情報保護學會論文誌
第11券 第6號, 2001. 12

분산환경에서 도메인-RBAC을 이용한 권한위임

이 상 하*, 채 송 화**, 조 인 준***, 김 동 규****

Delegation using D-RBAC in Distributed Environments

Sang-Ha Yi*, Song-Wha Chae**, In-June Jo***, Dong-Kyoo Kim****

요 약

분산환경의 정보보호를 위해 인증 및 접근통제 서비스는 필수적인 요구 사항이다. 권한위임은 개시자 편에서 중개자가 목적지 응용서비스의 접근권한을 허용하는 과정으로 분산환경에서 안전성을 유지하면서 서비스의 가용성을 증대시키는 방법이다. 단일 도메인 내에서 역할기반 접근통제(RBAC) 권한위임이 용이하게 제공된다. 그러나 다중 도메인 환경에서 도메인간 개시자가 중개자에게 문서공유를 위해 권한위임을 요청할 경우 권한남용 방지를 위해서 접근권리을 제한하여 처리할 수 있도록 권한위임 제약 기능이 필요하다. 본 논문에서 문서공유 문제를 RBAC으로 해결하고자 권한위임 제약을 하는 권한위임 뷰를 제안한다. 제안된 모델은 다중 권한위임에서 발생하는 문서과잉 노출을 방지할 뿐만 아니라 통신 및 시스템의 과부하 현상방지, 효율적인 보안관리가 가능하다.

ABSTRACT

Authentication and access control are essential requirements for the information security of distributed environment. Delegation is process whereby an initiator principal in a distributed environment authorizes another principal to carry out some functions on behalf of the former. Delegation of access rights also increases the availability of services offer safety in distributed environments. A delegation easily provides principal to grant privileges in the single domain with Role-Based Access Control(RBAC). But in the multi-domain, initiators who request delegation may require to limit the access right of their delegates with restrictions that are called delegate restriction to protect the abuse of privilege. In this paper, we propose the delegation view as function of delegation restrictions. Proposed delegation view model not only prevent over-exposure of documents from granting multiple step delegation to document sharing in multi-domain with RBAC infrastructure but also reduce overload of security administrator and communication.

keyword : *delegation view, delegate, multi-domain, document sharing, RBAC*

Ⅰ. 서 론

분산환경에서 인증 및 접근통제 서비스는 정보보호를 위해 필수적인 기능이다. 인증 서비스에 관하여 다양한 연구들이 논의되었다. 인증은 사용자나 프로세스의 신분을 밝히는 과정이고, 접근통제는 정당한 인증 후 특정 오퍼레이션이나 자원의 사용을 사용자 프로세스에게 허용하는 방법이다. 특히, 클라이언트는 응용서버에게 신임장을 전달하고 응용서버의 객체에 오퍼레이션을 요청한다. 응용서버는 접근통제

* 동서울대 정보통신공학과(shyi@haksan.dsc.ac.kr)
** 아주대학교 정보통신공학과 정보통신 및 시큐리티 연구실(portula@chollian.net)
*** 배재대학교 컴퓨터공학과 네트워크 & 보안연구실(injune@mail.paichai.ac.kr)
**** 아주대학교 정보 및 컴퓨터공학부 교수(dkkim@madang.ajou.ac.kr)

서비스의 매개변수로 클라이언트 데이터를 넘겨받아 서비스를 처리한다. 현재 대부분 응용 프로그램은 자신이 소유하고 있는 자원들을 자신들의 특정 응용에만 적용되는 제한적인 접근통제 방법을 사용하여 접근권한을 결정한다. 일반적으로 응용에 관련한 특정 접근권한 및 권한정보의 관리를 위한 기반구조를 제공하는 것이 대부분이다.

RBAC(Role-Based Access Control)[3,4,12] 기반의 분산환경에서 한 클라이언트 프로세스가 다른 응용서버 프로세스로 서비스를 요청한다고 하면, 클라이언트의 요청을 받은 응용서버가 또 다른 응용서버로 서비스 요청하여 서비스를 제공하는 관계가 성립한다. 이는 두 기업간의 조직을 가지고 있는 역할기반구조의 경우라면 분산환경의 접근권한은 한 사용자가 안전하게 다른 사용자에게 자원을 공유하기 위한 방법으로 일시적 접근권한을 양도하는 경우이다.[5,6,10] 이는 곧 권한위임의 관계로 성립한다. 분산 환경에서 권한위임은 응용서비스의 안전성을 유지하면서 가용성을 증대시키는 방법이다.[1]

상호간 안전한 접근권한의 양도로 생기는 일시적인 권한위임은 일련의 처리과정이 발생하는 경로가 생긴다. 다단계 권한위임이 발생할 경우, 권한위임의 연결고리에서 권한위임을 요청한 쪽을 개시자라 하고 요청을 받은 사용자는 중개자라 한다. 최종 권한위임을 처리하는 곳을 목적지 또는 목적지 서버라 한다.

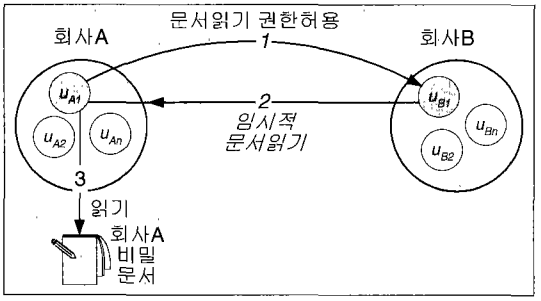
다중 도메인간 다단계 권한위임이 발생할 경우, 권한위임의 연결고리에서 의도하지 않는 정보의 과잉노출, 신뢰할 수 없는 중개자의 고의적인 위임정보의 변경 및 안전한 권한위임의 전달 방법이 문제가 된다.

본 논문은 분산환경에서 역할기반 접근통제 서비스를 기반으로 하고 있는 다중 도메인에 연관되어 있는 두 기업간의 문서공유에서 발생되는 문제점을 역할간의 권한위임으로 해결하기 위한 방안을 제시하고자 한다. 권한위임 연결고리에서 개시자가 중재자에게 권한위임을 요청할 경우 권한남용을 방지하기 위해서 개시자의 접근권리을 제한하여 처리할 수 있도록 권한위임 제약을 가해야 한다. 권한위임 제약을 가하기 위해 개시자 측에 권한위임 뷰를 두어 인터넷 분산환경에서 다양한 응용의 요구를 만족하는 융통성 있는 권한위임 뷰 모델을 제시한다.

본 논문은 분산환경에서 발생할 수 있는 기업간의 문서공유 문제에 대하여 2장에 설명하고, 3장은 역할기반 접근통제에서 발생할 수 있는 권한위임의 적용의 제약성을 기술하고, 4장에서는 분산환경 문서공유 문제를 역할기반 접근통제 권한위임을 통하여 해결하기 위한 권한위임 뷰 모델을 제시하고, 5장에서 결론을 맺는다.

Ⅱ. 연구의 동기 및 관련연구

분산환경에서 두 회사간의 접근통제 서비스 관점에서 문서공유를 위한 다음과 같은 환경이 있다. [그림1]에서 두 회사A, B의 관계에서 회사A의 관리자 및 일반 사용자가 자신의 비밀문서를 회사B의 특정 사용자에게 일시적으로 읽기 허용을 하여 문서를 읽을 수 있게 하는 문서공유 문제가 있다. 두 회사간에 어떤 문서를 볼 수 있는지 사전에 정확한 시간이 알려져 있지 않다고 하면 협상과정에서 어떤 문서는 볼 수 있을 수도 있고 없을 수도 있다고 가정한다.



[그림 1] 회사간 문서공유 문제

문서공유 문제의 단순 해결책으로 회사A의 시스템 관리자가 회사B의 어떤 사용자에게 필요시 필요한 각 문서에 대해서 읽기를 허용하는 접근통제 리스트(ACL : access control list)를 매번 수정함으로 이루어진다. 이러한 방법은 시스템 관리자 편에서 보면 너무 많은 수정 작업이 요구된다. 또한 시스템 관리자의 실수로 인해 ACL이 잘못된 상태로 생성될 수 있다.

앞에서 제시한 문서공유 문제는 권한위임 방안으로 OSF DCE(Open Software Foundation Distributed Computing Environment)[8] 환경에서 단편적인 방법이 제시되었다. OSF DCE의 문서공유 문제의 해결로서 개시자와 권한위임을 받은 중개자는 양쪽 모두 접근통제 리스트로부터 접근할 권한을 획득해야 한다. ACL은 초기 시스템 설정 단계에 시스템 관리자에 의해 모든 문서에 대해서 권한위임에 의하여 회사B의 자격으로 접근하여 중개자가 볼 수 있게 허용하도록 수정한다.

권한위임에 의해서 회사B의 위치에서 볼 수 있는 모든 문서에 대해서 회사A에 속한 개시자가 그 문서를 보려는 회사B의 자격을 제한하기 위해서 권한위임 제약을 획득해야한다. 권한위임 제약은 EPAC (Extended Privilege Attribute Certificate)을 생성하여 회사B로 넘겨준다.[8] 이 과정에서 발생되는 3가지 문제점을 살펴보면 다음과 같다.

첫째, 새로운 문서를 보려고 할 때 양쪽회사에 대해서 접근통제서버의 교환이 요구된다. 이는 접근통제 서버에서 허용되는 접근권한의 변경이 발생한다. 마찬가지로 권한위임 제약의 변경으로 인한 새로운 권한위임을 획득해야 한다. 이는 통신 트래픽 및 시스템의 과부하를 가져온다.

둘째, 회사A에 속한 또 다른 사용자가 또 다른 문서공유 작업을 위해서 회사B에 속한 동일한 사용자에게 접근을 허용할 수 있다. 즉, 다중 권한위임이 성립하는 회사A의 사용자는 회사B의 사용자 정보를 모르고 있는 상태이다. 두 번째 사용자는 자신의 권한위임 제약을 가지고 문서보기 가부처리를 하지 못한다. 그래서 이러한 경우가 빈번히 발생할 때 회사B는 모든 문서보기를 할 수 있다.

셋째, 관리자에 의해서 접근통제리스트는 권한위임 초기 및 종료과정에서 매번 수정되어야 한다. 이들 수정은 관리자의 추가적인 작업이 요구된다.

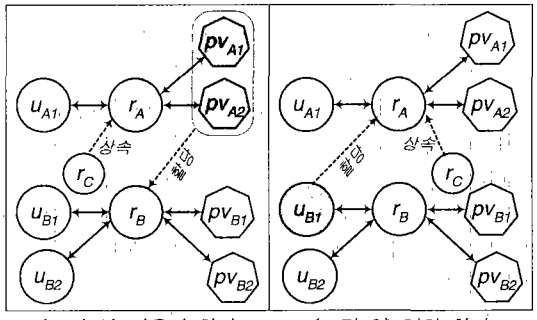
Ⅲ. 문서공유 문제의 RBAC 적용

앞장에서 제시한 문서공유 문제를 RBAC 기반으로 해결하기 위해 RBAC 모델이 가지고 있는 이점과 제한점을 살펴본다. 또한 다중 도메인 환경의 적용성을 고려한다. 역할기반 접근통제에서 역할 및 사용자에 관련된 접근권한은 자신이 연관되어 있는 접근권한을 획득하기 위해서 관련된 역할의 구성원이 되는 것이다.[13,14,15] RBAC에서 사용자의 권한위임은 개시자가 속한 역할의 능력을 권한위임 받으려는 다른 사용자가 속한 역할에서 개시자의 편에서 접근권한 능력을 처리할 수 있는 권한을 가짐으로 이루어진다.

RBAC 기반 계층적 역할구조에서 상위역할은 하위역할의 접근권한을 상속받는다.[7,11] 계층적 역할구조에서 사용자사이에 권한위임을 하기 위해서 복잡한 구성이 되고 여러 가지 경우를 고려해야 한다. 이런 구조에서 잘못된 권한위임은 무용적일 수 있고 다른 위험성을 내포하고 있다.

3.1 RBAC의 권한위임 과정

역할 계층구조에서 하향 권한위임만 의미를 가지며 사용자를 상위 역할로 할당하는 방법과 상위에 할당된 권한을 하위 역할로 할당하는 방법이 있다. 하위역할에 속한 사용자가 승진이 이루어진 것처럼 상위역할의 구성원으로 되는 것이 한 예가 된다.



〔그림 2〕 사용자 할당 〔그림 3〕 권한 할당

〔그림 2〕과 같은 조직체계를 가지는 역할 계층구조에서 권한위임에 의하여 사용자를 할당하는 방식으로 하위 역할 r_B에 속한 사용자 u_{B1}가 상위 역할 r_A의 구성원으로 될 경우 역할 계층의 상속에 의해서 의도하지 않는 역할 r_C의 권한까지 소유한다.

만약 〔그림 3〕에서 권한위임에 의하여 역할 r_B에 역할 r_A가 소유하고 있는 권한을 할당하는 방법은 r_B의 구성원인 u_{B2}에까지 의도하지 않는 권한위임이 발생한다. 위와 같은 방법으로 기존의 RBAC 모델[12]을 적용할 경우 권한위임을 제대로 반영할 수 없고 실세계에 발생하는 권한위임을 반영하기 위해서 다음과 같이 정의한다.

[정의 1] $r_A' = CreateR(r_A, Radmin(r_A), T_v)$

$r_A \geq r_A'$의 부분순서 관계를 만족하는 역할을 생성한다. r_A'은 새롭게 생성한 역할, $Radmin(r_A)$은 역할 r_A을 관리하는 개별 역할관리자, T_v는 새롭게 생성한 역할 r_A'의 시간속성을 가지는 유효기간이다.

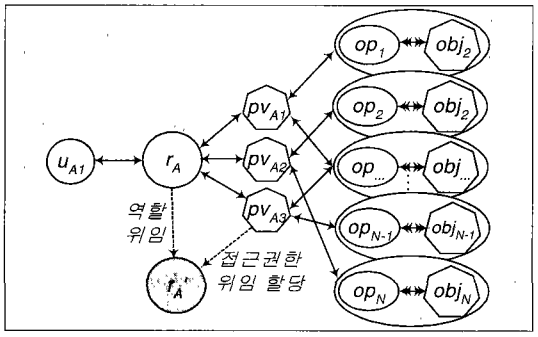
권한위임을 요청하는 개시자가 원래 할당된 역할 r_A의 상위에 해당하는 새로운 역할 r_A'을 생성하여 권한위임에 필요한 접근권한을 할당한다. 즉, 새로 생성한 역할에 부분적인 권한만 상속받도록 한다. 이와 같은 방법은 권한위임 시 새롭게 생성한 역할 r_A'가 유효기간의 속성을 가지고 있기 때문에 권한위임 취소가 용이할 뿐만 아니라 권한위임 제약의 적용이 간편하다.

RBAC 모델의 하향 권한위임은 사용자에게 직접 권한을 할당하는 것이 아니라 역할에 접근권한을 할당하기 때문에 사용자에게 직접 권한을 할당할 수 없다. 그래서 다음과 같은 정의에 의해 부분권한을 할당하는 것으로 한다.

[정의 2] 접근권한 (pv)은 연산 (op)과 객체 (obj)의 쌍으로 구성된다. 접근권한의 요소는 $2^{m\times n}$이다. 여기서, m은 op의 개수, n은 obj의 개수이다.

여기에서 접근권한 요소를 $pv=(op_m, obj_n)$로 표현한다. 여기에서 객체 (obj)는 시스템의 내적 자원(예, 파일, 디렉토리, 등)과 시스템의 외적 자원(예, 프린터, 디스크, CPU, 등)으로 구성된다. 연산 (op)은 객체 (obj)에 행해지는 연산으로 운영체제 측면에서 읽기, 쓰기, 실행으로 표현할 수 있고, 데이터베이스 측면에서 실행, 삽입, 삭제, 선택, 갱신 등을 의미한다.

[그림 4]에서 주어진 자원을 접근권한에 할당하고 접근권한 pv을 역할 r_A에 할당한다. 권한위임이 발생할 시 새로운 역할 r_A'을 생성하여 원래에 할당된 모든 권한 (pv_{A1}, pv_{A2}, pv_{A3}) 모두를 권한위임을 위해 할당하는 것이 아니라 부분적인 권한 pv_{A3}만을 역할 보안 관리자가 필요한 권한만 할당하도록 하여 최소 권한원칙을 준수하도록 한다.



[그림 4] 부분권한 할당

3.2 임무분리의 권한위임

상호 배타적인 접근권한을 가지는 역할사이에 상호역할간 권한위임은 판매부서장이 판매부서의 감사를 하기 위해서 감사부서의 구성원에게 권한위임을 할 수 있다. 감사부서의 구성원으로서 원래 역할 구성원의 역할뿐만 아니라 판매부서의 부서장의 역할을 가질 수 있다. 이는 상호 배타적인 역할간의 권한위임이 발생되었으므로 임무분리가 준수 되어야한다.

상호역할간 권한위임에서 임무분리는 주어진 특정 역할을 여러 계층의 사람이 수행할 경우에 발생한다.[2] 따라서 이러한 역할은 다수의 부분역할로 분리되어야 하고, 이렇게 분리된 부분역할이 접근권한이 허용된 사람에게만 할당함으로써 정보자원의 무결성을 보장할 수 있다. 즉, 분리된 역할이 그 역할을 할당받은 사람에 의해서만 수행되는 것을 보장해야한다. 따라서 각 개인은 자신에게 허용된 부분역할만을 수행할 수 있다.

특히, 동일 사용자가 동시에 수행하지 말아야 하는 상호 배타적 접근권한을 갖는 상호 배타적 역할 (M_R: Mutual Exclusive Role)에 대해서는 사용자가 이를 준수하면서 역할의 접근권한을 수행하도록 하여야 한다. 임무분리는 사용자에게 역할을 할당하거나 사용자가 역할을 수행하는데 있어서 정적 임무분리와 동적 임무분리로 구분할 수 있고[7] 접근권한을 사용하여 역할을 할당하면 임무분리의 표현 및 관리를 용이하게 할 수 있다.

따라서, 상호 배타적 역할은 상호 배타적 접근권한을 갖는 역할로써, 할당된 접근권한을 사용자가 동시에 수행하는 것을 방지해야 한다. 이에는 하나의 역할에 두개 이상의 상호 배타적 접근권한이 주어지는 경우와 두 개 이상의 역할에 상호 배타적 접근권한이 주어져서 수행되어야 하는 경우가 있다.

참고문헌[16]에서 상호 배타적 역할에 대해서 접근권한을 상속할 수 없도록 제약을 두고 있다. 따라서, 상호 배타적인 역할은 상호 배타적인 접근권한을 분리한 부분적인 권한만 역할간에 권한위임이 이루어져야 한다. [그림 4]에서처럼 상호 배타적인 접근권한이 부분적인 권한으로 분리할 수 없는 경우 권한위임이 이루어질 수 없으며 자신에게 할당된 고유 접근권한 형태로 표현된다.

Ⅳ. 문서공유 문제해결을 위한 RBAC의 방법

회사 조직간의 문서공유 문제를 RBAC 기반으로 해결하기 위해서 앞에서 제시한 제약들을 준수해야 한다. 모든 사용자의 환경은 공개키 기반(PKI : Public Key Infrastructures)에서 이루어진다. 모든 사용자 및 서버는 자신 도메인 인증당국(CA : Certificate Authority)에서 X.509 인증서를 발급 받아 보유하고 있다는 가정을 한다. 또한 조직체 내의 상위 수준에 있는 도메인 보안 관리자와 하위 관리자로 개별 역할 보안 관리자를 분리하여 관리하는 다계층 보안관리 구조를 가진다.

4.1 개별 역할 보안 관리자 수준의 권한위임 생성

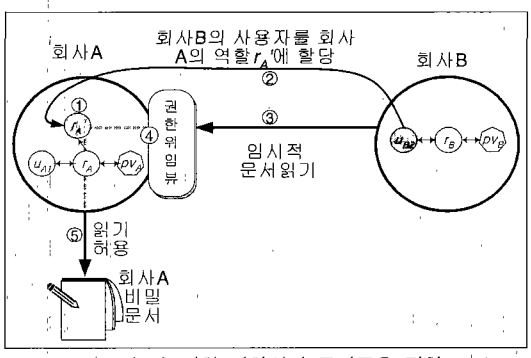
조직체내의 도메인 보안 관리자와 역할 보안 관리자를 분리하여 관리한다. 분리된 관리는 조직체 관리 및 개별 사용자의 관리 부하를 줄일 수 있다. 도메인 보안 관리자는 조직 전체의 보안정책을 통합관리하고 역할 보안관리자는 도메인 정책에 의해 초기에 도메인 관리자에 의해 부여된 접근권한들에 한하여 역할 내에서 발생되는 보안 문제를 관리한다. 이런 계층적인 관리구조는 분산환경에서 상호견제 관리로 관리의 과부하나 권한이 집중되는 현상을 방지할 수 있는 권한남용 방지 구조이다. 또한 권한위임 뷰의 정의는 다음과 같다.

[정의 3] 권한위임 뷰 목록은 $Veiwlist(r_i,\ T_v,\ Radmin(r_i),\{pv_i\},domain_map\uparrow)$이다.

권한위임 발생 시 새롭게 생성한 역할 r_i'을 기준으로 유효기간 속성 T_v, 역할 관리자 *Radmin*, 권한위임이 허용된 접근권한의 집합 pv_i, 도메인간의 매핑 테이블의 포인터 $domain_map\uparrow$를 가지는 역할 권한위임 뷰 목록이다.

Viewlist 데이터는 새로운 권한위임의 발생시 [정의4] 및 [정의5]에 의해서 갱신된다. 이 권한위임 뷰의 데이터에 따라 권한위임 제약이 부여되고 권한위임에 의한 접근통제가 이루어진다. 개시자는 권한위임 뷰는 제공하지만 ACL을 주지 않는 방안이다. 권한위임 뷰는 개시자 도메인에 위치한다.

[그림 5]은 두 조직간의 역할 계층구조 관점에서 보면 하향 권한위임이 성립한다. 그래서 회사A의 역할 r_A에 소속되어 있는 사용자는 회사B의 역할 r_B의 구성원인 사용자에게 문서를 보여주기 위해서 권한위임을 요청한다. 회사A의 사용자 u_{A1}는 개시자가 되고 회사B의 사용자 u_{B1}는 권한위임 중개자이고, 다시 회사A의 역할 r_A는 목적지가 된다.



[그림 5] 역할 권한위임 문서공유 절차

[그림 5]에서 두 회사의 역할간 권한위임 절차는 다음과 같이 이루어진다.

① [정의 1]에 따라 회사A의 역할 r_A는 새로운 역할 r_A'을 생성한다. 새롭게 생성한 역할은 일시적인 역할 r_A'로 원래의 역할 r_A가 가지고 있는 모든 권한위임을 하는 것이 아니라 보안 역할관리자가 부분적인 권한위임을 할당하고 권한위임 뷰에 등록한다.

② 새롭게 생성한 역할 r_A'에 회사A의 사용자 u_{A1}의 권한위임 요청에 의해서 회사B의 역할 r_B구성원 u_{B1}사용자를 할당하는 권한위임을 한다.

③ 회사B의 사용자 u_{B1}는 회사A의 역할 r_A'에 권한위임을 할당받았으므로 읽기 허용에 의한 문서 읽기를 시도한다.

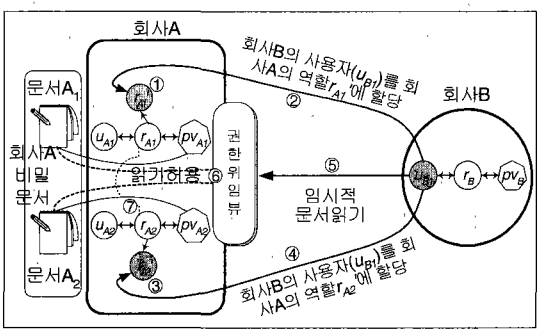
④ 회사B의 사용자 u_{B1}읽기를 시도할 경우 권한위임 뷰에 정당하게 등록된 사용자라면 ⑤의 절차에 의해서 읽기 허용을 한다.

회사A의 개시자는 회사B의 중개자에게 새롭게 생성한 역할만을 넘겨준 결과가 된다. 이는 RBAC 접근통제리스트는 회사A가 주도하고, 필요한 접근권한 뷰만 넘겨준 결과를 가진다. 역할 권한위임에서 세부 권한위임을 제어할 수 있는 방안은 권한위임 뷰를 통하여 역할기반 접근통제 서비스를 제공하는 것이 더 세밀하게 제어가 가능하다. 접근권한을 위한 권한위임 뷰를 주지만 ACL을 주지 않는 방법으로 개시자의 관점에서 통제가 가능하다. 이는 2장에서 제시한 문서공유 문제에서 첫째, 셋째의 해결책이 된다. 그렇지만 둘째의 다중 권한위임 문제는 남아있는 상태로 4.2절에서 해결 방안을 제시한다.

4.2 다중 권한위임

[그림 5]와 같이 이미 권한위임이 발생한 상태에서 회사A에 속한 임의의 다른 사용자가 회사B의 동일 사용자에게 문서공유를 위해 다른 문서에 대해서 권한위임을 할당한다고 하면 다중 권한위임이 발생한다. [그림 6]은 도메인간 공통상위 접근권한 상속(CS : Common Senior Inheritance)[16] 관계로 볼 수 있는 다중 권한위임이 발생할 경우 위임 절차는 다음과 같이 이루어진다.

① 회사A의 역할 r_{A1}는 새로운 역할 r_{A1}'을 생성한다. 새롭게 생성한 역할은 일시적인 역할 r_{A1}'로 원래의 역할 r_{A1}가 가지고 있는 모든 권한위임을 하는 것이 아니라 부분적인 권한위임을 할당한다.
② 새롭게 생성한 역할 r_{A1}'에 회사B의 역할 r_B 구성원 u_{B1}사용자를 할당하는 권한위임을 한다.
③ 회사A의 역할 r_{A2}는 새로운 역할 r_{A2}'을 생성한다. 새롭게 생성한 역할은 일시적인 역할 r_{A2}'로 원래의 역할 r_{A2}가 가지고 있는 모든 권한위임을 하는 것이 아니라 부분적인 권한위임을 할당한다.
④ 새롭게 생성한 역할 r_{A2}'에 회사B의 역할 r_B의 동일한 구성원 u_{B1}사용자를 할당하는 권한위임을 한다.
⑤ 회사B의 사용자 u_{B1}는 회사A의 역할 r_{A1}' 및 r_{A2}'에 권한위임을 할당받았으므로 회사A의 문서 A_1 및 A_2을 동시에 읽기를 획득한 경우가 발생한다. 그래서 두 문서 읽기 허용이 동시에 문서 읽기를 시도한다.
⑥ 회사B의 사용자 u_{B1}가 동시에 읽기를 시도하는 방지는 권한위임 뷰에서 접근허용 가부를 처리한다.
⑦ 회사A의 두 역할은 상호 배타적인 역할인지를 확인해야 하며 역할간 동적 임무분리를 준수하는지를 검증한다.



〔그림 6〕 다중 권한위임 관계

〔그림 6〕과 같이 다중 권한위임이 발생할 경우 권한위임 뷰가 없다면 회사B의 사용자 u_{B1}는 회사A의 비밀문서 보기를 통하여 문서과잉 노출을 하는 경우가 발생한다. 문서과잉 노출을 방지하기 위해 역할 계층구조에서 개별 역할 제약의 방안으로 방지할 수 없다.

두 도메인간의 다중 권한위임에 의해서 발생하는 문서공유 문제는 문서공유 허용을 해주는 회사A에 권한위임 뷰를 둠으로 해결하고자 한다. 단 초기에 권한위임 뷰는 회사B의 모든 사용자는 접근권한 통제 서버를 통하여 뷰를 획득할 수 있다.

매번 새로운 권한위임 요청 시 새로운 역할 r'을 생성한다. 생성한 역할에 필요한 권한만 할당한다. 다른 도메인에 권한위임 접근권한을 제어하기 위해서 권한위임 뷰에 등록은 다음과 같은 방법으로 이루어진다. 〔정의4〕의 방법으로 결정할 수 있다.

[정의 4] 새롭게 생성한 역할 r_j'라하고, 이미 권한위임 뷰에 등록한 역할 $r_1', \ldots, r_i'$라면, 권한집합 $P(r_j')$ 및 $P(r_1', \ldots, r_i')$라 한다.

$$P(r_j') = \{\langle op_1, obj_1\rangle, \langle op_2, obj_2\rangle, \ldots, \langle op_m, obj_m\rangle\}$$

$$P(r_1', \ldots, r_i') = \cup_{i=1}^{n} r_i'$$

$$= \{\langle op_1, obj_1\rangle, \langle op_2, obj_2\rangle, \ldots, \langle op_n, obj_n\rangle\}$$

$$P(r_j') \cap P(r_1', \ldots, r_i')$$

$$= \{\langle op_1, obj_1\rangle, \langle op_2, obj_2\rangle, \ldots, \langle op_t, obj_t\rangle\}$$

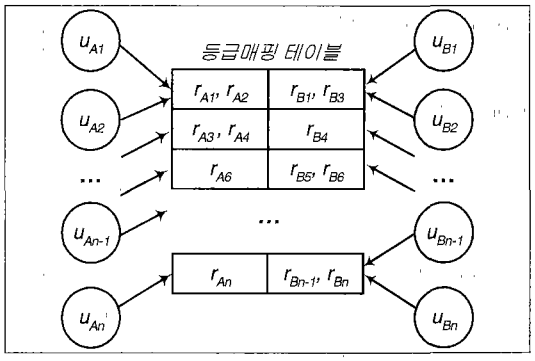
$$where \ o \le t \le n$$

만약 교집합 $P(r_j') \cap P(r_1', \ldots, r_i')$이 존재하면, 이 교집합에 해당하는 객체의 권한만 권한위임 뷰를 통하여 권한위임을 할당한다. 그러므로 중첩 객체가 존재하지 않으므로 문서 과잉 노출이 발생하지 않는다. 2장에서 제시한 둘째의 해결책이 된다. 문서 과잉 노출 방지는 할 수 있지만, 도메인 사이에 역할 기반 계층구조와 사용자를 결정하는 또 다른 문제가 발생한다. 이 문제를 4.3에서 다룬다.

4.3 도메인 대 도메인 역할 매핑

두 도메인사이에 서로 다른 조직체계의 다양한 역할 계층구조를 가지고 있다. 즉, 서로 다른 역할 구조를 가지고 있다. 다중 도메인 사이에 역할 계층구조의 모든 역할 매핑은 현실적으로 불가능하다. 이는 두 도메인사이에 필요한 계층의 등급구조를 조절함으로 간편 구조로 만들면 된다.

이는 매핑함수를 통하여 역할 매핑 등급 테이블을 생성하며 도메인사이에 접근통제 정책에 따라 결정을 할 수 있다. 결정된 매핑 등급 테이블은 권한위임 뷰에 등록한다. 〔그림 7〕는 등급 매핑 함수 테이블을 나타내는 그림이다. 매핑 등급함수는 정의5로 나타낸다.



〔그림 7〕 도메인간 역할 매핑

[정의 5] $T(Dom_1, Dom_2) = f_{Domain}(r_A, r_B)$, 위 식은 두 도메인간의 역할 매핑을 표현한다. f_{Domain}은 도메인 매핑함수로 도메인 이름과 역할을 입력값으로 한다. T는 매핑 테이블로 등급 및 도메인 이름 속성을 가진다. 다중 도메인 매핑을 표현하면 다음과 같다.

$$T(Dom_1, Dom_2, \ldots, Dom_n) = f_{Domain}(r_{A_{1,\ldots,n}}, r_{B_{1,\ldots,n}}, \ldots, r_{Z_{1,\ldots,n}})$$

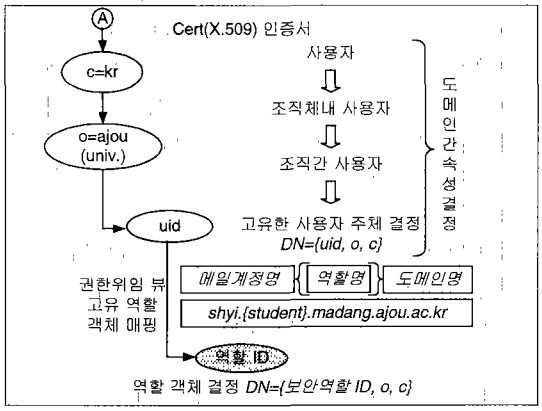
매핑 테이블 T에서 도메인 이름 속성(attribute) 및 등급의 튜플(tuple)을 가진다. 도메인 이름의 속성은 사용자의 권한의 등급을 결정하기 위한 출구(outgoing)를 의미하고 다른 하나는 입구(incoming)를 의미한다. 여러 개의 도메인에도 상관없이 역할 계층의 도메인간의 정책권한 매핑이 성립한다.

문제는 도메인간의 사용자의 ID를 결정하여 어떻게 사용자를 회사A의 역할로 위임을 할 것인가를 결정하는 문제이다. 도메인사이에 역할 및 사용자의 유일성은 이미 보유하고 있는 X.509 인증서 필드의 DN (Distinguished Name) 값을 가지고 결정한다.

〔그림 8〕은 X.509 필드의 DN 필드 값으로 도메인을 구분한다. 도메인의 구분이 결정되면 도메인 내의 사용자를 구분하는 것이 또한 문제가 된다. 그 필드내의 메일주소를 가지고 ID를 결정한다.

다음과 같은 방법으로 유일한 ID를 결정할 수 있다. 예를 들어 메일ID shyi@madang.ajou.ac.kr을 shyi.{student}.madang.ajou.ac.kr로 변환하여 ID로 사용한다. 이와 같은 ID는 인터넷망에서 분산 주소공간과 개개의 조직을 통제할 수 있는 어떤 형태의 도메인도 구분할 수 있으며 확장성을 가질 수 있다. 이는 확장성을 제공하여 분산 환경에서 인증서의 효율적인 탐색으로 관리 가능하다.

권한위임 취소(revocation) 문제는 권한위임 토큰 발생시 할당한 시간이 만료되면 자동적으로 권한위임 토큰이 만료된다.



〔그림 8〕 도메인간의 ID 결정

또한 개별 역할 관리자는 간편하게 권한위임 역할에 할당한 사용자를 그 사용자의 권한위임 취소 요청에 의해서 할당한 역할의 구성원으로 제거하면 자동으로 권한위임이 취소된다. 권한위임 토큰의 안전한 생성 및 전달은 4.3절에 다룬다.

4.4 X.509 인증서 기반 권한위임 토큰

〔표 1〕은 도메인간에 권한위임을 하기 위해서 권한위임 토큰을 생성하기 위한 알고리즘이다.

〔표 1〕 토큰생성 알고리즘

Input : 1) *Delegation Request:*($u_A \Rightarrow u_B$)
Output: 1) *Delegation Token:* ($DT_{A,B}$)
2) *Delegation View: Viewlist*
3) *return REJECT*

```
if Domain_single then
  if ∃Role_hierarchy(down) then
    if (r_A ≥ r_B) and (r_A, r_B) ∉ M_R then
      if validation time satisfy then
        partial := select pv_A
        create r_A' := partial
        return delegation_view := Viewlist
        return create DT_A,B
      else return REJECT
    else return REJECT
  else Domain_multi then
    goto〔그림8〕Ⓐ
endif
```

다중 도메인에서 도메인의 결정은 〔그림 8〕의 방

법으로 결정된다. 도메인이 결정이 되면 다음 알고리즘이 반복 수행한다.

개시자 u_A는 권한위임의 요청으로 권한위임 토큰을 안전하게 중개자 및 목적지에 전달하고 중개자의 위조나 수정을 방지할 수 있는 방안이 있어야 한다.

이를 위해 X.509 인증서에서 생성된 공개키 방식을 사용한다. 접근권한은 권한위임 비밀키 및 사용자 신분 두 가지의 조합에 기반을 두고 있다. 프로토콜은 다음과 같이 이루어진다.

권한위임 경로는 $u_A \Rightarrow u_B \Rightarrow u_A$로 이루어지는 폐쇄루핑(looping) 관계로 이어지는 문서공유 문제로 2가지 다른 단계로 분할이 된다.

1단계 : 개시자 u_A는 u_B로 권한위임을 시작하며 $u_A \Rightarrow u_B$로 표현한다.

2단계 : 중개자 권한위임 $u_B \Rightarrow u_A$로 개시자 자신에게 돌아오는 권한위임을 한다. 최종 목적지에서 권한위임에 의한 접근권한의 실행으로 각 단계는 분리되어 있고, 이들은 X.509 인증서의 공개키 방식의 전자서명 방법을 사용하여 개시자에 의해서 전자서명이 된다.

4.4.1 공개키 기반 토큰

권한위임에 의한 접근권한은 비밀 권한위임 키를 기반으로 한다. 정확한 순서는 다음과 같이 진행이 된다. 여기서 표기의 편의상 개시자, 중재자, 목적지의 각각의 사용자 및 프로토콜에 대해서 다음과 같이 기술한다.

- $A=\{u_{A1}, \ldots, u_{An}\}$, $B=\{u_{B1}, \ldots, u_{Bn}\}$
- $E_{Kpr_A}[m]$: 사용자A가 전자서명을 제공하기 위해 메시지 (m)을 A의 개인키 Kpr_A(private key)로 전자서명
- $E_{Kpu_B}\{m\}$: 사용자A가 메시지 비밀성을 위해서 메시지 (m)을 B의 공개키 Kpu_B(public key)로 암호화
- $k_{A,B}$: 사용자 A가 생성한 랜덤넘버(random number)에 해당하는 세션키
- $Pr_A(r_B)$: $k_{A,B}(A, r_A', r_B, B, D_t, data)$ 사용자A가 요청한 역할의 접근 권한위임 제약속성으로 세션키 $k_{A,B}$로 봉인된 값
- D_t : 권한위임 유효기간
- $cert_A$: 사용자A의 공개키(Kpu_A)을 포함한 X.509 인증서
- $h(m)$: 메시지 (m)의 해쉬 함수 값

1. $A \Rightarrow B : cert_A, E_{Kpu_B}\{DT_{A,B}\}, Sign_A$

개시자A는 X.509 인증서 ($cert_A$)을 취득한 상태에서 $cert_A$로부터 사용자A는 권한위임 키쌍 (Kpr_A, Kpu_A)을 획득한다. 또한 권한위임 토큰 $DT_{A,B} := \{A, r_A, k_A, B\ Pr_A(r_B), TS_{A,B}\}$을 생성하고, 권한위임 개인키 Kpr_A는 권한위임 토큰 $DT_{A,B}$의 전자서명용으로 사용하여 다음과 같이 전자서명을 한다. $Sign_A = E_{Kpr_A}[h(DT_{A,B})]$는 부인방지를 위함이다. A는 권한위임 토큰 ($DT_{A,B}$)을 B의 공개키로 암호화하여 $E_{Kpu_B}\{DT_{A,B}\}$로 전송한다. 신뢰성은 인증방법과 더불어 개인키 Kpr_A을 가지고 있다는 것을 보여줌으로 증명된다. 사용자A,B는 인증시 또는 권한위임 토큰을 전달하는 과정에서 사용자는 신뢰성 있는 대칭 암호화 세션키 $k_{A,B} := k_A \equiv k_B$를 획득하게 된다.

2. $B \Rightarrow A : cert_B, E_{Kpu_A}\{DT_{B,A}\}, Sign_B$

B는 자신의 인증서로부터 권한위임 키쌍 (Kpr_B, Kpu_B)을 획득한다. B는 권한위임 토큰 $DT_{B,A} := \{B, r_B, k_{B,A}, Pr_B(r_A), TS_{B,A}\}$을 생성하고, B의 권한위임 개인키 Kpr_B로 $Sign_B = E_{Kpr_B}[h(DT_{B,A}, Sign_A)]$ 전자서명 한다. B와 A는 신뢰성 있는 키교환 프로토콜을 사용하여 상호인증을 한다. B는 A의 공개키 Kpu_A로 암호화하여 토큰 ($DT_{B,A}$)을 전송한다.

권한위임 키쌍 (Kpr_A, Kpu_A), (Kpr_B, Kpu_B)의 연산은 PKI 기반 계층구조 키 생성의 아이디어에 기반 한다. 이 키쌍은 사용자의 신분을 대신하며 각각의 권한위임 토큰에 전자서명 하는데 사용된다.

권한위임 토큰 $DT_{A,B}$ 및 $DT_{B,A}$는 간단한 연결고리가 성립한다. 개인키 Kpr_B는 두 토큰을 함께 묶어서, 토큰의 전자서명 매개변수 중에 하나가 되고, 상호 추적할 수 있는 연결고리 키가 된다. 공격자는 토큰 $DT_{A,B}$ 및 $DT_{B,A}$을 위조 및 수정이 불가능하고, 또 다른 권한위임 경로를 구성할 수 없다. 이 정보보호의 안전성은 PKI의 X.509 인증서의 전자서명 방법 및 인증키 교환 프로토콜에 의존한다. 또한 권한위임 연결고리 경로는 추적(traceable)이 가능하다.

V. 결 론

두 회사 이상 상호관계가 있는 다중 도메인간 문서공유 문제를 해결하기 위한 방안으로 역할기반 접근통제의 권한위임 방안을 적용하였다. 참고문헌[8] DCE OSF 제시한 두 회사간의 문서공유 관련한 문제점을 회사의의 조직체계를 잘 반영할 수 있는 역할기반 접근통제를 통하여 2장에서 제시한 3가지의 문제를 해결하였다.

관리적인 과부하는 도메인 관리자와 개별 역할 관리자를 구분하여 관리함으로 도메인은 회사간의 정책에 관련한 관리 및 역할기반의 역할만 관리함으로 관리적 부담을 줄여준다. 또한 관리적인 부담을 줄이기 위한 방안으로 권한위임 뷰를 두어 상호역할 및 도메인간의 접근통제는 이 뷰를 통하여 통제한다. 권한위임 뷰는 융통성을 제공하고 개별 역할 관리자 단위로 정보의 변경이 가능함으로 동적인 접근통제 관리가 가능하다.

새롭게 생성한 역할에 개시자의 의도로 중개자에게 새로운 역할의 구성원으로 되기 위한 권한위임을 함으로 실제적인 권한위임이 허용된다. 실제로 접근권한위임 뷰는 주지만 ACL은 주지 않는 결과로 권한위임 취소 및 할당을 개시자가 주도 할 수 있다.

개시자가 다중 권한위임 발생을 허용하지만 권한위임 뷰를 통하여 객체에 대한 다중 권한위임 할당이 되지 못하게 방지함으로써 의도하지 않은 과잉 문서노출을 방지한다.

문서공유를 위한 권한위임이 실제 발생시 권한위임 뷰를 통하여 통제가 된다. 이 환경은 권한위임 취소가 개시자의 의도로 취소가 용이하기 때문에 중개자는 고의적인 행위 발생시 즉시 취소가 된다. 권한위임 토큰의 생성은 PKI 구조하에 권한위임 토큰이 생성되어 도메인과 사용자간에 발생하는 탐색구조는 디렉토리 구조를 그대로 반영한다.

본 논문에 제시한 문서공유 문제를 두 회사사이뿐만 아니라 다중 권한위임에서 문서과잉 노출 문제 해결방법이 다양한 인터넷 응용환경에 적용 가능하다. 향후 역할기반 접근통제에서 권한위임을 통한 분산환경에 적합한 PMI(Privilege Management Infrastructures)기반에 적용할 필요가 있다.

참 고 문 헌

[1] B. Clifford Neuman, "Proxy-Based Authorization and Accounting for Distributed System," In Proceedings of the 13th International Conference on Distributed Computing Systems, pp. 283~291, 1993. 5.

[2] D. Richard Kuhn, "Mutual Exclusion of Role as Means of Implementing Separation of Duty in Role-Based Access Control Systems," National Institute of Standards and Technology, Jun. 1996.

[3] David F. Ferraiolo, Janet A. Cugini, D. Richard Kuhu, "Role-Based Access Control(RBAC): Features and Motivations," 11th Annual Computer Security Application Conference, pp. 554-563, Dec. 1995.

[4] Emil Constantin Lupu, "A Role-Based Framework for Distributed Systems Management," University of London, Phd thesis, Jul. 1998.

[5] Ezedin Barka and Ravi Sandhu, "A Role-Based Delegation Model and Some Extension," 23rd National Information Systems Security Conference, pp. 101~114. Oct. 2000.

[6] Ezedin Barka and Ravi Sandhu, "Framework for Role-Based Delegation Models," 16th Annual Computer Security Applications Conference, Dec. 2000.

[7] Fang Chen & Ravi Sandhu, "Constraints for Role Based Access Control," In Proceedings 1st ACM Workshop RBAC, Sep. 1995.

[8] Jonathan T. Trostle, B. clifford Neuman, "A Flexible Distributed Authorization Protocol," Internet Society 1996 Symposium on Network and Distributed System Secu rity, pp. 43~52, May 1996.

[9] Karen R. Sollins, "Cascaded Authentication," In Proceedings of th 1988 IEEE Symposium of Research in Security and Privacy, pp. 156~163. Apr. 1988.

[10] M. Gasser and E. McDermott, "An Architecture for Practical Delegation in a Distributed System," IEEE Symposium on Security and Privacy, pp. 20~30, 1990.

[11] Matunda Nyanchama, "Commercial Integrity, Roles and Object Orientation," University of Western Ontario, Phd thesis, Sep. 1994.

[12] Ravi S. Sandhu, Edward J. Coyne, iiHal L. Feinstein, and Charles E. Youman, "Role-Based Access Control Models," IEEE Computer, Vol. 29, No. 2, pp. 38~47, Feb. 1996.

[13] Ravi Sandhu and Venkata Bhamidipati, "The ARBAC97 Model for Role-Based Administration of Roles : Preliminary Description and Outline," Proceedings of Second ACM Workshop on Role-Based Access Control, Nov. 1997.

[14] Serban I. Gavrila and John F. Barkley, "Formal Specification for Role-Based Access Control User/Role and Role/Role Relationship Management," NIST, Sep. 1999.

[15] W. A. Jansen, "Inheritance Properties of Role Hierarchies," 21th National Information Systems Security Conference, Oct. 1998.

[16] 이상하, 조인준, 천은홍, 김동규, "역할기반 접근통제에서 역할 계층에 따른 접근권한 상속의 표현," 한국정보처리학회 논문지 제7권 7호, pp. 2125~2134, 2000. 7.

〈著者紹介〉

이 상 하 (Sang-Ha Yi) 정회원
1987년 2월 : 울산대학교 전자계산학과 졸업(학사)
1991년 2월 : 아주대학교 대학원 컴퓨터공학과 졸업(석사)
1999년 2월 : 아주대학교 대학원 컴퓨터공학과(박사수료)
1991년~1992년 : (주)큐닉스컴퓨터
1993년~1999년 : (주)케이엔아시스템
2000년 3월~현재 : 동서울대 정보통신공학과 전임강사
〈관심분야〉 정보보호, 네트워크 관리 및 보안, 분산 처리 시스템 보안

채 송 화 (Song-Hwa Chae) 정회원
1997년 8월 : 아주대학교 컴퓨터공학과 졸업(학사)
1999년 8월 : 아주대학교 대학원 컴퓨터공학과 졸업(석사)
1999년~1999년 9월 : 한국통신프리텔
1999년~2001년 : 한국정보보호센터
2001년~현재 : 아주대학교 정보통신전문대학원 박사과정
〈관심분야〉 PKI, 무선통신 보안, 전자상거래

조 인 준 (In-June Jo) 종신회원
1982년 2월 : 전남대학교 계산통계학과 졸업(학사)
1985년 2월 : 아주대학교 전자계산학과 졸업(석사)
1999년 8월 : 아주대학교 대학원 컴퓨터공학과(박사)
1983년 9월~1994년 2월: 한국전자통신연구원 선임연구원
1991년 12월 : 전산조직응용기술사
1992년~현재 : 대한기술사회원
1994년~현재 : 배재대학교 컴퓨터공학과 교수
1995년~현재 : 한국정보보호학회 정보보호응용 연구회 전문위원
1997년~현재 : 한국정보처리학회 시스템 통합 연구회 학술위원
〈관심분야〉 정보보호, 컴퓨터네트워크, 전산조직응용

김 동 규 (Dong-Kyoo Kim) 정회원
1973년 2월 : 서울대학교 공과대학 응용수학과 졸업(학사)
1979년 2월 : 서울대학교 자연과학대학원 전자계산학과 졸업(석사)
1984년 : 미국 Kansas State University 전자계산학과(박사)
1986년~IEEE 802.4, 802.6, 802.10 Working Group Member
1979년~현재 : 아주대학교 정보 및 컴퓨터공학부 교수
Asiacrypt '96 조직위원장, 건설교통부 항공 교통관제소 신공항 교통관제 시스템 평가위원회 위원, 한국과학기술연구소 연구원, 한국통신학회 상임이사, 한국정보보호학회 부회장 역임
〈관심분야〉 컴퓨터 통신, 정보보호, 프로토콜 엔지니어링