International Journal of Security and Its Applications
Vol.8, No.5 (2014), pp.149-160
http://dx.doi.org/10.14257/ijsia.2014.8.5.15

Trust-based Access Control Model in Multi-domain Environment

Zhang Qikun[1], Wang Ruifang[1], Qu Jiaqing[2], Gan yong[1] and Zheng Jun[3]

[1]*Institute of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, P. R. China*
[2]*Shanghai Radio Equipment Research Insitute, 200090, P. R. China*
[3]*School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, P. R. China*
E-mail: wangruifang29@163.com

Abstract

Access control is a process which control users to execute some operations of access some network resource according to the users identify of attribution. This paper analyzes current access control model, and extends the RBAC (role based access control) model, and based on which we propose a trust based access control model in Multi-domain environment (MD-TRBAC). Design a heap-based trust management mechanism for MD-TRBAC Model, which is used to control the cross-domain access resource among different domains. The MD-TRBAC model can provide more security, flexible and dynamic access control mechanism, and therefore improve both the security and the reliable of authorization mechanism.

Keywords: *Trust, access control, multi-domain, heap*

1. Introduction

It is necessary to share resources and security interoperability among multiple trust domains in cloud computing environment. Access control (AC) is one of the key technologies to guarantee the security for interoperability among multiple trust domains. For some resources often not belong to the same security management domain, the security interoperability among multiple trust domains has the nature of cross-domain and dynamics. The traditional identity-based access control technologies are unable to meet the security needs of multi-domain access in cloud computing environment. Currently, the main access control technology in cloud computing are identity and access management (IAM) technology, but the IAM technology is not very effective to solve the cross-domain access control and authorization issues in cloud computing environment.

In collaborative systems each domain implements a different AC policy in its environment. Hence, an essential security requirement is to preserve secure inter-operation. Specifically, secure inter-operation requires that the principles of autonomy and security should be guaranteed, as stated in reference [1]. The principle of autonomy states that if an access is permitted within an individual system, it must also be permitted under secure inter-operation. On the contrary, the principle of security states that if an access is not permitted within a system, it must also be denied under secure inter-operation. In the existing literature, there are several approaches that preserve the principles of secure inter-operation in RBAC models. B. Shafiq [2] proposes an integer programming (IP)-based approach for optimal resolution of the examined conflicts. In reference [3] an inter-domain role-mapping approach based on the least privilege principle is suggested. Waleed W., *et al.*, [4] proposes an extended access control model based on attributes associated with objects and subjects. It incorporates trust

ISSN: 1738-9976 IJSIA

and privacy issues in order to make access control decisions sensitive to the cross-organizational collaboration context. Youna Jung, *et al.,* [5] propose the CPBAC (Community-centric Property Based Access Control) model, which extends the existing CRiBAC (Community-centric Role interaction Based Access Control) model for use in online social networks to support cooperation among users. D. Unal, *et al.,* [6] proposes a new formal security policy model for multi-domain mobile networks, called FPM-RBAC, Formal Policy Model for Mobility with Role Based Access Control. FPM-RBAC supports the specification of mobility and location constraints, role hierarchy mapping, inter-domain services, inter-domain access rights and separation of duty. Hongxin Hu, *et al.,* [7] proposes an approach to enable the protection of shared data associated with multiple users in online social networks. We formulate an access control model to capture the essence of multiparty authorization requirements, along with a multiparty policy specification scheme and a policy enforcement mechanism. Ruj, *et al.,* [8] proposed an access control scheme based on Multi-authority Attribute Based Encryption. Their objective is to provide fully distributed data access control by using several Distribution Centers (DCs). Chang Choi Junho Choi Pankoo Kim [9] proposes Onto-ACM (ontology-based access control model), a semantic analysis model that can address the difference in the permitted access control between service providers and users. Zou, *et al.,* [10] proposed imposing multi-grained constraints on the RBAC model in the multi-application environment and it shows the authorization process of the proposed model.

The contribution of this article, we combined with trust management mechanism to establish the heap-based access control model, the model is suitable to access control among multiple domains in cloud computing environment. It can dynamically adjust the access decisions according to the changes of trust degree, and it has a good safety supervision mechanism during the access control. The performance analysis shows that the proposed scheme is highly efficient. Specific contributions are as follows:

1) establishing the heap-based access control model;
2) proposing multi-domain access control mechanism in cloud computing environment;
3) analyzing the performance of multi-domain access control by experiment.

2. Preliminaries

The Bayesian decision theory: Assuming the overall probability distribution is $f(x,\theta)$, $\theta \in \Theta$ is the unknown parameter, sample drawn from the overall is $X_1,...,X_n$, parameter estimation can be derived as follows by using the sample and θ [11].Bayesian Estimation:

1) Take the unknown parameter θ as a random variable (or random vector), and before sampling take the already known information of θ as priori knowledge. Use a certain probability distribution $h(\theta)$ to represent such a priori knowledge, and this probability distribution $h(\theta)$ is called the "priori distribution" of θ. This distribution reflects the probability distribution of the information obtained about the unknown parameter θ before experiment.

2) Define the distribution function $f(x_1,\theta)...f(x_n,\theta)$ of the sample $X_1,...,X_n$ containing the parameter θ as the conditional distribution function of $(X_1,...,X_n)$ on condition of the given θ. So the joint probability density function of $(\theta, X_1,...,X_n)$ is $h(\theta)f(x_1,\theta)...f(x_n,\theta)$, and the marginal probability density of $(X_1,...,X_n)$ is $p(X_i,...,X_n)=\int_{\theta\in\Theta} h(\theta)f(X_1,\theta)...f(X_n,\theta)d\theta$.

3) Propose the conditional distribution function of θ on condition of the given $X_1, \ldots, X_n$ is:

$$h(\theta \mid X_1, \ldots, X_n) = \frac{h(\theta) f(x_1, \theta) \ldots f(x_n, \theta)}{p(X_i, \ldots, X_n)}$$

which is called "posterior probability density" of θ. The function represents the probability distribution of knowledge about θ after obtaining the sample $X_1, \ldots, X_n$; and comprehensively reflects the priori distribution of θ and the information brought by the sample.

4) Make the inference of θ by $h(\theta \mid X_1, \ldots, X_n)$.

3. The Heap-based Access Control Model

The heap is a specialized tree-based data structure that satisfies the heap property: If A is a parent node of B then the key of node A is ordered with respect to the key of node B with the same ordering applying across the heap. Either the keys of parent nodes are always greater than or equal to those of the children and the highest key is in the root node (this kind of heap is called max heap) or the keys of parent nodes are less than or equal to those of the children and the lowest key is in the root node (min heap). In other words, a binary heap is a complete binary tree which satisfies the heap ordering property. The ordering can be one of two types:

1) The max-heap property: the value of each node is less than or equal to the value of its parent, with the maximum-value element at the root.

Using this data structure, we construct a heap-based access control model. Each domain in the cloud networks negotiates a different trust degree value, according the trust degree values of different domains to construct a big root heap. Making the higher trust degree value of the domain is close to the top of the heap. Model structure is shown in Figure 1:

The model mainly consists of the following modules:

1) Authorization server: receiving a user's request and deciding whether allow the user to access the local resources according to its trust degree value.

2) Token generation module: packaging the user's request information, identity and trust degree value as a token.

3) shortest-path calculation module: According the user's source domain and he will access destination domain to calculate an access path in the heap.

4) Proxy module: receiving a token from its child nodes, getting a user's cross-domain trust degree value from the cross-domain trust table and written it to the token, and then passed the token to its father node.

5) cross-domain trust table: storing the different domains' trust degree value.

6) inner-domain trust table: storing the members' trust degree value in the inner-domain.

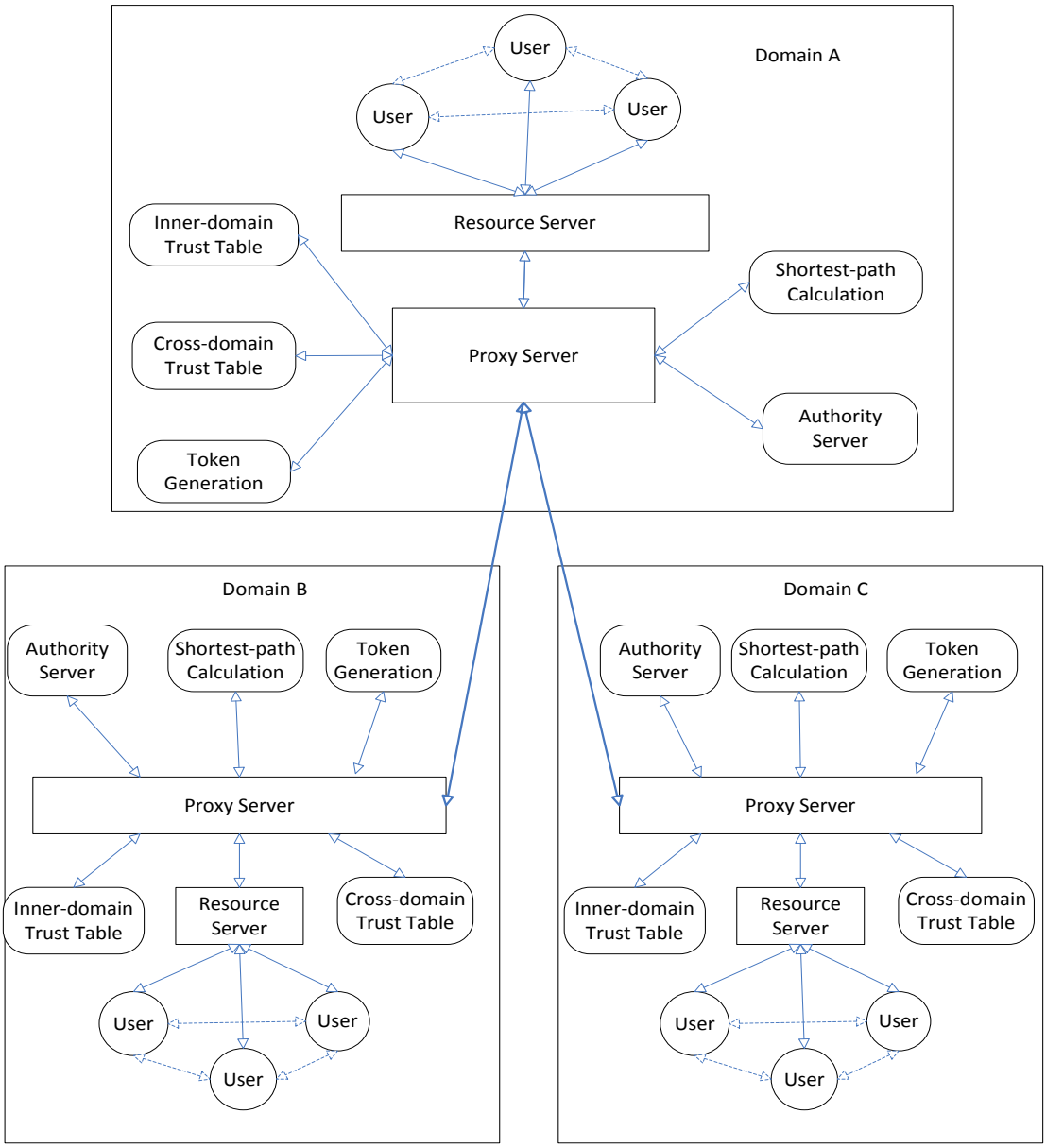

Figure 1. The Heap-based Access Control Model

4. The Calculation of Trust Degree Value

4.1 Calculation of the Trust Degree Value

Calculate the direct trust within domain by using Bayesian decision theory to estimate the success and failure rate of a certain service.

Assuming the interaction between node i and node j is random, the evaluation sequence of node i to node j is $ES_{ij}.Rat = \{es_{ij}^{1}.Rat, es_{ij}^{2}.Rat, ..., es_{ij}^{N}.Rat\}$, $ES_{ij}^{+} = \left\{ es_{ij}^{n} \middle| es_{ij}^{n} \in ES_{ij}^{n}, es_{ij}^{n}.rat = 1 \right\}$ expresses the positive evaluation sequence set of node i to node j ,

$ES_{ij}^{+} = \left\{ es_{ij}^{n} \middle| es_{ij}^{n} \in ES_{ij}^{n}, es_{ij}^{n}.rat = 0 \right\}$ represents the negative evaluation sequence set of node i to node j, supposing the number of positive evaluations is $Z_{ij} = |ES_{ij}^{+}|$ and the number of negative evaluations is $F_{ij} = |ES_{ij}^{-}|$. Suppose the probability of successful interactions is p and the failed is q, the Bayesian conditional expectation estimates of p and q will be

$$\tilde{p} = \frac{Z_{ij} + 1}{N_{ij} + 2}, \tilde{q} = \frac{F_{ij} + 1}{N_{ij} + 2}, \tilde{p} + \tilde{q} = 1$$

Therefore, node i can estimate the probability of success of the interaction between node j and itself.

Suppose node i and node j are not connected, $Z_{ij} = |ES_{ij}^{+}| = 0$, and $\alpha = Z_{ij} + 1 = 1$; $F_{ij} = |ES_{ij}^{-}| = 0$, and $\gamma = F_{ij} + 1 = 1$. The original probability density of p is $Beta(\alpha, \gamma) = Beta(1,1)$, which is evenly distributed on [0, 1], therefore $\tilde{p} = \frac{\alpha}{\alpha + \beta} = \frac{1}{2}$. When they are connected, which is to say that $\alpha = 2, \beta = 2$, $\tilde{p} = \frac{2}{2+2} = \frac{1}{2}$. With the increment of evaluation, node i will know more about node j, and p will be more accurate.

4.2 Calculation the Trust Degree Value of Each Entity in Inner- Domain

1) Calculation the Direct Trust Degree Value of Each Entity

Suppose Z_{ij} and F_{ij} each represents the number of positive evaluations and negative evaluations, $\alpha = Z_{ij} + 1$ and $\gamma = F_{ij} + 1$. Suppose p is the probability of successful and q is the probability of failed interactions of node i to node j, $E(h(p|\alpha, \gamma))$, $E(h(q|\alpha, \gamma))$ each means the mathematical expectation of Bayesian estimation. Then the calculation of direct trust can be as follows:

$$DTV_{ij} = \begin{cases} E(h(p|\alpha, \gamma)) - E(h(q|\alpha, \gamma)) = \frac{\alpha - \gamma}{\alpha + \gamma}, (\alpha > \gamma) \\ 0 \qquad\qquad\qquad\qquad , others \end{cases}$$

2) Calculation the Recommended Trust Degree Value of Each Entity

Calculation of recommended trust of node i to node j: When finding the direct trust table of node j, we construct recommendation network by recursively searching node k that is directly interacts with node j, calculate the recommended trust through the pass and synthesis relation of trust. In order to avoid finding too deeply, the recursion depth should be limited, so that the influence to calculation from trust path could be ignored at the same time.

Definition 1: Trust intensity represents the reliability of trust in progress of recommendation trust delivery; it reflects the main entity's belief degree of direct trust. Use I to represent the trust intensity, and $I \in [0,1]$.

Definition 2: Recommendation trust includes the direct trust value of object entity (calculated according to the entities that have direct interactions with the object entity) and trust intensity of direct trust value, which is to say that recommendation trust consists of the

direct trust value and trust intensity. Recommendation trust is represented as (T, I) and is called recommendation trust vector or trust vector in short.

(1)Delivery of trust relation

The recommendation trust will attenuate in progress of trust delivery, performs as the attenuation of trust intensity, as shown in Figure 2 (a). Suppose the trust value of k to j got from direct experience is T_{kj}, the trust value of i to k is T_{ik}, then the recommendation trust vector that k recommends to i is (T_{kj}, 1), after receiving the recommendation trust from k and the other entities, i synthesizes them and finally get the trust relation. The attenuation formula of trust intensity is: $I_{ij} = T_{ik} I_{kj}$. Then the recommendation trust vector of entity i to entity j is (T_{kj}, T_{ik}), which means that the trust value of k to j get from direct experience is T_{kj}, and we can get T_{ik} as the conclusion of credibility of i to j. When there are multiple intermediate entities, the process is as the same [12].

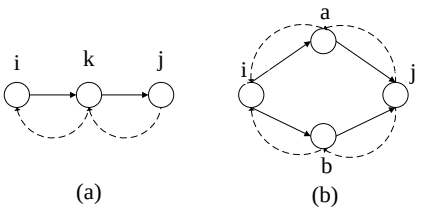

Figure 2. Trust Delivery and Synthesis

(2)Synthesis of the trust vector

To synthesize the trust vector is to respectively synthesize the direct trust value and trust intensity. Synthesize the direct trust value by taking strength as the weight of trust.

As shown in Figure 2 (b), according to the attenuation principle in previous section, upon the recommendation of intermediate entities a and b, i can get two recommendation trust vectors (T_{ij1}, I_{ij1}) and (T_{ij2}, I_{ij2}). Then based on the above analysis, the synthetic trust value of i to j is

$$IDTV_{ij} = \frac{I_{ij1} T_{ij1} + I_{ij2} T_{ij2}}{I_{ij1} + I_{ij2}}.$$

When there are multiple intermediate recommended entities in parallel, the synthetic trust value of i to j will be:

$$IDTV_{ij} = \frac{\sum_{k=1}^{n} I_{ij_k} T_{ij_k}}{\sum_{k=1}^{n} I_{ij_k}}$$

If the two intermediate recommended entities a and b have the same recommended trust values, I_1 and I_2, which means that there are two evidences to prove that the recommended conclusion is true, and the possibilities are $I1$ and $I2$, so for comprehensive consideration, the possibility (synthetic trust intensity) that the recommended conclusion is true is:

$I = 1 - (1 - I_1)(1 - I_2)$ .When there are multiple intermediate recommended entities in parallel, the synthetic trust intensity will be: $I = 1 - \prod_{k=1}^{n} (1 - I_k)$.

But if the recommended trust values from the intermediate recommended entities are different, we should firstly synthesize the direct trust value and then get the synthetic trust value. In this case, intermediate entities firstly synthesize the direct trust value by the trust values of themselves, and then calculate the synthetic trust value by using the above formula [13].

(3) The trust degree value of each entity in inner-domain

Calculation of direct trust between entities: By calculating the direct trust value and recommendation trust value, the trust value between entities can be calculated by the formula $TV_{ij} = \beta DTV_{ij} + (1 - \beta) IDTV_{ij}, (0 \le \beta \le 1)$ with an appropriate weighting factor β .

4.3 Calculation the Across-Domain Trust Degree Value of Each Domain

The calculations of the direct trust value and the recommended trust value across domains are as the same with the calculations within domain. The formula for calculating the cross-domain trust value is

$$DOMTV_{ij} = (\beta DOMDTV_{ij} + (1 - \beta) DOMIDTV_{ij})\alpha + (1 - \alpha)\varphi_i, (0 \le \alpha \le 1), 0 \le \beta \le 1)$$,

$DOMDTV_{ij}$ means the direct trust values between domains, $DOMIDTV_{ij}$ represents the recommended trust values of a certain domain, φ_i is the trust value of node i got from the trust table kept by proxy server in this domain.

5. Access Control in Multi-domain Environment

Accord the cross-domain trust degree values of each domain to establish a large toot heap. For each domain, the parent node is responsible for evaluating the cross-domain trust degree values. The process of cross-domain access control is shown in Figure 3. When the entity B in domain 3 who wants to access the resources of the domain 5, the process is as follows:

Step 1. The entity B sends his request information and its identity to authority server in the same domain. The authority server issues a certificate and a role B to her according to his request information and his trust degree value, and then entity B sends the certificate, role and request information to the proxy.

Step 2. The proxy gets the path information from the Shortest-path calculation module. The Shortest-path calculation module calculates the path of the destination domain that the entity B will be accessed, and then return the path to the proxy.

Step3. The proxy in domain3 sends the entity B‘s related information to the proxy in domain 5.The authorization server in domain 5 receives the information from the proxy in the same domain, and then calculates a minimum permission role B for the entity B according to the information of entity B and the security degree of resource.

Step 4. Authority server in Domain 3 creates a token for entity B and writes the role information, the weigh of role and the trust degree value of entity in the token, and then passes the token to his farther node. His father node writes the trust degree value of domain 3 in the token, and continues to upload the token, until pass the token to the most recent common ancestor node of domain 3 and domain 5 in the heap.

Step 5. Similar to step 4, the authorization server in domain 5 also creates a token, and writes the security degree of resource that will be accessed, the minimum permission of

resource and the trust degree value in the token, and also passes the token up layer by layer, until it is passed to the most recent common ancestor node of domain 3 and domain 5 in the heap. In each layer, the trust degree value of each domain also is written to the token.

Step 6. the domain1 is the ancestor node of domain 3 and domain 5, the authority server of domain1 calculates the trust degree value V3 of entity B in domain3 and calculate the security degree of resource R5 in domain5 by the trust degree algorithm of section 3.If V3>R5,the authority server of domain 5 assigns a role B to the entity B, and the entity B get a access permission of role B in domain 5.

Step 7. the authorization server monitor entity B's behavior, during the course of the entity B to access the resource in domain 5, if the entity has a illegitimate action to access the resource, then authorization server revokes his access permission, and passes a repatriation token along the path ②.Each domain received the repatriation token on the Path ② will reduce the trust degree value of domain 3.

Step 8. When the end of access resource, the session is terminated, and the role B of the entity B is also immediately be revoked by the authorization server in domain5, and then the administrator can write some of the audit information.

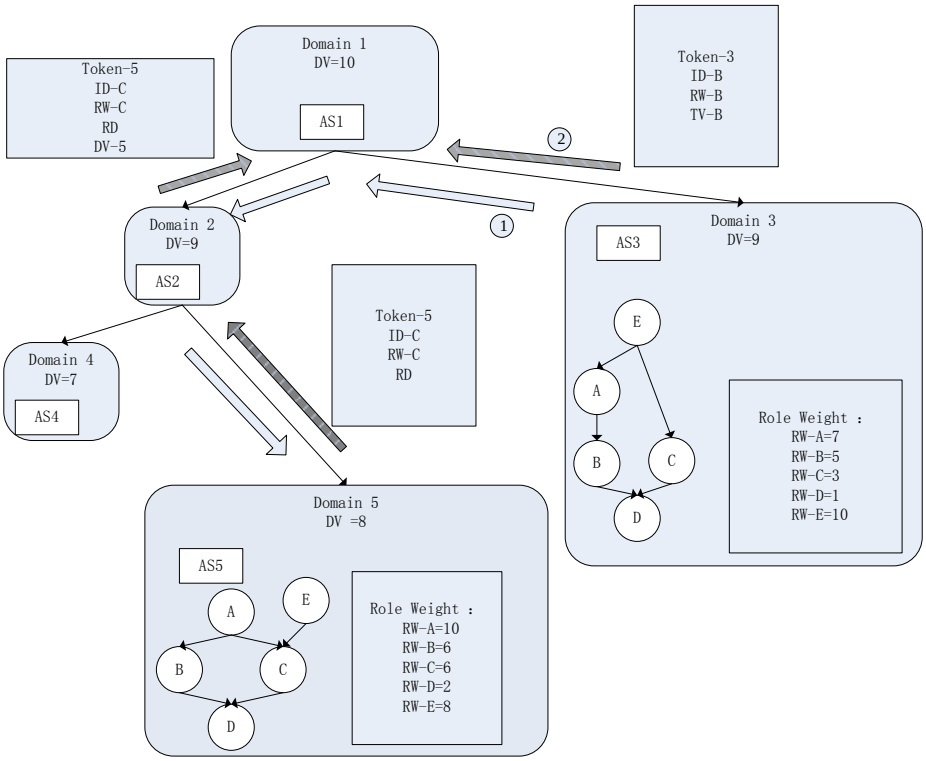

Figure 3. The Process of Cross-domain Access Control

DV is the trust degree value of domain; ID is the identity of entity; AS is the authority server of domain; RW is the weigh of role; TV is the trust degree value of entity; RD is the security degree of resource ①; is the information flow of passing certificate ②; is the information flow of passing token.

The pseudo-code is as follows:


```
requestPrim(target){
        path=askForPath(target); // calculate the access path
        createRequest(priv); // Create a request for a request
        createToken(credentials); // token initialization
        primeToken(path, LCA(this,target)); // the requester transmits his token
}
requestHandle(request,path){
        mini_role=calculate_mini_role(request);// calculate a minimum permission according to the
information of entity and the security degree of resource.
        createToken(mini_role); // Create a token for a role
        primeToken(path, LCA(this,request.source));// the resource domain passes his token
}
writeCredential(token){
    write (token, this, token.source); //write the trust degree value in the token
}
tokenJustify(tokenSource,tokenTarget){
        src=calculateTokenCredential(tokenSource);
        dst=calculateToken(Credential(tokenTarget); //compare the trust values of the two tokens
        if (src>=dst){allow access;}
        else {deny access;} // If the total trust value of the requester is higher than the trust value of
the resource, allow the requester to access, otherwise refuse it
}
```

6. Performance Analysis

In MD-TRBAC model proposed in this paper, we build both trust relations between entities and domains; take different algorithms to calculate their trust values according to their natures and characteristics, and finally make accurate assessment of their trust relations.

The scenario supposed in this paper is interactions between entities within a domain, in which a user aims to accessing an interested node, and it doesn't matter whether he wants to upload or download a resource he wants or even just a simple accessing. The concerns we care most are whether the source node is being recognized by the target node and the recognition accuracy. There are totally 40 nodes in this experiment, which is divided into two types: honest nodes, they use services provided by the network safely and rationally, and can accurately rate collaborations between entities; dishonest entities, they use the services unreasonably, and they may even cause threats to the service providers. The weight parameter β in trust formula is set to be 0.9, which means that entities pay more attention to the direct trust value of access nodes rather than indirect values from other entities.

Experiment 1: We observe the changes of trust relations between entity i and both honest entities and dishonest entities along with the increase of interactions. It is supposed that honest entities and dishonest entities have the same number in our experiment. Simulation parameters are shown in Table 1.

Table I. Simulation Parameters

Total number of entities	40
Honest entities	50%
Dishonest entities	50%
Original direct trust value	0.5
Threshold of trust	0.4
Weight parameter β	0.9

Figure 4 shows the trend of trust relation changes between entity i and both honest entities and dishonest entities along with the increase of interactions. Since the original direct trust value is 0.5 and it's above the threshold 0.4, so entities at the beginning can access each other. In Figure4, horizontal axis represents the times of interactions; vertical axis represents the trust value; green line represents the trust value changed trend of honest entities with the number of interaction, while blue line represents the trust value changed trend of dishonest entities with the number of interaction.

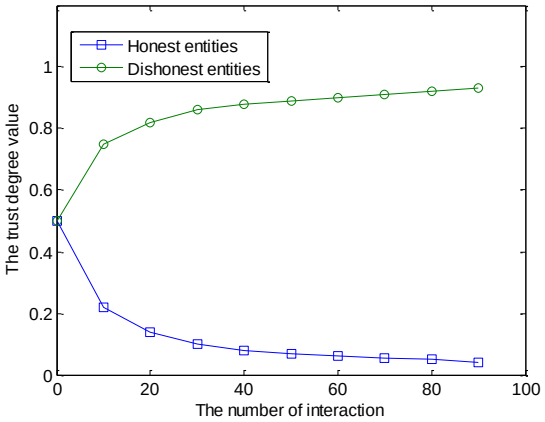

Figure 4. Trend of Trust Relations Along with Interactions Increase

As we can see in Figure 4, along with the increase of interaction times, trust values of honest entities is gradual increase while the trust values of dishonest entities is gradual decrease. As a result, the resource entity could pre-judge whether the access entity is honest or dishonest, and then it decide to permit or reject the access.

Experiment 2: Compare the accuracy of detecting malicious behavior in both MD-TRBAC model and EigenTrust Model. The simulation parameters are the same as experiment 1, Figure 5 shows the result:

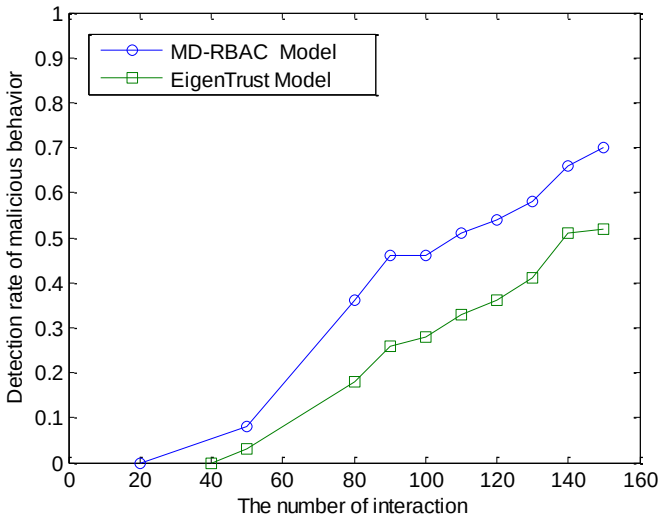

Figure 5. Comparison of Malicious Behavior Detecting Accuracies in Two Models

Blue line represents the accuracy of MD-TRBAC Model proposed in this paper, while the green line means the result of Eigen Trust Model. In Figure 5 shows that MD-TRBAC Model has faster convergence than Eigen Trust Model.

7. Summary

In this paper, we deeply analyzed several issues of RBAC model, which include the lack of commission control depth in a distributed environment, the inefficiency of cascading revocation of the authorization roles and the incapability of judging whether the commission violated the principle of RBAC model before it is done and so on. To deal with these problems, we proposed MD-TRBAC model, designed trust management mechanism of MD-TRBAC Model, which was used to control the access, established the credible authority commission tree and finally proposed the detection algorithm for implicit upgrade of the role's authority to avoid violation of the least privilege principle in RBAC model. The experiments and analyses prove the feasibility, effectiveness and safety of MD-TRBAC model.

Acknowledgements

This work is supported by National Natural Science Foundation of China under Grant No. 61272511, 61272038 and 61340059, the PhD Research Fund of the Zhengzhou University of Light Industry and National High-tech R&D Program of China (863 Program) (Grant No. 2013AA01A212).

References

[1] L. Gong and X. Qian, "Computational issues in secure interoperation", **(1996)**.
[2] B. Shafiq, J. B. D. Joshi, E. Bertino and A. Ghafoor, "Secure interoperation in a multi-domain environment employing RBAC policies", IEEE Trans. on Knowl. And Data Eng., vol. 17, no. 11, **(2005),** pp. 1557– 1577.

[3] L. Chen and J. Crampton, "Inter-domain role mapping and least privilege", in SACMAT ' 07: Proceedings of the 12th ACM symposium on Access control models and technologies, New York, NY, USA: ACM, **(2007)**, pp. 157– 162.
[4] W. W. Smari, P. Clemente and J.-F. Lalande, "An extended attribute based access control model with trust and privacy: Application to a collaborative crisis management system", Future Generation Computer Systems, vol. 31, **(2014)**, pp. 147– 168.
[5] Y. Jung, J. B. D. Joshi, "CPBAC: Property-based access control model for secure cooperation in online social networks", computers & security, vol. 4, no. 1, **(2014),** pp. 1 9 -3 9.
[6] D. Unal and M. U. Caglayan, "A formal role-based access control model for security policies in multi-domain mobile networks", Computer Networks, vol. 57, **(2013)**, pp. 330– 350.
[7] H. Hu, G.-J. Ahn and J. Jorgensen, "Multiparty Access Control for Online Social Networks: Model and Mechanisms", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, vol. 25, no. 7, **(2013)** July, pp. 1614-1627.
[8] S. Ruj, A. Nayak and I. Stojmenovic, "Distributed fine-grained access control in wireless sensor networks," in IPDPS, **(2011),** pp. 352–362.
[9] C. Choi,·J. Choi and·P. Kim, "Ontology-based access control model for security policy reasoning in cloud computing", J Supercomput, vol. 67, **(2014),** pp. 711–722.
[10] D. Zoua, L. Heb, H. Jina and X. Chenc, "CRBAC: imposing multi-grained constraints on the RBAC model in the multi-application environment", J Netw Comput Appl., vol. 32, no. 2, **(2009),** pp. 402–411.
[11] M. Zhimao, "Test and Research on Bayes with Dynamic Parameters", National University of Defense Technology, **(2009)**.
[12] Y. Gang, "Research and Implementation on Authorization Management in Inter-Domain Computing Environment", National University of Defense Technology, **(2006)**.
[13] L. Junguo, "A Dissertation Submitted to Huazhong University Technology", **(2007)**.

Author

Zhang Qikun, Vice Professor, Ph.D. Zhengzhou University of Light Industry , Zhengzhou. China. His research interests include information security and cryptography.

