a role changed, permissions could be added or deleted. The user would be authenticated by his identity and allowed access to the resources on the basis of the privileges assigned to the role which he possessed. This results in easier overall system maintenance. Also, it is very effective in the verification of security policies.

Advantages: Access control for many users is consolidated into a single role entry making it beneficial for easier overall system management. RBAC provides flexibility for a subject to have multiple roles or membership in multiple groups [11]. It has an integrated support for principle of least-privilege, separation of duties and central administration of role memberships and access controls [7].

Disadvantages: This method is suitable for a system with a limited number of users and roles and also, where the user's roles seldom change. However, extending this method across administrative domains, led to the difficulty in deciding a role's privileges. It does not consider the random and dynamic behavior of the users [11], making it unsuitable for use in cloud computing environment, which is dynamic in nature.

The identity-based access control (IBAC) methods, namely, ACL, MAC, DAC, and RBAC have sometimes been known as the authentication based control methods and require a tight coupling among domains. These methods provide coarse-grained access control, and are effective in distributed system, where there are only a set of users with a known set of services. Since the growth of the networks as well as the users is always on the rise, identity-based access control was found to lack the strength to support such a large development. Furthermore, IBAC was problematic for the distributed systems due to the difficulty in managing access to the system and the resources, and also, due to the susceptibility to errors. These were the shortcomings of traditional access control mechanisms. Access control using cryptographic methods is receiving greater attention, which is more flexible and robust as compared to the traditional access control mechanisms [12].

In order to provide fine-grained access control in a large, distributed and dynamic environment, such as, the cloud, the attribute-based access control (ABAC) was proposed.

Attribute-based Access Control (ABAC):

In ABAC, users are assigned attributes and access policies are formed by combining attributes to grant access to the data or the resources. Any user who needs to access the data must possess the set of attributes that satisfy the access policy defined. This ensures that ABAC is secure, scalable and flexible. Authentication of users takes place at the time of data access and at the time of request. Whenever a new user joins the cloud, the policies and the rules defined for accessing the data need not be modified as per the new user. If the user possesses the attributes that satisfy the access policy defined for data accessibility, permission is granted to the user to access the data, else, access to the data is denied.

In a way, ABAC is an extension of RBAC with features such as, delegation of attribute authority, decentralization of attributes and interference of attributes [10]. This makes ABAC more suitable for the cloud environment that consists of an enormous number of dynamic users, massive amount of storage, and also, dynamic and flexible constructions of networks.

ABAC consists of four entities, namely, the requestor, the resource, the service and the environment.

- *Requestor:* one who sends requests to the cloud and invokes actions on the service.
- *Resource:* one or more services act upon it.
- *Service:* software and hardware with a network-based interface and predefined set of operations.
- *Environment:* contains information that might be useful for taking access decisions.

Access policies are specified based on the attributes of all these four entities [13]. With this approach, the access control will be flexible enough to have multiple policies in multiple domains, which is, otherwise, not the case with traditional access control models, such as, the ACLs, which have their own security policy. In addition, it also provides scalability essential to large scale distributed system [10].

Advantages: This method is highly adaptable in a dynamic environment making it suitable to be used in cloud computing. It provides fine-grained access control and accountability.

Disadvantages: Though the policy configuration points in ABAC are increasing rapidly and are flexible, it leads to greater difficulty when expressing and comprehending the policy. It will require strong and comprehensive foundations for ABAC to flourish [14].

Role-based access control (RBAC) and Attribute-based access control (ABAC) use the cryptographic primitive known as Attribute-Based Encryption (ABE), which enables data and information to be encrypted under some access policy and then stored in the cloud. Here, users possess a set of attributes and are given the corresponding keys. Only those users having a matching set of attributes will be able to decrypt the data and the information stored in the cloud.

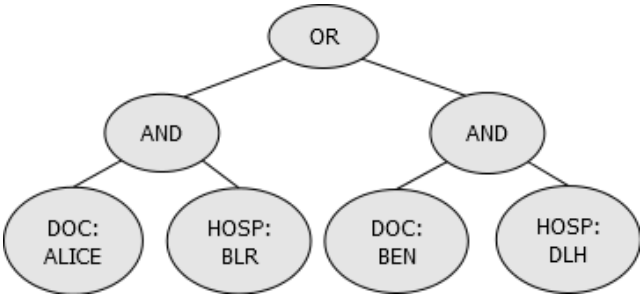

Fig.1. Example of Access Tree Structure in ABAC

Fig. 1 illustrates an access policy in the form of an access tree. An access control policy is a rule that is formulated as a Boolean formula over the attributes[12]. Data is encrypted using these access policies and it defines who can access a resource based on the attributes that a user possesses. It contains attributes and threshold gates such as, AND, OR, and n-of-m gates (where $n < m$). An access policy can be represented as an access tree in which leaves represent attributes and internal nodes represent logical gates, such as, AND, OR and n-of-m gates. Suppose Alice is a doctor (DOC) from Bangalore (Blr) hospital (HOSP) or Ben is a doctor (DOC) from Delhi (Dlh) hospital (HOSP). An access policy may specify that either Alice from Bangalore hospital or Ben from Delhi hospital may access a patient's health record. Such an access policy can be represented in the form of a Boolean function as:

$$(\text{DOC} = \text{Alice} \wedge \text{HOSP} = \text{Blr}) \vee (\text{DOC} = \text{Ben} \wedge \text{HOSP} = \text{Dlh})$$

where DOC and HOSP are the attributes and Alice, Blr, Ben, Dlh are the values for the attributes. The logical gates AND and OR are represented using the symbols $\wedge$ and $\vee$, respectively, in the Boolean function. In general, the access policy is of the form:

$$(a_1 \wedge a_2 \vee a_3) \wedge (a_1 \vee a_2 \vee a_3)$$

where a_1, a_2 and a_3 are the attributes and $\wedge$ and $\vee$ are the logical gates that aid in the formation of an access policy in attribute-based access control. The access policy indicates that any user possessing the attributes that satisfy the access structure specified is authorized to access the data in the cloud.

Attribute-Based Encryption (ABE):

ABE is an expansion of public key encryption that allows users to encrypt and decrypt messages based on user attributes. It comes in two complimentary forms, Ciphertext-Policy Attribute Based Encryption (CP-ABE) and Key-Policy Attribute Based Encryption (KP-ABE), depending on whether the access policy is associated with the ciphertext or the private key. ABE schemes have the advantages of providing a fine-grained access control [15]. ABE-based systems are well suited to provide user controlled-privacy, as users in these communities are already characterized by their attributes [16]. So, even if the storage on the server is compromised, the loss of information is minimal, because the access control policy is bound to the data and the users and the server no longer mediates access to files. Two main features of ABE are:

1. Capacity to address complex access control policies
2. No need to know the exact list of users in advance. Knowledge of the access policy is sufficient.

ABE must also satisfy the property of collusion resistance, meaning that even if multiple users collude, they should not be able to decrypt the ciphertext unless one of them was able to decrypt it completely by himself. This ensures only users with the right keys can have access to the information [17].

Ciphertext Policy Attribute Based Encryption (CP-ABE):

CP-ABE is a scheme in which attributes are associated with an access structure and secret key is associated with the ciphertext [5]. It is a promising cryptographic primitive for fine-grained access control of shared data and has several advantages such as security against ciphertext attacks, applicability to Key Policy Attribute Based Encryption (KP-ABE), size of the public key and the ciphertext being of the same size, etc.

This scheme is similar to RBAC and can be used for providing access control in many applications such as medical system. In this scheme, the access policy is determined by the data owner. Therefore, it is more suitable for access control applications that consist of four probabilistic polynomial time algorithms [18] as Setup, Encryption, Key Generation, and Decryption as shown in Fig. 2.

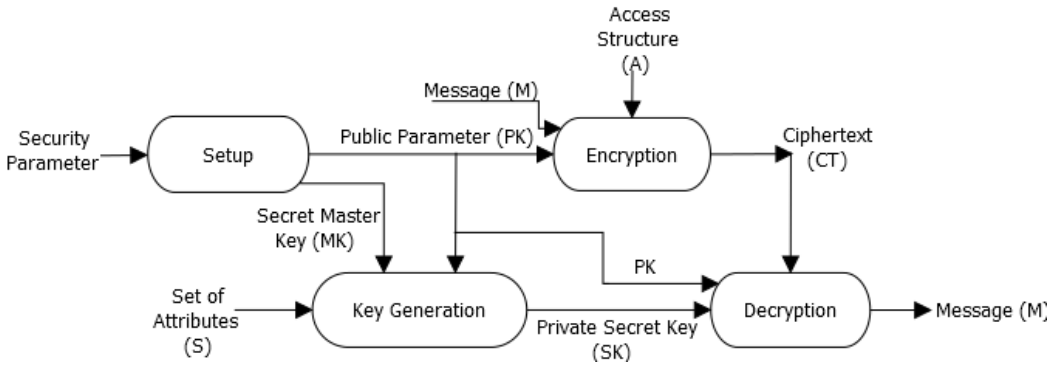

Fig.2. Ciphertext Policy Attribute Based Encryption (CP-ABE)

Fig. 2 can be described as follows: *Setup* algorithm takes as input a security parameter and produces a public key (PK) and a master secret key (MK). *KeyGen* algorithm takes as input the public key (PK), secret master key (MK) and a set of attributes (S) as input and produces a private secret key (SK) as output. The public key (PK), a message M, and an access structure (A) form the input to the *Encryption* algorithm and the output is a ciphertext (CT). *Decryption* algorithm takes as input the private secret key (SK), public key (PK) and the ciphertext (CT) as input and produces the message M as output [19].

Key Policy Attribute Based Encryption (KP-ABE):

KP-ABE is a public key cryptographic primitive for one-to-many communications [20]. In this scheme, each ciphertext is labeled by the encryptor with a set of attributes. Each private key is associated with an access structure, which specifies the type of ciphertext the key can decrypt. The KP-ABE is so named because the access structure is specified in the private key [5]. Each user is assigned an access structure that is usually defined as an access tree over data attributes. User is able to decrypt a ciphertext if and only if the data attributes satisfy his access structure [20].

This scheme also consists of four algorithms, namely, Setup, Encryption, Key Generation, and Decryption as shown in Fig. 3. KP-ABE finds its application in secure forensic analysis and pay-per-view TV system. The KP-ABE provides fine-grained data access control and efficient operations such as file creation/deletion and new user grant [5].

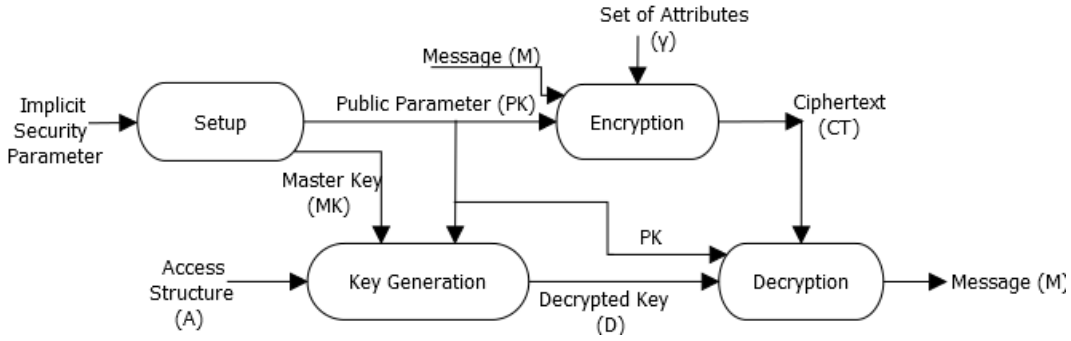

Fig.3. Key Policy Attribute Based Encryption (KP-ABE)

Fig. 3 can be described as follows: *Setup* algorithm takes as input a security parameter and produces a public key (PK) and a master key (MK). *KeyGen* algorithm takes as input the public key (PK), master key(MK) and an access structure(A) as input and produces a decryption key (D) as output. *Encryption* algorithm takes as input the public key (PK), a message M, and a set of attributes and produces the output as a ciphertext (CT). *Decryption* algorithm takes as input the decryption key(D), public key (PK) and the ciphertext (CT) as input and produces the message M as output.

4. Analysis

Based on the discussion presented in the previous section about the various access control methods, a comprehensive analysis of those methods has been provided in a tabular format (Table 1), to show who is responsible for providing access control in each of the methods. Along with the accountability information for each of the methods, their advantages and disadvantages and also, their applications have been listed out.

Table 1. Access Control Methods and Their Features

Method	Accountability	Advantages	Disadvantages	Applications
Access Control List (ACL)	System Administrator	Object's ACL makes is easy to determine modes of access a subject possesses, Revoking all accesses to an object is easier	Limited number of users, Not scalable, Not flexible	Windows NT/2000, UNIX-based systems, Novell's Netware
Mandatory Access Control (MAC)	System	Security levels assigned to objects and subjects controls disclosure of information	Limited number of users, Not flexible, Difficulty in implementation	Government and military applications or Mission critical data applications
Discretionary Access Control (DAC)	Data Owner	Focuses on fine-grained access control of objects through access control matrices and object level permits, Least access privilege implementation is easier	Susceptibility to Trojan Horse attacks, Difficulty in system maintenance and verification	Data-based Web applications
Role Based Access Control (RBAC)	Roles in the system	Subjects can have multiple roles or membership in multiple groups, Integrated support for principle of least privileges, Separation of duties, Central administration of role membership	Limited number of users and roles, Not scalable	Medical organizations, Academic institutions
Attribute-based Access Control (ABAC)	Attributes	Adaptable in dynamic environment, Provides fine-grained access control and accountability	Greater difficulty when expressing and comprehending the policy, Requires strong and comprehensive foundations to flourish	Government organizations, Health Care Systems, Airlines, Insurance, Telecommunications Carriers

The characteristics / features of the various access control mechanisms listed in Table 1 help in deciding which one would be a suitable access control method to be applied in a cloud computing environment.

Access control plays a very important role in providing security to the data stored in the cloud. The access control methods need to be flexible and scalable across multiple domains. In attribute-based access control, attributes are used to define the access policies and fine-grained access control is provided by granting differential access rights to a set of users. In other words, attributes take on the accountability and users are allowed access to the data depending on the attributes they possess. This makes the attribute-based access control mechanism independent of the system, system administrator, data owner or roles a user possesses. Users attributes themselves will be able to make the access decisions without knowing the user's identity. For example, any user who uploads a file on the cloud, may have an access policy which states that any user belonging to Academic department may access the data. In this case, the data owner who uploads the file, will not need to know the identity of the users who can access the data. This will help reduce the burden on the data owner. ABAC provides flexibility and scalability to the sensitive information in the cloud, along with

confidentiality and authentication of the users. From this discussion, it is clear that Attribute-based Access Control is most suitable to cloud environment.

5. Conclusion

Data in the cloud is prone to attacks by users as well as servers on which it is stored. Due to this reason, security and integrity of the sensitive information have been the major concerns among the cloud users. To provide security to the data, only authorized users are allowed to access the data, and it can be enforced through the access control mechanisms. A clear understanding of the access control methods helps in deciding which one is suitable in a cloud computing environment. In this paper, a comprehensive review of the various previous and current access control methods has been presented, from which, it can be stated that, attribute-based access control is suitable in a cloud environment. Attribute-based access control provides fine-grained access control by granting differential access rights to a set of users and allows flexibility in specifying the access rights of individual users. It is an emerging research area to provide security and integrity to the data and also, achieve fine-grained access control in a cloud environment.

References

[1] RajaniKanth Aluvalu, Lakshmi Muddana, "A Survey on Access Control Models in Cloud Computing", Springer International Publishing Switzerland 2015 S.C. Satapathy et al.(eds.), Emerging ICT for Bridging the Future – Vol. 1, Advances in Intelligent Systems and Computing 337, DOI: 10.1007/978-3-319-13728-5_73 pp. 653.

[2] D. Richard Kuhn, Edward J. Coyne, Timothy R. Weil, "Adding Attributes to Role-based Access Control", IEEE Computer Society, vol. 43, No. 6, pp. 79 – 81, (June 2010).

[3] Jin Li, Gansen Zhao, Xiaofeng Chen, Dongqing Xie, Chunming Rong, Wenjun Li, Lianzhang Tang, Yong Tang, "Fine-grained Data Access Control Systems with User Accountability in Cloud Computing", Proc. 2nd IEEE International Conference on Cloud Computing Technology and Science, IEEE Computer Society, (2010).

[4] Sushmita Ruj, Milos Stojmenovic, Amiya Nayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds", 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 556 – 563.

[5] Vipul Goyal, Omkant Pandey, Amit Sahai, Brent Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data", ACM CCS'06, Alexandria, Virginia, USA, (October 30 – November 3, 2006).

[6] Sushmita Ruj, Amiya Nayak and Ivan Stojmenovic, "DACC: Distributed Access Control in Clouds", International Joint Conference of IEEE TrustCom – 11, IEEE Computer Society, pp. 91 – 98, (2011)

[7] Ryan Ausanka-Crues, "Methods of Access Control: Advances and Limitations".

[8] Natarajan Meghanathan, "Review of Access Control Models for Cloud Computing", David C. Wyld (Eds): ICCSEA, SPPR, CSIA, WimoA – 2013, pp. 77–85, 2013.

[9] Ravi S. Sandhu and Pierangela Samarati, "Access Control: Principles and Practice", IEEE Communications Magazine, September 1994.

[10] Abdul Raouf Khan, "Access Control in Cloud Computing Environment", ARPN Journal of Engineering and Applied Science, Vol 7, No.5 May 2012.

[11] P. G. Shynu, K. John Singh, "A Comprehensive Survey and Analysis on Access Control Schemes in Cloud Environment".

[12] Sascha M¨uller, Stefan Katzenbeisser, and Claudia Eckert "Distributed Attribute-Based Encryption", LNCS 5461, pp. 20–36, 2009, Springer-Verlag Berlin Heidelberg 2009.

[13] ByungRae Cha, JaeHyun Seo, Jong Won Kim, "Design of Attribute-Based Access Control in Cloud

Computing Environment", Proceedings of the International Conference on IT Convergence and Security 2011, Springer Science, +Business Media B. V. 2012.
[14] Xin Jin, Ram Krishnan, Ravi Sandhu, "A Unified Attribute-Based Access Control Model Covering DAC, MAC and RBAC".
[15] Cheng-Chi Lee, Pei-Shan Chung, Min-Shiang Hwang, "Survey on Attribute-Based Encryption Schemes of Access Control in Cloud Environment", International Journal of Network Security, Vol. 15, No. 4, PP. 231 – 240, July 2013.
[16] Piretti et al., "Secure Attribute-Based Systems", CCS'06, October 30–November 3, 2006, Alexandria, Virginia, USA. Copyright 2006 ACM.
[17] Subhashini Venugopalan, "Efficient Multi-level Threshold Attribute Based Encryption".
[18] Yong Cheng, et al., "Efficient Revocation in Ciphertext-Policy Attribute-based Encryption based Cryptographic Cloud Storage", Journal of Zhejiang University-SCIENCE C (Computers & Electronics) ISSN 1869-1951 (Print); ISSN 1869-196X (Online), Zhejiang University and Springer-Verlag Berlin Heidelberg 2013.
[19] John Bethencourt, Amit Sahai, Brent Waters "Ciphertext-policy attribute-based encryption", IEEE Symposium on Security and Privacy", Pages 321–334, 2007.
[20] Shucheng Yu, Cong Wang, Kui Ren, Wenjing Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing", INFOCOM '10 Proceedings of the 29th Conference on Information Communications, Pages 534-542.

Authors' Profiles

Madhura Mulimani graduated in Computer Science Engineering from Karnatak University Dharwad, and completed M. Tech in Computer Network Engineering from Visvesvaraya Technological University, Belagavi in 2005. She is currently pursuing her Ph. D. in Visvesvaraya Technological University, Belagavi. Her research topic is Security in Cloud Computing. Her other areas of interest include Network Security and Information Security.

Rashmi R. Rachh graduated in Electrical and Electronics Engineering from Karnatak University Dharwad, and completed M.Tech.in Computer Science and Engineering from Manipal Academy of Higher Education, Manipal in 2003. She obtained her Ph.D. from Visvesvaraya Technological University, Belagavi in 2013. Her research interests include Cloud Computing, design of efficient VLSI architectures for cryptosystems and cryptanalysis. Presently, she is working as Associate Professor in the department of Computer Science and Engineering, Visvesvaraya Technological University, Belagavi.

How to cite this paper: Madhura Mulimani, Rashmi Rachh,"Analysis of Access Control Methods in Cloud Computing", International Journal of Education and Management Engineering(IJEME), Vol.7, No.3, pp. 15-24, 2017.DOI: 10.5815/ijeme.2017.03.02