PROCEEDINGS OF SCIENCE

Trust Model Based on Role and Attribute in Cloud Environment

Cong Wang[1]
School of Science, Beijing University of Posts and Telecommunications
Beijing, 100876, China
E-mail: wangcongcherish@gmail.com

Ronghua Li[2]
China Mobile Communications Corporation
Beijing, 100032, China
E-mail: lironghua@chinamobile.com

Yijie Shi[3]
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
Beijing, 100876, China
E-mail: yijieshi2000@bupt.edu.cn

Jie Zhang
School of Science, Beijing University of Posts and Telecommunications
Beijing, 100876, China
E-mail: jiezhang@bupt.edu.cn

In the light of the problem that the current trust model is mostly role-based trust model, yet this trust model doesn't take attribute and cross-tenant access into account, which is not suitable to the dynamic multi-tenant cloud environment. In this paper, we propose a trust model based on role and attributes, including subject attributes, resource attributes and environment attributes; and this model includes single-tenant and multi-tenant trust sub-models. Analysis shows that the proposed trust model can resolve the problem that when attributes fulfill the requirements but the users' or roles' credibility is low. User Role Assignment and Role Permission Assignment will reduce the security of tenant's data and trust problem between tenants when the users access tenants. So it can be applied to access control in multi-tenant cloud environment.

CENet2017
22-23 July, 2017
Shanghai, China

[1]Speaker, Corresponding Author
[2]This study was supported by the Ministry of Education-China Mobile Research Foundation (Grant No.: MCM20150501)
[3]Corresponding Author

PoS(CENet2017)068

1.Introduction

At present, the trust model in cloud environment is mostly used in Role-Based Access Control (RBAC). In the existing trust model applied to access control based on the role's encryption in cloud environment[1-4], it is divided into the data owner-role model and the user-role model, in which, the former is used to help data owner evaluate the credibility of the role from two aspects, namely the individual trust and the succession trust; then the trust evaluation is adopted to decide whether to store its encrypted data in the cloud for a specific role. The user-role model is used to help the role evaluate the credibility of the user from two aspects, namely the direct trust and the recommended trust, and decide whether to grant membership to the user according to the user's credibility. It doesn't consider attributes, including subject attributes, resource attributes and environment attributes. Moreover, it's constructed just based on RBAC framework, which is not suitable to the dynamic cloud. But this trust model doesn't take the cross-tenant access problem into account, and not suitable to access control in multi-tenant cloud environment. In the cross-tenant role-based trust model[5-7], there are two ways for users in tenant A across access tenant B' resources, allocate B's role for the users in A, or allocate B'role' son role for A's role. In this sense, if it meets one of the definitions of trust relationship, role permission assignment or role hierarchy, the cross-tenant access can be available. While taking the cross-tenant access problem into account, it's also constructed just based on RBAC but fails to consider attributes. The trust relationship and the way of cross-tenant access its defined are not suitable for access control based on role and attribute. Moreover, this trust model is only a formal role-based trust model and the specific calculation of trust value is not completed.

In this paper, we propose a trust model based on role and attribute, including single-tenant and multi-tenant trust sub-models. Chapter I introduces the research background and Chapter II introduces related basic knowledge. Chapter III and IV introduces the single-tenant and multi-tenant trust sub-models based on role and attribute are elaborated. The final chapter analyzes the proposed trust model and makes conclusion.

2.Basic Knowledge

2.1Role-Based Access Control（RBAC）

In 2001, the National Institute of Standards and Technology (NIST) proposed the standard RBAC reference model[8], NISTRBAC. It is divided into three models: Basic RBAC, Level RBAC and Constrained RBAC. It consists of seven basic elements: Users, Roles, Sessions, Objects, Operations, Role Permission Assignment and User Role Assignment. The basic idea is to establish the many-to-many relationships between users and permissions through the roles, so the users can obtain the permissions to the resources [8-10].

2.2Trust Model

Assume that $X=\{x_1, x_2, .., x_n\}$ be the collection of n feedbacks of the transactions, and have r positive feedbacks and s negative feedbacks. Now set a possible feedback for the next transaction to x_{i+1}, then the possibility that x_{i+1} is positive is expressed as follows [2]:

$$\varepsilon(r,s)=\frac{r+\alpha}{r+\alpha+s+\beta} \tag{2.1}$$

The trust value's computation Formula (2.1)

Most Bayesian trust uses the assumed parameters $\alpha=\beta=1$ [11-15].

2.3Symbol Specification

Symbol	Meaning	Symbol	Meaning	Symbol	Meaning
DO	Data Owner	**URA**	User Role Assignment	**SA**	Subject Attribute
RH	Role Hierarchy	**RPA**	Role Permission Assignment	**PA**	Permission Attribute
U	User	**RBAC**	Role-Based Access Control	**EA**	Environment Attribute

Table 1: Symbol Specification

3.Single-tenant Sub-Trust Model Based on Role and Attribute in Cloud Environment

In this paper, we present a trust model based on role and attribute in cloud environment, which includes single-tenant and multi-tenant trust sub-models. The single-tenant trust sub-model refers to the trust sub-model in one internal tenant so as to resolve the problem that when attributes fulfill the requirements but users' or roles' credibility is low, URA and RPA will reduce security of tenant's data. It includes the trust of role to user and the trust of DO to role.

3.1Trust of Role to User

The role computes the user's trust value according to the following three aspects, and the role determines whether to conduct URA by the relation of the trust value and pre-set threshold.

3.1.1Trust Based on User's Behavior T_1

The role computes the user's trust value according to the interactive history between the user and the role. When the user once joines the role, the role can call out the user's trust records from the central database. Marking the role that the user requests to join as R_K or K, then the user's trust record is $H_{K\to U}=(R_K, V_{K,U})$, where $V_{K,U}=(p,q)$ is the trust vector, p denotes the resources'number that the user accessed as the role of K, and q denotes illegal events that the user participated in, which means disclosure of resources to the users that doesn't belong to the role, or making authorization on the data that is not in its range. Then, $T_1=\varepsilon(p-q,q)$.

3.1.2Trust Based on User's Reputation T_2

The role computes the user's trust value according to interactive history between the user and other roles. Assuming that in tenant A, the user once joins n roles except K, marked as $\{R_1, R_2, .., R_n\}$, the trust records of each role to the user can be recorded as $H_{1\to U}, H_{2\to U}, ..., H_{n\to U}$, where $H_{i\to U}=(R_i, V_{i,U}), V_{i\to U}=(p_i, q_i), i=1,2,\ldots,n.$ Then, $T_2=T(V_{R,U}), V_{R,U}=\sum_{i=1}^{n} V_{i,U}$.

3.1.3Trust Based on Attribute T_3

The role makes judgement if SA (including ID, Password, Token, etc.) and EA (Time, IP, etc.) are matched with the corresponding role. If so, $T_3=1$; otherwise, $T_3=0$.

3.1.4Calculation of Trust Value

According to above three aspects, if $T_3=1$ and set the weight of T_1 and T_2 as $w_{1,}w_2$, then in tenant A, the trust value of role to user is $T=w_1T_1+w_2T_2$, $w_1+w_2=1$,where T_1 and T_2 are computed by the trust value's computation Formula (2.1). If $T_3=0$, then $T=0$.

3.2The Trust of DO to Role

DO computes role's trust value according to the following three aspects, and DO determines whether to conduct RPA by the relation of trust value and pre-set threshold.

- Trust Based on the Role Itself $T_1^{'}$ [2]: DO computes the role K's trust value according to the interactive history between DO and role K. If the tenant A(DO) has interactive history with role K, that is, A once gives permission to role K, then A can call out role K's trust records $H_{K\rightarrow R}=(O_K,V_{K,R}),V_{K,R}=(p,q)$ from the central database, then $T_1^{'}=T(V_{K,R})$.

- Trust Based on the RH $T_2^{'}$ [2]: DO computes the role's trust value according to the interactive history between DO and the role's son. In tenant A, it contains the trust based on K's son itself and trust based on K's son's RH, set their weight as $w_{21,}w_{22}$, then $T_2^{'}=w_{21}T_{21}+w_{22}T_{22,}w_{21}+w_{22}=1$.

- Trust Based on attribute $T_3^{'}$: DO judges if PA, including operating attributes(read, write, update and delete etc.) and resource attributes (type and secret level of DO's resource), and EA (time, length of access time and IP, etc.) are matched with the role. If so, $T_3^{'}=1$; otherwise, $T_3^{'}=0$.

According to above three aspects, if $T_3^{'}=1$, set the weight of $T_1^{'}$ and $T_2^{'}$ as $w_1^{'},w_2^{'}$, then in tenant A, the trust value of role to user is $T^{'}=w_1^{'}T_1^{'}+w_2^{'}T_2^{'}$, $w_1^{'}+w_2^{'}=1$, where $T_1^{'}$ and $T_2^{'}$ are computed by the trust value' computation Formula (2.1). If $T_3^{'}=0$, then $T^{'}=0$.

4.Multi-tenant Trust Sub-Model Based on Role and Attribute in Cloud Environment

The multi-tenant trust sub-model solves the trust problem between tenants when the users access across tenants, which can be applied to access control in multi-tenant cloud environment. If a user in tenant A wants to access tenant B's resources, B needs to decide whether A's user is trustworthy, which is related to the trust problem between tenants. B is the party that initiates the trust, known as the trustor, and the tenant A is the trusted party, known as the trustee. There are three ways to access across tenants:

- The user in A, who wants to access B's resource, requests joining in the corresponding role of B.
- The role in A requests to become the corresponding role in B, and all users of the role in A can access the corresponding resources of role in B.
- The role in A requests to become the ancestral role of one or a few roles in B, and all users of the role in A can access the corresponding resources of role in B.

In the first way, the corresponding role in B needs to determine whether the user that issued the request is trustworthy, and the problem becomes an expansion of trust of role to user in multi-tenant environment. In the second and third way, B (DO) needs to determine whether the role in A that issued the request is trustworthy, and the problem becomes an expansion of trust of DO to role in multi-tenant environment.

4.1Trust of Role in B to User in A

We describe the calculation of trust value of the corresponding role in B to the user that issues a request in A from the following three aspects.

- Trust Based on User's Behavior T_1 : In the multi-tenant trust model, the role in B computes the user's trust value according to their interaction history, and the method of calculation is the same as the trust of role to user in one internal tenant in 3.1.1, the difference is, here, the role refering to B's role.

- Trust Based on User's Reputation T_2 : In the multi-tenant trust model, T_2 refers that the role computes trust value according to the interaction history between the user and other roles from three aspects: trust of A's role to the user, trust of B's role to the user except the role as requested to join in, and other tenants' role to the user, denoted respectively as $T_{21,}T_{22,}T_{23}$. Their method of calculation is the same as the trust of role to user in one internal tenant in 3.1.2, just change the role into roles in corresponding tenant. Set the weight of $T_{21,}T_{22,}T_{23}$ as $w_{21,}w_{22,}w_{23}$. If one of them doesn't have trust records, set the corresponding weight as 0, and adjust the other weights to ensure that their sum is 1. Then $T_2=w_{21}T_{21}+w_{22}T_{22}+w_{23}T_{23,}w_{21}+w_{22}+w_{23}=1$.

- Trust Based on attribute T_3 : The role in B judges if SA (including ID, Password, Token, etc.) and EA (Time, length of access time, IP, etc.) are matched with the corresponding role. If so, the trust based on attribute $T_3=1$, otherwise, $T_3=0$.

According to above three aspects, if $T_3=1$ and set the weight of T_1 and T_2 as $w_{1,}w_2$ then the trust value of the role in B to the user in A is $T=w_1T_1+w_2T_{2,}w_1+w_2=1$,where T_1 and T_2 are computed by the trust value' computation Formula (2.1). If $T_3=0$, then $T=0$.

4.2The Trust of Tenant B to Role in A

We describe the calculation of trust value of B to the role K in A which wants to join in B's role or become one or more roles' ancestor from the following four aspects. These two cases belong to the trust of DO to role.

- Trust Based on the Role Itself $T_1^{'}$: If tenant B (DO) has interactive history with the role K, that is, B once gives permissions to role K, then B can call out role K's trust records $H_{K\to R}=(O_K, V_{K,R}), V_{K,R}=(p, q)$ from central database, then the trust based on the role itself $T_1^{'}=T(V_{K,R})$.
- Trust Based on the Role's Reputation $T_2^{'}$: It refers to the trust of other tenants except B to role K. If K is once given corresponding roles or ancestor roles in tenant $t_1, t_2, \ldots, t_m$, then B can call out interactive history $\{H_{1\to K}, H_{2\to K}, \ldots, H_{m\to K}\}$ between $t_1, t_2, \ldots, t_m$ and K, $H_{i\to K}=(t_i, V_{i,K}), V_{i\to K}=(p_i, q_i)$. Then the trust based on the role's reputation $T_2^{'}=T(V_{t,K}), V_{t,K}=\sum_{i=1}^{m} V_{i,K}$.
- Trust Based on RH $T_3^{'}$: It contains the role's trust based on RH in A, B and other tenants, T_{31} , T_{32} and T_{33} respectively. Their weights are set as $w_{31,}w_{32,}w_{33,}$ and $w_{31}+w_{32}+w_{33}=1,$, where T_{31} contains trust based on the role's son itself, reputation, RH in A, $T_{311,}T_{312}$ and T_{311} respectively. Their weights are set as $w_{311,}w_{312,}w_{313}$, then $T_{31}=w_{311}T_{311}+w_{312}T_{312}+w_{313}T_{313}$. T_{32} and T_{33} are similarly available. Therefore, the trust based on TH $T_3^{'}=w_{31}T_{31}+w_{32}T_{32}+w_{33}T_{33}$.
- Trust Based on attribute $T_4^{'}$: B judges if PA, including operating attributes(read, write, update and delete etc.) and resource attributes(type and secret level of DO's resource), and EA (time, length of access time and IP, etc.) are matched with role K in A; if so, the trust based on attribute $T_4^{'}=1$; otherwise, $T_4^{'}=0$. In the case when the role of A wants to become one or several roles' ancestor in B, the corresponding PA and EA are determined according to more than one role. In the other case when the role of A wants to become the corresponding role in B, the corresponding PA and EA are determined according to the corresponding role.

Based on the above four aspects, set the weights of $T_1^{'}, T_2^{'}, T_3^{'}$ as $w_1^{'}, w_2^{'}, w_3^{'}$. Inthe two cases when the role of A wants to become one or several roles' ancestor in B and the role of A wants to become the corresponding role in B, the weights are set differently and $w_3^{'}$ in the former case is more than the latter. If $T_4^{'}=1$, the trust of tenant B to role in A $T^{'}=w_1^{'}T_1^{'}+w_2^{'}T_2^{'}+w_3^{'}T_3^{'}, w_1^{'}+w_2^{'}+w_3^{'}=1$. If $T_4^{'}=0,$ then $T^{'}=0$.

5.Conclusion

In this paper, we propose a trust model based on role and attribute, including single-tenant and multi-tenant trust sub-models. It takes attributes into account, including subject attributes, resource attributes and environment attributes. In the trust of role to user, the trust of DO to role and the trust between tenants, they not only include the trust based on user's or role's behavior, the trust based on user's or role's reputation and the trust based on RH, but also include the trust based on attribute. By introducing the attributes, the proposed trust model is more applicable in dynamic cloud environmentA; by introducing trust, we can solve the problem when the attributes meet the requirements but the users' or roles' credibility is low, URA and RPA will

reduce the security of tenant's data. In addition, by introducing the multi-tenant trust sub-model, we can solve the trust problem between tenants when the users access across tenants; therefore, the trust model based on role and attribute can be applied to access control scheme multi-tenant cloud environment. The result shows that it is a more optimized trust model in the cloud environment.

References

[1] Lan Zhou, Vijay Varadharajan and Michael Hitchens. *Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage*. IEEE Transactions on Information Forensics and Security, 2013(12) :1947-1958.

[2] Lan Zhou,Vijay Varadharajan and Michael Hitchens. *Trust Enhanced Cryptographic Role-Based Access Control for Secure Cloud Data Storage*. IEEE Transactions on Information Forensics and Security,2015(10):2381-2395.

[3] R.K.Banyal,V.K.Jain and Pragya Jain. *Dynamic Trust Based Access Control Framework for Securing Multi-Cloud Environment.* Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies,2014.

[4] Sudip Chakraborty, Indrajit Ray. *TrustBAC - Integrating Trust Relationships into the RBAC Model for Access Control in Open Systems,* SACMAT,2006: 49-58.

[5] Tang B,Sandhu R. *Cross Tenant trust models in cloud computing*. proceedings of 14th IEEE Conference on Information Reuse and Integration(IRI),2013:129-136.

[6] Tang B,Sandhu R. *A Multi-Tenant RBAC Model for Collaborative Cloud Services*. Eleventh Annual Conference on Privacy, Security and Trust, 2013:229-238.

[7] Mohamed Amine Madani, Mohammed Erradi, Yahya Benkaouz. *Access Control in a Collaborative Session in Multi Tenant Environment.* 2015 11th International Conference on Information Assurance and Security,2015.

[8] DAVID F, Sandhu R and Richard D. *Proposed NIST Standard for Role-Based Access Control.* ACM Transactions on Information and System Security, Vol. 4, No. 3, August 2001: 224–274.

[9] David F.Ferraiolo and D.Richard Kuhn. *Role-Based Access Controls.* 15th National Computer Security Conference ,1992:554 - 563.

[10] Ravi S. Sandhu, Edward J. Coynek, Hal L. Feinsteink and Charles E. Youmank. *Role-Based Access Control Models.* IEEE Computer, 1996(2): 38-47.

[11] Lik Mui,Mojdeh Mohtashemi, Ari Halberstadt. *A Computational Model of Trust and Reputation for E-businesses.* Proceedings of the 35th Hawaii International Conference on System Sciences, 2002.

[12] M. Blaze, J. Feigenbaum, and J. Ioannidis. *The KeyNote Trust Management System Version 2.* Internet Society, Network Working Group. RFC 2704,1999.

[13] M. Blaze, J. Feigenbaum, and J. Lacy. *Decentralized Trust Management.* In Proceedings of 17th IEEE Symposium on Security and Privacy,Oakland, California, USA, May 1996: 164–173

[14] P. Bonatti and P. Samarati. *Regulating Service Access and Information Release on the Web.* In Proceedings of the 7th ACM COnference on Computer and Communication Security, Athens, November 2000:pages 134–143.

[15] Audun Josang. The Beta Reputation System. 15th Bled Electronic Commerce Conference eReality: Constructing the eEconomy Bled,2002: 324-337.