IJournals: International Journal of Software & Hardware Research in Engineering
ISSN-2347-4890
Volume 4 Issue 3 March, 2016

Data Security Based HABE Algorithm for Multi-level Hybrid Cloud

LI Na[1]; DONG Yunwei[1]; JING Hongli[2]; CHE Tianwei[3]; ZHANG Yu-chen[4]

The School of Computer Science and Technology , Northwestern Polytechnical University ,Xi'an ,China[1];Beijing Topsec Technology Co. Ltd., Beijing , China[2]; Beijing Jianyin Investment technology development Co. ,Ltd, Beijing,China[3]; Information Engineering University, Zhengzhou ,China[4]
tao_lina@163.com[1]; yunweidong@nwpu.edu.cn[2]; jinghongli@sina.com[3]; tianweiche@163.com[4]; zycxz@126.com[5]

ABSTRACT

According to the security problems of ciphertext sharing in hybrid multi-level cloud such as data security, privacy protection and so on, This article designed and realized solution in hybrid multi-level cloud based on security control solution and property encryption by safety management strategy user privacy protection; on the basis of HABE solution of hierarchical multi-level authorization center, we have designed a secure, fine-grained BSP-HABE model and solution based on security control solution by safety management strategy made by data sender, authorization center and management mechanism using XACML to set environment and solution constraints.

Keywords: data security, HABE, multi-level authorization, xacml.

1. Introduction of data security in the hybrid multi-level cloud

Cloud computing[1] is a new type of computing mode, by which computing resources may be stored in a shared configurable pool and visited via an available convenient and on-demand network. Due to the strong position of cloud computing service providers in the provision of services and weak position of users in receiving services, it may cause serious information asymmetry. On the one hand, when information is transferred to the cloud computing service provider under the mode of cloud computing, it would be impossible for enterprises themselves to comprehensively control information and cloud computing details of the service provider. On the other hand, as cloud computing [2] is designed for multi-party users, the service provider will not show the key information in cloud computing to users in consideration of security. In conclusion, security has become a key problem that constrains the development of cloud computing.

In order to solve the problems concerning hybrid multi-level cloud data protection, this paper makes research on security of data storage and access.

2. Problems of data security in the hybrid multi-level cloud

2.1 Illegal Access of Cloud Service Providers

The users in the hybrid multi-level cloud value their task and the data confidentiality very much. The users cannot control the task and confidential data any more when provided to the providers of cloud computing service, which, however, the cloud computing service can access. The IT administrator and other staff of the cloud service provider may make use of it without permission or leak the confidential data. If leaked or tampered illegally, the consequences can be unimaginable.

2.2 Problems of Cross-level and Cross-cloud Data Leakage

In the hybrid multi-level cloud environment, the data visitors come from cloud systems of different level, while the way of access and control varies according to different levels. Thus it certainly will result an inconsistent intensity and access policy against the same data[3]. Further, the current access and cont

rol policy and method usually aimed at local only. So, when the data is shared among different cloud s ystems, it cannot implement the security policy of o ne cloud or multi cloud and cannot make a joint co ntrol of the users' access. More seriously, when use rs in low-level cloud visit users in high-level, it will lead to a protection intensity cut of the high-level c loud resources due to the lower access and control i ntensity of the low-level cloud.

2.3 Share of Multistage Ciphertext under Open Environment

In current cipher technology[4], whatever symmetric cryptography or asymmetric cryptography, the enc ryption party must know the shared secret key or p ublic key of the decryption party. But under open e nvironment such as hybrid multi-level cloud, the us ers cannot get the information of the decryption par ty in advance when they make a data hosting. If get , it can only do a one-to-one data encryption. One-t o-many encryption and decryption is impossible w hile the decryption party is unidentified.

3. Solution on Data Security in the Hybrid Multi-level Cloud

A cryptography based on property can realize encry ption and decryption according to the property of th e users' identification. Thus a One-to-many secure data sharing can be realized.

The property-based cryptography has four features and advantages. First, when encryption, only the pr operty is conducted a safe handling. There is no bu siness with the number and identification of the use r in the system, which greatly reduces the system o verhead, and thus protects the user privacy; second, when decryption, only users met requirement of ci pher property can decrypt correctly, which assures t he secure storage of the data; third, due to the key, random polynomials, and random number of the gr oup users in the system are related, illegal attacker cannot combined with different users' key, which c an effectively prevent the collusion attack; fourth, i t can support a flexible access and control policy of logical and threshold operation such as and, or, an d non. These characteristics make it a good applicat ion prospect in the field of group key management, fine-grained access and control, directional radio, a nd privacy protection, especially cloud computing.

4. Analysis and Design of the Model

According to the hierarchical HABE (Hierarchica l Attribute based Encryption) design concept, multi -level authorization centers in the hybrid cloud suc h as the root authority CA, the regional authority C A, and the authorization center AA are introduced . As per classification the centers will accordingly do jobs of key distribution and properties authentic ation .etc, meanwhile the root CA is responsible for passing the signature guarantee delivery of hierarc hical trust chain to the next level authority. To esta blish a PMC (Property Management Center) to ma intain the global attribute list and response to the re sponses of the CA when beyond the access and con trol property in current cloud ; considering the char acteristics of HABE, multilevel hybrid cloud data c haracteristics and the actual cloud access mode, at l ast a multi-stage and multi-authority (BSP-HABE) model, after the authority CA set the access meth od and the control strategy in combination of the X ACML, was designed to meet the cross-level, cross -domain data access and user privacy protection. T his model is based on the access and control of the user role and the host of the file permissions, and th e property description of the circumstances and dat a resources in the security framework of multilevel hybrid cloud.

4.1 Safety Classification Model

Safety Classification (BSP-HABE) model is compo sed of property management center PMC, the cloud service provider CSP, multi-level authorization ce nter and cloud users. Safety Classification (BSP-H ABE) model as shown in Figure 1, the following ar e their roles:

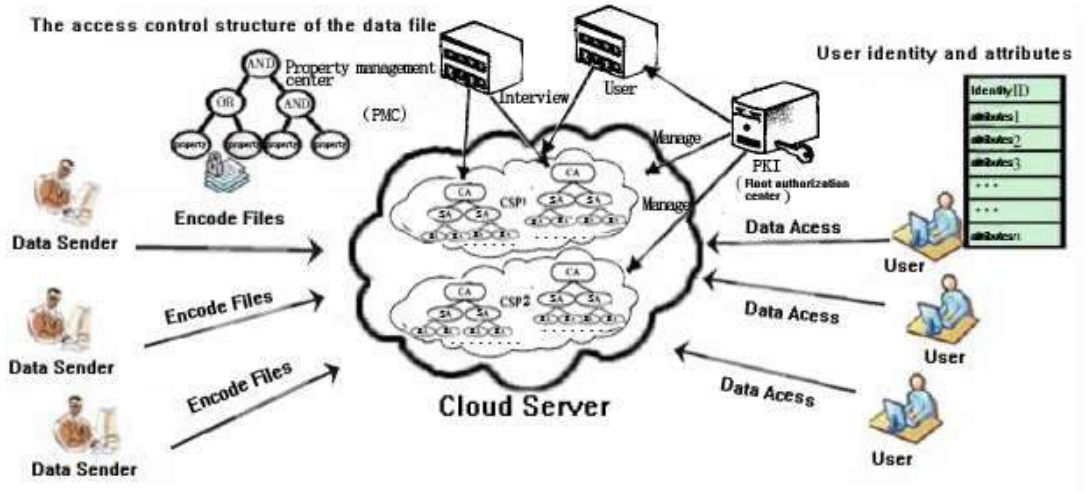

Fig 1: BSP-HABE model

(1) The public key infrastructure (PKI): to issue digital certificate to all the authorization center CA, authorization sub-center SA and users under cloud environment.

(2) Property management center (PMC): to maintain the attribute list of the whole system within the cloud environment, and response the request of cross regional attribute list of the authorization management center (CA).

(3) The cloud service provider (CSP): to provide data storage access for all users of the cloud environment, it will "honest but curious" store the data in the cloud, and strictly fulfill the specific service.

(4) The cloud authorization management center (CA): it consists of multiple authorization management sub-centers. It can authenticate users, add or delete new users, and conduct the authorized access and control of the ciphertext by public key and private key certificate mechanism.

(5) Cloud authorization sub-center (SA): to provide authorization hierarchy tree for the DataSender, and conduct the authorized access and control of the ciphertext.

(6) Users: it consists of DataSender and User. The DataSender will set the access strategy tree according to the authorization level architecture tree provided by the SA, and store the encrypted data in the cloud server. While the User will request to access to the data stored in the cloud server.

4.2 Realization of BSP-HABE Scheme

4.2.1 File Creation

There are two steps to create a ciphertext file. The first step is the DataSender authentication; the second step is that to upload the encrypted data files after superiors authorized center examinated. Specific procedure is as follows:

- DataSender Authentication

(1) DataSender generates random numbers r1, composes the request packets with the private ID and institution ID, signs with the private key of current institution, encrypts with the public key issued by the CA , and then sends to the CA within the cloud. Specifically stated as:

ECA-pk (ECompany-sk (r1, DataSender-id, file query information), agency ID)

(2) After receives the request, the CA starts with the request packet to obtain the agency ID, looks for agencies to obtain the corresponding public key. After decrypts with the public key and the CA private key ,it accesses request packets r1, self-generates random number r2, then it collects DataSender-id and all the IDs within agency to generate a request packet, signs with the private key of CA, encrypts with the public key issued by the SA , and then sends to the SA within the cloud. Specifically stated as:

ESA-pk (ECA-sk (r1, r2, DataSender-id, a collection agency ID)

(3)the SA decrypts the ciphertext with private key , calculates e (gDataSender-id, gr2), generates a request-reply packet with the result and the random number r1,and then sent back to DataSender after encrypted with the private key of SA. Specifically described as:

ESA-sk (r1, e (gDataSender-id, gr2))

(4) DataSender uses SA private key to decrypt, verifies r1 to complete the DataSender authentication.

- To establish Access Control Policy Tree and Data Encryption

In this model scenario, the access control policy consists of three parts, one is the XML access structure defined by t

he datasender; the second is the authorized access policies set by the authorization center , such as access permissions to the dense document (secret, secret, top secret) and the control requirements of the respondents users' departments, positions, roles, permissions and so on; the third is a strategy formed by actual operating environmental constraints and policy constraints (Extensible Access Control Markup Language) in multi-stage hybrid cloud .

The DataSender made a request of data encryption to SA according to the object's property. After review of the security policy (such as judgments of users' level and roles, issued data classification associated permission judgments, etc.), if the property of the respondents beyond its scope of jurisdiction, then SA will made a request to CA, and CA will communicate with PMC to obtain the associated property list, and send back property list of DataSender;

DataSender develops access and control policies, procedures as follows:

(1) Authorized center constructed a authorization level policy tree T in accordance with cloud users' departments, roles, permissions and file security classification, run the Setup algorithm to generate the public system parameters and the private key. (2) DataSender will restrict visitors by specifying the conditions (i.e. access structure)that must be met to qualify respondents ,and thereby generate an XML file; DataSender develops environmental and policy constraints, generates constraint control strategy using XACML ; symmetrically encrypt the file by way of hybrid encryption. A random number is generated as a symmetric key used to encrypt data files form a data ciphertext; ciphertext property based encryption algorithm to encrypt CP-ABE ciphertext.

First select the order of the system as a prime number p of the group G , G_T , g is a generator of the group G , while select bilinear mappings $e: G \times G \to G_T$. Select the security parameter κ determines the size of the group. Meanwhile define Δi, s, where $i \in Z_p$, S is a set, whose elements belong Z_p . Authorities attribute the authorization level domain ζ and data access policies publisher DataSender attribute domain Ψ, regarded all of its elements using the function $H:\{0,1\}^* \to Z_p$ mapped to Z_p .

(1) Authorized center constructed a authorization level policy tree T, completed system initialization

Authorized centers executed the spanning tree algorithm [184], according to the top-down approach in order for each node in the tree-order $k_x - 1$ polynomial x definition p_x . For any s, $\forall s \in z_p$, so $p_{root} = s$, according to the algorithm, randomly selected $k_{root} - 1$ points defined p_{root} . For other nodes x, the algorithm predetermined tree $p_x(0) = p_{parent(x)}(index(x))$ and a random selection $k_x - 1$ of the other points to complete the definition p_x . When the algorithm is run to the tree leaf node x, it provides $p_x(0) = p_{parent(x)}(index(x))$ as the unique attribute node x.

Authorized centers arbitrarily selected randomly $y \in Z_p$, so that $g_1 = g_y$, for any $g_2 \in G$, any randomly selected $t_1, t_2, \ldots t_{n+1}$ from G and definitions $N = \{1, 2, 3, \ldots n+1\}$ .Then followed by the definition of function $T(x) = g_2 x^n \prod_{i=1}^{n+1} t_i^{\Delta i, N(X)}$, which can be seen as a function of n-th order polynomial h. Authorized centers run Setup (n) algorithm to obtain the following public parameters:

$$PK = \{T, g_1, g_2, t_1, t_2, \ldots t_{n+1}, e(g_1, g_2)^{\alpha s}, g^{\beta s}\}$$

Corresponding master key:

$$MK = \{y, s, g^a, \beta\}$$

(2) Custom access and control policy tree $\tilde{T}$ of DataSender and message encryption

DataSender similar process with the authorization center, data released by DataSender run spanning tree algorithm, to create an access control policy tree $\tilde{T}$, each node x to define a $k_x - 1$ tree-order polynomial q_x . Algorithm $\forall r \in Z_p$, and set $q_{root}(0) = r$, then the algorithm then randomly selected $k_{root} - 1$ points to define q_{root} . For other nodes x, the algorithm predetermined tree $q_x(0) = q_{parent(x)}(index(x))$ and a random selection of $k_x - 1$ the other points to complete the definition q_x .

Then, the DataSender selected subset of attributes $r_y \in \Psi$ and $v_i \subseteq \ell$, according to the authorization center authorization level push down the tree T, to create an acce

ss control tree $\tilde{T}$, perform encryption algorithm on data M to obtain the following ciphertext:

$$CT = (\tilde{T}, \tilde{E} = Me(g_1, g_2)^{\alpha\beta\gamma}, E_0 = g^{\beta\varsigma\gamma}, E' = g^{q_y(0)}, E'' = \{T(i)^{q_y(0)}, \forall y \in r_y and i \in \nu_i\})$$

4.2.2 Ciphertext Access

The process of file creation is consistent, the user applies identity authentication to authentication center. The specific authentication process is as follows:

The user generates a request packet including their own identity ID and Affiliations ID , then signs with the institutions private key, at last sends back to CA after CA public key encryption. Specific packet is as follows:

ECA-pk (ECompany-sk (user-id), agency ID).

Authentication center obtains institution ID from the encrypted packet, and obtains the corresponding institutions public key and their own private decryption from query. The identity of User is authenticated. If it is proper legal, the user will at first use their private key to sign to submit to the authorization center, then use public key from authorization center to send the encrypted user ID. The corresponding packets is EAA-pk(ECA-sk(user-id)). After certification, the CA will evaluate and determine the security policy . The specific process is as follows:

(1) The user will apply a request to the policy enforcement point (PEP) to access the data

(2) Policy Enforcement Point (PEP) will respond, require users to provide their own authentication information.

(3) The user will send the collected evaluate information to the Policy Enforcement Point (PEP).

(4) Policy Enforcement Point (PEP) send a request of user data access and evaluate information to (Policy Decision Point) PDP for decisions.

(5) (Policy Decision Point) PDP determines if the system environment meets the access request according to XACML constraint control file. If anyone is not satisfied, then (Policy Decision Point) PDP will be sentenced to deny the request.

(6) (Policy Decision Point) PDP will inform PEP about decision results.

(7) Policy Enforcement Point (PEP) received PDP decision results. If it is enabled, the user could obtain the permission of the document encryption, then the user can decrypt.

Decryption process is as follows:

Authorization center performs Private key generation algorithm to the generate the private key and the ciphertext to send to the user, and then user runs the decryption algorithm to decrypt the data.

(1) Authorization center generates private key. For different user ID, $\forall d \in Z_p$, authorization center depending on different user attributes, run their own private key generation algorithm to generate respective private key:

$$SK_d = \{g^{\frac{\alpha+\beta}{\beta}}, Dx = g_2^{dp_x(0)} T(i)^{r_x}, R_x = g^{r_x}, \forall i \in \upsilon_i\}$$

, where $\forall r_x \in Z_p$, with the corresponding attribute i. The CA will save it safely by using a secure manner (e.g. certificate mechanism encryption, safety traffic channels or to get face to face)

(2) The user decrypt the ciphertext. When User is accessing the ciphertext data, after judging whether to meet control constraints according to the security policy , authorization center will send the ciphertext CT and the corresponding private key SKd. The user runs decryption algorithm to decrypt. If the point x is a leaf node, then the order $i = att(x)$, and then continue to run the following functions to decrypt:

$$Decrypt = \begin{cases} \frac{e(D_x, E')}{e(R_x, E'')} = \frac{e(g_2^{dp_x(0)} T(i)^{r_x}, g^{q_x(0)})}{e(g^{r_x}, T(i)^{q_x(0)})} = e(g_2, g)^{dp_x(0) q_x(0)} \\ \perp \end{cases}$$

If the tree for non-T non-leaf node x, repeated until the final decryption functions is the leaf nodes: If all nodes z are the children nodes x, then run the decryption function $Decrypt(C, D, z)$ to decrypt, the result is stored in F_z . S_x belongs to the set of arbitrary order is k child node z, and meet the decryption function $Decrypt(C, D, z)$.

If this set is not present, then the node does not meet the requirements, i.e. indicating that the node is also difficult to meet the requirements of the authorization level of the tree T. If not, run the following equation:

$$F_x = \prod_{z \in S_x} F_z^{\Delta_{iS'_x}(0)}, where \begin{cases} i = index(z) \\ S'_x = \{index(z) : z \in S_x\} \end{cases} =$$

$$\prod_{z \in S_x} e(g,g)^{dp_x(0)q_{x'}(0)(\Delta_{i,S'_x}(0))} =$$

$$\prod_{z \in S_x} e(g,g)^{dqx'(0)\, p_{parent(z)}(index(z))(\Delta_{i,S'_x}(0)} =$$

$$\prod_{z \in S_x} e(g,g)^{dq_{x'}(0)\, p_{x(i)(\Delta i,S'_x}(0))} =$$

$$e(g,g)^{dsq_{x'}(0)}$$

Similarly, at the $\tilde{T}$ root node, returns a $e(g,g)^{dsr}$ .Decrypted plaintext is calculated by the following formula recovery:

$$\tilde{C}/(e(g^{\frac{\alpha+\beta}{\beta}}, g^{\beta sr})/e(g,g)^{dsr}) = M$$

4.3 Security Analysis of the Scheme

（1）Security analysis of data storage

Data storage involves the security of the data ciphertext and the key ciphertext after encryption, and of the key generation.

For the security of data ciphertext and key ciphertext , because which are due to using symmetric algorithm (AES) to encrypt the data file is safe, the main consideration is the security of the key. The key uses CP-ABE encryption and decryption algorithm. According to the literature [5] it has been proved that the CP-ABE is safe, then if the set of attacker's properties do not meet the requirements to access tree T, it cannot restore the symmetric key to decrypt and thus will not be able to access the data files.

CP -ABE scheme also has anti-conspirator characteristics. When more than one users make a collusion attack, even if their set of attributes conspired together can meet the access tree T, however, due to the characteristics of user's private key generation algorithm, different users have different random number t, so these unauthorized Users will not be able to get conspiracy and ciphertext decryption key, which ensures data confidentiality. In this scheme the key generation of property encryption is by the way that authentication center (e.g. CA, SA) provides DataSender with a authorization level policy tree. The DataSender developed its access and control policy tree according to the requirements their own, and then generated the attribute encryption algorithm based on key. So users who do not meet the authorization and access control policies cannot get to decrypt ciphertext, otherwise it is contrary to the DBDH difficulties.

(2) Proof of forward and backward secrecy security

The scheme has guaranteed the forward and backward secrecy security of the cloud server data storage for the user to add and revocation.

Proof: The program uses event-triggered, add new users to store data in the cloud is encrypted before the number, so new users can not access the data already stored in the ciphertext, so after ensuring the safety. For revocation of the user, the center has been authorized for authorization and access control policies have been adjusted to revoke a user has visited the control does not meet the requirements, thus ensuring backward security. Likewise the user to adjust the problem, if the user is limited to inter-agency mobility within a licensed center, not privileges revoked, and distributes only the correspondingly different set of attributes, attribute-level meet long visit control policy, further access to the data, otherwise prohibited actions .

(3) Security Analysis of the collusion resistance

This scheme can resist user collusion attacks.

Proof: the collusion of the scheme is divided into two categories, which are the attacks of the authorization level architecture of authorization center and the attacks of the access and control tree of Dataowner.

Authorization level architecture authorization center of attack resistance, authorization policy tree is formulated by the authorization center, if you seek joint properties (in the entire property encryption application environment, safety provided these users have their own set of attributes and authorization policies cannot grade completely), even though these conspiracy attribute set match after a joint user access level, you may wish to set up a user to obtain the private key is $D_x = g_2^{dp_x(0)} T(i)^{r_x}$, which is obtained by another user of a private key $D_x' = g_2^{d'p'_x(0)} T(i)^{r_x}$ .Due to the d is not the same, the results of these co Users can not decrypt seek tree T, can not match the authorization policy level, you can not access the data. Relevant literature [6] [7]also resist the attack of such collusion, so the authorization level strategy of authorization center construction is safe.

The premise of launching an attack against the access and control tree of Dataowner is that firstly the user must match the authorization policy level which is enacted by authorization center collusion. Similar to authorization level architecture attack of the authorization center, the corresponding private key of conspiracy to obtain any user is $(g^{\frac{\alpha+d}{\beta}}, D_x = g_2^{dp_x(0)} T(i)^{r_x})$, any other user to obtain th

e private key is $(g^{\frac{\alpha+d'}{\beta}}, D_x = g_2^{d'p'_x(0)}T(i)^{r_x})$, d as an a uthorized center to generate random numbers, which corre spond to only one user. Since users correspond to differen t random number d is not the same, px corresponding to di fferent users are different, so it does not have the conditio ns of decrypting the ciphertext.

Using common parameters ($e(g_1,g_2)^{\alpha s}, g^{\beta s}$) can also att ack the access and control tree of DataSender. The Safety of the access and control policy defined by DataSender is determined by the CP-ABE. The literature [8] [9] have been proved that CP-ABE is safe by been able to resist Certific ate collusion attack.

From the above analysis, we can conclude that the joint pr operty of collusion attack implemented by unauthorized u sers and users not meeting specific authorization level can not have the conditions of decryption and thus can not get the plain data, so the scheme can resist collusion attacks.

5. Conclusion

Based on the HABE basis, this paper introduces multi-lev el authorization center CA and established Property Mana gement Center (PMC) to ensure the delivery of the multile vel chain trust and the response of system global property list; it also uses the advantage of XACML language in the development of constraint and control policies to drawn u p a control security policy supervised by three regulatory authorities which is constitute of the data sender ,authoriz ation center and management institutions. Then in this pap er we designed an authorization center model (BSP-HAB E) and scheme with multi-faceted strategy of control const raints and multi-level fine-grained security, which increas ed the security and flexibility of cloud storage ciphertext a ccess; at the same time on the premise of ensuring the safe ty, the re-encryption of the ciphertext and policy updates a fter user is deleted will transfer to the cloud server to perf orm, reducing the computational cost of datasender to mee t the applications demand of cross-cloud, cross-level data access and security policies.

6. REFERENCES

[1]. Si Tian-Ge, Tan Zhi-Yong, and Dai Yi-Qi A Security Proof Method for Multilevel Security Models[J]. Journal of Computer Research and Development, 2008,45(10): 1711-1717 (in Chinese)

[2]. PANDEY S, WU L, GURU M S, et al. A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments[C]//IEEE International Conference on Advanced Information Networking and Applications , 2010 : 400-407.

[3]. GB/T 17859-1999. Classified criteria for security [S]. BEIJING: Standards press of china,1999 (in Chinese).

[4]. CAO Qi , WEI Zhibo , GONG Wenmao. An optimized algorithm for task scheduling based on activity based costing in cloud computing[C]//Bioinformatics and Biomedical Engineering, 2009 : 1-3.

[5]. PUCHINGER J, GUNTHER R. RAIDL, et al. The multidimensional knapsack problem : structure and algorithms[J]. INFORMS Journal on Computing , 2010 : 250-265.

[6]. Sandhu R S, Coyne E J, Feinstein H L. Role-based access control models[J]. IEEE Computer, 1996,29(2):38-47.

[7]. Zhai De-Gang, Xu Zhen,Feng Deng-Guo. Violation of static mutual exclusive role constraints in dynamic role transition[J]. journal of computer research and development. 2008, 45(4):677-683(in Chinese).

[8]. CloudSim a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms[J]. Software − Practice & Experience archive, 2011 : 23-50.

[9]. WANG Chao,CHEN Xing-yuan,LI Na. An access control mode based on information flow graph[C] Proceedings of the International Conference on Computational Intelligence and Security. SANYA,CHINA,2011,998-1000