T-RBAC에 기반한 사용자 수준의 다단계 위임기법

나민선[0] 박 석
서강대학교 컴퓨터학과 데이타베이스 연구실
(margaret, spark)@dblab.sogang.ac.kr

Multi-Step Delegation Based On Task-Role-Based Access Control Model

Min-Sun Na Seog Park
Database Research Lab., Department of Computer Science, Sogang University

요 약

RBAC은 역할 계층구조에서 권한의 계승과 의무분리와 같은 제약조건을 다룸으로써 접근권한의 관리를 수월하게 하고 기업환경을 잘 반영할 수 있는 장점이 있다. 하지만 RBAC은 현실세계의 기업환경에서 빈번히 이루어지는 권한의 위임을 제대로 구현하지 못한다는 문제점을 가지고 있다. 본 논문에서는 자신의 고유역할 뿐만 아니라 상위 역할로부터 위임 받은 위임역할에 대해서도 새로운 위임 역할을 생성함으로써 역할계층 구조상의 다른 역할의 사용자에게 다른 과업을 할당해 줄 수 있도록 하여, 최소 권한의 원칙을 만족하는 다단계 위임을 구현하였다. 위임 시에 생길 수 있는 보안 문제를 해결하기 위해서 역할단위가 아닌 과업단위의 위임으로 제한하고, 과업단위의 의무분리를 적용하였으며, 위임할 수 있는 과업을 규정하고 최하위 역할을 지정하였다. 기존 다단계 모델에서 제안된 기법과의 비교를 통해서 본 논문에서 제안된 기법이 실제 기업에서 이루어지는 다단계 위임을 타당하게 구현할 수 있음을 보인다. 또한 T-RBAC을 기반으로 ARBAC97을 적용해서 제안된 기법을 모델링하고 Prototype을 구현하였다.

1. 서 론

RBAC(Role-Based Access Control)은 MAC과 DAC의 대안으로 상업적인 영역에서 주목 받고 있다. RBAC의 중심 개념은 권한이 역할에 부여되고, 사용자는 조직 내에서의 적절한 책임과 자격에 맞는 역할에 할당된다는 것이다. 따라서 사용자는 역할의 멤버가 됨으로써 역할의 권한을 할당 받게 된다. 이렇게 함으로써 사용자가 역할의 멤버에 할당되거나 회수될 때 권한 관리를 매우 단순화시킬 수 있다는 장점이 있다.

본 논문은 기업환경에서 사용자 수준의 위임 시에 발생할 수 있는 권한의 오용과 남용 그리고 정보의 유출 등의 보안 문제 등을 해결하기 위한 다단계 위임기법을 제시한다.

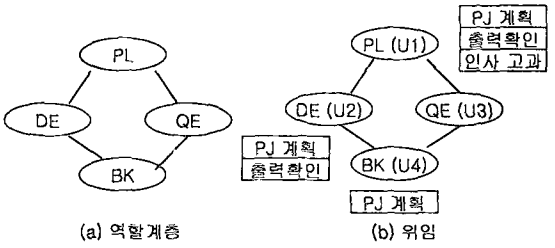
본 논문은 위임의 개념을 역할 담당자의 부재로 인한 백업의 개념과 역할 분배의 개념의 위임으로 제한을 둔다. 즉, 위임이란 역할 담당자의 부재로 그 역할을 수행할 수 없는 경우에 다른 사람이 수행할 수 있도록 자신의 권한의 일부를 임시적으로 부여하는 것 뿐만 아니라, 자신의 역할의 일부를 다른 여러 사람에게 분배함으로써 일의 병행 수행성을 높여 업무의 효율성을 증대 시킬 수 있는 업무의 위임으로 정의한다.

2. 본 론

2.1 연구동기

기업내의 조직에서 일어나는 위임의 상황에 대해 알아보고, 위임을 하는데 필요한 요구사항을 알아보고 기존 모델에서 만족시키지 못한 문제점에 대해서 해결하겠다.

현실세계(real world)에서 다단계 위임이 일어나는 상황은 다음 [그림 1]과 같다.



[그림 1] 현실세계의 다단계 위임의 예

RBAC에서 이러한 현실세계의 다단계 위임의 상황을 구현하는 것은 쉽지 않다. RBAC에서는 사용자가 관리자에 의해서 할당 받은 역할을 통해서 객체에 대한 권한을 부여받기 때문에 사용자는 자신의 역할에 속한 권한 전체를 위임할 수 밖에 없다. 따라서 정보유출 등의 보안 문제가 발생하고 최소 권한의 원칙에 위배가 된다.

그러므로 현실세계에서 일어나는 다단계 위임을 반영할 수 있는 기법이 필요하다. 이러한 다단계 위임을 시스템에서 구현하기 위한 요구사항은 다음과 같다.

1. 사용자 수준에서 위임이 일어날 수 있어야 한다.
2. 역할 전체 혹은 부분을 위임/회수할 수 있어야 한다.
3. 위임역할에 사용자를 할당/회수할 수 있어야 한다.
4. 자신의 역할 뿐만 아니라 상위 역할로부터 위임 받은 역할의 과업을 다른 사용자에게 위임할 수 있어야 한다.
5. 다단계 위임을 제한하여야 한다.
6. 상위 역할의 멤버는 위임역할을 즉시 회수할 수 있는 권한을 가진다.

7. 위임으로 인해 발생할 수 있는 최소 권한의 원칙의 위배, 의무분리 정책의 파괴, 정보 유출 등의 보안 문제를 해결해야 한다. 권한을 위임하려는 사람은 위임 받는 사람을 그들의 업무적 책임과 권한에 따라 결정해야 한다.

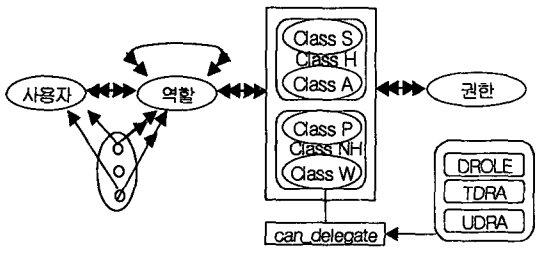
2.2 제안한 다단계 위임 기법

사용자에게 역할단위로 권한을 부여하는 RBAC모델이 아닌 일의 기본 단위인 과업(Task) 단위로 권한을 부여하는 T-RBAC모델에서 다단계 위임을 제시한다. 과업단위의 위임을 할 수 있게 되면 최소 권한 원칙의 파괴, 정보 유출(비밀성)의 보안 문제를 해결할 수 있고, 다단계 위임 시 각 역할마다 과업을 다르게 위임할 수 있다. 또한 과업 단위의 의무분리를 적용함으로써 RBAC에서의 위임보다 일의 분배를 효율적으로 할 수 있다.

사용자 수준에서 다단계 위임을 구현하기 위해서 보안 관리자에게만 주어졌던 역할 생성 및 파괴(ROLE), 역할에 과업을 지정 및 회수(TRA), 사용자를 역할에 지정 및 회수(URA) 하는 권한을 일반 사용자에게도 주어서 위임 역할 생성 및 파괴(DROLE), 위임역할에 과업을 지정 및 회수(TDRA), 사용자를 위임역할에 지정 및 회수(UDRA) 할 수 있도록 하였다. 이러한 권한을 관리 하기 위해서 보안 관리자 권한의 관리에 적용했던 ARBAC97을 이용해서 위임역할을 관리한다. 그러므로 T-RBAC에 ARBAC97을 적용함으로써 의무분리의 원칙, 최소 권한의 원칙, 정보유출(비밀성)을 보다 쉽고 효율적으로 관리할 수 있다[5].

DROLE, TDRA, UDRA 의 권한을 T-RBAC 모델에 적용하기 위해서, T-RBAC 모델의 클래스를 재분류하였다. 위임은 특정한 사람에게 자신의 역할의 일부를 위임하는 것이므로, 위임의 특성상 역할 계층상에서의 상속이 되어서는 안되기 때문에, 역할 계층상에서 상속이 되는 과업 class H 와 상속이 되어서는 안 되는 과업 class NH 의 2 개의 클래스로 재분류하였다.

일반 사용자가 생성한 위임역할이 역할 계층상에서 상속이 되지 않도록 하기 위해서, class NH 에 can_delegate 과업을 지정한다. T-RBAC 모델에 can_delegate 과업을 지정한 그림은 [그림 2]와 같다.



[그림 2] T-RBAC 모델 클래스 재분류 및 can_delegate 과업 지정

2.2.1 can_delegate 과업

- 위임역할 생성 및 파괴(DROLE)

사용자가 위임을 하려면 새로운 위임역할을 생성하여야 한다. 다단계 위임을 하기위해서 생성할 수 있는 위임 역할은 자신의 고유역할과 상위 역할로부터 받은 역할 두 가지이다.

위임 역할의 관리자는 위임역할을 생성한 사용자와 그 상위 역할의 사용자이며, 위임역할을 파괴할 수 있는 권한을 가진다. 이는 위임역할이 더 이상 필요하지 않을 때나 악의적으로 사용되고 있을 경우 즉각적으로 회수하기 위함이다.

- 위임역할에 과업을 지정 및 회수(TDRA)

위임역할을 생성한 후에 사용자는 과업을 지정 및 회수할 수 있다. 또한 위임역할의 관리자 즉, 위임역할을 생성한 사용자의 상위 역할 사용자에 의해서도 위임역할에 과업을 지정 및 회수할 수 있다.

위임역할의 부모역할에 지정된 과업만이 될 수 있으며, CDT를 참조하여 하위로 위임할 수 없는 과업은 지정할 수 없다. 위임 역할에 역할전체의 권한을 지정하는 것이 아니라 일의 기본 단위인 과업을 지정한다. 이로써 사용자에게 필요한 최소 권한만을 줄 수 있다. 또한 과업 단위의 의무분리 정책을 적용해서 업무의 효율을 높이는 효과를 얻을 수 있다.

- 위임역할에 사용자를 지정 및 회수(UDRA)

위임역할을 생성한 후에 사용자를 위임역할에 지정 및 회수할 수 있다. 또한 위임역할의 관리자 즉, 위임역할을 생성한 사용자의 상위 역할 사용자에 의해서도 위임역할에 사용자를 지정 및 회수할 수 있다. 위임역할에 지정될 수 있는 사용자는 위임역할의 하위역할에 지정된 사용자뿐이다. 이때 위임역할에 지정된 과업과 위임역할에 지정하려는 사용자가 속한 역할의 과업과 의무분리관계에 있다면 그 사용자는 위임역할에 지정될 수 없다.

2.2.2 다단계 위임의 제한

본 논문에서는 과업단위로 다단계 위임을 할 수 있는 최하위 역할을 선택하여 다단계 위임을 제한하였다. 최하위 역할의 선정은 그 역할에 속한 사용자의 업무와 책임 등을 고려하여 선정된다. 과업단위로 위임을 할 수 있는 적절한 최하위 역할을 선정함으로써 최소한의 권한과 정보 유출의 보안 문제를 좀 더 완벽하게 지킬 수 있고, 작업의 분배를 효율적으로 할 수 있는 장점이 있다.

다단계 위임을 제한하는 정책은 보안 관리자에 의해서 CDT테이블로 작성 된다. 이 예는 [표 1]과 같다.

[표 1] CDT(Can Delegate Table) 테이블

팀장(PL)	PJ 계획	BK연구원(BK)
팀장(PL)	출결확인	DB(DE), 품질 연구원(QE)
팀장(PL)	인사고과	팀장(PL)

3. 분석

3.1 기존 모델과의 비교분석 및 평가

기업환경에서 사용자 수준의 다단계 위임을 구현한 기존의 모델인 Shim 모델, RDM2000 모델과 비교 분석한다.

Shim 모델은 RBAC96모델을 기반으로, 역할을 작업의 부분집합으로 세분화하여 역할의 전체·부분 위임을 구현하였다. 보안 관리자와 상위 역할의 멤버에 의해서 위임을 감독하며, 위임으로 인해서 발생할 수 있는 의무분리 정책의 파괴, 정보 유출 등의 보안의 문제를 사용자와 역할 속성에 제한을 둠으로써 해결하였다[2].

이러한 Shim 모델은 위임역할을 생성할 시에 두개의 역할(DR, DE)을 자동으로 생성해야 하는 번거로움과, 상위

역할로부터 위임 받은 역할에 대한 위임역할을 생성할 수 없는 문제점이 있다. 또, 역할속성의 개념이 모호하고, 같은 도메인의 사용자라도 위임역할을 지정 받을 수 있는 사용자와 위임역할을 할당해서는 안 되는 사용자의 구분이 불명확하다. 권한의 회수 시에 위임역할의 부분 권한을 회수하는 방법이 제시되지 않았다.

RDM2000 Model 은 RBAC96 모델을 기반으로, 사용자-역할 할당을 하는데 있어서 끊임없는 보안 관리자의 개입으로 사용자-역할 할당의 관리의 노력이 증가하게 되는 문제를 해결하고자, ARBAC97 을 이용하여서 보안 관리자의 개입 없이 사용자 수준에서 위임을 할 수 있도록 하였다. 또 RBDM0 Model 을 기반으로 역할계층 구조상의 위임과 다단계 위임을 구현하였다. 그리고 위임의 rule-based 언어를 이용하여서 can_delegate 를 정의하였다 [4].

이러한 RDM2000 모델은 위임역할을 생성하지 않고 기존의 역할계층상의 위임을 하기 때문에 역할의 부분 위임을 제시하지 못했고, 역할 전체의 권한을 위임함으로써 정보 유출, 권한의 남용과 같은 보안의 문제가 발생할 소지가 크다. 권한의 회수도 역할 전체의 권한만 할 수 있다.

이들과 비교하여 제안한 모델은 다음과 같은 장점이 있다.

- 다단계 위임에 필요한 위임 과업을 부여함으로써 사용자 수준에서 쉽게 다단계 위임이 구현된다.
- 상위 역할로부터 위임 받은 역할에 대해 새로운 위임역할을 생성할 수 있게 함으로써 최소 권한의 원칙을 지키면서 업무의 분배를 효율적으로 할 수 있다. 역할단위의 위임보다 일의 능률을 좀 더 올릴 수 있다.
- 무결성 보장을 위한 의무분리 정책을 과업단위로 수행함으로써 역할의 작업 병행 수행율을 높이고, 역할단위로 적용할 때보다 좀 더 유연성을 제공한다.
- 위임역할의 상위 역할 사용자에게도 위임역할을 회수할 수 있는 권한을 줌으로써 위임을 감독하도록 하여 비밀성을 보장한다.

하지만 제안한 다단계 위임 기법이 보안요구사항을 만족하는 분산환경에서의 사용자 수준의 위임을 구현하기 위해 추가되거나 수정되어야 할 사항에 대한 연구가 필요하다.

3.2 정량적 분석

본 논문에서 제시한 다단계 위임기법으로 생성된 위임역할의 수와 기존의 다른 위임기법으로 생성된 위임역할의 수를 비교함으로써 위임역할 관리의 오버헤드가 많지 않음을 보인다.

[그림 1(a)]와 같은 역할계층을 가진 부서에서, 역할 PL 의 과업 PJ계획, 출결확인을 하위역할 DE, QE 의 사용자 U2, U3에게 지정하고, 또 PJ 계획만을 역할 BK 의 사용자 U4에게 지정하고자 하는 시나리오의 경우에 각 모델에서 생성할 수 있는 위임역할의 수는 다음 [표 2]와 같다. [표 2]에서 볼 수 있듯이 제안한 기법에서 생성한 위임 역할의 수는 기존 모델에서 생성한 위임역할의 수와 같다. 그러므로 위임역할을 관리하는데 오버헤드는 없을 것이다.

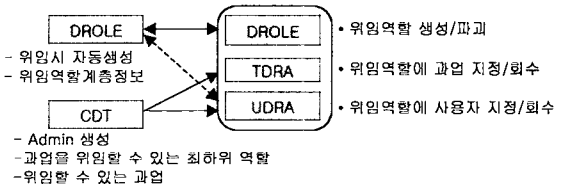
또한 앞의 시나리오에서 Shim 모델은 PL'' 를 역할 PL 의 사용자 U1이 생성해야 하는 반면에, 본 논문에서 제시한 기법에서는 DE역할이 PL'' 를 생성할 수 있어서 업무분배의 효율성이 높은 장점이 있다. RDM2000 모델에서는 위임이 불가능 하다.

[표 2] 위임역할 수 비교

	위임하는 역할	위임하는 사용자	위임받는 역할	위임받는 사용자	생성한 위임역할	총 생성한 위임역할 수
Shim 모델	PL	U1	DE	U2	PL'	2
	PL	U1	QE	U3	PL'	
	PL	U1	BK	U4	PL''	
RDM 2000						위임 불가
제안한 기법	PL	U1	DE	U2	PL'	2
	PL	U1	QE	U3	PL'	
	DE	U2	BK	U4	PL''	

3.3 Prototype 구현

T-RBAC 미들 웨어 시스템에 다단계 위임을 하기 위한 Delegation Tool Prototype을 구현하였다. 이는 다음[그림 3]과 같다. 사용자는 접근제어 대상이 되는 BK21홈페이지를 웹 브라우저를 통해 접근하게 된다. 각 정보는 테이블로 저장되며, T-RBAC 엔진은 사용자가 요청한 웹 페이지가 접근 가능한지의 여부를 테이블에 저장된 정보를 보고 결정하여 접근이 가능하면 해당 웹 페이지를 보여주며 접근이 불가능하면 "Access Denied" 에러 메시지를 보여준다.



[그림 3] Delegation Tool의 Prototype

4. 결론

T-RBAC모델에 사용자 수준의 다단계 위임 기법을 제시하고, 위임 시 발생가능 한 보안문제를 해결하였다. 또한 제안한 기법의 Prototype을 웹 환경에서 구현하였다. 향후 분산환경에서 다단계 위임을 구현하기 위한 추가 연구가 필요하다.

5. 참고문헌

[1] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, Charles E. Youman, "Role-Based Access Control Models", *IEEE Computer*, 29(2): pages 38-47, February 1996.

[2] 심재훈, 박석, "역할기반 접근제어에 기초한 사용자 수준의 위임기법", 통신정보보호학회 논문지, Vol.10, No.3, pages 49-62, 2000.9.

[3] 박석,오세종, "기업환경을 위한 과업-역할기반 접근제어 모델", 한국통신정보보호학회 논문지 11권1호, 2001.2.

[4] Longhua Zhang, Gail-Joon Ahn, Bei-Tseng Chu, "A Rule-Based Framework for Role-Based Delegation", SACMT'01, pages 153-162, May 3-4, 2001.

[5] 조병철, 박석, "역할기반 접근제어에서 ARBAC97의 역할관리 기법을 적용한 위임", 한국통신정보보호학회 종합학술발표회 논문집, 제10권1호, pages 840-849, 2001. 2.