The Design and Simulation of a New Dynamic Credit and Role based Access Control Strategy

Haibo Gao, Wenjuan Zeng
College of Information Science and Engineering, Hunan International Economics University, Changsha, China
37802055@qq.com

Xiaohong Deng
College of Applied Science, Jiangxi University of Science & Technology, Ganzhou, China dxh_lizi@sohu.com

***Abstract*—In order to solve the shortage of the access control method based on role and credit, a novel dynamic credit and role based access control method is proposed. The method controls an entity's access behavior through its credit value, and introduces penalty and feedback mechanism to dynamically adjust entity's credit value. In order to manage an amount of entities effectively, entities with different credit values are divided into different roles in system, and roles are assigned to corresponding privileges respectively. Experimental simulation results showed that the proposed method has good controllability of entity's credit value and manipulability to protect system's resources.**

***Index Term*—role, credit, access control, penalty, feedback**

I. INTRODUCTION

Role-based access control strategy (RBAC) mainly considers how to authorize privileges to users, but don't paid attention to the operational behavior of users that has got system's access privileges. So, it may lead to serious security risks. For example, if the control system endows user's access authorization is not reasonable, which does not meet the principle of "least privilege", in other words, the authorized privilege is excessive. In addition, even if the authorized privilege is reasonable, but the "legitimate" user has malicious operation, it will damage and reveal of the system resources. The biggest drawback of RBAC is that user's privilege is relatively fixed and lack of dynamic adjustment [1-3]. In addition, it is unable to control user access behavior. Therefore, it will lose the protection of system resources.

Credit comes from the human social life, has been widely used in the economic, political and social fields [4-6]. The concept of credit in computer system is the evaluation of reliability degree of a resource access user and its behavior. The introduction of credit mechanism can promote users to regulate their own behavior when they access system resources, if their access behavior did not affect the integrity and security of resources, then they will get a good reputation, and will be trusted by the control center. Therefore, they will have a high credit value. On the contrary, there will get a low credit value. That user has low credit value will access system resources difficult, even can't login in system. Research on credit based access control (CBAC) strategy has important meaning to the system's resource protection under the network environment, the credit mechanism makes the visitors' access behavior controllable, and visitors' credit can be dynamic adjusted along with its own access behaviors. In addition, the credit mechanism makes the individual pays more attention to its credit, and strengthens the credit concept in the social behavior. CBAC makes up the shortage of RBAC method, and has great economic, social, and technology value.

But the CBAC's disadvantage is that once the credit value is set, it is difficult to adjust. E.g., if an entity with high credit value has illegal access behaviors, its credit can not be adjusted. In this paper, the dynamic credit and role based access control model (DCRBAC) is proposed to solve the problem of CBA and RBAC methods. In the proposed method, each resource visitors have their own credit value, which can quantizes their own access behavior, and can be dynamically adjusted. Each resource visitor's credit value in system belongs to a credit level, and different credit level corresponds to a different role.

II. THE MODEL OF DCRBAC

A. The Definition of Model

The access control model of DCRBAC is shown in Fig.1. Firstly, each resource visitor's credit value can be computed according to the credit calculation strategy, and then be divided to a credit value level. Secondly, the visitor's role is assigned according to the access control policy, and the different roles have different privileges. Finally, visitors which have obtained some privileges can access system's resources.

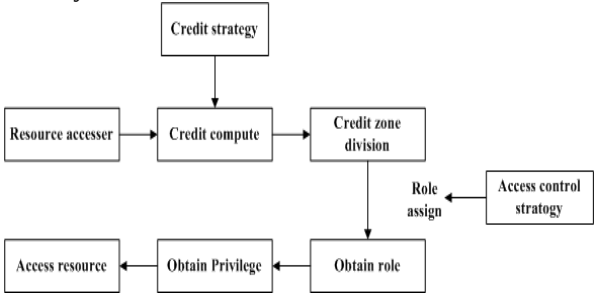

Figure 1. The overall model of DCRBAC

doi:10.4304/jcp.9.2.506-511

Definition 2.1 Resource Access User(RAU). RAU is the subject of resource access, can be denoted as $RAU = \{rau_1, rau_2, \cdots, rau_n\}$.

Definition 2.2 Resource. Resource is the information protection object in access control system, is also the object of RAU, represented by R, $R = \{r_1, r_2, \cdots, r_n\}$.

Definition 2.3 Request for Services(RS). The resource access users request for services, can be denoted by *RS*. $RS = \{< aru_i, r_i > | 1 \le i \le num\}$, *num* is the total number of resources can be accessed.

Definition 2.4 Access Privilege(AP). The resource access user has operation privileges to object. *AP* denotes the set of privileges, AP={Read, Write, Copy, Execute, Refuse}.

Definition 2.5 Role. Role is a set of users that have the same operate privileges. Once a role defined, can not be modified frequently. When a user is assigned to a role, it plays this role and has some responsibility. Role is delegated to privilege, and each role has a group of privileges. Role also is the bridge between user and privilege, user belongs to Role, and Role has privileges. $Role = \{Role_1, Role_2, \cdots, Role_n\}$.

Definition 2.6 Authorization. Authorization is assign some privileges to role, set *A* is the set of Authorization, $A = \{a_1, a_2, \cdots, a_n\}$. Authorization is a quad tuple, can be denoted by $< AP, Role, R, condition >$, if the *condition* is satisfied, the *AP* is assigned to *R*, or else refuse.

Definition 2.7 Division. Divide the different *RAU* to the correspond role, set *D* is the set of division, $D = \{d_1, d_2, \cdots, d_n\}$.

Definition 2.8 Credit. Credit is divided into static and dynamic credit [7], represented by *SC* and *DC* respectively, *SC* is the credit got by resource visitor when the authorization and division was completed. However, *DC* is the credit updated by system through auditing the access behavior of resources visitors (such as reward credit value is positive, punish credit value is negative), total credit value is denoted by $TC = SC + DC$.

Definition 2.9 Credit Threshold(CT), Credit Threshold is the threshold value to divide credit level, there are three different thresholds: $CT_1 < CT_2 < CT_3$.

Definition 2.10 Credit Level. Credit Level has four grades, and different credit level is got according to *RAU*'s total credit *TC* and *CT*. The "distrust" corresponds to credit level 1, "basic trust" is credit level 2, "trust" is 3 and "fully trust" is 4.

The proposed access control strategy can be represented by $< RAU, Role, AP, TC, D, R >$. The system access control center assigns *AP* to *R*, and divides different *RAU* to *D*. In an actual access procedure, the visit behavior of *RAU* on resource *R* is determined according to the general trust credit value of *TC*.

B. The Mechanism of Access Control

The access control mechanism considers how to implement the access control policy [8], this section uses an office automation system or management information system under open environment as an example, and explains the DCRBAC methodology applied to the digital management information system based on Web, a feasible scheme is schematically shown in Fig. 2.

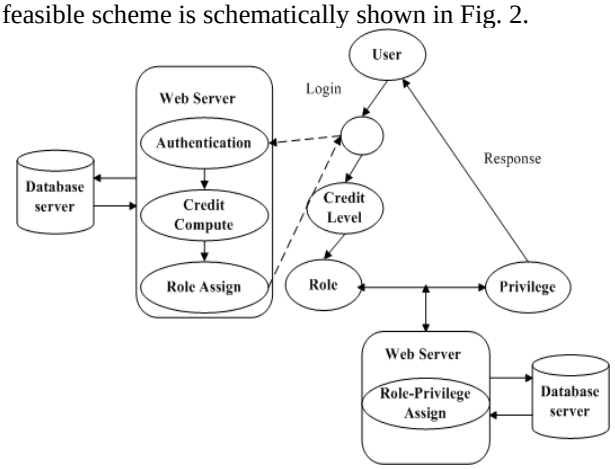

Figure 2. The implementation scheme of DCRBAC

The specific steps of access control mechanism are shown as follows:

(1)User provides a login request through the remote network or the enterprise internal network;

(2)Web server authenticates the user's identity, check the database's user name and password, if match, pass the identity authentication; or else, refuse;

(3)Web server computes the user's new credit value according to the current user's old credit value and historical records of access behaviors;

(4)Web server assigns the current user to a predefined role according to the user's credit value;

(5)Web server reads the corresponding relation table between roles and privileges in the database, gives corresponding privileges of role to the user, thus, the preparation of user access resources is completed;

(6)The user requests for access resource, and Web server determines whether the operation authority request is legal or not. If legal, the current operation is accepted, and records the operation in legitimate data access log;

(7)If the user request for the resource operations over its role's privileges, this operation is rejected, and record this operation in the illegal data access log.

C. The Division of Credit Level and Authorization

Table I gives RAU's credit level, conditions and access privileges. Here, Defuse denotes RAU can not access R, Read, Copy, Execute, and Write represents the resource R can be read, copied, and written.

TABLE I.
RAU'S CREDIT LEVEL, CONDITION AND ACCESS PRIVILEGE

No.	Credit level	Condition	Access privilege
1	Distrust	$TC<CT_1$	Defuse
2	Basic trust	$TC>=CT_1$ & $TC<CT_2$	Read
3	trust	$TC>=CT_2$ & $TC<CT_3$	Read、Copy and Execute
4	Fully trust	$TC>=CT_3$	Read、Copy、Execute and Write

The privilege control center has four different authorization levels: refuse, weak, strong and fully authorization, these levels is correspond to credit levels, their relation can be shown in Table II.

TABLE II.
THE RELATION OF AUTHORIZATION AND CREDIT LEVEL

Authorization level	Credit level
refuse	distrust
weak	basic trust
strong	trust
fully	fully trust

The authorization algorithm of access control based on credit and role as follows:

Input: 1. *AP,Role, R*

2. *RAU,* $< aru_i, r_i >$

Output: fully, strong, weak and refuse

Begin

1、if $AP = \varnothing \,||\, Role = \varnothing \,||\, R = \varnothing$

Return;

2、else if $r_i \notin R$

Return;

3、else read the aru_i 's TC

4、if TC<CT_1

Then refuse;

5、else if TC>=CT_1 & TC< CT_2

Then weak;

6、else if TC>=CT_2 & TC< CT_3

Then strong;

7、else fully;

End

D. The Computation of Credit Value

(1) Computing method

The most direct and simple method to compute the credit value (*CV*) of entity is using the times of normal access to resources to divide the total times of access. The total access times is equal to the sum of normal access times *N* and abnormal access times *UN*, the user's credit value can be represented by Eq.(1) as follows [9]:

$$CV = \frac{N}{N + UN} \tag{1}$$

According to the reward and punishment mechanism and practical experience accumulation of an entity's credit value, the decline's degree caused by abnormal access behavior is more than the increase's degree caused by normal access increased greatly. The formula of computing entity's credit value is shown in Eq.(2) [10]:

$$CV = \begin{cases} \dfrac{N}{N + UN} - \dfrac{1}{1 + e^{1/UN}}, UN \le N \\ 0, UN > N \end{cases} \tag{2}$$

In viewing of the entity's credit value is closely related to the credit value of last access time, this paper presents a new formula to calculate the credit value of current access entity by Eq.(3), and then get the final value through doing the weighted average and the credit value of last visit in the system database, the formula is as follows:

$$CV' = (1 - \alpha) \times old\_CV + \alpha \times new\_CV \tag{3}$$

In Eq.(3), $0 \le \alpha < 1$, If α infinitely closes to 0, illuminates the credit value's update is slow. A possible situation is that a user recently visit behavior is normal, and access operations are close; if α is close to 1, represents an entity's credit value updates fast, the corresponding situation is that entity has not recently visited behavior regularity. CV' denotes the final credit value, old_CV denotes the old credit value when the entity accesses system in last time, new_CV denotes the credit value is calculated by an entity's access record in system.

In general, the access behaviors of most visitors are regular in the actual access control system. So, α is usually taken as 0.125. Of course, its value can be dynamically adjusted according to the system's actual situation.

(2) Initialization method

According to the above-mentioned calculation method of entity's credit value, entity's credit value is related to its old credit value stored in the system database and history access record. Therefore, for a new registered entity User_A, the User_A credit value is 0, so, it could not access system. In this case, the access control policy should be given an initial value to User_A. In our method, in order to make the entity User_A to get the corresponding service, a minimum trust "basic trust" level and a value CT_1 is assign to User_A. By this, the User_A has the chance to get system's service, and this method also ensures the system's security. Because it makes a not well-known visitors have minimal access system, meets the principle of "least privilege" in information security field.

(3) Reward and penalty method

The award mechanism of entity's credit value can be realized through two aspects:

1) Increase old_CV;

2) Increase the normal access times *N*.

In our access control policy, each entity's access privileges depend on its credit value directly, so the most direct and effective way is to change the entity's credit value stored in the system database.

Another method is aim at a resource visitors authority only need a small degree promotion, but his normal access times *N* has not meet the requirements, you can achieves target by increasing *N*. It is not difficult to conclusion that, compare with the first method, the credit value increase degree is much low, this method belongs to the credit value's small adjustment.

Correspondingly, the punishment mechanism of entity credit value can also be achieved through two aspects:

1) Decrease old_CV;

2) Reduce the abnormal access times *UN*.

The first one is not routine method, if the system control center finds a trusted user's access behaviors damages the system resource, even this user's *UN* value

is small, we can directly reduce its old_CV to decrease its credit value. Another method is a common method, for most visitors, they have not single malicious access behavior in particular, so its credit value should be accumulated through his abnormal access times reducing.

(4) recovery method

In order to avoid deadlock state occurs, e.g. an entity with credit value 0 will be permanent refused by system. So a feasible entity's credit value recovery mechanism in the actual access control system is required. The specific mechanism is as follows:

1) For the entity with credit value 0, sets a timer T, if the time is more than T and the entity has not in the trusted state, and then we manually or automatically restore this entity's credit value. Generally, the restored credit value is equal to CT_1, namely basic trust threshold.

2) System records an entity's credit value recovery times, set a upper bound of MAX, if an entity credit recovery times equal to MAX, it proves the entity belongs to a malicious one, this type entity's credit value will be refused to recover. So, this type entity is classified into a blacklist.

III. Simulation Results and Discussions

Our simulation experiment selects four different service requesters as User_A, User_B, User_C, User_D (their credit level are distrust, basic trust, trust and fully trust, respectively), and their initial credit values are 0.3, 0.5, 0.7 and 0.9. Set the three thresholds $CT_1 = 0.4$, $CT_2 = 0.6$, $CT_3 = 0.8$.

In the simulation, the four users randomly request for services 30 times. The statistical analyses results are shown in Table III.

TABLE III.
THE RESULTS STATISTICS OF USER REQUEST FOR SERVICES

User name	Success	Refuse	Successful precentage
User_A	2	28	6.67%
User_B	14	16	46.67%
User_C	23	7	76.67%
User_D	28	2	93.33%

It can be seen from Table III that the request success rate of the entity whit high credit value is obviously high, because it has much more privileges. The distrust resource visitor User_A just has only 2 times success in 30 times requests, and the success rate of less than 10%. It said that the system has good control of some operations beyond of privileges, and the credit based access control strategy has good effect to control the resources usage.

TABLE IV.

THE ACTION STATISTICS OF USER'S ACCESS

User name	Total times	Normal times	Abnormal times
User_A	2	1	1
User_B	14	11	3
User_C	23	20	3
User_D	28	26	2

In the random services request, we track and evaluate the historical records of successful response of the service, and record the different times access behavior of the four users in the database table history. Table IV gives the normal and abnormal access times in access behavior for four different users.

In order to prove the correctness and validity of the calculation formula of the entity's credit value, using the simulation tool Matlab to simulate the credit value calculation formula. Suppose an entity credit value equals to 1, and the successful access times are increased from 1 to 15, and abnormal access times are changed from 1, 2, 5, 10, to 15. The test results as shown in Fig.3.

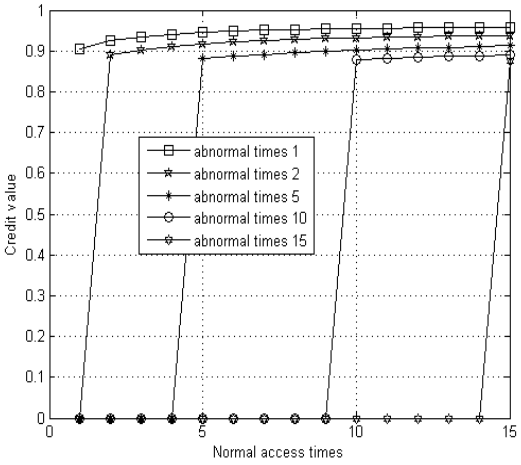

Figure 3. The impact of entity credit under different normal access times

It can be seen from Fig.3 that if the abnormal access times is fixed, entity's credit value increases with the increase of the normal access times, but the growth rate is smaller. It represents a visitor's credit value will be more and more big through the continuous accumulation of normal access times, but the growth rate is small. At the same time, if the normal access times is fixed, entity's credit value decreases with the increase of the abnormal access times, and the decrease rate is very obvious.

The entity's credit value computation not only depends on the normal and abnormal access times, but also is directly related to the weight coefficient α. In order to illustrate the influence of coefficient α to the entity's credit value, we also supposes a entity with credit value 1, and its abnormal access times is fixed to 2, but the normal access times is changed from 1 to 15 with linear growth. The coefficient α is set to 0.125 and 0.25 respectively, and the experimental results as shown in Fig.4.

From Fig.4, we can see that, the entity's credit value is relatively large when α is small, and increases with the increase of α. The reason is that when α is small, the increase of normal access times would cause entity's credit value increase, although abnormal access times can lead to credit value becomes small, and the decline degree is greater than the increased degree, However, α 's value makes the effects not significant.

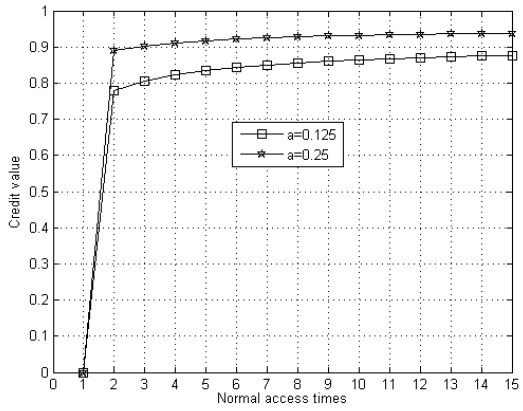

Figure 4. The impact of entity credit under different parameter α

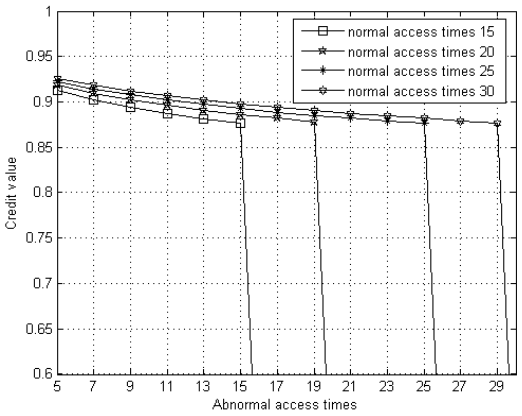

Figure 5. The impact of entity credit under different abnormal access times

Fig.5 shows the influence of abnormal access times on the entity's credit value. The abnormal access times is increased from 5 to 33 with step 2, while the normal access times is set to 15, 20, 25 and 30, respectively. The initial credit value of entity is 1, and $\alpha = 0.125$. It can be seen from figure 3.3 that the entity's credit value becomes small with the increase of the abnormal access times when the normal access time is certain, and the decrease degree is high. It represents that if a resource visitor continuously has abnormal access behaviors, its credit value will become smaller and smaller, and even will drop to 0, cause denial of service. At the same time, when the abnormal access times is certain, the more normal access times, the higher entity credit value.

In order to verify the effect of punishment mechanism on the credit value of entity, we choose an entity with the credit value 1 as the experimental object ($\alpha = 0.75$). Fig.6. shows the entity's credit value under the condition of no punishment mechanism and penalty mechanism. It can be shown from Fig.6. that if no abnormal access behavior appears (in figure, the abnormal access appears when the normal access times more than 50, and the normal access times increase 5 times, the abnormal access times increases 2)the entity's credit value is always 1. But when the abnormal access occurs and lack of punishment mechanism, the entity's credit value decreased slightly, while in case of punishment mechanism, the entity's credit value decreased significantly.

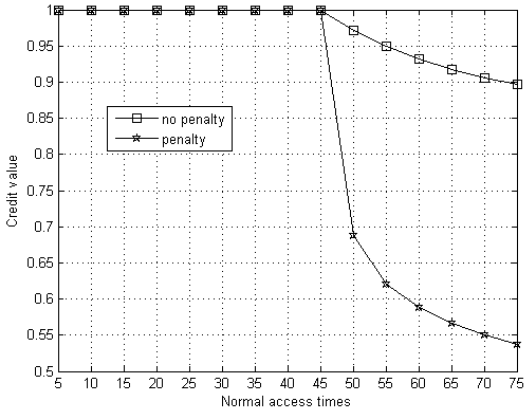

Figure 6. The impact of entity credit under penalty scheme

IV. Related Works

Access control is an important element in network security. Role-based access control is one of the most popular and widely deployed access control model. Meanwhile, trust or named credit provides a new direction for access control in open network environments. In recent years, more and more researchers pay attention to the study of access control and present many new methods. We make a review of this related works as follows.

Kumar in literature [11] uses formal concept analysis based on mathematical lattice and order theory to design a RBAC, and derive a dyadic formal context from the triadic security context that represents role-based access permission and perform attribute exploration. The proposed method has been demonstrated that it is feasible on a health care ad hoc network. Peeters et al in literature [12] introduce a single point of failure that is vulnerable to relay attacks. A threshold-based distance-bounding protocol that distributes a user's private key among various personal devices improves system security and reliability. A trust ecosystem for mobile devices based on trusted computing principles is presented in literature [13]. The trust ecosystem model from the latest PC platform provides a reference for the evolution of a trust model for the diverse mobile device market. By balancing the trusted computing burden between the mobile device and network and separating service delivery/access control decisions and error reporting from remediation, the processing in the network is hugely simplified, and efficiency in the use of network resources is increased. Based on the dynamic nature of trust, literature [14] studies the temporal and spatial characteristics in the security of society and proposes the concept of scenario trust in which four factors are considered: access time, place, history behavior and risk control strategy. Experimental results show that this model exhibits good scalability and can meet the need of dynamic access control in open network environments.

Attacking to the different network environment, researchers provide different access control model. Ruj et

al. propose a decentralized security framework for smart grids that supports data aggregation and access control [15], this is the first work on smart grids, which integrates these two important security components (privacy preserving data aggregation and access control) and the first paper which addresses access control in smart grids. Lin et al. provide a trust-based access control mechanism for cloud computing considering its multi-domain aspects [16-17]. Access control in local domain directly applies RBAC model combined with trust degree, whereas in multi-domain it contains the conception of role translation. Li et al [18] proposes an access control model for negative authorization to provide the user with the ability and flexibility of specifying the objects to which access is not desired through the means of negative authorization in cloud computing environment. The literatures [19-20] study the new risk management model and credit risk model using data mining technique.

V. CONCLUSION

The proposed DCRBAC strategy can meet the original design requirements, the resource access user's privileges can be controlled by its credit value, because credit value can be dynamically adjusted and touched with user's access behavior, the proposed method can effectively prevent the abnormal access behavior of legitimate users in the system. In addition, the credit access control strategy presented in this paper mainly depends on the security mechanism, and the data integrity based on the mature relational database technology. So, our method is high feasible, and can be applied to practical system.

ACKNOWLEDGEMENTS

The authors would like to thank anonymous reviewers for their valuable comments and suggestions that have helped us to improve the quality of the paper. This research was financially supported by Scientific Research Fund of Hunan Provincial Education Department No.10C0914.

REFERENCES

[1] A. Lazouski, F. Martinelli, P. Mori. Usage control in computer security: a survey[J]. Computer Science Review, 2010, 4(2):81-99.

[2] J. B. D. JOSHI, E. BERTINO, A. GHAFOOR. An analysis of expressiveness and design issues for the generalized temporal role-based access control model[J]. IEEE Transactions on Dependable and Secure Computing, 2005, 2(2):157-175.

[3] Z. J. Wang, Q. Liu. Extended access control model based on RBAC[J]. Computer Engineering and Applications, 2005, 41(35): 23-25.

[4] X. N. Ma. Formal Description of Trust-based Access Control[J]. Physics Procedia, 2012, 33:555-560.

[5] S. N. Ma, J. S. He, F. Gao, X. G. Sun. A trust-based dynamic access control model[J].Journal of Information and Computational Science, 2010, 7(10):2165-2173.

[6] S. N. Ma, J. S. He, F. Gao. An access control model based on multi-factors trust[J]. Journal of Networks, 2012,7(1):173-178.

[7] S. Guo, X. P. Lai. An access control approach of multi_security domain for web service[J]. Procedia Engineering, 2011,15: 3376-3382.

[8] H. Koshutanski, A. Lazouski, F. Martinelli, P. Mori, Enhancing grid security by fine-grained behavioral control and negotiation-based authorization[J]. International Journal of Information. Security, 2009,8 (4):291-314.

[9] N. Li, W. H. Winsborough, J. C. Mitchell. Distributed credential chain discovery in trust management[J].Journal of Computer Security, 2003, 11(1):35-86.

[10] Y. Sun, W. Yu. Information theoretic framework of trust modeling and evaluation for ad hoc networks[J]. IEEEE Journal on Selected Areas in Communications, 2008, 19(7):l716-l730.

[11] C. A. Kumar. Designing role-based access control using formal concept analysis[J]. Security and Communication Networks, 2013, 6(3):373-383.

[12] R. Peeters, D. Singelee, B. Preneel. Toward more secure and reliable access control[J]. IEEE Pervasive Computing, 2012,11(3):76-83.

[13] D. Howry, Y. Shah, A. U. Schmidt. Evolution of trust system from PCs to mobile devices: building secured and trusted ecosystems[J]. IEEE Vehicular Technology Magazine, 2013,8(1):70-80.

[14] G. Y. Lin, Y. Y. Bie, M. Lei. Trust based access control policy in multi-domain of cloud computing[J]. Journal of Computers, 2013, 8(5):1357-1365.

[15] S. H. Ma, J. S. He, X. B. Shuai. An access control method based on scenario trust[J]. International Journal of Computational Intelligence systems, 2012,5(5):942-952.

[16] S. Ruj, A. Nayak. A decentralized security framework for data aggregation and access control in smart grids[J]. IEEE Transaction on Smart Grid, 2013, 4(1):196-205.

[17] G. Y. Lin, Y. Y. Bie, M. Lei. Trust based access control policy in multi-domain of cloud computing[J]. Journal of Computers, 2013, 8(5):1357-1365.

[18] X. H. Li, J. S. He, T. Zhang. Negative authorization in access control for cloud computing[J]. International Journal of Security and its Applications, 2012,6(2):307-312.

[19] S. Islam, S. H. Houmb. Towards a framework for offshore outsource software development risk management model[J]. Journal of Software, 2011, 6(1):38-47.

[20] K. N. Fang, H. Huang. Variable selection for credit risk model using data mining technique[J]. Journal of Computers, 2011, 6(9):1868-1874.

Haibo Gao, born in NingXiang Hunan in 1979 male, Han, master's degree, lecturer, main research direction: network & information security.

Wenjuan Zeng, born in NingXiang Hunan in 1979 female, Han, master's degree, lecturer, main research direction: web information processing & data mining.

Xiaohong Deng, born in TianMen Hubei in 1982 male, Han, master's degree, lecturer, main research direction: digital watermarking.