Model of Domain based RBAC and Supporting Technologies

Zan Yang
Institute of Command Automation, PLAUST, Nanjing 210007, China
China Institute of Electronic Equipment System Engineering, Beijing 100141, China
Email: woshiyangzan@163.com

Lin Yang
China Institute of Electronic Equipment System Engineering, Beijing 100141, China
Email: Yanglin61s@yahoo.com.cn

Xiang-yang Luo
China Institute of Electronic Equipment System Engineering, Beijing 100141, China
State Key Laboratory of Information Security (Institute of Information Engineering, Chinese Academy of Sciences), Beijing 100093, China
Email: xiangyangluo@126.com

Lin-ru Ma
China Institute of Electronic Equipment System Engineering, Beijing 100141, China
Email: Malinru61s@yahoo.com.cn

Bao-sheng Kou and Kun Zhang
Chinese troop of 61046, Beijing 100097, China
Email: {Kbs121, 61046}@163.com

***Abstract*—Nowadays the significance of the access control technology in service oriented network is increasingly highlighted. The RBAC access control model has a variety of advantages such as easy management and high efficiency. With the expansion of the network scale, a network must be divided into autonomous multi-domains for convenient management. However, there is still a lack of studies on the domain based RBAC model as the barrier of applying the RBAC to multi-domain environment, and the corresponding supporting implementation technologies for the domain based RBAC are also in weak. In this paper, we proposed a model of domain based RBAC (D-RBAC) to better adapt to the security requirements of the multi-domain environment. We firstly introduced the domain concept and model and then gave a formal description to the proposed D-RBAC model. Secondly, we designed feasible implementation architecture for the D-RBAC model and based on this architecture we proposed two supporting technologies. The fuzzy role mapping method according to user's attributes has strong description abilities for role assignment and the convenience of realization. The dynamic collaboration domain construction framework can greatly improve the efficiency of inter-domain access control. The proposed D-RBAC model and the related supporting technologies can obviously facilitate the application of RBAC in multi-domain environment**

***Index Terms*—domain based, RBAC, implementation architecture, role mapping, dynamic collaboration**

I. INTRODUCTION

With the continuous development of network technology, information systems have more requirements in sharing and interoperability, so the service-oriented construction mode of information systems attracts more and more attention, which is based on open and unified standards to publish information systems as services, sharing or reusing information resources in distributed heterogeneous environment, helping to break the barriers between different information systems, realizing the free transformation of resources and information between the networks, making better use of resources in different locations of the network. Web service technology is one of the main technologies to achieve this architecture [1].

Access control is the most intuitive security control technology for the services [2-3]. Based on rules, access control mechanism makes decisions to the service requests, avoiding abuse and destruction of services, protecting the integrity and confidentiality of data. At present, common researches related to access control include role based access control (RBAC), attribute based access control (ABAC), trust management (TM), context based access control (CBAC) and so on. Among all of them, RBAC [4-5] was applied in much more occasions, as its relatively easy management, high efficiency, and its adaption ability to various security requirements. RBAC can be used to construct some more optimized access control mechanisms as well combined with trust, attributes or other factors [6-7]. Early access control studies were carried out under the assumption of "in a single global domain", which means that the access control mechanisms are familiar to all users. So it is easy to

doi:10.4304/jcp.8.5.1220-1229

construct a harmonious and unified access control rule set based on the users' characteristics and all security requirements. However, with the continuous expansion of service-oriented network, it is difficult to have a centralized security management institution taking the same job [8]. So a distributed multi-domain mode will be more suitable to the large-scale network environment.

At present, there are many constrained models based on RBAC [9-10]. However, when applying the RBAC to large-scale distributed networks, there is still a lack of discussion on the domain-based RBAC model which can adapt to complex security needs. Although [11] and [12] introduced the concept of domain, but the definition of domain-based RBAC model and feasible realization mechanism have not been included. This paper is based on the consideration of characteristics of distributed multi-domain network environment, gives a definition of domain concept and model, formally describes the proposed domain based RBAC model—D-RBAC and its implementation architecture, briefly introduces some supporting technologies including the inter-domain role mapping and dynamic collaboration domain construction.

The paper was divided into five parts: Section 2 introduces the related work; Section 3 discusses the proposed models of domain and domain-based RBAC; Section 4 describes the implementation architecture and related supporting technologies; Section 5 gives a full-text summary and an outlook for the future work.

II. Related Works

The earliest mature role based access control model is the RBAC96 model proposed by Sandhu et al [4]. According to the definition of this model, the central notion of RBAC is that the permissions are associated with roles and users are associated with appropriate roles in accordance with their responsibilities and qualifications. This greatly simplifies the management of permissions. A user may play multiple roles, while a role may be assigned to multiple users. A user establishes a session during which the user activates some subset of roles that he is a member of to complete some work. The RBAC basic model RBAC0 was defined as follows:

- U, R, P, S, users, roles, permissions and sessions respectively;
- $PA \subseteq P \times R$, a many-to-many permission-to-role assignment relation;
- $UA \subseteq U \times R$, a many-to-many user-to-role assignment relation;
- User: $S \rightarrow U$, a function mapping each session s_i to the single user $user(s_i)$ (constant for the session's lifetime);
- Roles: $S \rightarrow 2^R$, a function mapping each session s_i to a set of roles $role(s_i) \subseteq \{r | (user(s_i), r) \in UA\}$ (which can change with time) and session s_i has the permissions $\bigcup_{r \in role(Si)} \{p | (p, r) \in PA\}$.

RBAC96 also provided three advanced models in the same time. To reflect the real situation and facilitate the management, RBAC1 based on the basic model defines a partial order relation on the role set, namely the role hierarchy, meaning a senior role can inherit all the permissions of junior roles subordinated by it except private permissions. Similarly, if a user is assigned to a senior role in a session, he has been implicitly assigned to all the junior roles subordinated by the senior role. RBAC2 introduces the concept of constraints to the basic model, such as separation of duties, least privilege and so on. RBAC3 is the consolidation of RBAC1 and RBAC2 model. In addition, RBAC96 provides an ARBAC model with administrator roles.

Then Ferraiolo and Kuhn introduced RBAC2000 model as the recommended standard of NIST, with four models at different levels [5]. Afterwards, a lot of constrained models based on RBAC, such as temporal RBAC (TRBAC) model [9], general temporal of RBAC (GTRBAC) model [10] and so on was proposed, defining periodic role enabling, temporal dependencies among roles etc. These studies mainly focused on how to improve the role system and role assignment. But facing the requirements of applying RBAC to the multi-domain network, there is still a lack of study on domain-based RBAC model and related supporting technologies. Literature [11] introduced the concept of domain on the basis of role system, encapsulated each part of the distributed system into domains to achieve decentralized management, but did not propose a domain-based RBAC model and a feasible realization mechanism. Literature [12] offered a domain-based access control infrastructure for distributed collaboration environment, mainly focused on domain management infrastructure, fine grain domain-based access control rules and inter domain collaborations etc. But the literature also does not propose a domain-based RBAC model.

When the network is divided into domains to be managed distributed, it is considerable crucial to take cautious access control for the users from external domains, which is a key supported technology to achieve the domain-based RBAC model. To solve the problem of inter-domain access control, Kapadia et al proposed an IRBAC model [13], promoting some basic concepts, such as security context, inter-domain role mapping, role mapping policy and so on. Fu introduced a method which carries out role mapping through negotiation mechanism based on the trust of two-party domains [14]. Literature [15] offered a method which resolved role mapping by the comparison of the attributes of users assigned to different roles. Literature [16] described a method which mapped a local role in a domain to an equivalent global role for implementation of access control by comparing the security levels of services and global roles. However, the existing methods of role mapping still lacked depiction to the uncertainties and fuzzy phenomena in role management. About the technology of dynamic collaboration domain construction, literature [17] introduced the principles of virtual organization construction and literature [18] described a construction method of virtual organization based on the consideration to trust and other factors. But there is still an imperative need to a dynamic collaboration domain construction technology which can adapt to the D-RBAC model and is

easy to achieve. This paper will meet the deficiencies of the existing researches to make a meaningful research.

III. Domain Model and the D-RBAC Model

In this section we proposed a domain-based RBAC model—D-RBAC, according to the demand characteristics of applying RBAC to the multi-domain network environment. We firstly introduced the domain concept and model, and then formalized the D-RBAC model, as the basis of related supporting technologies.

A. The Domain Concept and Model

Firstly, we would introduce the concept and model of domain. It is difficult to establish a centralized security management institution in a large-scale service-oriented network, not only for the efficiency, but also because of many potential problems, such as the difficulty to satisfy complex global security requirements and single point of failure. Many existing security management institutions often make security policies aimed at their own domains for the purpose of management.

A domain is a logical realm governed by a set of common security policies [8]. Domain boundaries can depends on the natural boundaries of the network, the organization boundaries, community of interest (COI) etc. Similar to the hierarchies of organizations, hierarchies can also exist among different domains. Domains in each level can have their own security management. The range overlapping may exist among different domains, as shown in Fig. 1 . The domain classes can be divided into:

1) Single-Machine Domain: contains the resources and policies that exist in the user's computer under the control of the only user;

2) Network Boundary Domain: contains the entities and policies within the natural network boundaries;

3) Organization Domain: contains the entities and policies belonging to the same organization;

4) Community of Interest Domain (COI): contains the entities belonging to different organizations or natural network realms but sharing the same interest or responsibility and security policies in COI;

5) Global Domain: includes entities and policies within the entire network realm.

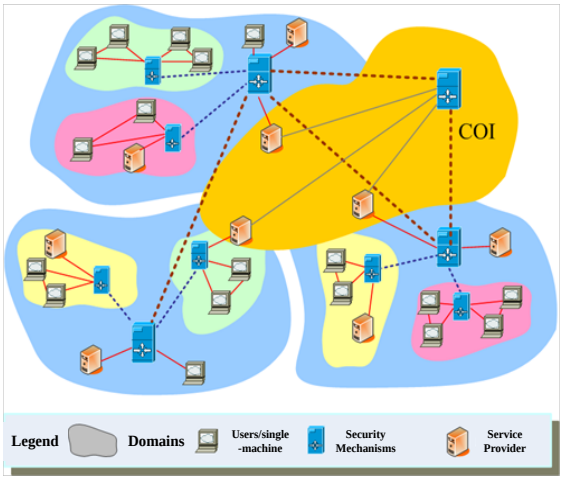

Figure 1. Examples of domains

The common domain model can be formally described as following:

● D, U, S, M, P, domains, users, services, management institutions and security policies respectively;

● $d=\{U_d, S_d, M_d, P_d\}$, a domain contains the users, services, management institution and policies belonging to it;

● $DL \subseteq D \times D$, partial order relations on D, showing the hierarchy among domains, also known as the containment relations, writing as $\geq$;

● $\forall d_1, \forall d_2, U_{d2} \in U_{d1} \wedge S_{d1} \in S_{d2} \wedge M_{d1} \blacktriangleright M_{d2} \wedge P_{d1} \blacktriangleright P_{d2} \Longrightarrow (d_1, d_2) \in DL \wedge d_2 \geq d_1$, any Domain 1 and any Domain 2, if all users of Domain 2 are the users of Domain 1 and all services of domain 1 are the services of Domain 2 and the management institution of Domain 1 submits to that of Domain 2 and the policies of Domain 1 obey to that of Domain 2 then the Domain 1 and Domain 2 have a hierarchical relationship and Domain 2 contains Domain 1;

● $\forall d_1, \forall d_2, \exists u, \exists s, (u \in U_{d1} \wedge u \in U_{d2}) \vee (s \in S_{d1} \wedge s \in S_{d2}) \Longrightarrow d_1 \vdash d_2$, If there is a user u which belongs to Domain 1 and Domain 2 at same time or a service s which belongs to Domain 1 and Domain 2 at same time, then Domain 2 and Domain 1 are overlapped.

B. The Domain Based RBAC Model—D-RBAC

As an easily-managed access control model with high usability, RBAC has many advantages, such as:

1) Disjoint the permissions and users. Users are assigned to appropriate roles closely related to the concept of user group. Roles can correspond to the specific semantics in practical application scenarios.

2) RBAC has the high efficiency of the access control enforcement.

3) RBAC can provide a variety of models to adapt to variety of security needs.

4) Independent from security control mechanisms, RBAC is a strategy-neutral model.

Thus, based on the familiarity to the users and security requirements of domain, it is suitable and efficient to adopt the RBAC in a domain. However, it is unrealistic to build a global role system and role assignment rules in the overall area to satisfy all the domain's security requirements. Various services concern about different characteristics of users, such as whether the user is geographically close to itself, then the users with a certain attribute should be assigned to the roles individually rather than be merged with other similar users. This will greatly expand the space of role system. Another problem is that some roles have the permissions in the community of interest and they simultaneously belong to their organization. Then inappropriate inheritance of permissions in the role hierarchy will occur sometimes.

To adapt to the characteristics of the multi-domain distributed management, based on RBAC3 model and the domain model mentioned before, this paper designs a domain-based RBAC model (D-RBAC), which can be formally described as following:

● U_d, R_d, P_d, S_d, SS_d, PT, PA_d, UA_d, the users, roles, permissions, services, sessions, permission types, permission-to-role assignment relationships, user-to-role assignment relationships of Domain d respectively;

● $\forall p, s(p)\in S_d \Longrightarrow p\in P_d$, any permission p, if the specified target service in p belongs to Domain d, then the permission p belongs to the Domain d too;

● PT={inheritable, non-inheritable}, permissions can be divided into two types—the "inheritable" meaning the permissions of junior roles can be inherited by the relevant senior roles, and the "non-inheritable" meaning the permissions of junior roles cannot be inherited by relevant senior roles;

● $PA_d\subseteq P_d\times R_d\times PT$, many-to-many permission- to-role assignment relations in Domain d, can be divided into the inheritable ones and the non-inheritable ones;

● $UA_d\subseteq U_d\times R_d$, many-to-many user-to-role assignment relations in Domain d;

● $\forall ua, r(ua)\in R_d \Longrightarrow ua\in UA_d$, any user-to-role assignment relation ua, if the role in ua belongs to Domain d, then ua belongs to Domain d too;

● $\forall pa, p(pa)\in P_d \wedge r(pa)\in R_d \Longrightarrow pa\in PA_d$, any permission-to-role assignment relation pa, if the permission in pa belongs to Domain d and the role in pa belongs to Domain d, then pa belongs to Domain d too;

● $\forall u, \exists ua, u(ua)=u \wedge r(ua)\in R_d \Longrightarrow u\in U_d$, any user u, if there is a user-to-role assignment relation ua, the user in ua is u and the role in ua belongs to Domain d, then the user u belongs to Domain d;

● $\forall r, \exists d, \exists d', r\in R_d \wedge d\geq d' \Longrightarrow r\in R_{d'}$, any role r, if r belongs to Domain d, and there is a Domain d' contained by Domain d, then r belongs to Domain d' too;

● $\forall ua, \exists d, \exists d', ua\in UA_d \wedge d\geq d' \Longrightarrow ua\in UA_{d'}$, any user-to-role assignment relation ua, if ua belongs to Domain d, and there is a Domain d' contained by Domain d, then ua belongs to Domain d' too;

● $RH_d\subseteq R_d\times R_d$, a partial order relation on the role set of Domain d, namely the role hierarchy, writing as $\geq$;

● User: $SS_d\rightarrow U_d$, a function mapping each session s_i of Domain d to the single user user(s_i) of Domain d (constant for the session's lifetime);

● Roles: $SS_d\rightarrow 2^{Rd}$, a function mapping each session s_i of Domain d to a set of roles role(s_i) $\subseteq\{r|(\exists r'\geq r)[(\ user(s_i), r')\in UA_d]\}$ (which can change with time) and session s_i has the permissions $\cup_{r\in role(si)}\{p|(\exists r''\leq r)(\ p, r'')\in PA\}$.

The D-RBAC Model is showed in Fig. 2. .

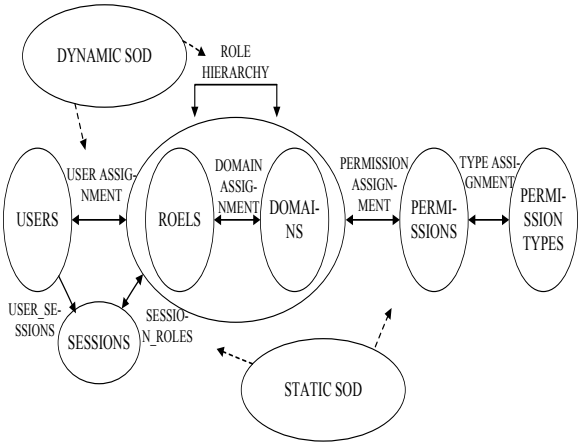

Figure 2. The D-RBAC model

In the model of D-RBAC, users, roles, permissions, services, permission-to-role assignment relationships, user-to-role assignment relationships, and sessions belong to respective domains. The security management institution creates roles in the domain, establishes rules of permission-to-role assignments and user-to-role assignments. If a user can be assigned to a role in a domain by hitting an existent rule, then it is the native user of this domain, otherwise it is an outer-domain user. If a role belongs to a senior domain, then it is the role of all junior domains subordinated by the senior domain, and it can be identified by all these junior domains. A user-to-role assignment rule established by a senior domain is also the rule of all junior domains subordinated by the senior domain. The permission-to-role assignment rule established in a domain is commonly owned by the domain separately. The role hierarchies in a domain can be divided into two kinds: the inheritance hierarchies and the activation hierarchies [5]. Focused on the problems like some permission of roles in the community of interest potentially be inappropriate inherited by the senior roles in organization domains, in the process of permission-to-role assignment, two permission types—the inheritable and the non-inheritable, are banded with the permissions. The meanings of inheritable and non-inheritable like mentioned before. This is an effective way to prevent the leakage of permissions.

D-RBAC model can fully satisfy the security needs of the multi-domain environment. It is convenient to build the role system and role assignment policies in a domain adapted to the local security requirements, and to establish a security context in the domain. Generally speaking, at the beginning of setting up a domain and the security policies in it, the users allowed have the permissions in the domain should be assigned to the inner roles, and these users can be classified as the native users.

IV. IMPLEMENTATION TECHNOLOGIES

Based on the D-RBAC model, this section designed the corresponding feasible implementation architecture, and proposed an inter-domain role mapping method and a dynamic collaboration domain construction framework as the supporting technologies to achieve D-RBAC in multi-domain environments.

A. The D-RBAC Implementation Architecture

Actually in the realization of D-RBAC model, users connect to permissions using the roles as medium through two kinds of security rules: user-to-role assignment rules and permission-to-role assignment rules. When a user attempts to request a service, he should firstly log on a portal which can be a proxy to request the identity authentication system in the domain of logging on to issue an identity token for the user. The identity token contains the basic identity information of the user. This information can be obtained from the identity management infrastructure in network. In this paper we suggests to establish an overall unified identity management infrastructure as the basis of multi-domain network security environment, or it will be difficult to audit or make security evaluations to a users in the whole area if the user don't have a fixed identity. However, whether the users belong to a domain is decided by whether there is a role assignment rule for him in this domain. Because a service may simultaneously belongs to different domains, so when a user with an identity token request a service, the request message needs to specify the context was in which domain, or it has been imposed when the service was published.

User's request should be blocked by the security agent in front of service. Then the agent requests the policy decision system in the service's located domain to make a judgment for whether allowable. The policy decision system firstly verifies the issuer of identity token. If the verification is passed, it will request the identity authentication system in the same domain to provide the role and other attribute information of user. The user-to-role assignment rules in a domain should be stored in the identity authentication system's database. Identity authentication system queries this database based on user's identity. If the database contains the rule for the user, then the user is a native user has native roles. Otherwise, it is an outer-domain user and the identity authentication system will request the identity management infrastructure to inform which domain the user belongs to. And then the identity authentication system will send all these information back to the policy decision system. It is noteworthy that the roles of a senior domain belong to its junior domains as well, so the user-to-role assignment rules of a senior domain should be propagated to its junior domains. At the same time the junior domains' identity authentication systems can ask their senior domain's identity authentication system for assistance as well. If the user is a native user, the policy decision system will query its database, the permission-to-role assignment rules should be stored in this database, and then a whether or not allowable judgment should be made based on the user's role and other attribute information. If the user is not a native user, the policy decision system will acquire necessary information from the user's located domain to map the user's outer-domain role to a native role, then a judgment to this user will be made. At last the policy decision result will return to the security agent, if allowed the user can access the service.

A cross-domain access control implementation architecture is shown in Fig. 3. .

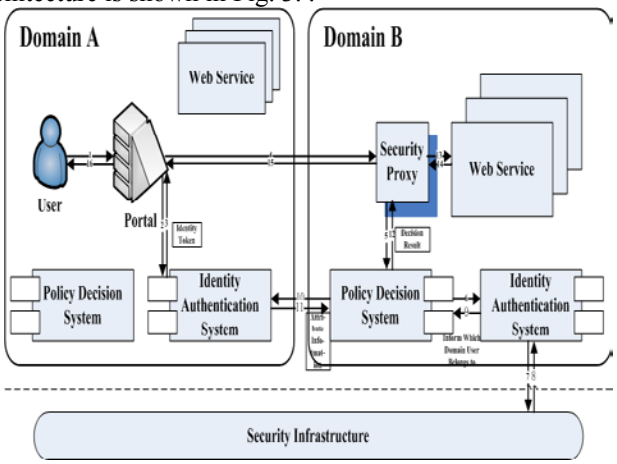

Figure 3. Implementation architecture

B. Role Mapping Based on Fuzzy Theory

In the D-RBAC runtime, it will face the situation that the access control needs to be performed to a strange outer-domain user. As there is no relevant user-to-role assignment rule in native domain under this circumstance, it is necessary to map the user's outer-domain role to a native role and make a decision to the service request according to the existent rules. So Role mapping is a basic supporting technology in the realization of D-RBAC model. Literature [14-16] introduced a number of role mapping methods. We believe that the attribute information should be used as the most fundamental basis for the role mapping to outer-domain users. This not only accords with the natural mentality, but also has a powerful description ability and convenience of realization. Literature [15] proposed a method of role mapping method based on comparison of users' attributes. However, all the mentioned methods lack the considerations on the uncertainty and fuzziness of role mapping. Fuzzy theory can offer mathematical description languages and instruments to research and deal with the fuzzy phenomena [19-20]. The following proportion presented a method using the fuzzy set theory to deal with the user's attributes information for role mapping.

Based on the D-RBAC implementation architecture described above, the role mapping for outer-domain users should be completed by the policy decision system, which uses the automated trust negotiation to request the identity authentication system in the stranger user located domain to provide the user's various needed attributes. In this way, the user's role in the user's located domain, can be seen as one of attributes as references too. Different domains concern different terms of attributes based on their responsibilities and role systems. Take a bank domain as an example. The roles could be assigned to strange outer-domain domain users include the Ordinary User, Corporate Representative, and VIP User so on. A bank domain may pay more attention to attributes like Gender, Age, Position, Profession, and Account Balance so on. A university may have roles including Student, Teacher, and Professors so on, for its outer-domain users,

and it may more concern about attributes like Age, Education Degree, Professional Title, and Work Experience so on. It should be explained that the attributes needed by role mapping process sometimes cannot be all obtained, so those attributes cannot be obtained will be ignored, but the quantity of attributes must meets the minimum condition, otherwise the role mapping cannot be performed. The policy decision system should firstly maps every item of attributes provided by the user's located domain to a value within a certain range; this involves the creation of a global semantic system. For example, Education Degree can be classified into the 4 degrees below: Undergraduate, Bachelor Degree, Master Degree and Doctor Degree, can be mapped to 1, 2, 3 and 4, so the universe about the Education Degree attribute item is {1, 2, 3, 4}. Another example, Positions can be classified into Temporary Worker, Staff, Department Head and Corporate Leader, can be mapped to 1, 2, 3 and 4, and the universe about the Position attribute item is {1, 2, 3, 4}. At last, a numerical value set $\{a_1, a_2 \ldots a_n\}$ can be obtained to describe a user's attribute set $\{A_1, A_2 \ldots A_n\}$, and a_n belongs to the universe U_n of A_n. Policy decision system will map the user to one in the native role set $\{R_1, R_2 \ldots R_m\}$ based on the user's attributes. The following part introduced the definition of fuzzy set and membership function.

Definition 1: A fuzzy set $\underline{R}$ on the universe U is the set allows its members to have different degrees of membership, called membership function $\mu_{\underline{R}}$, writing as:

$$\mu_{\underline{R}}: U \rightarrow [0, 1]$$

For any $u \in U$, there is $u \rightarrow \mu_{\underline{R}}(u)$, $\mu_{\underline{R}}(u) \in [0, 1]$, $\mu_{\underline{R}}(u)$ is the degree of u belongs to $\underline{R}$.

Applying the fuzzy set theory, we could regard every universe of the attribute items as the universe of a "close to a role" fuzzy set. For every role R_i, in each attribute item A_k's universe U_k, its "close to" fuzzy set membership function $\mu_{\underline{ik}}(u)$ $(u \in U_k)$ can be assured, indicating the probability that a user who has an attribute value u in the attribute item A_k is close to the role R_i. Take a user's attribute value set $\{a_1, a_2 \ldots a_n\}$ respectively substitute into $\mu_{\underline{11}}(u) \ldots \mu_{\underline{mn}}(u)$. Then a probability matrix Z_{mn} representing the user's close degree to every role by every attribute item could be obtained. According to Z_{mn} we could finally calculate the probabilities $(p_1, p_2 \ldots p_m)$ respectively denoting the degrees of the user is close to a certain role in $\{R_1, R_2 \ldots R_m\}$, where $p_i = Z_{i1} + Z_{i2} + \ldots + Z_{in}$. If the user is most likely close to role b, that is $p_b = MAXof(p_1, p_2 \ldots p_m)$, then the user should be assigned to the role b. The user's attributes based fuzzy role mapping algorithm of this method is showed in Fig. 4. .

Fuzzy Role Mapping Algorithm

Input: $\mu_{11}(u) \ldots \mu_{mn}(u)$, $a_1 \ldots a_n$, Z_{mn}

Process:

(a) For i=1, i=i+1, i≤m;
(b) For j=1, j=j+1, j≤n;
(c) $z_{ij} = \mu_{ij}(a_j)$;
(d) For k=1, k=k+1, k≤m;
(e) $p_k = z_{k1} + z_{k2} + \ldots + z_{kn}$;
(f) p_b = MAXof $(p_1 \ldots p_m)$.

Output: R_b

Figure 4. Fuzzy role mapping algorithm

To establish the membership functions, one should draft a subjective and rough function firstly. Common membership functions include the normally distribute type, the partial small type and others. Then in the practical application process the initial membership function need to be modified and revised through continuous study and test. To establish the membership functions in each kind of attribute for each role adapt tovarious different application fields will be the future work of this paper.

C. The Dynamic Collaboration Domain Construction

When a user sends an inter-domain service request, the policy decision system of the service's located domain will take a long processing delay to deal with role mapping; the efficiency is difficult to be guaranteed. Generally, users should join the frequent-accessed service's located domain based on interest or responsibility. However, in the network running phase, users may still need to access outer-domain services frequently sometimes for the requirement of temporary tasks, which may result to an unbearable processing delay. Except the single service, in the service-oriented network, the service composition composed by s series of service can be provided for the users too, which may result to a much more longer processing delay because several times of role mapping should be dealt with. So beside the static manner of domain division, it is necessary to offer a temporary dynamic collaboration domain construction method, so that multiple users and multiple services can join one domain for the purpose of collaboration. Without role mapping for inter-domain service or service composition access, the efficiency will be greatly improved.

Based on the D-RBAC implementation architecture, referring to the existed studies to the virtual organization [17-18], we proposed a feasible and concrete framework of dynamic collaboration domain construction. A brief introduction to the framework is given as follows. Dynamic collaboration domain is formed by the participated users and services based on the mutual trust for the sake of efficient collaboration, requiring the inter-

domain neutral and credible temporary domain management institutions as the infrastructure. Dynamic collaboration domains should be registered and managed in the management institutions with independent identity authentication systems. When a user wants to create a dynamic collaboration domain, the user should send invitations to interesting services. The invited service requests the policy decision system in its own domain to map a native role to the user, and then it should make a decision that whether can join the collaboration domain according to its trust management policies, if allowed it should reply to the inviter and sends the user-to-role assignment rule for the user to a identity authentication system in a temporary domain management institution, this means the user will be assigned to a fixed role in a period of time. The user then registers the dynamic collaboration domain in the temporary domain management institution and this user will be the initiator of the domain. Thereafter, when the user needs to access the service in same dynamic collaboration domain, it will be indicated in the request message that this access is an internal access in the collaboration domain. Then the policy decision system used for the service's access control will no longer make a role mapping to this user, but will get the user's user-to-role assignment rule from the identity authentication system of the registered temporary domain management institution. The service should periodically adjust the role assignment rules, and periodically judge that whether can keep on participating in the dynamic collaborative domain accordance its trust management policies. If the trust environment status does not match the conditions, the service will quit the collaboration domain.

A User can check out whether the service needed to be accessed frequently in future is already in the registered dynamic collaboration domain to decide to join in the domain or not. When he wants to join the collaboration domain, the user should send requests to all the existent services in the domain. These services will request their native domain's policy decision system to map a role for the user, and judge whether can allow the user to join the domain based on their trust management policies. If the request is allowed by all the services, the user can be registered to join the collaboration domain and all the local user-to-role assignment rules for the user will be released to the temporary domain management institution. When in a dynamic collaboration domain there already are some involved services, the initiator of this domain has the right to invite other services to join this domain as well. Because one service may be combined with other services to form a service composition, so it is necessary to know each other's role between each pair of services, else they need to map each other's role in their own domain and release the two role assignment rules to the temporary domain management institution. When the initiator of the dynamic collaboration domain sends dissolution request to the registered temporary domain management institution, or when all the services have exited from the domain, the life of the dynamic collaboration domain comes to an end. The dynamic collaboration construction topology diagram is showed in Fig. 5. The hollow spots represent the users in figure and the solid spots represent the services.

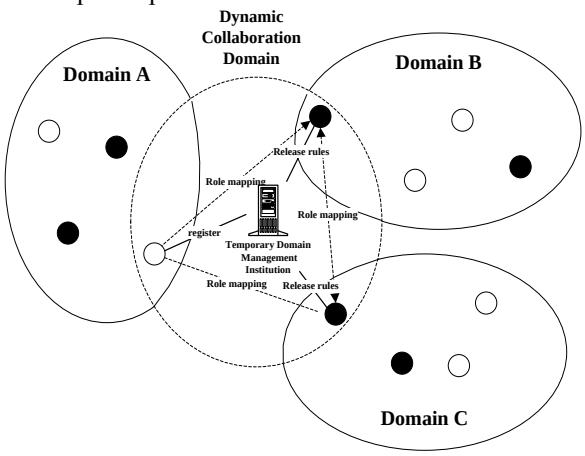

Figure 5. Dynamic collaboration topology

V. EVALUATIONS

An experimental platform was designed to evaluate the proposed D-RBAC model and corresponding supporting technologies in functionality and performance. We used the service-oriented manner to realize the identity authentication system, the policy decision system and the temporary domain management institution, in order to facilitate the publication and access. The services and security control mechanisms were developed and running in Java environment. We use NetBeans as the development tool, Linux+JAX-WS as the runtime environment, Mysql as the information storage database, Servelet standardized filtration mechanism as the services' security proxy. For the identity and policy information expression, we used the SAML standardized OpenSAML-J library and the XACML library realization of the ppzian project.

This paper designed a number of test cases for the function evaluations, three of these test cases were briefly described as follows:

1) A user assigned to a role of the senior domain in a domain hierarchy requests to access the service in the junior domain. The identity authentication system in the junior domain needs to ask for the user's user-to-role assignment rule in the senior domain's identity authentication system and then allow its request according to the role-to-permission assignment rules. Because of the domain hierarchy, the role of the senior domain simultaneously belongs to the junior domain. So when a user can be assigned to a role in the senior domain accesses a service in the junior domain, the identity authentication system in the junior domain should ask the senior domain whether the user-to-role assignment rule is existent in it, or the user-to-role assignment rule was already propagated to the junior domain.

2) A user in a domain request to access a service in another domain, the policy decision system in the service's located domain gets the user's attribute information from the identity authentication system in the user's located domain, to map a native role for the user.

Then the user can obtain corresponding permissions. Because the service request in an inter-domain request, so the identity authentication system in the service located domain cannot find the user's role assignment rule in its database. The policy decision system should acquire necessary information of the user in the user's located domain to map a role for the user.

3) A user invites a service in another domain to jointly construct a dynamic collaboration domain for the efficiency. The service requests the policy decision system in its domain to make a decision whether to allow based on trust management policies, if allowed it will map a role for the user and release the user's user-to-role assignment rule to the temporary domain management institution. When another user requests to join this collaboration domain, all the existent services in the domain should make a judgment to the request and do the role mappings. The inter-domain service access control will consume a lot of time to deal with the role mapping. The dynamic collaboration domain method can efficiently save the processing time. The users and services participate in a joint dynamic collaboration domain and the user-to-role assignment rules are released to the domain management institution and the authentication process will be dealt in this single domain.

The test results showed that the outcomes of these three test cases were both consistent with the expectation, achieved good effects.

We also tested the performance of proposed D-RBAC implementation architecture and corresponding supporting technologies. To simulate the multi-domain interaction environment, two domains were divided. One computer ran the user's browser and the portal, one computer ran the identity authentication system and the policy decision system in user's located domain, One computer ran the services and the security agent, one computer ran the identity authentication system and the policy decision system in service's located domain, one computer ran the temporary domain management institution and its identity authentication system. All the computers were configured as 2.6G Pentium CPU, 2G RAM and Inter 1000Mbps Ethernet Card. The test consisted of three situations, (1) the user accesses the service in the same domain without role mapping; (2) the user accesses the outer-domain service through role mapping; (3) the user and the service are in one dynamic collaboration domain, the user's role is already allocated in this domain. We tested the access control process time delay for 100 times in these three situations respectively. For the role mapping, the user needed to be mapped to one of the 5 different roles based on 5 items of attributes. The respective average time delays of these three situations were showed in Fig. 6. .

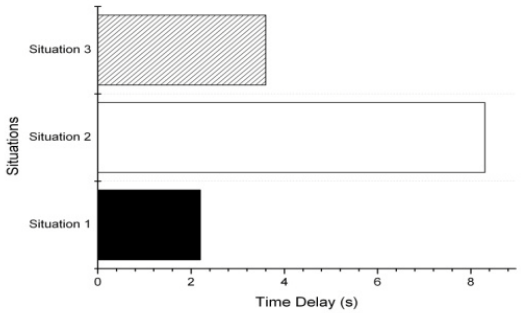

Figure 6. Average time delays in three situations

It can be seen that in situation (1), the access control process without role mapping consumed a minimum time delay, because the identity authentication system in the service's located domain just only needed to acquire the user's role assignment rule in its own database. In situation (2), the access control process with role mapping consumed a maximum time delay, because the identity authentication system in the service's located domain could not find any role assignment rule for the user, the policy decision system needed to acquire necessary information in the user's located domain and made a role mapping process. in situation (3), the access control process without role mapping in a temporary dynamic collaboration domain consumed a medium time delay, because the service request had indicated the context was in a temporary dynamic collaboration domain and the policy decision system in the service's located domain could get the user's role assignment rule from the identity authentication system belonging to the temporary domain. All the time delays were within a reasonable range. The construction of collaboration domain effectively reduced the time delay of access control process. We also compared the performance of the proposed methods in this paper and the methods in [14], because the literature [14] adopted the similar access control scene and the similar experimental environment. In the situation that the service access is in a common domain, the related access control process in this paper and the process in [14] consumed a similar average time delay, because they used the similar mechanisms. In the situation that the access control process needed to deal with role mapping, because the method in [14] required the interactive negotiation, but the attribute-based method in this paper simply depended on the user's attribute information provided by the user's located domain, so the related average time delay of access control process in this paper is just seventy-eight percent of the average time delay in [14]. In the situation that the access control process was needed to be imposed on an inter-domain service request and some methods were applied to improve the efficiency, because although [14] used the role mapping results caching method, but the security judgments and the verification of role mapping results were also needed to be done, our paper adopted the dynamic collaboration domain construction method and the role assignment process was only needed once, so efficiency was improved and the related average

time delay of access control process in this paper is just fifty-three percent of the average time delay in [14].

So the work of this paper made a significant improvement to the previous related researches not only in functionality but also in performance. The D-RBAC model and the supporting technologies can obviously facilitate the application of RBAC in multi-domain environment.

VI. Conclusion and Future Work

It is necessary to take effective access control for services in the service-oriented network. Role-based access control, assigning users to different roles, associates permissions with roles to offer an efficient, easy-managed security control method can be adaptable to different security requirements. With the continuous expansion of the network, a distributed multi-domain management mode will be more suitable to the complex security requirements. At present, studies on RBAC are mostly focusing on combining with variety of factors to optimize the model or security constraints. However, there is still a lack of research on domain-based RBAC model for the application of RBAC to large-scale multi-domain network environment. Corresponding supporting technologies for the model are still poor too.

Firstly, this paper, based on the characters of multi-domain autonomous environment, discussed the domain concept and formalized model, promoted a domain-based RBAC model—DRBAC and then gave a formal description to the model. The model established attribution relationships to the domains, users, roles and services so on, defined the native user and outer-domain user, taking measures to adapt to multiple security requirements in multi-domain environment.

Secondly, facing to the potential obstacles to realize the D-RBAC, this paper studied corresponding supporting technologies. We designed the feasible implementation architecture for the D-RBAC model firstly. Then to solve the problem of inter-domain role mapping, this paper, focusing on the uncertainty and fuzzy phenomenon, proposed a role mapping method based on fuzzy set theory, which takes users' attributes as the judgment basis with a strong description ability and convenience of realization. To solve the efficiency problem of frequent inter-domain access, based on the implementation of D-RBAC model, this paper introduced a dynamic collaborative domain construction framework based on the infrastructure of temporary domain management, then users and services can collaborate freely in the domain without role mapping.

However, our work in this paper still lacked advanced considerations and depictions to the D-RBAC model to satisfy more complex security requirements; when deal with the fuzzy role mapping, the membership functions in each kind of attribute for each role based on various different application fields were still needed to be assured; There is still a demand for further theoretical breakthrough in the area of dynamic domain construction. All above mentioned will be our future works.

Acknowledgment

We sincere thank for financial support from the National Natural Science Foundation of China (Grant No. 60902102, 61170032) and the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDA06030601). We thank the experts and other anonymous reviewers for their helpful comments.

References

[1] E. Bertino, L. Martino, F. Paci and A. Squicciarini, "Security for Web Services and Service-Oriented architectures", Springer, 1st Edition, 2010.

[2] R. S. Sandhu and P. Samarati, "Access control: principles and practice", Communication Magazine, vol. 9, pp. 40-48, Sept. 1994.

[3] S. Vimercati, S. Foresti and P. Samarati, "Recent advances of access control", Handbook of Database Security, Jan. 2008.

[4] R. S. Sandhu, E. J. Coyne, H. L. Feinstein and C. E. Youman, "Role based access control models", IEEE Computer Society, vol. 29, pp. 38-47, Feb. 1996.

[5] R. S. Sandhu, D. Ferraiolo and R. Kuhn, "The NIST model for Role-based access control: towards a unified standard", Proceedings of the fifth ACM workshop on Role-based access control, New York, 2000.

[6] D. Kuhn, E. Coyne and T. Weil, "Adding attributes to Role-based access control", IEEE Computer, vol. 43, pp. 79-81, Jun. 2010.

[7] N. H. Li, J. Mitchell and W. Winsborough, "Design of a Role-based Trust-management framework", SP '02 Proceedings, Washington, pp. 114-130, 2002.

[8] Defense Information Systems Agency, "A security architecture for NET-CENTRIC Enterprise Services", Version 0.3, Mar. 2004.

[9] E. Bertino and P. Bonatti, "TRBAC: a temporal Role-based access control model", ACM Transactions on Information and System Security, New York, vol. 4, Aug. 2001.

[10] S. Piromruen and J. B. Joshi, "An RBAC framework for time constrained secure interoperation in multi-domain environment", WORDS '05 Proceedings, pp. 36-45, Pittsburgh, Feb. 2005.

[11] X. L. Shi, Y. Fang, Y. Zhang, Y. L. Li and L. P. Sun, "A Role-based access control model in distributed environment", Journal of Sichuan University, vol. 44, pp. 303-307, Apr. 2007.

[12] Y. Demchenko and C. Laat, "Domain based access control model for distributed collaborative applications", e-Science '06 Proceedings, Amsterdam, pp. 24-24, 2006.

[13] A. Kapadia, J. Al-mohtadi, R. Campbell and D. Mikunas, "IRBAC 2000: secure interoperability using dynamic role translation", Technical Report in Illinois University, 2000.

[14] C. S. Fu, N. Xiao, Y. J. Zhao and T. Chen, "Negotiation-Based Dynamic Role Transition in Data Access across Multi-VOs", Journal of Software, vol. 19, pp. 2754-2761, Oct. 2008.

[15] A. Kamath, R. Liscano and A. E. Saddik, "User-Credential based role mapping in multi-domain environment", PST '06 Proceedings, New York, 2006.

[16] G. Geethakumari, D. Negi and D. Sastry, "A cross – domain role mapping and authorization framework for RBAC in Grid systems", International Journal of Computer Science and Applications, vol.06, pp. 1-12, 2009.

[17] Y. Demchenko, L. Gommance, C. Laat, M. Steenbakkers, V. Ciaschini and V. Venturi, "VO-based dynamic security associations in collaborative grid environment", CTS 2006. International Symposium on, Amsterdam, pp. 38-47, 2006.

[18] P. Robinson, Y. Karabulut and J. Sap, "Dynamic Virtual Organization management for Service Oriented enterprise

applications", Collaborative Computing: Networking, Applications and Worksharing, Karlsruhe, 2005.
[19] G. J. Klir, B. Yuan, "Fuzzy Sets and Fuzzy Logic: theory and applications", Prentice Hall PTR, Finland, 1995.
[20] W. Tang and Z. Chen, "Research of subjective Trust Management model based on Fuzzy Set theory", Journal of Software, vol. 14, pp. 1401-1408, 2003.

Zan Yang was born in Tianjin, China, Jun. 1983. He is a Ph. D. student in the Department of Computer Science and Technology, Institute of Command Automation, PLAUST, Nanjing, China. Currently 2 years degree is earned. His major field of research includes network security, service oriented computing. He is simultaneously an engineer of the Institute of EESEC of China, Beijing, China.

Lin Yang was born in Hefei, China, Feb. 1970. He is a Ph. D. in Computer Science and Technology. His major field of research includes network security, broadband communication network. He is a researcher in the Institute of EESEC of China, Beijing, China since he was graduated.

Xiang-yang Luo was born in Hubei, China, 1978. He is a Ph. D. in Computer Science and Technology. His major field of research includes information security, digital watermarking. He is a researcher in the Institute of information engineering of China.

Lin-ru Ma was born in Xian, China, Oct. 1978. She is a Ph. D. in Computer Science and Technology. Her major field of research includes network security. She is a researcher in the Institute of EESEC of China, Beijing, China since she was graduated.

Bao-sheng Kou was born in Beijing, China, Aug. 1972. He received the Bachelor Degree in Computer Science and Technology in 1994. His major field of research includes command automation and network security. He is an engineer in the China troop of 61046, Beijing, China.

Kun Zhang was born in Xinxiang, China, Aug. 1980. He received the Bachelor Degree in Cipher Technology in 2002. His major field of research includes network security. He is an engineer in the China troop of 61046, Beijing, China.