A REVIEW ON ACCESS CONTROL MECHANISM FOR SECURE CLOUD

[1]BIRRU DEVENDER, [2]Dr. SYED ABDUL SATTAR

[1] Research Scholar, Rayalaseema University, Kurnool, Andhra Pradesh,India
[2]PhD (ECE), PhD (CS) Director R&D, professor of ECE. Nawab shah Alam Khan college of engineering & Technology, new malakpet, malakpet, Hyderabad. T.S. India.

ABSTRACT The relationship between users and resources is dynamic in the cloud, and service providers and users are generally not in the same security domain. Identity-based security (for example, discretionary or mandatory access control models) cannot be used in an open cloud environment, where each resource node may be unfamiliar or not even familiar. Know. Users are normally identified by their attributes or characteristics and not by their default identities. A dynamic access control mechanism is often required to perform cross-domain authentication. In this article, we will focus on the following three broad categories of access control models for cloud computing: (1) role-based models; (2) attribute-based encryption templates and (3) multi-tenancy templates. We will review the existing literature on each of the above access control models and their variants (technical approaches, features, applicability, advantages and disadvantages) and identify future directions of research to develop access control models for cloud computing environments.

KEYWORDS: Access Control Models, Role-based Access Control, Attribute-based Encryption Model.

1. INTRODUCTION

The three models of cloud computing service delivery are: Software as a Service (SaaS) in which cloud clients use the applications of the ISP; Platform as a Service (PaaS) where customers distribute their self-created applications on a development platform provided by a cloud service provider; and Infrastructure as a Service (IaaS), where cloud customers praise IT, storage, and network capacity from the cloud service provider. The paradigm of cloud computing is associated with security issues at both the provider and consumer levels. While providers want to ensure that their resources and services are used only by authorized users; consumers want to ensure that their data is securely stored in the cloud and that servers are not compromised.

Access control is a key aspect of information security that is directly related to key features such as privacy, integrity, and availability. Cloud service providers must provide the following basic access control features: (i) Control access to cloud service features based on the specified policies and the level of service purchased by the customer. (ii) Controlling access to consumer data from other consumers in multi-tenant environments. (iii) Control access to normal user functions and privileged administrative functions. (iv) Maintain an accurate access control policy and up-to-date information on the user's profile.

Access control models can traditionally be classified into three types: (1) Discretionary (2) Mandatory and (3) Role-based. In the Discretionary Access Control (DAC) model, the owner of the object decides access permissions for other users and sets them accordingly. The UNIX operating system is a classic example of the discretionary access control model. For example, the object (that is, the owner of an object) can specify which permissions (read / write / execute) members of the same group can have and which permissions can have all the others. DAC models are typically used only with legacy applications and provide significant management overhead in the modern multi-user and multi-application environment typical of distributed systems such as the cloud. Mandatory Access Control (MAC) models ignore the need for resource mapping and are therefore more adaptable to distributed systems than DAC models. The MAC model is generally used in multi-level security systems. Here, access permissions are decided by the system administrator and not by the subject. In a multi-level MAC model, each subject and each object is identified with a classification security level (for example, Unclassified, Classified, Secrets, and Top Secret). The Bell LaPadula model recommends the "no-read" rule and the "no-write" rule to keep the information confidential. The Biba model recommends rules of "non-writing", "non-reading" and "non-executing-up-or-down" to maintain the integrity of the information. In a role-based access control model (RBAC), a user has access to an object based on the role assigned in the system. Roles are defined based on the job functions. Authorizations are defined on the authority of the work and the responsibilities of the work. Operations on the object are called based on permissions. RBAC models are more scalable than discretionary and mandatory access control models and better suited for cloud computing environments, especially when service users can not be tracked with a fixed identity.

The relationship between users and resources is dynamic in the cloud, and service providers and users are generally not in the same security domain. Identity-based security (for example, discretionary or mandatory access control models) cannot be used in an open cloud environment, where each resource node may be unfamiliar or not even familiar. know. For example, it can be seen that cloud users, especially at the SaaS level, access services via the Internet through various means such as mobile phones, laptops or PDAs; therefore, it is not possible to identify users via fixed IP addresses. In such situations, traditional firewalls cannot be used to filter packets based on users' fixed IP addresses. In a cloud, users are normally identified by their attributes or characteristics and not by their default identities. Therefore, dynamic access control is required to perform cross-domain authentication.

2. RELATED RESEARCH

For the grid computing and cloud computing paradigms, there is a common need to be able to define the methods by which consumers discover, request and use the resources provided by third-party central structures and also implement highly parallel calculations. Distributed that works on these resources. The networks came into effect in the mid-1990s to solve large-scale computing problems on a network of resource-sharing machines that would provide the same computing power at affordable prices as with expensive supercomputers and large dedicated groups. at that time. A grid can typically include processing, storage, and network resources from multiple geographically dispersed organizations and these resources are normally considered heterogeneous with availability and dynamic capabilities. The network's two main concerns were interoperability and security, as resources come from different administrative domains with different local and global resource utilization policies, as well as different hardware and software configurations and platforms. Most grids use a batch-planned calculation model with appropriate strategies to enhance the identification of the correct credentials based on the batch tasks that will be performed for accounting (for example, the number of processors required, the number of duration of assignment, etc.) and for security purposes.

Condor is a centralized workload management system that is suitable for calculating intensive work in local closed network environments. Its resource management mechanism is similar to that of UNIX (Discretionary Access Control), with additional access modes in addition to traditional read and write permissions. Legion uses an object-oriented approach where all files, services, and devices are treated as objects and are accessible through the functions of these objects. Each object can define its own access control policy, typically performed using the access control list and authentication mechanisms, in a default MayI function called before any other function of the object. The Globus Grid Toolkit (GT) provides mechanisms for translating users' network identities into local identities (which in turn can be verified by resource providers using the appropriate local access control policies) and also enables user certificate delegation from many different sites.

With the Single Sign-On mechanism (eg OpenGrid Service Infrastructure, OGSI), users can only log in once and access multiple sites on the network because programs can be allowed to access resources on behalf of a user and can delegate them to other programs. OGSI works with resource brokers (eg Gruber) who act as application points for distributed policies to apply both local use policies and service level agreements global. sites to share effectively across multiple sites. The authors propose a multi-policy access control model (ABMAC) based on flexible attributes for grid computing systems in which each autonomous domain can have its own security policy. ABMAC is based on the idea of integrating the individual authorization decisions received for resource / service access requests (all identified with their characteristics or attributes) according to the security policy of each domain and to arrive to a final decision using a combination algorithm that can be adapted to resource / exploitation constraints. The ABMAC approach is more scalable than the development of a subset of individual domain policies and the evaluation of the user's demand for access to resources on the basis of this superset.

3. ROLE-BASED ACCESS CONTROL MODEL

In a role-based access control (RBAC) model, the role of a user is assigned based on the concept of minimum privilege, which is the role with the least permissions or functionality required for the job to be performed. run. The role-based access control model (TRBAC) has been considered a valid model for cloud computing environments where traditional static access control models cannot be used as discretionary, mandatory, or simple models based on roles. . TRBAC can dynamically validate access permissions for users based on assigned roles and the task that the user must perform with the assigned role. Tasks can be classified as workflow tasks (those that must be executed in a particular order) that require active access control and non-workflow activities (those that can be executed in any order) that require passive control accesses. Active role-based access control based on workflow activities is time-sensitive and the access permissions assigned to users performing these tasks change dynamically over time, depending on the order in which the tasks are to be performed. It must be ensured that a user has the minimum privileges required to perform a task with a particular role and that no role can be assigned to two or more tasks at the same time. Another variant of the role-based access control proposed for cloud computing environments is the access-based-based access control (ARBAC) model, where certain attributes and values are assigned to the data object to be protected; a user with a specific role must send appropriate values for those attributes and access the objects after proper validation by the service provider. An ARBAC model based on a fine-grained key has been proposed, in which private keys or symmetric keys are used to encrypt / decrypt the attribute values defined for the data objects to be protected.

Bertino and others have proposed the temporary-RBAC (TRBAC) model that enables and disables a role at run time based on user requests. The authors argue that in some applications, some roles must be static and remain active, while only users and permissions are assigned dynamically. In this context, they proposed a generalized TRBAC model (GTRBAC) that supports the activation of roles instead of the enabling role. A role is said to be activated if at least one user assumes this role. GTRBAC supports the activation and deactivation of constraints on the maximum active duration granted to a user and the maximum number of activations of a role by a single user in a given time interval. The authors present an XML-based RBAC policy specification framework for applying access control in dynamic XML-based Web services. However, GTRBAC and X-RBAC are not able to provide reliable and context-sensitive access control (essential to dynamic Web services, characteristic of cloud computing environments) and are based solely on access control based on the identity. or capacity. The authors propose an improved hybrid version of the X-RBAC and GTRBAC models, called the X-GTRBAC model. X-GTRBAC relies on trusted third-party certification (such as any PKI certification authority) to assign roles to users. X-GTRBAC also considers context (such as time, location or environmental status when access requests are made) to directly influence the level of trust associated with a user (as part of the user profile) and incorporates it into its access control decisions. Access privileges for a user / role are based on the threshold (that is, the confidence level) established based on the requestor's access patterns; If the user seems to deviate from his or her usual profile, the user's trust level is automatically reduced to avoid potential abuse of privileges. This real-time feature of X-GTRBAC adapts to web cloud environments with different customer activity profiles.

4. ATTRIBUTE-BASED ENCRYPTION (ABE) MODEL

Attribute-based encryption (ABE) is more appropriate (compared to traditional public key infrastructure or identity-based encryption) to protect data privacy and confidentiality in a cloud computing environment. ABE is useful when the data source does not know the recipient's identity or public key; but knows only certain attributes of the recipient. For example, imagine the user Alice who wants to communicate with her former classmates, but does not know their email addresses. ABE identifies a user with a series of attributes. Sahai and Waters (SW) offer ABE as follows: given a secret key of a set of omega attributes, you can decrypt an encrypted text encrypted with a public key based on a set of ω attributes "only if sets ω and ω "Overlap sufficiently determined by a threshold value t. The SW schema also proposes the use of an access-based policy to decide which attributes are required to decrypt a message. An example for the access tree can be: Class2005 ∧ (∨ MyCollege MyTeacher) implies that any user who graduated in 2005 class at MyTeacher or MyCollege meets politics.

As the EBA scheme extension, proposed in the literature are two variants: the ABE system (ABE-KP) based on the key policy and the EBA system (CP-ABE) based on political cryptogram. In KP-ABE, the cipher text is associated with a set of attributes and the secret key is associated with the access tree. The cryptographic part has no control over who has access to the data and can only define the set of descriptive attributes needed to decrypt the cipher text. There is a trusted authority that generates the secret key, provided that the user sends the appropriate values for the attributes that make up the access tree. In CP-ABE, the access of the tree is associated with an encrypted text and a cryptographic part determines the criteria according to which the data can be decoded, while the secret key is associated with a set of attributes. The CP-ABE scheme has been exploited for efficient implementation of the authorization model as a service providing users (content owners) with a single access control point to set permissions on data belonging to multiple services.

An extension of the naive KP-EBA and CP-ABE regimes for multiple systems, which are typical of cloud computing environments, would require each user to maintain the attribute or access tree issued by different authorities, and it is necessary a global authority that can check attributes between different organizations and release appropriate secret keys for all users of the system. However, this global authority is likely to be attacked and can become a bottleneck in an Internet cloud environment. Another major challenge is the possibility of collusion between multiple users (including those whose attributes have been revoked) with attributes from different authorities to gain illegal access to the data. The authors proposed a KDC (Key Distribution Center) approach to distribute the decryption key to owners and users of data that are assigned a specific set of attributes, which is encrypted with the data by the owner. ; users with the corresponding set of attributes can retrieve data from the cloud. The attribute-based cryptography model applied here is secure collusion because it is based on bilinear pairing on elliptic curves; two users can not decode data that none of them has individual access rights. The KDC-based access control model is more likely to become a single point of failure (especially when used with one or more KDCs in the cloud) and incurs significant control and management costs as the number of users increases and cloud provider.

The authors propose a model of access control based on the multi-authority ABE, adapted to cloud computing environments. According to this scheme, each user is assigned a unique global user identifier (UID) and each user is assigned a unique authorization identifier (AID). The UID and AID are issued by a Certificate Authority (CA) approved by the different authority domains. To prevent two users from colliding together to gain illegal access to the data, the certificate authority's UID must be used in conjunction with secret keys issued by different authorities to decrypt the data. The authors propose an efficient method of attribute revocation in multi-authority CP-ABE systems using proxy encryption. The CA-based system is more distributed than the KDC-based approach; a KDC must also be online to distribute keys to users, while a CA must not always be online.

5. CONCLUSION

We identify the following guidelines for future research on access control models in cloud computing environments: (1) Develop access control models based on attributes based on attributes so that role assignments and role are assigned constructs separately using policies applied on the attributes of users, roles, objects and the environment; and attribute-based role assignment and role authorization rules must be applied in real time to enforce access control decisions. (2) Develop a role-based and position-based control model that is embedded when applying the cloud policy (thus preventing disclosure of the user's identity, role, or position directly to a cloud remote cloud server may not be fully reliable) and enable / enable the role only when the user is in logical locations (calculated from actual locations using specific mapping functions) located in the spatial boundary of a role. (3) Explore hardware-software security co-design so that the access control and access control mechanisms implemented in the software are integrated into a new hardware architecture and virtualization capabilities that can help protect the confidentiality and integrity of data and resources, even when the powerful underlying hypervisor can be compromised. (4) Mitigate internal threats to data and resources from the point of view of the unreliable cloud provider administrator and the employee of the victim organization who exploits the cloud's weaknesses for unauthorized access. (5) Integrate the relationship of trust and reputation in access control models for a better quality of service in the cloud.

REFERENCES

[1] L. Popa, M. Yu, S. Y. Ko, S. Ratnasamy and I. Stoica, “CloudPolice: Taking Access Control out of the Network,” Proceedings of the 9th ACM Workshop on Hot Topics in Networks, October 2010.

[2] S. Oh and S. Park, “Task-role-based Access Control Model,” Information Systems, vol. 28, no. 6, pp. 533-562, September 2003.

[3] H. A. J. Narayanan and M. H. Gunes, “Ensuring Access Control in Cloud Provisioned Health Care Systems,” Proceedings of the IEEE Consumer Communications and Networking Conference, 2011.

[4] S. Sanka, C. Hota and M. Rajarajan, “Secure Data Access in Cloud Computing,” Proceedings of the 4th IEEE International Conference on Internet Multimedia Services, December 2010.

[5] S. Yu, C. Wang, K. Ren and W. Lou, “Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing,” Proceedings of the 29th IEEE International Conference on Information Communication, pp. 534-542, 2010.

[6] E. E. Mon and T. T. Naing, “The Privacy-aware Access Control System using Attributed-and Role-based Access Control in Private Cloud,” Proceedings of the 4th IEEE International Conference on Broadband Network and Multimedia Technology, pp. 447-451, October 2011.

[7] V. Goyal, O. Pandey, A. Sahai and B. Waters, “Attribute-based Encryption for Fine-Grained Access Control of Encrypted Data,” Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89-98, 2006.

[8] J. Bethencourt, A. Sahai and B. Waters, “Cipher text-Policy Attribute-Based Encryption,” Proceedings of the IEEE Symposium on Security and Privacy, pp. 321-334, 2007.

[9] K. Yang and X. Jia, “Attribute-based Access Control for Multi-Authority Systems in Cloud Storage,” Proceedings of the 32nd IEEE International Conference on Distributed Computing Systems, pp. 536-545, 2012.

[10] T. Ristenpart, E. Tromer, H. Shacham and S. Savage, “Hey, You, Get off my Cloud: Exploring Information Leakage in Third-Party Compute Clouds,” Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 199-212, 2009.

[11] D. Nurmi, R. Wolski, C. Grzegorczyk, S. Soman, L. Youseff and D. Zagorodnov, “The Eucalyptus Open-Source Cloud-Computing System,” Proceedings of the International Symposium on Cluster Computing and the Grid, pp. 124-131, 2009.