Delegation Management in Service Oriented Decentralized Access Control Model

Houxiang Wang, Ruofei Han, Xiaopei Jing, and Hong Yang
Information and Electric College, Naval University of Engineering, Wuhan, China
hrf_402@sina.com

Abstract—Net-Centric Environment (NCE) and Service Oriented Architecture (SOA) are new emerging concepts that influence development of information systems. They bring in high risk as well as high sharing. In order to ensure the security of the future military systems, a service oriented decentralized access control (SODAC) model is proposed in previous work. The management to delegation is investigated in further for SODAC model. The entire delegation mechanism and process is discussed, and the three most concerned issues in delegation are well fulfilled with the proposed mechanism.

Index Terms—net-centric, service oriented, access control, delegation management, trust negotiation

I. INTRODUCTION

Delegation is an important issue in access control area. It is an activity that one party hands over its authority to another to accomplish certain task [1]. So the authorization is controlled by the delegation provider according to his own policy, which is out of the range of centralized access control from some point of view. However, there are still several issues that are concerned widely in behalf of the delegation provider, including that:

- Minimum authority delegation. Only the necessary authority for the target task should be delegated, so that the risk of authority leakage can be minimized.
- Time limit to delegation period. Once the target task is accomplished, the delegation should be banished. Otherwise, it may be misused by the delegation requester.
- Control on re-delegation. The permission for delegation requester to re-delegate the authority he achieved from delegation must be constrained in behalf of the original delegation provider, or the authority leakage will be inevitable.

Many researches have been done on these problems [2], but few of them take all the three into consideration. With the development of network and information technology, nowadays systems are almost designed according to service-oriented architecture (SOA), both in commercial or military applications. The latest architecture product, DoD Architecture Framework version 1.5 (DoDAF v1.5), proposed by department of defense of USA, has absorbed the spirit of SOA to guide development of military systems in Net-Centric Warfare (NCW) [3].

Though the Net-Centric Environment (NCE) enables high resource sharing and collaboration between information systems, it also increases the possibility of malicious attack and information leakage. On the other hand, resources and control authorities are both highly distributed in this open environment. To better manage authentication and authorization in this environment, we have proposed a service oriented decentralized access control (SODAC) model in our previous work [4]. In this research, we will further investigate the delegation management in SODAC model. The entire delegation mechanism and concerned issues will be discussed in it.

II. RELATED WORK

A. SODAC model

SODAC model is proposed to meet security requirements of the complicated changefully NCW. It simplifies all the elements involved in the access process into “Entity” and “Service”. Service stands for independent function or working process; Entity stands for all the other participating elements in the service. Since these participating elements can be service invoker in one working process, or service provider in another, we named them “Entity” uniformly. For better management, the uniform “Entity” can be divided into several subclasses according to practical applications. Entities are described with “Attribute”, and Services are constrained with “Policy”, which are both subclass of “Entity”. The structure of SODAC model for military application is shown in Figure 1.

Service composition is a mechanism to issue high level functions that take existing services as a part. SODAC model absorbs spirit from workflow management in task-and-role based access control to manage service composition. There are many benefits to introduce composition in. Firstly, it takes full advantage of previous achievements, so that no repeated development is needed, and much more complicated application would be feasible. Secondly, it facilitates developers to compose new functions rapidly according to user’s requirement. Lastly, it shares system load with others, which will also make full use of resources all over NCE. However, the composing service may not be stored locally, even be issued by another domain. In order to protect the information involved in the collaboration, the collaborating parties should come to an agreement on accessing rules through trust negotiation, and delegate corresponding authorities to the other side, which is the most concerning issue in this research.
B. Membership-based negotiation

If the two sides involved in access or delegation process are unfamiliar, they have to negotiate in behalf of each side before collaboration. In our previous work, we have proposed a membership-based access control mechanism for trust negotiation [5].

The membership-based access control classifies external subjects into several groups according to predicted accessing manner. Each group is associated with several particular services, which is provided to members of the group, and with particular policy that is used to constrain subjects to join into the group. Unlike the role in role-based access control (RBAC), which is used to describe responsibility and ability of users in internal working process [6], group is used to describe the level of service that can be provided to its member, and the corresponding conditions on them. So, group is also a set of services and a set of policies. Also, the group set is constructed with various relationships, as shown in Figure 2. The circle stands for policy, and the space surrounded by circle stands for group. If a user wants to become member of a group, he has to satisfy the policy, as the dashed line going across the circle from outside to inside.

When an accessing subject invoked a service whose corresponding membership he does not have, a negotiation route will be produced to guide him joining into the group according to his attributes and group structure. Take the “$G_r \subseteq G1 \cup G2$” in Figure 2. for example. Suppose that a subject wants to invoke a service provided to members of $G_r$, and he is an entirely new customer who even does not have membership of $G1$ and $G2$, then the negotiation route will guide him to be member of $G1$ and $G2$ in the first step as required in $P1$ and $P2$, and to be member of $G_r$ in the second step to satisfy requirement of $P_r$. So the policies, $P1$ and $P2$, works just as the prepositive policy of $P_r$. If the subject can not pass $P1$ and $P2$, $P_r$ will not be disclosed to him. However, if he passes $P1$ and $P2$ but does not satisfy $P_r$, he can be authorized to invoking services provided to members of $G1$ and $G2$, which may be restricted issue of the target service.

However, the negotiation for delegation may be different in some way, but the main mechanism is also applicable.

Figure 1. Structure of SODAC model

Figure 2. Membership described with group relationship
III. DELEGATION MANAGEMENT IN SODAC

In SODAC model, accessing to resources must be performed though invoking the corresponding service. However, the authorization to service invocation lies on authentication of the invoking entity’s attributes according to service policy. So the delegation of authority is turned to be transfer of attribute certificate in the end. Membership can also be described as an authority attribute, which can be introduced here to simplify the delegation authorization.

A. Virtual attribute

In order to constrain the ability of the delegation requester, we introduce in a new kind of attribute, named virtual attribute.

Definition 1 (Virtual Attribute): a derivative attribute issued by delegation provider to delegation requester to indicate delegated authority.

As shown in Figure 3, the virtual attribute is just a certificate assertion without attribute value, and the provided accessing service is just an invoking interface to that of original service. Accessing services of virtual attribute are also issued by delegation provider according to the authority delegated to delegation requester, which is just a subset of the provider’s authority.

B. Delegation constraint

With virtual attribute, we have restricted the delegation requester from achieving the practical authority himself. In order to place more constraints on the delegation, the delegation provider can issue additional policy on the accessing service. The management to virtual attribute is referring to the template-instance management of task-role-based access control (T-RBAC) [7]. With that, the delegation provider will reserve records of all the delivered virtual attributes. When an attribute request arrives, the delegation provider will first recognize the corresponding virtual attribute that it comes from, and then control the service process according to policy of the original service and that of particular delegation service.

The SODAC model absorbs idea from usage control (UCON) that authentication can be configured to be performed before, simultaneous with or after the service is ongoing [8]. Every policy is attached to an authentication point (AP) along the service route, as shown in Figure 4. So that, we can easily define policy to revoke delegation dynamically as soon as its validity expire.

For example, if the delegation provider wants to set a policy to constrain the times that delegated authority can be used. It can be attached to an authentication point set after the service is accomplished. So the policy will be authenticated every time after the service invocation is finished, and once the authentication returns false, the virtual attribute will be revoked.

C. Delegation chain

As discussed in the introduction, the permission to re-delegate authority must be controlled. However, the situation still exists that another party C may request relegation from party B whose own authority is delegated from party A, especially in SOA. With service composition, it is an ordinary form that some service invoking other one in its working process, but is a composing part of another service simultaneously, known as service chain.

Since the actual accessing authority belongs to the original delegation provider, the eventual delegation confirmation should be controlled by it. But the delegation process should be mediated by other delegation party to protect information of the original provider. As the same as many previous research in re-delegation, we introduce in delegation chain to constrain permission of re-delegation.
As shown in Figure 5, \( E_0, E_1, \ldots, E_n \) are entities; \( VA_1, VA_2, \ldots, VA_n \) are virtual attributes; \( OA \) is the corresponding original attribute; \( p_1, p_2, \ldots, p_n \) are policies. When a new entity request delegation from some delegating party, it will transfer the request along the delegation chain back to original entity. If the original entity approves the request, he will produce a new virtual attribute derived from the virtual attribute be requested, and send it back to delegation requester along delegation chain. The new delegated virtual attribute is restricted by all the policies assigned to its previous ones in delegation chain. So, the longer the delegation chain is, the higher restriction will be set for the new delegation requester. The original delegation provider can also define delegation negotiation policy previously to restrict length or width of the delegation chain. The delegation chain is restored and maintained only at the original delegation provider’s side. Once a virtual attribute is revoked, all the other ones derived from it should be revoked too.

### D. Delegation process

When the delegation requester is unfamiliar to the delegation provider, a negotiation is necessary to determine whether to approve it or not. The situation is similar to trust negotiation in regular service invoking. The difference is that the final certificate sent back to requester is not membership, but virtual attribute. We introduce in a particular delegation service at delegation provider’s side. All the services for attributes of the entity can be invoked from the delegation service. So that, different negotiation route can be produced according to the attribute requested. The delegation process is shown in Figure 6.

![Figure 6. Sequence diagram of delegation process](image)
IV. CONCLUSION

The NCE is a typical decentralized environment, resources and authorities are distributed all over the involved entities. With SOA, it is easy to share resources with others by issuing services. But the authority to access resources must be controlled to protect sensitive information. In this research, we investigated in the management to delegation at the base of SODAC model, and take the three widely concerned issues in consideration. With the proposed mechanism, we could protect the actual authority from leaking out, control delegation according to the practical requirement, and constrain its validity and re-delegation.

This work is just a stepped accomplishment of our research on SODAC model. We only proposed mechanism to control delegation, practical implementation and validation will be investigated in future work.

REFERENCES


