International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 2, February 2013 ISSN 2319 - 4847

Cross Bread Role based Access Control For Extended Security At Azure in Cloud Computing

Parminder Singh[1], Sarpreet Singh[2]

[1]Research fellow, Department of Computer Science and Engineering, Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, INDIA
[2]Assistant Professor, Department of Computer Science and Engineering, Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, INDIA

ABSTRACT

Cloud computing provides a computer user access to Information Technology (IT) services i.e. applications, servers, data storage without requiring an understanding of the technology or even ownership of the infrastructure. Privacy and Security are big issues in Cloud Computing. The cloud environment is a large open distributed system. It is important to preserve the data, as well as, privacy of users. Various access control models are in use, including the most common Mandatory Access Control (MAC), Discretionary Access Control (DAC) and Role Based. Access Control (RBAC).This paper proposes the security enhancement in Role Based Access Model using Reference Ontology to restricts the number of user per role, number of transaction per user and adds the feature of backup and restoration. The advantages to add these features to enhance the more security on Cloud Computing and data will never loss in case of Cloud crash. This paper proposes a reference ontology framework for access control in a cloud to facilitate the design of security system and reduce the complexity of system design and implementation.

Keywords: Cloud Computing, Security, Role-based Access, Ontology.

1. INTRODUCTION

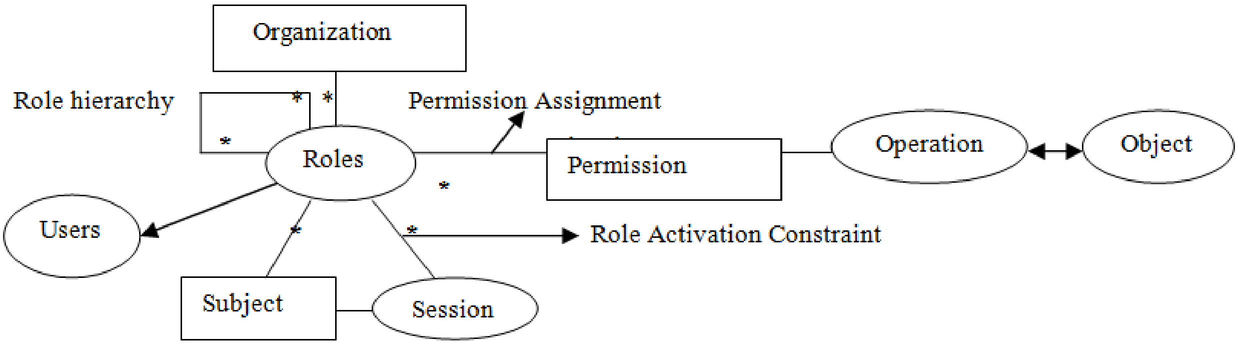
Cloud Computing is development and application modification in this modern world. In the previous time we use to create applications on the local server and we also use to keep them on the local server. If the local server that is the local system crashes then the entire system and my application crashed automatically. It was getting into a huge problem all over the world. To overcome this problem, the concept of cloud computing was brought into action. Brand Software Companies like Google, Microsoft, Face book started their own cloud over which now these days, data is available in bulk. A security architecture document should be developed that defines security and privacy principles to meet business objectives. Documentation is required for management controls and metrics specific to asset classification and control, physical security, system access controls, network and computer management, application development and maintenance, business continuity, and compliance. A design and implementation program should also be integrated with the formal system development life cycle to include a business case, requirements definition, design, and implementation plans. Design reviews of new changes can be better assessed against this architecture to assure that they conform to the principles described in the architecture, allowing for more consistent and effective design reviews.

2. RELATED WORK

2.1. Mandatory Access Control (MAC):- MAC was associated with the Bell-LaPadula Model of multi-level security in 1973. Bell-LaPadula model describes methods for assuring Confidentiality of information flows. The mandatory access control (MAC) model counters these threats by controlling access centrally. An ordinary user cannot change the access rights a user has with respect to a file, and once a user logs on to the system the rights he/she has are always assigned to all the files he/she creates. This procedure allows the system to use the concept of information flow control to provide additional security. For best practices, MAC policy decisions are based on network configuration. Loosely defined access control model in which user has an access to resources given by an administration .Only administrator can assign permissions to access objects and subjects. Administration define the access policy and usage which cannot be modified or change by the user. In policy administrator can define who has access to which files and programs. In other words access controls are managed by the administrator only. MAC is the main access control model used by the intelligence agencies and military to maintain policy access restrictions. MAC is primary developed for purposes where confidentiality is more important than integrity. MAC is considered a good model and straight forward for commercial system that operate in environment like financial institutions where risk of attack is high.

2.2. Discretionary Access Control (DAC):- At Discretionary Access Control (DAC) is a user-centric access control model in the sense that a file owner determines the permissions that are assigned to other users requiring access to the file. There is no central control so this model is easy to implement in a distributed applications on the Web. This model is based on the resource ownership. File Password might be a simple form of Discretionary Access Control (DAC), where password created by file owner requires accessing the file. In Linux, the permission to access the file is general form of DAC. DAC is controlled by the root/administrator or owner rather than being hard coded into the system**.** This model is implemented using Access Control List (ACL) associated with resource that identifies the user who can access the resources and authority the user is allowed in referencing the resource. This type of control is discretionary in the sense that subjects can manipulate it, because the owner of a resource, in addition to the security administrator, can identify with what authority who can access the resource. The main drawback of this model is that they fail to recognize the difference between computer programs and human users. To provide the safety analysis in DAC there is state transition system based meta-formalism control schemes and also presenting the algorithm for deciding safety with running time O (n3) in Graham-Denning scheme.

2.3. Role Based Access Control (RBAC):- RBAC in which permission are associated with roles and users are assigned to appropriate roles. Mandatory Access Control (MAC), Discretionary Access Control (DAC) proved to be problematic for distributed systems and managing the access to resources and system become hard so new access model is introduced known as Role Based Access Control (RBAC). Role Based Access Control (RBAC) using Reference Ontology describes a RBAC model using a role ontology for Multi-Tenancy architecture for specific domain. Ontology transformation algorithms are provided to compare the similarities of different ontology. It helps to reduce the complexity of system design and implementation.



Three primary rules are defined for RBAC:
1. Role assignment: A subject can exercise permission only if the subject has selected or been assigned a role.
2 .Role authorization: A subject's active role must be authorized for the subject. With rule 1 above, this rule ensures that users can take on only roles for which they are authorized.
3. Permission authorization: A subject can exercise permission only if the permission is authorized for the subject's active role.

3. PROBLEM FORMULATION

Role-based access control (RBAC) is a Ontology and it is a combination of mandatory and discretionary access control. It is complete Architecture. In the role-based access control model, a role is typically a job function or authorization level that gives a user certain privileges with respect to a file and these privileges can be formulated in high level or low level languages. RBAC models are more flexible than their discretionary and mandatory counterparts because users can be assigned several roles and a role can be associated with several users. To Create a Architecture which can provide the users of this system an access control through which they can access the content of the system. The administrator of the system can be providing access control by the users of facility. And then RBAC system has been implemented. Our objective and motive is to create an Advance RBAC to enhance the security of entire application. Our objective may also include reducing the burden of administrator of the system. By contrast, with the RBAC approach, access privileges are handled by assigning permissions in a way that is meaningful, because every operation has a specific pre-defined meaning within the application. In an RBAC model, a user's role is not mutually exclusive of other roles for which the user already possesses membership. The challenge of RBAC is the contention between strong security and easier administration. A RBAC system has two phases in assigning a privilege to a user: in the first phase, the user is assigned one or more roles; and in the second phase, the roles are checked against the requested operations. In RBAC, permissions are associated with roles rather than users, thus separating the assignment of users to roles from the assignment of permissions to roles. Users acquire access rights by their roles, and they can be dynamically re-assigned or removed from roles without changing the permissions associated with roles. The number of roles is typically much smaller than the number of users. Roles may have a hierarchical structure, and it reflects the organization's lines of authority and responsibility. Our strategy is to use RBAC model to control MTA in cloud. Intuitively, one may find that ontology information can be used to map users to roles and build up the role hierarchy. It proposes a reference ontology framework, in which users can search ontology database given a specific domain to find out relevant candidate role hierarchy templates, further get the corresponding policies associated with the templates to help with their own designs. The authorization to access a file relies on a set of rules that are specified formally and are used to decide which users have the permissions required to access a file. Permission allows a user to perform a number of well-defined operations on a file. For example, the security administrator of our collaborative web application can choose to schedule automatic virus checks with an anti virus application. In this way, the application gets assigned the privilege of scanning all the hard disks and memory on the computers on the network (system) with the aim of eliminating viral threats. Permissions imply a hierarchy of access rights and so users are assigned roles that define what permissions they can exercise and in what context. The MTA has increased the security risk due to the sharing of software, data and data schemas by multiple tenants. As these collocated tenants may be competitors, if the barriers between tenants are broken down, one tenant may access another tenant's data or interfere with their applications. The cloud providers are responsible for ensuring that one customer cannot break into another customer's data and applications. To overcome all this we can provide role-based access control (RBAC). An added drawback that RBAC faces is that roles can be assigned in ways that create conflicts that can open up loopholes in the access control policy. However, in the cloud, due to multi-tenancy architecture (MTA), data from multiple clients are stored and managed by the same software. When the software makes a mistake, potentially millions of clients may access private data of other clients. Furthermore, data stored in a cloud may be available to cloud administrators and they may access or modify data for their own benefits.

3.1 Problem in current RBAC:-

1. There is nothing mentioned in current RBAC that how many user would be their per Role. This may lead to a hacking environment. Suppose somebody hacks the role which we have generate [although cloud computing is very secure]. When the hacker user will try to access the content of the current RBAC, then normal RBAC will not restrict him from accessing the content because it allow to do so.

2. Suppose we have made some restriction over the generation of new ID of the specific role. Then also there is the possibility for the hacker to hack the existing ID and to hack the entire transaction. To overcome this problem we can put restriction of number of transaction per day. So that in any case if any existing Id gets hacked a minimum amount of damaged.

3. Third problem is that, RBAC concept sends data directly to the cloud computing sever, he never keeps a copy for any kind of backup or so. Suppose if administrator is creating something, he might make a mistake and the data could be in incorrect security format which again leads to the data security threat.

To overcome these problems, we can create "New Advance RBAC Architecture" system which a kind of Ontology which can keep a backup of the data which is getting send to the cloud server and to restrict the number of users per role. For this purpose we will have to implement security policies to a local cloud sever just to make sure that the data which is getting stored over the main cloud server has a backup for restoration, if something goes wrong, also if the number of users per role exceeds, the admin of the system should get a alarm or something so that he can come to know that security threats has attacked the system. New features for Advance RBAC are Limit over the number of user per role, Limit over the number of transaction per day or per hour, Keep backup data for restoration, Increase the security.

3.2 Using Ontology For RBAC:-

Ontology, a conceptual structure which contains knowledge in a domain and their relationships, provides useful and valuable information for cloud computing. It specifies a conceptualization of a domain in terms of concepts and their relationships, which is used to generate a commonly agreed vocabulary for information exchange without ambiguity. In general, according to the semantic in a specific domain, roles are defined as a combination of the official positions, job functions, and etc. For example, in an IT company, typical official positions could be that of the ordinary member, group manager, regional manager and etc. Functions represent the user's daily duties such as being a developer, testing engineer and etc. Additionally the organizational unit to which a user belongs is used as an access control criterion for certain applications.

4. RESULT & DISCUSSION

The main purpose of this paper is to examine the Role-Based Access Control security model from a theoretical perspective. To make this possible, the basic theory needed to understand the model will be investigated and briefly discussed. The theoretical approach should, if possible, discussion an ontology that can be used to express the security relations for any system. Such an ontology should be able to describe access restrictions as well provide an abstraction from real systems. The specification for interoperation in distributed environment is introduced. The works include a definition of ontology to describe the concepts and a declaration of rules to explicit the relationship between concepts.

The ontology based approach can express security policy with semantic information and provide a machine interpretation for descriptions of policy in open distributed environment. At the restoration of ontology Access control generally suggests that there is an active user and/or application process, with a desire to backup the data .For simplicity, we will here after refer to an entity as a user and a data object as a file. Access control typically involves two steps: authentication and authorization. In order to authentication can active user, the distributed system needs some way of determining that a user is in who he/she claims to be. A password is an example of a standard authentication method. On the other hand, A role is a set of operations that a user is allowed to perform on a data object(s). A user can have more than one role and more than one user can have the same role. Partial orderings are used to order the permissions associated with a set of security policies. And At Information flow control allows the access control system to monitor the ways and types of information that are propagated from one user to another. A security system that implements information flow control typically classifies users into security classes and all the valid channels along which information can flow between the classes are regulated by a central authority or security administrator.

5. CONCLUSION

This paper proposes a RBAC model using reference ontology with enhancement in policy control. Firstly it restricts the number of user per role and number of transaction per user to enhance the security. Secondly there is concept of backup and restoration of data in local server that helps to make the availability of data even if cloud crash. The new methodologies and Ontologism has been introduced time to time for the same purpose. One of the ontology is called RBAC system. Role Based Access Control is an architecture which provides the authority to restrict the user if he is not allowed to go on with the content .It has been affective in a lot of manner. This architecture saves the data from the unauthorized use of the data. The admin panel has all the rights to restrict the user of accessing the data and back again he can again edit the access rights of the user.

REFERENCES

[1] A.C. O'Connor and R.J. Loomis (December 2010) (PDF). Economic Analysis of Role-Based Access Control Research Triangle Institute.
[2] A. Das and D. Grosu, “Combinatorial auction-based protocols for resource allocation in grids,” Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International, 2005.
[3] C.S. Yeo and R. Buyya, “A taxonomy of market-based resource management systems for utility-driven cluster computing,” Software: Practice and Experience, vol. 36, Nov. 2006.
[4] D. Friedman, “The double auction market institution: A survey,” The Double Auction Market: Institutions, Theories, And Evidence, J. Rust, ed., Westview Press, 1993.
[5] D.R. Kuhn (1998). "Role Based Access Control on MLS Systems Without Kernel Changes" (PDF). Third ACM Workshop on Role Based Access Control.
[6] D.S. Diamond and L.L. Selwyn, “Considerations for computer utility pricing policies,” Proceedings of the 1968 23rd ACM national conference, New York, New York, USA: ACM Press, 1968.
[7] Ferraiolo, D.F. and Kuhn, D.R. (October 1992). "Role-Based Access Control"(PDF). 15th National Computer Security Conference (2009).
[8] “Google App Engine” http://code.google.com/appengine/
[9] I. Foster and C. Kesselman, “The grid: blueprint for a new computing infrastructure,” Oct. 1998.
[10] I.E. Sutherland, “A futures market in computer time,” Communications of the ACM, vol. 11, Jun. 1968.
[11] J. Shneidman, C. Ng, D.C. Parkes, A. AuYoung, A.C. Snoeren, A. Vahdat, and B. Chun, “Why Markets Could (But Don’t Currently) Solve Resource Allocation Problems in Systems,” Challenges, 2005.
[12] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, Above the Clouds : A Berkeley View of Cloud Computing, 2009.
[13] “Microsoft Windows Azure” http://www.microsoft.com/windowsazure/
[14] N. Nisan, S. London,O. Regev, and N. Camiel, “Globally distributed computation over the Internet-the POPCORN project,” Proceedings. 18th.
[15] P. Cramton, Y. Shoham, and R. Steinberg, Combinatorial Auctions, The MIT Press, 2005.
[16] P. A. Loscocco, S. D. Smalley, P. A. Muckelbauer, R. C. Taylor, S. J. Turner, and J. F. Farrell. The Inevitability of Failure: The Flawed Assumption of Security in Modern Computing Environments In Proceedings of the 21st National Information Systems Security Conference, Oct. 1998.
[17] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger, “Economic models for resource management and scheduling in Grid computing,” Concurrency and Computation: Practice and Experience, vol. 14, 2002
[18] S. Clearwater, Market-Based Control: A Paradigm for Distributed Resource Allocation, World Scientific, 1996.