Design on role-based multi-area access control method in electric power unified application platform system

Cheng Zhou*, Jian Shi

China Electric Power Research Institute, NARI Road.No.8, Nanjing, China, 210003

Received 1 October 2014, www.cmnt.lv

Abstract

With the further promotion of smart grid and the concentration of business systems, the State Grid Corporation put forward higher information security protection requirements. This paper proposes a Role-based Multi-area Access Control Method (RMACM), which provides a generalized and effective mechanism of security management in Electric Power Unified Application Platform System. RMACM provides a set of items constraint specifications. These constraint specifications are organized to form a construction, and an enact process is proposed to make it scalable and flexible to meet the need of diversified service application systems. Concerned on the problem that the standard role-based access control mechanism does not consider the implementation in multi-area secure, RMACM erases the downward information flow by extended rules of read and write and some authorization constraints while still keeping the expressive power and flexibility of standard RBAC, which makes up the limitations when applying standard RBAC on multi-area systems.

Keywords: Permission Management; Electric Power Unified Application Platform System; Role based Access Control

1 Introduction

Under the tendency of integrating dispatching with control in recent years, the access control of electric power Unified Application Platform system (UAP) becomes more and more complicated. For this reason and the defects in traditional discretionary access control, this paper proposed a multi-area access control method based on UAP system.

At present, the role-based access control is a widely used authorized model. Particularly in the field of commercial applications, organizing roles according to work authority and user accessing to the role of the members according to work responsibilities can effectively increase authorized flexibility and convenience. Sandhu's RBAC96 is the core part of the American National Standards [1], Sandhu divide the role model into four-concept family as RBAC0, RBAC1, RBAC2 and RBAC3. RBAC0 pointed out the minimum demand basis model, which supports role system. RBAC1 increases role hierarchy conception based on the RBAC0. RBAC2 increase role constraints based on the RBAC0. RBAC3 includes all of the above three models. But restrictions on RBAC in the United States national standards, only explain the role's separation of static duties and separation of dynamic duties norms, and do not explain role norms under mandatory access.

On the other hand, Mandatory access control is a usual method to control the one-way flow of information and prevent the illegal flow of information. A number of information systems such as electric power Unified Application Platform system need both the mandatory access control and the flexibility of role access mechanism. Nyanchama and Osborn [3] use information flow analysis method to simulate MAC in RBAC, which need verify system log. In role mechanism of Demurian [7], the method has not been realized flexibility and expression of integrity role mechanism. All these methods [3-8], which realize the mandatory access in role-based access control, have failed to maintain intuitive, flexibility and expression of RBAC under situation of control information flow.

This paper proposes a Role-based Multi-area Access Control Method (RMACM), which provides a generalized and effective mechanism of security management in Electric Power Unified Application Platform System. The validity and reliability of the proposed method has verified by results of applying it in actual project [9-17].

2 Design of RMACM

RMACM can be expressed as: using the set of role can express any subset in the (U, WS, OP) space; that is to say, the universal of access control of the mapping $F\!: U \times WS \times OP \to \{0,1\}$ in (U, WS, OP) space can be expressed as, the arbitrary subset expressed as Eq . (1),

* Corresponding author's E-mail: zhoucheng3@epri.sgcc.com.cn;

$$D = \{(u_i, ws_i, op_i) \mid u_i \in U, ws_i \in WS, \\ op_i \in OP, and F(u_i, ws_i, op_i) = 1\} \tag{1}$$

in the (U, WS, OP) space can be expressed by the model. RMACM establish access control strategy according to a service hierarchical structure, thus the access control strategy is more direct viewing.

Definitions (predefined sets standards) USERS, ROLES, OPS, OBS and SESSIONS are respectively user set, role set, operation set, object set and session set of the system. Their meanings have been defined by RBAC standard.

Definitions (classified grades set, CLASSES). Classified grades index is the sensitivity of the data. if we use integer to indicate classified grades index, LOW indicates that the minimum classified grades, the highest classified grades is HIGH and LOW ≤ HIGH, then $\forall l \in$ CLASSES $\Rightarrow$ l ∈ [LOW..HIGH]. CLASSES is all classified grades sets.

Definitions (scope, CATEGORIES). The $I=\{CAT_1, CAT_2, \ldots, CAT_n\}$ is the system divides category sets, according to a characteristic. CAT_i ($1 \leq i \leq n$) is a category name. Scope $CATEGORIES=\{C_1, C_2, \ldots, C_k\}$($k \geqslant 1$) is a subset of I .That is $CATEGORIES \subseteq I$.

Definitions (user roles distribution, UA). $UA \subseteq USERS \times ROLES$, It is the distribution relations of many-to-many mapping the user to the role. $Assigned_users(r)=\{u \in USERS|(u,r) \in UA\}$will map the role r (excluding succession role) to the user set, $assigned\_users(r{:}ROLES) \rightarrow 2^{USERS}$.

Definitions (user authority set, PRMS). $PRMS=2^{(OPS \times OBS)}$, is a user's authority set.

Definitions (the role of authority arrangement, *PA*). $PA \subseteq PRMS \times ROLES$ is arrangements relations of many-to-many mapping the authority to the role. $Assigned\_permissions(r)=\{p \in PRMS \mid (p,r) \in PA\}$ will map the role r (excluding succession role) to the authority set, $assigned\_permissions(r{:}ROLES) \rightarrow 2^{PRMS}$.

Definitions (authority mapping, *Op* and *Ob*). $Op(p{:}PRMS) \rightarrow \{op \subseteq OPS\}$ maps the operation set of authority *p*. $Ob(p{:}PRMS) \rightarrow \{ob \subseteq OBS\}$ maps the object set of *p*.

Definitions (role inheritance relationship, RH). $RH \subseteq ROLES \times ROLES$ is *ROLES* the partially ordered set structure, which is known as the inheritance relationship and is expressed as →. $r_1 \rightarrow r_2 \Rightarrow authorized\_permissions(r_2) \subseteq authorized\_permissions(r_1)$. $Authorized\_users(r)=\{u \in USERS| \, r' \rightarrow r, (u, r) \in UA\}$ maps the role r and all roles' users of inheriting the role r, $authorized\_users(r{:}ROLES) \rightarrow 2^{USERS}$. $Authorized\_permissions(r)=\{p \in PRMS| \, r' \rightarrow r, (p, r') \in PA\}$ maps the role r and all roles' authorities of inheriting the role r, $authorized\_permissions(r{:}ROLES) \rightarrow 2^{PRMS}$.

Definitions (session mapping). $Session\_users(s{:}SESSIONS) \rightarrow USERS$ maps the session s to the corresponding user. $Session\_roles(s) \subseteq \{r \in ROLES|(\exists r' \rightarrow r) \, (session\_user(s), r') \in UA\}$ maps the session to the corresponding role sets, $session\_roles(s{:}SESSIONS) \rightarrow 2^{ROLES}$.

Session_roles (s) maps all roles than can be used in session s (including the role of inheritance).

Definitions (Conversational authority). $Avail\_session\_perms(s) = \bigcup_{r \in session\_roles(s)} assigned\_permissions(r)$ is the authority that can be used in a conversation, $avail\_session\_perms(s{:}SESSIONS) \rightarrow 2^{PRMS}$.

Definitions (security level, *LABELS*). Security level $LABELS \subseteq CLASSES \times 2^{CATEGORIES}$ is classified grades and the range combination. $Class(l{:}LABELS) \rightarrow \{class \subseteq CLASSES\}$ maps the classified grades of security level l. $Category(l{:}LABEL) \rightarrow \{category \subseteq CATEGORIES\}$ maps the range of security level.

Such as the security level *l* = {confidential, (Ministry of Personnel, Ministry of Finance)}, *Class (l)* = secret, *Category (l)* = (Ministry of Personnel, Ministry of Finance).

Definitions (of dominated relationship, CH). $CH \subseteq LABELS \times LABELS$ is a partially ordered set structure level on the security level *LABELS*, which is known as the dominated relationship and is expressed as ≧, $l_1, l_2 \in LABELS$, $l_1 \geqq l_2 \Rightarrow Class(l_1) \geqslant Class(l_2) \wedge Category(l_1) \supseteq Category(l_2)$. $list\_categories(llist{:}2^{LABELS}) \rightarrow 2^{CATEGORIES}$ maps the security level set ranges, $\forall categ \in CATEGORIES$, $\forall llist \in 2^{LABELS}$,$categ \in list\_categories(llist) \Rightarrow \exists c \in CLASSES, (categ,c) \in llist$.

Such as the security level set *llist* = {(confidential, the Ministry of Personnel), (top secret, Ministry of Finance)}, *list_categories* (*llist*) = {Ministry of Personnel, Ministry of Finance}.

Definitions (user security level arrangements, USA). User security level arrangements $USA \subseteq USERS \times LABELS$ is the many-to-many mapping relation between users and security levels . $user\_label(u{:}USERS)=\{l \in LABELS|(u, l) \in USA\}$ maps user security level sets, $user\_label(u{:}USERS) \rightarrow 2^{LABELS}$.

Users can arrange multiplicity security level in RMACM.

Definitions (role security level arrangements, RSA). Role security level arrangements $RSA \subseteq ROLES \times LABELS$ is the many to one mapping relation between roles and the security level. $Role\_label(r{:}ROLES)=\{l \in LABELS|(r, l) \in RSA\}$ maps the security level of r, $role\_label(r{:}ROLES) \rightarrow \{l \in LABELS\}$. $Role\_label\_RH(r{:}ROLES) =\{l \in LABELS|\exists r', r \rightarrow r' \wedge$

$(r', l) \in RSA\}$ maps r and the security level of the role that r inherits, *role_label_RH*(*r*:*ROLES*)→2^{LABELS}.

Although the role and security level is the many to one mapping relation (a role only is arranged a security level), *role_label_RH* can be mapped multiple security levels by inheritance.

Definitions (object security level arrangements, OSA). Object security level arrangements OSA $OSA \subseteq OBS \times LABELS$ is the many to one mapping relation between objects and the security level. *Object_label* (*ob*:*OBS*)=$\{l \in LABELS \mid (ob,l) \in OSA\}$ maps the security level of r, *object_label*(*ob*:*OBS*) → $\{l \in LABELS\}$.

Definitions (Session classified grades arrangements, SCA). Session classified grades arrangements $SCA \subseteq SESSIONS \times CLASSES$ is the many to one mapping relation between Sessions and the classified grade. *Session_class*(*s*:*SESSIONS*)=$\{c \in CLASSES \mid$（*s*, *c*）$\in SCA\}$ maps the classified grades of session s, *session_class*(*s*:*SESSIONS*)→$\{class \subseteq CLASSES\}$.

Sessions only have classified grades and don't have the limit of range.

Definitions (static security level separation, SSC). Static security level separation $SSC \subseteq ((2^{\text{CATEGORIES}} \times N)$ is the set of the binary tuple (*categs*, *n*) ,*categs* is the set of ranges , natural number n ≥ 2, and SSC requires that cross ranges of roles which a single user can undertake can not be equal to or greater than n at *categs* set, namely: $\forall(categs,n) \in SSC, \ \forall u \in USERS \Rightarrow |categs \cap list\_categories\,(\bigcup_{u \in assigned\_users(r)} role\_label\_RH(r))| < n$.

For example, SSC = ({devices department, purchasing department}, 2), a person can only undertake the role within the scope of devices department and purchasing department, but not undertake the role across two departments. However, there is not such a restriction for security level of users (it is only a qualification).

Definitions (static separation of duties, SSD). Static separation of duties $SSD \subseteq ((2^{\text{ROLES}} \times N)$ is the set of binary tuple (rs, n), rs is the set of roles, natural number n ≥ 2, and *SSD* requires that users cannot arrange equivalent or more than n roles at *rs* set at the same time, namely: $\forall(rs,n) \in SSD, \ \forall t \subseteq rs\text{: } |t| \geqslant n \Rightarrow \bigcap_{r \in t} authorized\_users(r) = \varnothing$.

Definitions (dynamic security level range separation, DSC). Dynamic security level range separation $DSC \subseteq ((2^{\text{CATEGORIES}} \times N)$ is the set of binary tuple (*categs*, n), categs are scopes of sets, natural number n ≥ 2, DSC requires the security level range of active roles in a single session cannot be equal to or greater than n of categs sets, namely: $\forall(categs,n) \in DSC, \ \forall s \in SESSIONS, \ \forall r \in ROLES \Rightarrow |categs \cap (\bigcup_{r \in session\_roles(s)} list\_categories(role\_label\,(r)))| < n$.

For example, DSC = ({devices department, purchasing department}, 2), active roles in a single session cannot cross the scope of these two departments.

Definitions (dynamic separation of duties, DSD). Dynamic separation of duties $DSD \subseteq ((2^{\text{ROLES}} \times N)$ is the set of binary tuple (rs, n), *rs* is the role set, natural number n ≥ 2, and DSD requires users cannot activate more than n roles in the *rs* set, namely：$\forall(rs,n) \in DSD, \ \forall s \in SESSIONS, \ \forall rs \in 2^{\text{ROLES}}, \forall role\_subset \in 2^{\text{ROLES}}, role\_subset \subseteq rs, role\_subset \subseteq session\_roles(s) \Rightarrow |role\_subset| < n$.

Definition: inheritance of role, denote role as *X*, *Y*.

$X = \{(u_i, ws_i, op_i) \mid u_i \in U, ws_i \in WS, op_i \in OP, H_1(ws_i, op_i) = 1, G_1(u_i) = 1\}$

$Y = \{(u_j, ws_j, op_j) \mid u_j \in U, ws_j \in WS, op_j \in OP, H_2(ws_j, op_j) = 1, G_2(u_j) = 1\}$ If

$W_2 = \{(ws_j, op_j) \mid ws_j \in WS, op_j \in OP, H_2(ws_j, op_j) = 1\}$ is subset of

$W_1 = \{(ws_i, op_i) \mid ws_i \in WS, op_i \in OP, H_1(ws_i, op_i) = 1\}$ and

$V_1 = \{u_i \mid u_i \in U, G_1(u_i) = 1\}$ is subset of $V_2 = \{u_j \mid u_j \in U, G_2(u_j) = 1\}$

then claim that *X* inherit *Y*, denote as $X \geqslant Y$, specially has:

$X > Y$,when $W_1 \neq W_2$ or $V_1 \neq V_2$;

$X = Y$,when $W_1 = W_2$ and $V_1 = V_2$。

In definition of role inheritance, V_1 , V_2 , W_1 , W_2 respectively are role *X*, *Y* related user sets and permission set. In space (U, WS, OP) , W_1 , W_2 respectively be projection generated by *X*, *Y* on the plane constitutes by axis *WS* and *OP*. role inheritance include two attributes: (1) transitivity. namely if $X \geqslant Y$ and $Y \geqslant Z$, then $X \geqslant Z$, which *X*, *Y*, *Z* are roles. (2) Underactivity or called irreversibility. Namely if $X > Y$, then $Y \ngtr X$, which *X*,*Y* are roles.

Definition: Class Privilege Authorize (PA) is a connection class between role and service, an instance is a two tuples (r, s), represent the authorization relationship that role *r* to service *s*. denote as pa(r, s). Each two tuple (r, s) in class is not redundant.

Definition: Class Status Authorize (SA) is a connection class between actor and role, an instance is a two tuples (a,r), denote that actor *a* is authorized some role *r*. each two tuple (a,r) in class is not redundant.

The definition of responsibility separation class is a self-association class in role class, an instance is a two tuples (role1, role2), represent conflicts of interest relations between the two roles. Responsibility separation has two type: static responsibility separation (SSD) and dynamic responsibility separation (DSD). The former stipulate that an actor cannot simultaneously be authorized two kind of roles, the latter stipulate that an actor can authorize this two roles. However, SSD and DSD are mutual, namely a SD relation cannot be static and dynamic at the same time, obviously, SD relation non-reflexive, symmetry, not transitivity.

Inheritance role for the organization and management of large role in regulating the service authority (PA to the Service) and the authorized user specification (from SA to the Actor). For the two roles, the role of inheritance and separation of duties are mutually exclusive; and separation of duties could be derived from the role of super-role.

To simplify the calculation of service as the role of authority and the management authority, in accordance with the succession of semantic roles may determine a set of rules so that the role can automatically inherit the role of the super-powers norms, but also allows users to automatically grant super role, and automatically activated. Each PA SA or an object that contains all the powers of constraint conditions. We first definition of similar size and power of the same concept. Similar is the meaning of power, the two-power p1 and p2, if the PA category, for the same services; If the SA category for the same role:

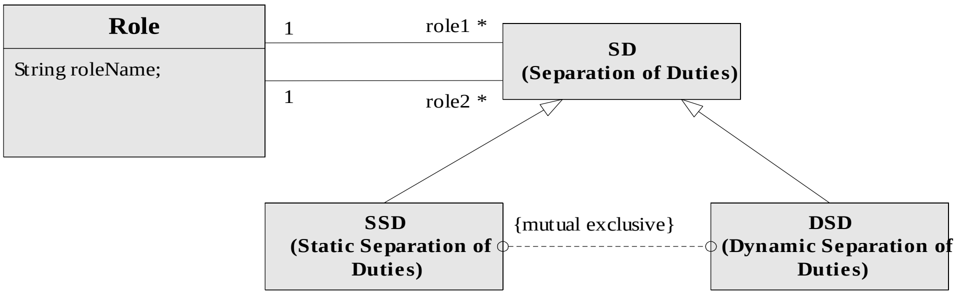


FIGURE 1 Separation relation of duties between two roles be divided into static and dynamic

- p1 power smaller than p2, if and only if p1 Context meet the restrictive conditions, p2 bound conditions are true, but also state that p2 Context of constraint conditions are true, p1 bound by conditions not true.
- p1 power larger than p2, if and only if p2 power smaller than p1.
- power p1 and p2 equivalent, if and only if p1 Context meet the restrictive conditions, p2 bound by the conditions for real, and vice versa.

3 Actors authorized service norms of RMACM

C1. SA targets a specific context that existed under the conditions of a service are entitled to play a role in the role:

The CC (sa (a, r). Context_cond) that the current conditions meet Context: sa (a, r). Context_cond. , A role that in the right context conditions cc r role.

C2. If a r undefined role of a role, a mandate norms, the role from the role of his son's role in regulating authority derived from. Derived from the following three-step algorithm that:

Step1. First obtain a role r all the actors have authorized a set of roles:

Corollary: If A is blank, the role of actors in a r without authorization.

Step2. Then from A to filter out the entire indirect role:

Corollary: If A non-empty set, then B is empty.

Step3. Final from B to be derived from authorized norms:

C3. Abstract role cannot be granted direct actors. If an abstract role that was awarded a role, is a necessary condition, was awarded the role of the role:

C4. If a definition of the role of a regulating role of the authority, the role of the normative authority to rewrite the (override) derived from its role of the normative, and rewrite the norms should have greater powers:

C5. Role of a given role does not exceed the number of authorized the role of the base:

4 Norms stage role of RMACM

C6. A rp (a, r) targets in the current context cc in that r a role in the current context has been one cc of a role to activate, and if the current context cc r meet on the role of a service s regulating authority, rp (a, r) s right to the service object:

The rp (a, r). Context that rp (a, r) Object Context parameter; R in that role in the Context cc s right to access services (defined in the standard C14); said rp (a, r) in the current context of the right to call cc s services.

C7. Abstract role cannot be directly activated. A role that is abstract activation is a necessary condition for a non-abstract role of activated:

C8. Activation of a role does not exceed the number of activated its base:

C9. A role to be activated at the same time DSD relations with the two roles:

C10. A role in a context of the conditions under activate a role, the necessary conditions are in the same context conditions also activated its ultra-role:

C11. A role in the current context, a cc a role in the activation r, the necessary conditions exist sa (a, r), and the current conditions meet sa Context (a, r) Context constraints, as well as the activation of the role of r Context bound:

The CC (sa (a, r). Context_cond) that meet the current context sa (a, r) of the restrictive conditions, CC (r.context_cond) that meet the current context of the activation of the role of r restrictive conditions.

C12. A role in the same role in the Context of activation of up to a certain time:

C13. Separation of duties (SD) static separation of duties (SSD) and dynamic separation of duties (DSD) of two types:

C14. For the two roles, static separation of duties (SSD) and dynamic separation of duties (DSD) Exclusive:

C15. Any role cannot be authorized for the two SSD relations with the role:

Theorem. Activation of the role of the base does not exceed the activation of the role of super-base:

Theorem. Role of the two cannot activate the role of relations with SD:

Prove: reduction to absurdity. Sd assumptions (r1, r2) exist, C13, we can see the ssd (r1, r2) or dsd (r1, r2) exist. If dsd (r1, r2) exist, the C9 and contradictions; if ssd (r1, r2) exist, C15, sa (a, r1) and sa (a, r2) cannot simultaneously exist, and this conflicts with the C11. It is not hypothetical.

Theorem. A role in the current context, a cc in the right s of a service, there is a sufficient condition for the activation of the role of being a r, in the context of the role of the service cc s right to call:

Among them, to a role that in the current context of the service cc s right.

Prove: the C6, we can see that

Rp that (a, r) in the current context of the right to call cc s services, the actors in the current context a cc in the service s right, so Has been formed.

5 Analysis and implement

Theorem: Arbitrary subset in the (U,WS,OP) space can be expressed by a union of a group of roles.

Proof: Make $D=\{(u_i,ws_i,op_i)\,|\,u_i\in U,ws_i\in WS,op_i\in OP,$ $F(u_i,ws_i,op_i)=1\}$ arbitrary subset in the (U,WS,OP) space, B is a collection of D' values from U .Arbitrarily admitted $u_i\in B$, $i=$ 1,2,… n, make

$D_i=\{(u_i,ws_j,op_j)\,|\,ws_j\in WS,op_j\in OP,$ $F(u_i,ws_j,op_j)=1\}=\{(u_k,ws_k,op_k)\,|$ $u_k\in U,ws_k\in WS,op_k\in OP,$ $H_i(ws_k,op_k)=1,G_i(u_k)=1\}$ of which:

$H_i(ws,op)=F(u_i,ws,op),ws\in WS,op\in OP$;

$$G_i(u)=\begin{cases}1, & if \quad u=u_i,\ u\in U,\\ 0, & if \quad u\neq u_i\end{cases}.$$

Obviously, $D=D_1\cup D_2\cup\cdots\cup D_n$,namely D can use a collection of a group of roles to indicate, therefore, a group of roles can express an arbitrary subset in the (U,WS,OP) space ,that is to say using role-based access control model can implement arbitrary Access Control Strategy, which proves RMACM is universal.

In RMACM, the concept of roles unifying users and authority allows users and authority have natures belonging to roles, for example the nature of inheritance. That is, using the concept of the role and natures can implement various access control strategies.

Suppose a user compute a set $A=\{u_i\,|\,u_i\in U,G_A(u_i)=1\}$ of and an authority set $B=\{(ws_i,op_i)\,|\,ws_i\in WS,op_i\in OP,H_B(ws_i,op_i)=1\}$, A default has authorities established by the mapping H_A: $WS\times OP\rightarrow\{0,1\}$, B default has users identified by the mapping $G_B:U\rightarrow\{0,1\}$. Therefore, sets $V=\{(u_i,ws_i,op_i)\,|\,u_i\in U,ws_i\in WS,op_i\in OP,$ $G_A(u_i)=1,H_A(ws_i,op_i)=1\}$ $W=\{(u_j,ws_j,op_j)\,|\,u_j\in U,ws_j\in WS,op_j\in OP,$ $G_B(u_j)=1,H_B(ws_j,op_j)=1\}$ are the sets of users and authorities in the initial state of the system, in accordance with the definition of the role, V and W are role. Also suppose R_1 and R_2 role, according to the definition of succession, if the user u_i inherits the role R_1, not only u_i is assigned authorities associated with the role R_1 , but also R_1 is associated with u_i . The authorities which are inherited by the role R_2 will be assigned to the role R_2, for this reason at the same time the users associated with the role R_2 are assigned these authorities, that is, these authorities associate with the users belonging to the role R_2.While the inheritance relation between authorities between embodies the hierarchy and sub-groups of the authority. In this way, users and authorities can be organized into a hierarchal structure. Maintaining a complete inheritance relation, managers can assign authorities to roles and the hierarchal relation of management roles, and associate users and roles.

Shown as Figure 2, in the implementation of RMACM, Users apply for entry to activate corresponding roles, and then carry on a session. After identity verification, according to the user's role and implement rules the engine of RMACM conforms whether the user's session activated successfully or not, and if it fails then the system rejects the user to login, or if it successes then the engine of the RMACM activates corresponding roles and determines the class of the session. When this session apply for a visit to the object resources, the engine of

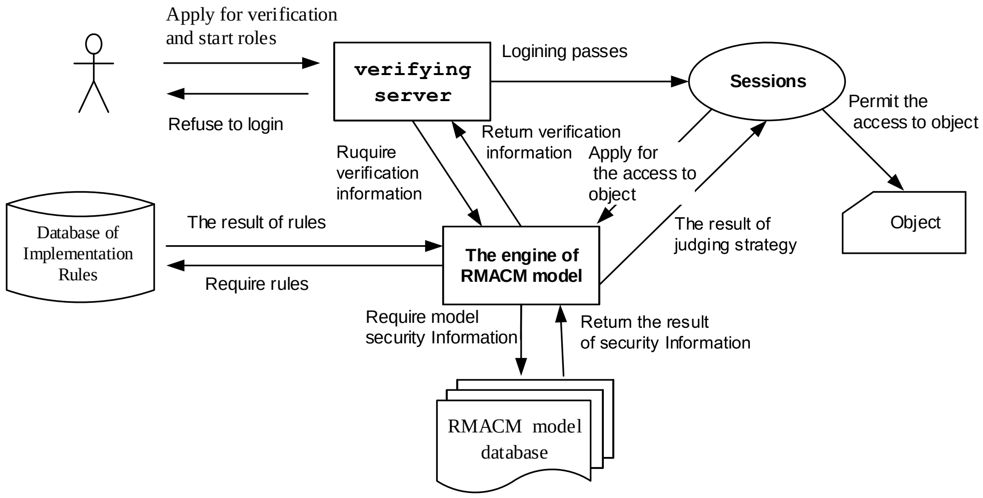

Apply for verification
and start roles
Refuse to login
verifying server
Logining passes
Sessions
Permit the access to object
Object
Ruquire verification information
Return verification information
Apply for the access to object
The result of judging strategy
Database of Implementation Rules
The result of rules
Require rules
The engine of RMACM model
Require model security Information
Return the result of security Information
RMACM model database

FIGURE 2 The implementation of RMACM

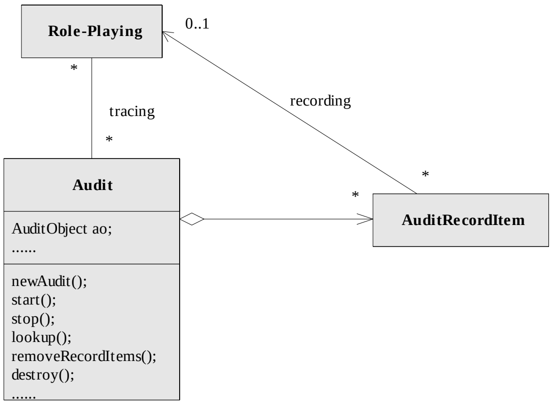


FIGURE 3 The implementation of audit class in RMACM

RMACM according to the type of this request in accordance with reading strategies and writing strategies to determine the access permission of this session.

In RMACM, RP monitoring reflects the current context state of the RP objects, and controls their life periods. The general operational design to implement RP monitoring is as follows:

RP.list (); / / list all current RP objects

RP.list (role (Role) r) / / list all current the RP objects, which activate the role r

RP.list (Actor a) / / list all the RP objects which the user a holds

rp.trace (); / / trace the current context state of a RP object rp

rp.deactivate (); / / deactivate or suspend the current activities of a RP object rp

rp.reactivate (); / / re-activation current activities of a RP object rp

rp.remove (); / / terminate and eliminate the life period of a RP object rp

RP audit is to automatically record the context state and service events called by a operator a, or a role r, or a particular service s in rp(a, r) lifetime, for the purpose of inspection or recovery systems. We establish an Audit class, each object is a type of record, and contains a group of audit record items(Figure.3).the lifetime of an Audit object is shown in the Figure 4 of a state diagram.

Specifically, the general operation of RP audit can be designed as follows:

NewAudit (Actor a) / / for a certain user a to create a certain audit

NewAudit (role (Role) r); / / for a certain role r to create an audit NewAudit (Service s) / / for a particular service s to create an audit NewAudit (Actor a role (Role) r); / / for a certain user a and a certain

Role r to create an audit, the user a has been granted to the role r

Audit.start (); / / start all audits

Audit.start (); / / start a certain audit

Audit.stop (); / / stop all audits

audit.stop () / / stop a certain audit

Audit.lookup () / / look up all audit record items

audit.lookup (); / /look up a certain record item

audit.removeRecordItems (); / / remove all record items of a certain audit

Audit.destroy (); / / destroy all audit record items

audit.destroy (); / / destory a certain audit record item

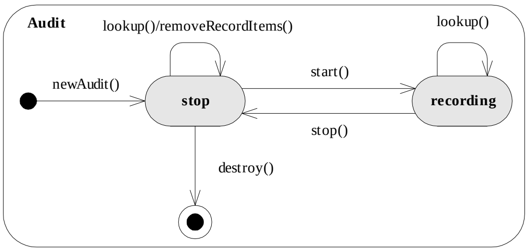


FIGURE 4 The state diagram of an Audit object' lifespan

6 Conclusions

RMACM provides a generalized and effective mechanism of security management in Electric Power Unified Application Platform System. RMACM provides a set of items constraint specifications. These constraint specifications are organized to form a construction, and an enact process is proposed to make it scalable and flexible to meet the need of diversified service application systems. Concerned on the problem that the standard role-based access control mechanism does not consider the implementation in multi-area secure, RMACM erases the downward information flow by extended rules of read and write and some authorization constraints while still keeping the expressive power and flexibility of standard RBAC, which makes up the limitations when applying standard RBAC on multi-area systems.

References

[1] *Role Based Access Control* American National Standard for information Technology BSR INCITS 359[S] Draft 4/4/2003

[2] Park J S, Sandhu R, Ahn G 2001 Role-Based Access Control on the Web *ACM Transactions on Information and System Security* **4**(1) 37-71

[3] Nyanchama M, Osborn S 2006 Modeling Mandatory Access Control in Role-Based Security Systems *IFIP Workshop on Database Security Proceedings of the ninth annual IFIP TC11 WG11.3 working conference on Database security IX* 129-44

[4] Osborn S, Sandhu R, Nunawer Q 2000 Configuring Role-Based Access Control To Enforce Mandatory And Discretionary Access Control Policies *ACM Transaction on Information and System Security* **3**(2) 85-106

[5] Osborn S 2007 Mandatory access control and role-based access control revisited *Proceedings of the second ACM workshop on Role-based Access Control* 31-40

[6] Papazoglou M P 2003 Service-Oriented Computing: Concepts, Characteristics and Directions *Fourth International Conference on Web Information Systems Engineering(WISE 03)* 10-12

[7] Demurjian S 2007 Implementation of Mandatory Access Control in Role-based Security System *CSE367 Final Project Report. Computer Science & Engineering* 06269-3155

[8] Mavridis I, Pangalos G, et al 2007 eMEDAC: Role-based Access Control Supporting Discretionary and Mandatory Features *Proc. of 13th IFIP WG 11.3 Working Conference on Database Security* 63-78

[9] Vinoski S 2003 Toward Integration: Integration With Web Services **7**(6) 75-7

[10] Peltz C 2003 Web Services Orchestration and Choreography *IEEE Computer* **36**(10) 46-52

[11] Ferraiolo D F, Sandhu R, Gavrila S, et al 2001 Proposed NIST standard for role-based access control *ACM Trans on Information and System Security* **4**(3) 224-74

[12] David G, Shang W Ch, An Ch H, et al 2004 Rainbow: Architecture-Based Self-Adaptation with Reusable Infrastructure *IEEE Computer* **37**(10) 46-54

[13] Nicol D M, Sanders W H, Trivedi K S 2004 Model-based evaluation: From dependability to security *IEEE Transactions on Dependable and Secure Computing* **1**(1) 48-65

[14] Ross J W, Westerman G 2004 Preparing for utility computing: the role of IT architecture and relationship management *IBM Systems Journal* **43**(1) 5-19

[15] Foster I , Kesselman C, Nick J M , Tuecke S 2002 Grid services for distributed system integration *IEEE Computer* **35**(6) 37-46

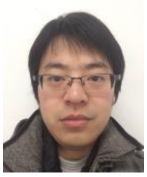
[16] Perrey R, Lycett M 2003 Service-oriented architecture *Proceedings of the Symposium on Applications and the Internet Workshops* 116-119

[17] Buyya R, Venugopal S 2004 The gridbus toolkit for service oriented grid and utility computing: An overview and status report *Proceedings of the 1th IEEE International Workshop on Grid Economics and Business Models* 19-66

Authors

CHENG ZHOU , 1981.06, Nanjing County, Jiangsu Province, P.R. China

Current position, grades: Information Security researcher of The Smart Grid Research Institute, China.
University studies: He received his B.E. in department of communication engineering from SouthEast University in China.
Scientific interest: His research interest fields include Network Security.
Publications: more than 10 papers published in various journals.
Experience: more than 5 years experience in network security and protection, has completed five scientific research projects.



JIAN SHI , 1983.08, Nanjing County, Jiangsu Province, P.R. China

Current position, grades: Information Security researcher of The Smart Grid Research Institute, China.
University studies: He received his B.E. in department of software engineering from TianJin University in China.
Scientific interest: His research interest fields include Network Security.
Publications: more than 5 papers published in various journals.
Experience: more than 5 years experience in network security and protection, has completed three scientific research projects.