한국항행학회 논문지 제12권 제5호 2008년 10월

유비쿼터스 환경을 위한 상황 인식 접근제어

Context awareness Access Control for Ubiquitous Environment

신동욱*, 황유동**, 박동규**

Dong-Wook Shin*, Yu-Dong Hwang* and Dong-Gue Park*

요 약

본 논문에서는 유비쿼터스 환경을 위해 컨텍스트 정보를 이용한 역할기반 접근제어 모델을 제안한다. 컨텍스트 정보를 이용한 접근제어의 기본적인 개념은 어떠한 정보나 객체에 접근할 수 있는 권한을 역할에 배정하여, 사용자를 그 역할에 할당함으로써 정보나 객체에 접근할 수 있는 권한을 획득할 수 있도록 하는 것이다. 그러므로 사용자는 자신에게 할당된 역할과 그 역할이 지니고 있는 권한에 맞는 정보 또는 객체에 접근하게 됨으로써 안전하게 정보나 객체의 이용을 보장할 수 있도록 한다. 본 논문에서 제안하는 모델은 GEO-RBAC 모델에서 부족한 제약 사항을 고찰하여 새로운 제약 조건을 제시하고 다양한 경우에서의 역할 충돌을 고려하였다. 또한 GEO-RBAC 모델에 적합한 위임 모델을 제안하기 위하여, 기존 위임 모델의 장/단점을 분석하고 이를 개선하여 GEO-RBAC에 적합한 위임 모델을 제안한다.

Abstract

This paper proposes role base access control model that use context information for ubiquitous environment. Concept of access control that use context information assigns permission that can approach in some information or object in part. And do so that can assigned user in part to it and acquire permission. So it can approach in information or object. Therefore, user approaches in information or object in assigned role, and the role that is allocated to own is having. So, do so that can secure information or utilization of object safety. Proposal model investigated lacking restriction item in GEO-RBAC model. So, it considered that present new restriction condition and role conflict in various case. Also, to GEO-RBAC model proposed suitable model, analyzed old model's advantage, shortcoming. And it presented proposal model to GEO-RBAC because improving this.

Keywords : Access control, RBAC, GTRBAC, Role Graph Model, Delegation

I. 서 론

인터넷과 웹이 활성화됨으로써 사용자는 문서, 디렉토리, 데이터베이스, 웹 페이지 등과 같은 자원들을 액세스하는 것이 훨씬 더 쉬워졌다. 그러나 이로 인하여 네트워크의 인증, 자원들을 액세스하기 위한 권한 허가, 데이터의 정책과 보안 그리고 보안 시스템의 무결성과 같은 몇 가지의 중대한 보안 문제들이 생기게 되었다.

정보 보안은 시스템들이 인증(authentication), 접근

* 이지케이텍(ezCaretech co. Ltd.)
** 순천향대학교 정보보호학과(Dept. of Information Security Eng, SoonChunHyang University)
** 순천향대학교 정보통신공학과(Dept. of Information and Communication Eng., SoonChunHyang University)
· 제1저자 (First Author) : 신동욱
· 투고일자 : 2008년 9월 23일
· 심사(수정)일자 : 2008년 9월 24일 (수정일자 : 2008년 10월 22일)
· 게재일자 : 2008년 10월 30일

제어(access control), 무결성(integrity), 신뢰성(confidentiality), 그리고 부인(non-repudiation)과 같은 5가지의 중요한 서비스를 제공하도록 요구한다. 이 중 접근제어는 컴퓨터내의 자원, 통신 자원 및 정보 자원 등에 대하여 사용, 변경, 조회 등의 작업을 할 수 있는 능력을 가능하게 하거나 제한할 수 있는 수단으로 식별 및 인증된 사용자만이 허가된 범위 내에서 시스템 내부의 정보에 대한 접근을 허용하는 기술적 방법이다. 접근제어를 위해 개발된 보안 정책으로는 임의 접근 통제(DAC : Discretionary Access Control)[1], 강제적 접근 통제(MAC : Mandatory Access Control), 역할 기반 접근 통제(RBAC : Role Based Access Control)[2, 3] 및 행위 기반 접근 통제(ABAC : Activity Based Access Control)[4,5] 모델과 기업 환경에 적합한 과업-역할 기반 접근 통제 모델(T-RBAC : Task-Role Based Access Control)[6] 모델 등이 있다.

그러나 이들 모델들은 모두 기업 환경에 대한 애플리케이션에서 시간 제약에 따른 자원의 사용제한을 하지 못한다는 제약이 있고, 기업 환경에서 반드시 필요하고 빈번히 발생할 수 있는 사용자 대 사용자 위임, 역할 대 역할 위임, 다단계 위임, 다중 위임 등의 기능을 제공하지 못한다는 제약이 있다. 또한 유비쿼터스 환경에서 발생할 수 있는 객체나 공간에 대한 접근제어를 제공하지 못한다.

본 논문에서는 시간(기간과 주기)에 따른 제약으로 자원의 사용을 제한할 수 있는 temporal constraints를 고려한 GEO-RBAC[18]과 GTRBAC (Generalized Temporal Role Based Access Control)[7,8,9] 모델의 Fine-grained Role-based Delegation 모델[10], 그리고 역할 계층을 그래프로 표현하여 멀티도메인의 개념과 역할 관리개념을 용이하게 할 수 있는 Role Graph Delegation 모델[11]을 적용하여 사용자 대 사용자(user to user) 권한 위임과 역할 대 역할(role to role) 권한 위임이 이루어질 수 있고, 사용자 또는 역할에 위임된 권한을 다른 사용자 또는 역할에 다시 위임할 수 있는 다단계 위임, 권한을 여러 사용자 또는 역할에 위임할 수 있는 다중 위임 그리고 상향 위임 기능을 제공한다. 또한 권한의 속성에 따라 6가지의 역할로 나누어 위임과 상속에 제한을 둘 수 있는 위임 모델을 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 역할 기반 접근제어 모델과 위임모델에 대해서 살펴보며, 3장에서는 제안 모델의 구성요소와 특징에 대해서 살펴보고 4장에서는 논문의 결론에 대해 논의한다.

Ⅱ. 관련 연구

이 장에서는 기존 연구들을 재검토하고 그들의 특성을 분석한다.

2-1 GTRBAC 모델

- GTRBAC 모델(Generalized Role Based Access Control Model)은 TRBAC 모델(Temporal Role Based Access Control)을 확장한 모델로, 역할의 사용과, 역할 - 권한 할당, 역할 활성화를 포함한 주기적이고 지속적인 시간의 제약 집합을 위한 명세를 포함한다. GTRBAC 모델의 특징을 정리하면 다음과 같다.

- Temporal constraints on role enabling/ disabling : 이 제약은 지연시간 또는 일정 기간 동안 사용자 - 역할 또는 역할 - 권한에 할당된 역할이 가능하도록 한다.

- Temporal constraints on user-role and role-permission assignments : 지정된 지연시간 또는 기간 동안 사용자와 권한을 역할에 할당한다.

- Activation constraints : 이 제약은 사용자들이 역할을 활성화 할때 제한을 한다. 명세된 가간동안 역할의 활성화를 제한하거나 세션 상에서 역할의 활성화 수를 제한한다.

- Run-time events : 런타임 이벤트 들은 관리자가 GTRBAC 이벤트들을 동적으로 시작하거나, 역할 활성화 제약들 또는 기간을 가능하도록 한다.

- Constraint enabling expressions : GTRBAC 모델은 가능 또는 불가능하게 하는 기간 제약들과 역할 활성화 제약들을 포함한다. 기간 제약들은 사용자 - 역할 할당관계와 역할 - 권한 할당관계들에 의해 역할이 가능하게 한다.

- Triggers : 트리거 들은 다양한 임시 이벤트들 사이의 종속성을 표현하기 위하여 트리거 프레임 웍을 제공하여 시스템에 의해 동적으로 변화하는 접근제어 요구사항에 적절히 대응할 수 있다.

TRBAC 모델뿐만 아니라 GTRBAC 모델에서도 임시 제약들과 역할 계층 사이의 상호작용이 중요한 문제이다. GTRBAC 모델은 역할들에 대한 임시 제약들의 존재로 I(permission - inheritance - only hierarchy), A(role - activation - only hierarchy), I-A (permission - inheritance - activation hierarchy) 역할 계층과 같은 부분 역할 계층이 존재하고 이들 역할 계층은 역할의 활성화/비활성화 제약과 시간 제약을 이용하여 제한된 상속 기능을 제공한다. I 역할 계층은 상위 역할에 사용자가 할당되고 역할이 활성화 되면 세션 상에서 하위 역할의 활성화와 상관없이 모든 권한이 상위 역할로 상속되는 역할 계층이고 A 역할 계층은 상위 역할에 할당된 사용자가 하위 역할을 활성화 할 수 있는 역할계층이다. I-A 역할 계층은 I 역할 계층과 A 역할 계층의 혼합형 부분 역할 계층이다. GTRBAC 모델에서는 역할 계층에서 최소 권한 원칙에 위배되지 않도록 하기 위하여 활성화 가능한 역할 집합을 계산 하여야 하는데 특히 I 역할계층, A 역할계층, I-A 역할계층이 혼합되어있을 경우 활성화 가능한 역할의 집합을 계산하는 것은 매우 복잡하게 된다. 이러한 여러 역할 계층이 혼합된 역할계층에서 사용자에게 할당된 역할에 의해 활성화 될 수 있는 역할의 집합을 UAS(uniquely activable set)이라 한다. UAS는 단일 세션에서 사용자에 의해 활성화 될 수 있는 역할들의 집합을 말하고 역할 계층을 통하여 사용자에 의해 활성화 될 수 있는 역할집합을 결정할 수 있도록 해줌으로써 최소 권한 원칙을 유지할 수 있도록 도와준다. 또한 GTRBAC 모델은 역할 활성화/비활성화와 이벤트 제약, 트리거를 이용하여 기존 모델에서는 불가능했던 접근 권한의 동적 활성화와 응용 레벨 제약의 명세를 필요로 하는 워크플로우(workflow)를 고려할 수 있게 되었고, 시간(기간과 주기)에 따른 제약으로 자원의 사용을 제한할 수 있게 되었다.

2-2 Role Graph 모델

Role Graph 모델[11]은 RBAC 모델을 근간으로 하고 있으며, 역할과 역할 관계 그리고 역할 모델 관리를 위해 그래프로 모델을 표현하고 있다. Role Graph 모델은 다음과 같은 특징을 갖는다.

1. MaxRole : 그래프의 최상위에 있으면 하위 역할의 모든 권한을 갖는다. MaxRole에는 일반역할을 할당하지 않는다.
2. MinRole : 시스템에서 최소 권한 집합을 갖는다. 이것은 빈 집합이 될 수 있다.
3. 모든 역할은 MinRole로 부터의 길을 갖는다.
4. 모든 역할은 MaxRole 까지의 길을 갖는다.
5. 어떤 두 개의 역할이 있고 (r1, r2) r1이 상위역할, r2가 하위 역할이라면 r2에서 r1 으로의 길을 가지고 있어야 한다.

Role Graph 모델은 Group Graph, Role Graph, Privileges의 세 개의 Panel로 나누어 서로에 대한 할당관계를 보다 쉽게 보여주고 있으며, MaxRole과 MinRole을 갖는 역할계층을 두어 트리 구조 역할 계층과 역트리구조 역할계층이 갖는 장점을 모두 취합하였다. 또한 MaxRole에 의하여 중앙 집권적 관리 역할을 갖는다. 다음 그림 1은 Role Graph 모델의 표현이다.

그림 2는 Role Graph 모델의 관리 도메인 표현이다. 관리 도메인과 관리 역할에는 다음과 같은 특징이 있다.

- 관리 도메인과 관리 역할

관리 도메인의 두 가지 타입 : ① 기본 도메인, ② 정규(보통) 도메인

① 기본 도메인 : 역할의 전체 조직을 표현, 시스템 보안 관리자(SSO)에 의해 관리됨.

② 정규 도메인 : MinRole을 제외한 유일 역할의 하위 역할을 포함. 유일 역할은 도메인의 역할 계층의 맨 위에 있다. (유일 역할은 역할의 job으로 구분). 정규 도메인을 구분하기 위해 도메인 indentifier 또는 도메인 ID를 사용.

관리 도메인은 역할 그래프의 부분 집합이다. D로 표시하며 역할의 집합을 포함한다. 관리자 역할은 (name, 권한집합, 관리 도메인 형태의 역할 집합)으로 되어있으며, 관리 역할에 할당되어진 사용자에 의해 관리되어진다. 관리 도메인과 관리 역할사이의 관계는 다대다 관계이다.

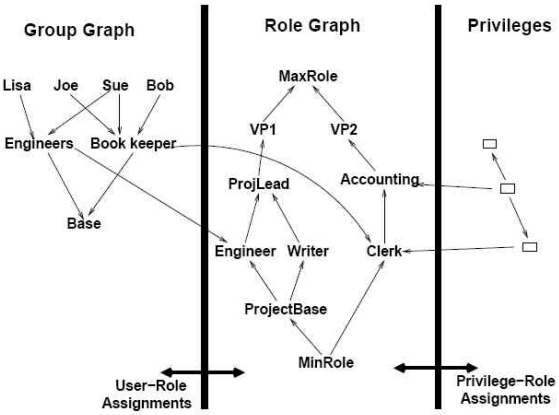

그림 1 Role Graph 모델
Fig 1. Role Graph model

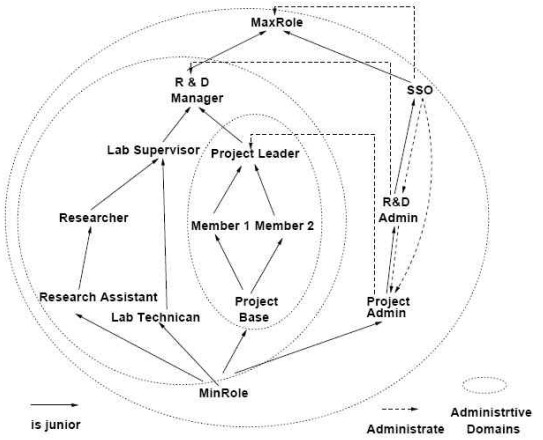

그림 2 Role Graph 모델의 관리 도메인 표현
Fig 2. Management domain of Role Graph model

2-3 GEO-RBAC 모델

GEO-RBAC(Geometry Role Based Access Control) 모델은 유비쿼터스 환경에 적합한 공간과 객체에 대한 접근제어 모델이다. GEO-RBAC 모델에서는 크게 세션, 역할, 사용자, 권한으로 나누어진다. 권한은 공간 객체와 객체의 동작으로 나누어지며, 역할은 역할 스키마, 역할 인스턴스로 나누어진다. 마지막으로 세션은 실제 위치 매핑 기능과 세션들로 나누어진다. GEO-RBAC 모델에서는 OGIS(Open GIS Consortium)을 기반으로 모든 지형을 데이터베이스화 시켜 정밀한 위상적 논리 공간을 만들고, 사용자의 위치를 위치 인식 단말기나 휴대폰으로 인식하여 물리 지형의 위치를 논리공간에 매핑 시키는 방법을 사용한다. 이렇게 만들어진 논리 공간에 정밀한 공간 범위를 설정하고, 이 범위에 역할을 할당하여 사용자가 정해진 범위의 공간 안에서 자원의 사용을 제한할 수 있도록 한 모델이다. GEO-RBAC 모델의 특징을 정리하면 다음과 같다.

1) 사용자(Users)

사용자는 로봇, 소프트웨어 에이전트, 컴퓨터 또는 지능을 가진 자율적인 대리인이나 사람이 될 수 있으며, 유비쿼터스 환경 내의 정보를 사용하는 주체이다. 보통은 사람을 지칭하는 것으로 하나의 주체는 한 명의 사람에 대응된다.

2) 공간 인식 객체(Spatially aware objects)

지형(feature)에 각각 인식할 수 있는 이름을 두고 그 이름으로 신원을 확인한다. 그 지형은 치수를 가지며, 기하학으로서 표현된다. 지현은 크게 공간과 비 공간으로 나누어지게 되는데, 공간은 길, 호수 도시 등 지형을 나타내며, 비공간은 자동차, 비행기 등 지형형태로 나타낼 수 없는 것들을 말한다. 또한 지형이 어떠한 위치에도 속하지 않았다면 비공간이다. 기존의 GEO-RBAC 모델에서는 OGIS(Open GIS consortium)[14]을 이용하여 지구의 모든 지형은 데이터베이스화한 논리지형에 GPS나 모바일 단말기에서 얻은 정보를 실세계와 논리지형에 매칭시켰으나, 본 논문에서는 zigbee 센서를 이용하여 실세계의 데이터를 논리 지형화 시킨다.

3) 역할(Roles)

GEO-RBAC 모델에서 역할은 가장 중요한 개념이며 역할을 두 가지로 나누어 구분한다.

가. 공간 역할(Spatial role)

공간 역할이란 역할 인스턴스라고도 불리며 역할 r과 역할의 공간 범위 e를 한 쌍으로 묶어 실제 적용될 역할을 말한다.

예) <r, e> (r=role, e=extent)

공간역할에는 역할이름, 역할 범위를 포함하고 있으며 공간 제약을 이행하는 집합이다. 역할 범위는 역할이 사용자가 권한을 행사할 수 있는 장소의 경계선을 의미한다. 또한 공간 역할은 공간 역할과 비공

간 역할로 나누어지는데, 공간 역할은 역할의 공간 경계가 정확한 시맨틱을 갖는 공간 지형인데 반해 스키마 역할 이름과 같은 이름을 갖는다.

나. 역할 스키마(Role Schema)

역할 스키마는 역할 스키마가 실행되기 위한 공간 인식 조직 기능의 집합으로 공통된 어떤 성질들을 정의한다. 역할 스키마는 공간 역할 집합을 위한 공통된 이름뿐만 아니라 역할이 가능하게 할 수 있는 공간 제약도 정의하고 있다.

예) 역할 범위의 지형타입, 논리적 지형의 지형타입, 실제위치와 매핑한 지역 타입 등

역할 스키마의 특징으로서 역할이 가능한 공간제약과 사용자의 행동위치를 작은 점으로 표시하며, 역할 범위의 이름에 의해서 완전히 신원이 파악되고 역할 스키마가 권한을 할당할 수도 있다.

예) Rschema=<TaxiDriver, Road-Network, Point On Road, mPoint On Road>

TaxiDriver : 역할의 이름.

Road-Network : 역할범위.

Point On Road : 논리적 위치

mPoint On Road : 실제위치를 매핑시켜 놓은 위치.

4) 권한(Permissions)

권한은 Operation과 Objects 두 가지로 이루어져 있다.

예) PRMS = <Ops, Obs> (Ops = operations, Obs = objects)

5) 세션(Sessions)

세션은 기본적으로 EnableSessionRoles와 SessionRoles 그리고 SessionUsers 으로 나누어진다.

가. EnableSessionRoles

사용자가 역할범위 공간에서 논리적 위치와 맞으면 EnableSessionRoles이다.

나. SessionRoles

사용자가 GEO-RBAC 서버에 로그인 하였지만, 역할 범위 공간의 논리적 위치에서 벗어나면 SessionRoles 이다.

다. SessionUser

사용자가 실세계와 논리지형에 매칭이 된다면 SessionUser 이다.

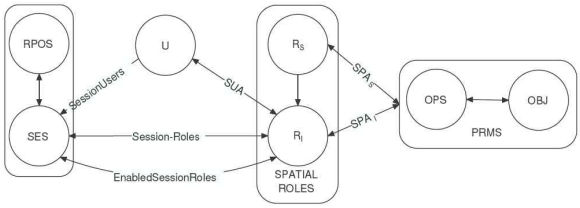

그림 3 Core GEO-RBAC
Fig 3. Core GEO-RBAC

5) GEO-RBAC 역할 계층(GEO-RBAC Hierarchy)

GEO-RBAC 모델에서는 역할 인스턴스와 역할 스키마 각각의 역할계층이 존재하며, 그것은 역할 스키마의 정의에 따르고 있다. 가령 역할 스키마에 정의된 역할계층에서 r1이 최상이 역할이고 그 밑으로 r2, r3의 하위 역할이 있다면 역할 인스턴스 역시 같은 역할 계층을 갖는다. 이는 역할 스키마가 역할 인스턴스를 정의하고 있기 때문이다. 하지만 GEO-RBAC 모델의 역할 계층은 RBAC96 모델과 같은 일반적인 역할계층으로 표현하고 있어 위에서 언급한 내용과 같은 문제가 생긴다.

2-4 Temporal constraints를 고려한 GEO-RBAC 모델

temporal constraints를 고려한 GEO-RBAC 모델에서는 기존의 GEO-RBAC모델[17]에 시간 제약을 추가한 모델이다. 시간 제약의 관념은 GTRBAC의 정의를 따르고 있다. 이 모델의 시간 제약에 따른 정적, 동적 의무 분리는 다음과 같다.

의무 분리 제약 정의는 기본적으로 역할의 집합이 상호배타적 관계를 맺고 있는 것이다. 하지만 상호배타적 관계를 갖지 않는 역할들이 특수한 경우에 의해 제약이 필요한 경우 기존의 GEO-RBAC 모델의 제약으로는 불가능한 상황을 temporal constraints를 사용함으로써 좀 더 강력하고 유연한 제약 정책을 보여줄 수 있다. 간단한 예를 들어 u1과 u2라는 사용자와 공간 s1을 가정한다. 이 두 사용자가 공통으로 사용할 수 있는 공간 s1을 개인 프라이버시에 의해 사용자 u1과 u2가 나누어 갖길 원하는 경우나 근무 시간이 겹칠 때 이해 충돌이 발생하는 경우, temporal

constraints 정책에 따라 시간별 사용시간을 정해준다면 쉽게 해결 할 수 있을 것이다. 제안된 모델에서 시간 제약의 관념은 GTRBAC의 정의를 따르고 있고 (set, n)type의 의미는 정의된 집합이 상호 배타적 관계를 갖지 않는 역할관계를 나타낸다.

temporal constraints를 고려한 GEO-RBAC 모델의 정적 제약특성은 다음과 같다.

① SIP(Static Instance-based Periodicity) 제약과 SID(Static Instance-based Duration) 제약 : 인스턴스 레벨에서 정의되며 (role_instance_set, n)⊥ & (I, P, SSoD) 그리고 (role_instance_set, n)⊥ & ([I, P|D], Dx, SSoD)로 표시한다. 활성화 된 역할의 기간 제약은 전체 활성 기간 제약과 활성화 최대 기간 제약으로 다시 나눌 수 있는데 전자는 주어진 기간에서 역할 활성화 기간 대한 제약이며, 후자는 역할이 활성화 할 수 있는 최대 기간에 대한 제약이다. 이 제약은 나머지 제약들에도 적용된다. SIP/D 제약의 목적은 역할 인스턴스 집합에 정의된 역할에서 n개 또는 더 많은 역할 인스턴스가 주어진 시간 또는 기간 동안 실행되는 것을 금지한다. 역할은 같은 스키마 이거나 다른 스키마의 인스턴스일수도 있다.

예) RoleSet = {Doctor(Hosp1), Doctor(Hosp2)}를 가정하면, 제약 (RoleSet, 2)⊥ & ([1.1.2006, ∞], WorkingDaysOfWeek),({Doctor(Hosp1), Doctor(Hosp2)})) ∈ SIP의 의미는 사용자가 Hosp1과 Hosp2 두 곳에서 2006년 1월 1일부터 ∞까지 의사가 될 수 없다. 제약 (RoleSet, 2)a 의 의미는 Nurse가 근무하는 동안 병동 Dep1과 Dep2 에서 활성화 되어 질 수 없다.

② SSNSP(Static Schema-based Non-Spatial Periodicity) 제약과 SSNSD(Static Schema-based Non-Spatial Duration) 제약 : 스키마 레벨에서 정의되며 (role_schema_set, n)⊥ & (I, P, SSoD) 그리고 (role_schema_set,n)⊥ & ([I, P|D], Dx, SSoD)으로 표시한다. SSNSP/D 제약의 목적은 role_schema_set에서 n개의 스키마로부터 n개의 역할 인스턴스가 주어진 시간 또는 기간 동안 실행될 수 없도록 한다.

예) 역할 스키마 Do = <Doctor, Hospital, Sector, mSector>를 가정하면, 제약 (Do, 2)⊥ & ([1.1.2006, ∞], WorkingDaysOfWeek),({Do})) ∈ SSNSP 제약의 의미는 사용자가 여러 개의 병원에서 2006년 1월 1일부터 ∞까지 의사가 될 수 있다는 의미이다.

③ SSSP(Static Schema based spatial Periodicity) 제약과 SSSD(Static Shema based spatial Duration) 제약 : 스키마 레벨에서 정의되며, (r1, r2, rel)⊥ & (I, P, SSoD) 그리고 (r1, r2, rel)⊥ & ([I, P|D], Dx, SSoD)로 표시한다. SSSP/D 제약의 목적은 두 개의 역할 스키마가 어떠한 위상적 관계를 갖는 경우 사용자는 주어진 시간 또는 기간 동안 하나의 역할만을 수행할 수 있게 한다.

예) 역할 Doctor와 Manager의 Do와 Ma스키마를 가정한다. 제약 (Do, Ma, Equal)⊥ & ([1.1.2006, ∞], WorkingDaysOfWeek),({Euqal}) ∈ SSNSP의 의미는 사용자는 같은 병원에서 주어진 시간 또는 기간 동안 Doctor와 Manager가 동시에 될 수 없다는 의미이다.

temporal constraints를 고려한 GEO-RBAC 모델의 동적 제약 특성은 다음과 같다.

① DIAP/D(Dynamic Instance-based Activation Periodicity/Duration) 제약과 DIEP/D (Dynamic Instance-based Enabling Periodicity/Duration) 제약 제약과 제약 : 인스턴스 레벨에서 정의되며, (role_instance_set, n)f & (I, P, DSoD) f ∈{a, e} 그리고 (role_instance_set, n)f & ([I, P|D], Dx, DSoD) f ∈ {a, e}로 표시한다.

- f = a인 경우 : activation time
- f = e인 경우 : enabling time

DIA/E 제약의 목적은 role_instance_set에 명세 되어진 것 중 한 세션에서 n개의 역할 또는 더 많은 역할이 주어진 시간 또는 기간 동안 인스턴스의 실행을 금지한다. 예)인스턴스 RoleSet={Nurse(Dep1), Nurse(Dep2)}를 가정하면, 제약 (RoleSet, 2)a & ([1.1.2006, ∞], WorkingDaysOfWeek), (Nurse(Dep1), Nurse(Dep2))의의미는 Nurse가 근무하는 2006년 1월 1일부터 ∞까지의 시간동안 병동 Dep1과 Dep2 에서 활성화 되어 질 수 없다.

② DSNSAP/D(Dynamic Schema-based Non-Spatial Activation Periodicity/Duration) 제약과 DSNSEP/D (Dynamic Schema-based Non-Spatial Enabling Periodicity/Duration) 제약 : 스키마 레벨에서 정의되며, (role_schema_set, n)f & (I, P, DSoD), n≥2, f∈{a,

e} 그리고 (role_schema_set, n)f & ([I, P|D], Dx, DSoD), n≥2, f∈{a, e}로 표시한다.

DSNSAP/D와 DSNSEP/D 제약의 목적은 role_schema_set에서 n개의 세션으로부터 n개 또는 더 많은 역할 인스턴스가 주어진 시간 또는 기간 동안 활성/가능 할 수 없게 한다. 예) 간호사 역할을 위한 스키마 Nu를 가정한다. activation time 제약({Nu}, 2)a & ([1.1.2006, ∞], WorkingDaysOfWeek), (Nu)은 간호사가 다른 병동에 할당 될 수 있는 여건이지만 시스템과 상호 작용하는 세션에서 2006년 1월 1일부터 ∞ 시간까지 하나 이상의 병동에서 활동할 수 없다.

③ DSSAP/D(Dynamic Schema-based Spatial Activation Periodicity/Duration) 제약과 DSSEP/D(Dynamic Schema-based Spatial Enabling Periodicity/Duration) 제약 : 스키마 레벨에서 정의되며 (rs1, rs2, rel)f & (I, P, DSoD) f∈{a, e} 그리고 (rs1, rs2, rel)f & ([I, P|D], Dx, DSoD) f∈{a, e}로 표시한다. DSSAP/D와 DSSEP/D 제약의 목적은 역할의 범위가 공간 관계 rel을 만족할 때 rs2 스키마와 rs1 스키마로부터 역할 인스턴스를 구분 짓는다. 예) 역할 Doctor와 Patient에 반응하는 Do와 Pa스키마를 가정한다. 제약 (Do, Pa, Equal) & ([1.1.2006, ∞], WorkingDaysOfWeek), (Equal)의 의미는 의사역할이 활성화 중일 때 같은 병원에서 주어진 시간 또는 기간 동안 환자 역할로 활성화 할 수 없음을 의미한다.

기존모델의 제약 조건에 비교하여 temporal constraints를 고려한 GEO-RBAC 모델은 다음과 같은 특성을 갖는다.

- 역할 스키마 레벨에서의 역할 시간, 기간, 주기 제약 고려.
- 역할 스키마 레벨에서의 역할 enable/activation 시의 시간, 기간, 주기 제약 고려.
- 역할 인스턴스 레벨에서의 역할 시간, 기간, 주기 제약 고려.
- 역할 인스턴스 레벨에서의 역할 enable/activation 시의 시간, 기간, 주기 제약 고려.

Ⅲ. 제안 모델

3-1 제안 모델의 목적

유비쿼터스 컴퓨팅 환경이란 사람과 컴퓨팅 기기 및 환경이 서로 상호 작용하여 컴퓨터가 스스로 사람의 필요사항을 알아서 처리하여 인간의 일상 활동을 지원하거나 향상시키는 것을 목표로 하는 컴퓨팅 환경이다.이 중 context 정보의 관리는 유비쿼터스 환경을 현실 세계에 구현함에 있어 가장 중요한 핵심 기술 중의 하나이다. 이를 위해 유비쿼터스 환경을 context 사이의 연관 관계로 표현하는 다양한 context 모델링 기법이 연구되고 있으며, 또한 context 모델을 이용하여 개발자는 필요한 context 정보를 이용하는 어플리케이션을 설계한다. 다양한 보안정책을 지원할 수 있는 역할기반 접근제어 모델은 차세대 환경에 가장 적합한 보안모델로 평가받고 있다. 하지만 기존에 연구 되었던 context 접근제어 모델들은 공간에 대한 접근제어와 객체에 대한 접근제어, 그리고 그 모델에 따르는 위임 기법과 권한 속성을 정의하지 않고 있어 실제 적용하여 사용하기에는 적합하지 않다. 따라서 본 논문에서는 공간접근제어에 적합한 GEO-RBAC 모델을 이용하여 그에 따른 다양한 제약조건을 고려한 모델인 Temporal constraints를 고려한 GEO-RBAC[17] 모델에 다양한 위임 기법을 적용한 context awareness 접근제어 모델을 제안한다.

3-2 제안 모델의 역할 계층

GEO-RBAC 모델의 역할계층은 RBAC과 같은 일반적인 역할계층의 구조를 띠고 있어 GTRBAC 모델의 역할계층으로 GEO-RBAC 모델의 역할계층을 확장한다. GTRBAC 역할계층의 특징으로는 역할들에 대한 임시 제약들의 존재로 I(permission - inheritance - only hierarchy), A(role - activation - only hierarchy), I-A (permission - inheritance - activation hierarchy) 역할 계층과 같은 부분 역할 계층이 존재하고 이들 역할 계층은 역할의 활성화/비활성화 제약과 시간 제약을 이용하여 제한된 상속 기능을 제공한다. I 역할 계층은 상위 역할에 사용자가 할당되고 역할이 활성화되면 세션 상에서 하위 역할의 활성화와 상관없이 모

든 권한이 상위 역할로 상속되는 역할 계층이고 A 역할 계층은 상위 역할에 할당된 사용자가 하위 역할을 활성화 할 수 있는 역할계층이다. I-A 역할 계층은 I 역할 계층과 A 역할 계층의 혼합형 부분 역할 계층이다.

GTRBAC 모델에서는 역할 계층에서 최소 권한 원칙에 위배되지 않도록 하기 위하여 활성화 가능한 역할 집합을 계산 하여야 하는데 특히 I 역할계층, A 역할계층, I-A 역할계층이 혼합되어있을 경우 활성화 가능한 역할의 집합을 계산하는 것은 매우 복잡하게 된다. 이러한 여러 역할 계층이 혼합된 역할계층에서 사용자에게 할당된 역할에 의해 활성화 될 수 있는 역할의 집합을 UAS(uniquely activable set)라 한다. UAS는 단일 세션에서 사용자에 의해 활성화 될 수 있는 역할들의 집합을 말하고 역할 계층을 통하여 사용자에 의해 활성화 될 수 있는 역할집합을 결정할 수 있도록 해줌으로써 최소 권한 원칙을 유지할 수 있도록 도와준다. 그림 4는 GEO-RBAC 역할계층의 확장 모델을 표현한다.

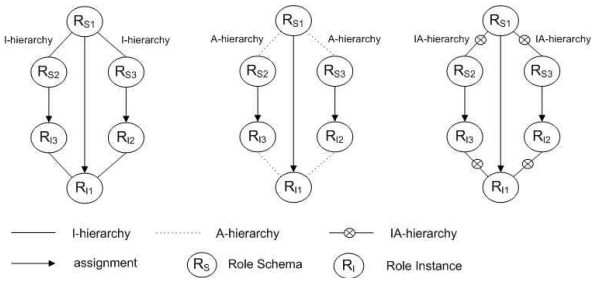

그림 4 GEO-RBAC 역할계층의 확장 모델
Fig 4. Extension model of GEO-RBAC role hierarchy

3-5 제안 모델의 권한 위임 모델

제안 모델에서는 기존 위임 모델인 Role Graph Delegation 모델의 그래프 표현, 다단계 위임, 다단계 취소, 관리 역할의 장점을 취하고, Fine-grained Role Delegation 모델에의 다양한 계층으로의 위임을 표현하였다. Role Graph Delegation 모델은 Fine-grained Role Delegation 모델에 비하여 비교적 쉽고 강력한 위임정책을 보여주었지만 I, A, IA 계층을 지원하지 않아 유연성 면에서 떨어지고, Fine-grained Role Delegation 모델은 RBAC(Role Based Access Control) 모델에 적합한 하향, 상향 위임정책과 GRTBAC(Generalized Temporal Role Based Access Control)을 기반으로 권한의 특성에 따른 역할분할을 보여주었지만, 다단계 위임과 다단계 취소에 대한 정책을 표현하지 못하고 있어 기존의 위임 정책의 기능을 지원하지 못한다. 이에 제안 모델에서는 위임모델의 그래프적인 표현, 다단계 위임, 다단계 취소, 관리 역할, 상/하향 위임, 혼성계층에서의 위임정책을 표현할 수 있도록 하였다.

1. 상향 위임

Role Graph Delegation 모델에서 가장 기본이 되는 요소는 MinRole과 MaxRole을 갖고 MinRole에서 MaxRole까지의 패스를 갖는다는 점이다. 이는 역할 계층의 방향성을 표시할 뿐만 아니라 각 역할 도메인들에 대한 관리 역할 계층에 대한 표현과 위임 역할에 대한 관리 역할의 표현에도 중요하다. 하지만 이는 I, A, IA 계층에 대한 표현과 상향 위임에 대한 표현이 불가능 하게 만든다. 따라서 MinRole과 MaxRole은 갖되 패스를 갖지 않고 I, A, IA 계층에 대한 이해구성으로 계층을 형성한다. 단 MinRole과 MaxRole은 존재한다. 상향 위임이 필요한 경우의 예로는 역할 P에 할당된 존이라는 사용자가 아파서 업무를 수행하지 못할 경우, 그의 업무를 대신할 능력을 가진 사용자가 필요하게 된다. 또한 P의 하위 역할이 존재하지 않을 경우, 그의 상위 역할이 업무를 대신할 수 있어야할 때, 상향 위임이 필요하게 된다.

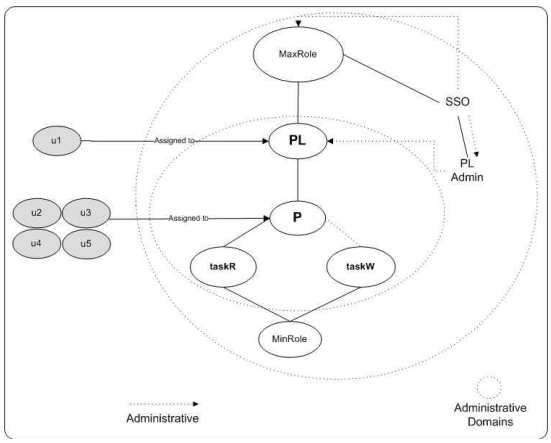

그림 5 제안 모델의 역할 그래프 표현
Fig 5. Role Graph of proposed model

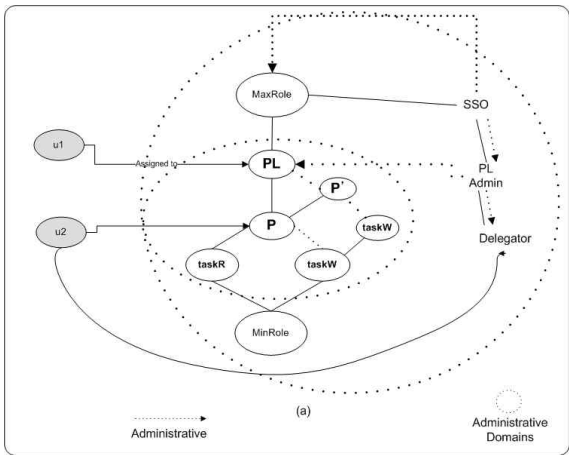

그림 6 제안 모델의 상향 위임
Fig 6. Upward delegation of proposed model

그림 5는 제안 모델의 역할계층 표현이다. 관리 역할이나 MinRole, MaxRole를 갖는 것은 Role Graph Delegation 모델의 것과 동일하다. 차이점은 역할 계층의 표현이다. 위의 그림에서 역할 PL과 P 사이, 역할 P와 taskR 사이는 I-계층으로 형성되어 있으며, 역할 P와 taskW는 A-계층으로 형성되어 있다. 기존의 Role Graph Delegation 모델과는 달리 I, A, IA 계층으로 계층을 구성할 수 있다는 점이 가장 큰 차이점이다. 그림 5를 바탕으로 그림 6에서는 상향 위임을 표현한다.

제안 모델의 위임 모델 생성은 위임하려고 하는 주체가 Delegator 역할에 할당되면서 시작된다. Delegator 역할은 관리 역할의 하위 역할이며, 위임 역할의 생성, 사용자-역할 할당 취소, 권한 취소 등의 관리를 맡게 된다. 위임하려고 하는 하위 역할이 Delegator 역할에 할당되면서 P역할의 위임 역할인 P'를 만들고 P와 I-계층으로 형성된다. taskW 역할도 P 역할과 같은 방법으로 위임 역할을 만들고 생성되어진 빈 역할 P'와 taskW'는 I-계층에 의해서 하위 역할인 P와 taskW의 권한을 상속한다. 그리고 위임역할들은 그들의 오리지널 역할의 관계와 같이 A-계층으로 형성되고 이를 PL 역할과 다시 A-계층을 형성하여 권한을 위임한다.

2. 제안 모델의 I-계층 하향 위임

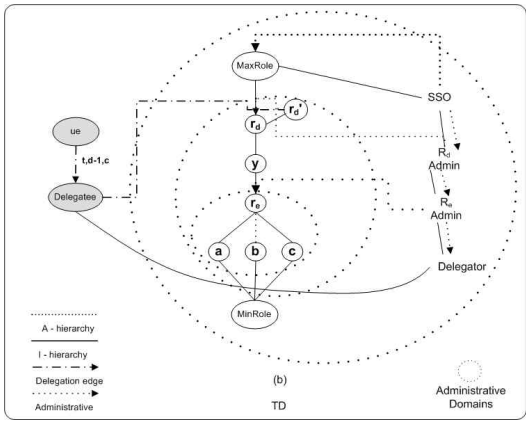

그림 7 제안 모델의 I-계층 하향 위임(단일단계)
Fig 7. I-Hierarchy downward delegation of proposed model (single step)

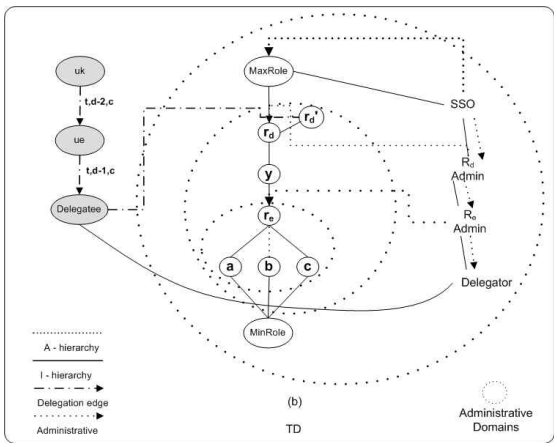

그림 8 제안 모델의 I-계층 하향위임(다단계)
Fig 8. I-Hierarchy downward delegation of proposed model (multi step)

그림 7은 제안모델에서의 I-계층 하향 위임을 보여준다. 역할 rd에 할당된 ud가 하위 역할인 ue에게 권한을 위임한다. 먼저 Delegator 역할을 생성하고, 위임자인 ud는 Delegatee 그룹을 생성한다. Delegatee 그룹을 생성하는 이유는 사용자 ue가 역할 rd에 직접 할당되는 것을 방지하기 위해서이다. 다음 rd의 위임 역할인 rd'를 rd의 상위 빈 역할로 만든다. I-계층에 의하여 rd'는 rd의 권한을 상속하게 된다. Delegatee 그룹은 Delegator 역할에 할당되고, ue의 그룹을 Delegatee 그룹에 할당한다. 이때 ue의 그룹에 위임 에지 라벨(t, d-1, c)을 만들어 표시한다. 위임 에지는 (t, d, c)로 이루어져 있으며, t는 위임 시간, d는 위임 깊이, c는 위임 제약 조건이 된다. 그림 3.6은 단일단

계 위임이며, 위임 에지의 생성으로 다단계 위임이 가능하게 된다. 그림 8은 I-계층 하향 위임에서의 다단계 위임을 표현한 것이다. 만일 그림 9에 사용자 uk를 할당한다면 uk를 ue에 할당하고 위임 에지(t, d-2, c)를 생성하면 된다. 이로써 Fine-grained Role Delegation 모델에서는 표현하지 못하였던, 다단계 위임을 표현할 수 있고, Role graph 모델에서 단순 상속계층이 아닌 I, A, IA 계층을 표현할 수 있다. 그림 9는 전체 위임에 관한 표현이며, 그 밖에도 부분 전체 위임과 부분 블록 할당 위임, 부분 권한 전체 위임, 부분 권한 블록 할당 위임 등으로 위임을 표현할 수 있다.

3. 제안 모델의 A-계층 하향위임

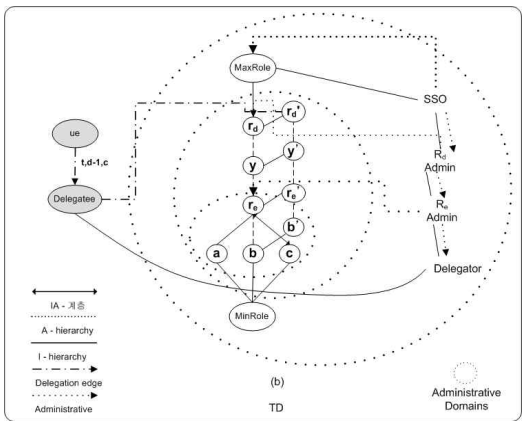

그림 9 제안 모델의 A-계층 하향위임
Fig 9. A-Hierarchy downward delegation of proposed model

그림 9는 제안모델의 A-계층 하향 위임이다. 역할 rd에 할당된 ud가 하위 역할인 ue에게 권한을 위임한다. 먼저 Delegator 역할을 생성하고, 위임자인 ud는 Delegatee 그룹을 생성한다. 다음 rd의 위임역할인 rd'를 rd의 상위 빈 역할로 만든다. 이때 I-계층 위임과 차이점은 rd의 하위 역할들이 A-계층으로 형성되어 있기 때문에 하위역할들이 활성화가 되어야지만 권한을 획득할 수 있다는 것이다. 그러므로 각각의 하위역할들도 I-계층으로 위임역할을 만들고 각 위임역할들이 A-계층으로 형성된다. 다단계 위임은 I-계층 하향 위임의 방법과 동일하다. 그 밖에도 부분 전체 위임과 부분 블록 할당 위임, 부분 권한 전체 위임, 부분 권한 블록 할당 위임 등으로 위임을 표현할 수 있다.

4. 제안 모델의 IA-계층 하향 위임

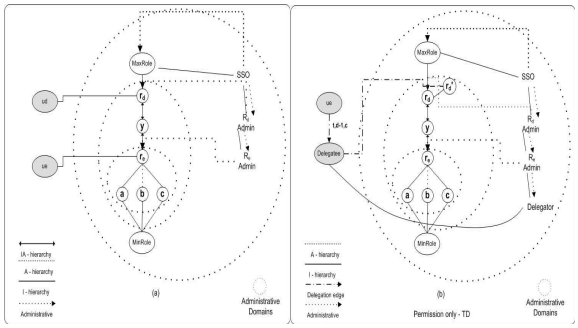

그림 10 제안 모델의 IA-계층 하향위임(권한)
Fig 10. IA-Hierarchy downward delegation of proposed model (permission)

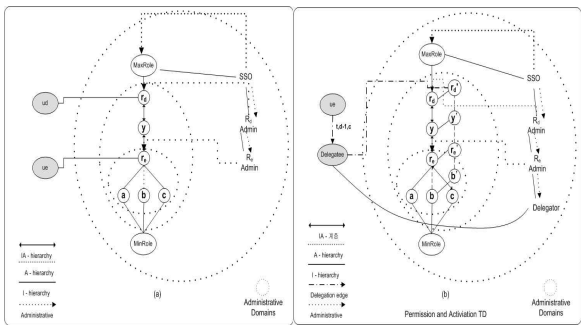

그림 11 제안 모델의 IA-계층 하향위임(활성화)
Fig 11. I-Hierarchy downward delegation of proposed model(activation)

그림 10은 제안 모델의 IA-계층에서의 하향 위임을 보여준다. IA-계층에서의 위임은 두 가지 경우로 나눌 수 있는데, 첫 번째로 권한(활성화)위임이고(그림 10), 두 번째로 권한-활성화 위임이다.(그림 11) IA-계층의 경우 I-계층의 속성과 A-계층의 속성 두 가지 모두를 가지고 있으므로, 위임에서 속성에 맞추어 IA-계층 자체 속성에 의한 위임과 I 또는 A 계층에 의한 속성에 따라 위임을 진행할 수 있다. 단계 위임은 I-계층 하향 위임의 방법과 동일하며, 부분 전체 위임과 부분 블록 할당 위임, 부분 권한 전체 위임, 부분 권한 블록 할당 위임 등으로 위임을 표현할 수 있다.

5. 제안 모델의 혼성계층 상향 위임

그림 12는 제안모델의 혼성계층 상향 위임이다. 그림 6의 상향 위임과 같은 방법으로 위임이 되지만 차이점은 다양한 계층을 지원한다는 점이다. 또한 Block assign을 통해 위임 기간 동안 전체 권한이 위

임되는 것을 방지할 수 있으며, 혼성계층으로 역할계층의 유연성을 높일 수 있다. 그 밖에도 부분 전체 위임과 부분 블록 할당 위임, 부분 권한 전체 위임, 부분 권한 블록 할당 위임 등으로 위임을 표현할 수 있다.

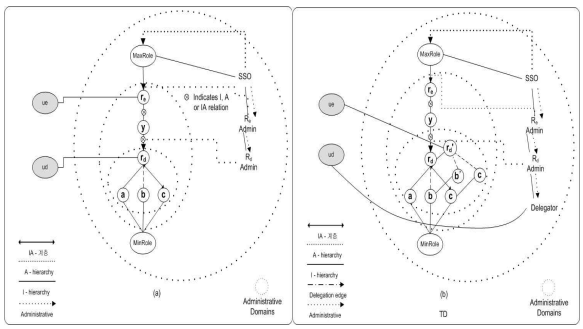

그림 12 제안 모델의 혼성계층 상향 위임
Fig 12. Mixed-Hierarchy upward delegation of proposed model

6. 제안 모델의 위임 취소

위임의 방법만큼이나 중요한 것이 위임의 취소이다. 기존의 각 위임모델들은 자신들의 특성에 맞게 부분 위임 취소와 전체 위임 취소 스키마를 가지고 있다. 하지만 GTRBAC 위임 모델은 단일단계 취소만 지원하고 있어, 부분 위임 취소와 다단계 위임 취소를 지원하지 않는다. 따라서 Role Graph Delegation 모델의 부분 위임 취소를 이용하여 GTRBAC 위임 모델에서는 지원하지 못했던 부분 위임 취소와 다단계 위임 취소를 지원한다. 위임 취소의 기본동작은 위임 에지를 지우는 것이다. 위임 에지가 삭제되면 위임은 더 이상 존재하지 않게 된다. 위임 취소 스키마는 다음과 같은 동작으로 이루어진다. 먼저 취소를 요청한 사용자나 역할로부터 시작하여 위임 에지를 획득한다. 다음 위임 에지에 명시된 위임 깊이(d : depth)로부터 위임의 깊이를 획득하고 취소하고 싶은 만큼의 깊이를 삭제한다.

그림 13은 정해진 깊이를 갖는 위임의 취소이다. 사용자 A는 B와 C에게 A의 권한을 위임하였다. 하지만 B로부터 시작된 권한 만 취소할 경우 B로부터 위임받은 D, E, F는 에지가 삭제되고 위임은 취소된다. 만약 깊이가 없다면 위임자로부터 시작된 에지를 삭제한다. 그림 14는 무한 깊이를 갖는 위임 취소를 보여준다. A는 그림 13과 마찬가지로 두 개의 위임 에지를 가지고 있지만, 마찬가지로 D 또한 두 개의 에지를 가지고 있다. 하지만 D 위임의 시작위치는 다르고 *(무한)위임을 갖는 속성과 제한 위임 속을 갖는 에지를 갖기 때문에 무한 속성의 위임 쪽을 취소할 수 있게 된다. 그림 15는 무한 위임과 제한 위임의 두 가지 속성을 갖게 되는 경우의 취소이다. B는 무한 위임을 줄 수 있어도 문제가 생기지 않는 반면 D가 무한 위임을 하게 되면 문제가 발생하는 것을 고려해 D가 제한 위임을 갖게 되었을 경우, 에지 또한 다른 속성의 에지를 가져야 한다. D가 무한 위임을 하지 못하는 경우라면 당연히 제한 위임속성을 가지고 있어야 하고 에지에는 위임 깊이를 표시해야 하기 때문이다. 중요한 점은 E 또한 두 개의 에지를 가지고 있고 같은 제한 위임의 속성이지만, 위임이 시작된 위치가 다르고 위임이 시작된 곳으로부터의 깊이를 가지고 있기 때문에, B->D->E에 이르는 위임에지를 삭제할 수 있다.

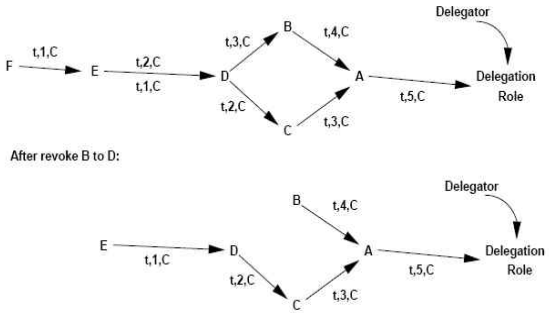

그림 13 제안 모델의 제한 깊이 위임 취소
Fig 13. Delegation revoke of proposed model(limited depth)

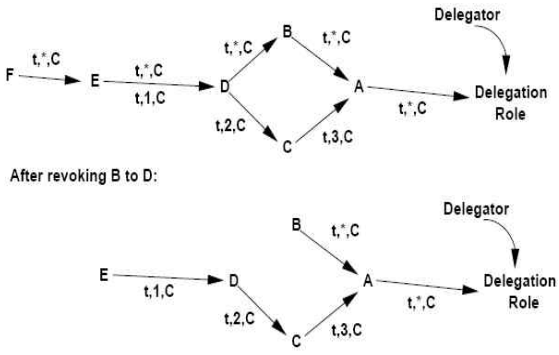

그림 14 제안 모델의 무한 깊이 위임 취소
Fig 14. Delegation revoke of proposed model(infinite depth)

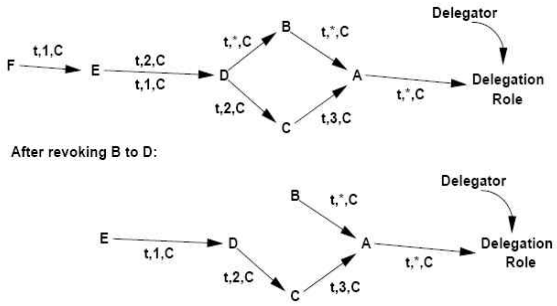

그림 15 제안 모델의 깊이 변화 위임 취소
Fig 15. Delegation revoke of proposed model(changes depth)

표 1 제안 위임 모델과 기존 위임 모델의 특징비교
Table 1. Compare the characteristics of the delegation model

		PBDM	GTRBAC Delegation	Fine-grained Role-based Delegation	제안 모델
일반 역할 및 권한 고려		○	○	○	○
역할의 제한적 상속	역할의 활성화 제약과 유효 시간 제약	×	○	×	○
	부역할 이용	×	○	×	○
역할 활성화 유효 시간 및 기간 적용		×	○	×	○
위임	사용자 대 사용자	×	○	○	○
	역할 대 역할	×	○	○	○
	다단계	×	×	○	○
	위임된 권한이 할당된 역할의 유효 시간 및 기간에 따른 제약	×	○	○	○
	혼성계층에 따른 위임	×	○	×	○
	상향위임	×	○	×	○
	부분위임	×	○	○	○
	사용자의 위치 정보에 따른 위임 제약	×	×	○	○

표 1에서는 기존의 위임모델들이 여러 가지 조건에서의 문제를 고려하지 않았음을 알 수 있으며, 제안 모델은 다양한 조건과 상황을 고려하여 위임을 적용할 수 있다. 대표적으로 기존의 위임모델에서는 고려하지 않았던 상향위임과 사용자의 위치 정보에 따른 위임제약, 혼성계층에 따른 위임을 지원한다.

Ⅴ. 결 론

본 논문에서는 GEO-RBAC 모델을 이용하여 유비쿼터스 환경에 적합한 접근제어 모델과, 그에 따른 제약 조건, 위임 모델을 제안하였다. 제안 모델은 유비쿼터스 환경에 적합한 GEO-RBAC 모델을 기본으로 기존의 GEO-RBAC 모델에서 지원하지 않았던 정적/동적 상황에서의 시간/기간 제약 추가, 역할 계층을 확장한 temporal constraints를 고려한 GEO-RBAC 모델에 다양한 권한 위임경우를 고려하여 역할 계층을 그래프로 표현하고, I, A, IA 계층에서의 상/하향 위임, 다단계 위임, 전체 위임, 부분 위임, 전체 취소, 부분 취소를 가능하게 하였다. 이로써 기존의 유비쿼터스 환경을 위한 접근제어에서 지원하지 않았던 공간과 객체에 대한 접근제어와 다양한 환경에서의 제약과 위임모델을 지원하여, 기존 접근제어 모델의 성능을 개선시켰다.

향후 연구로는 권한의 속성을 기반으로 하여 세부적인 부역할로 나누어 제안 모델의 접근제어 정책에 적용해야 할 것으로 사료되며, 제안 모델을 실제 유비쿼터스 환경에 적용하여 보다 편리한 접근제어를 구현하기 위한 연구도 필요한 것으로 사료된다.

참 고 문 헌

[1] C.P.Pfleeger, Security in Computing, second edition, Prentice-Hall International Inc, pp.290-315 , 1997

[2] R.S.Sandhu and E.J.Coyne and H.L.Feinstein and C.E.Youman “Role-Based Access control Models”, *IEEE Computer*, vol. 29, pp.38-47, 1996

[3] D.Ferraiom and J.Cugini and R.Kuhm “Role - based Access Control(RBAC) : Features and motivations”, *Proc. of 11th Annual Computer Security Application Conference*, 1995

[4] Dagstull and G.Coulouris and J.Dollimore “A Security Model for Cooperative work : a model and its system implications” *Positions paper for ACM European SIGOPS Workshop*, 1994

[5] R.K.Thomas and R.S.Sandhu “Task-based

Authorization Controls(TBAC) : A Family of Models for Active and Enterprise-oriented Authorization Management" *Proc. of the IFIP WF11.3 Workshop on Database Security,* 1997

[6] S. Oh and S. Park "Task-Role Based Access Control (T-RBAC): An Improved Access Control Model for Enterprise Environment", *Proceedings of the 11th International Conference on Database and Expert Systems Applications,* pp. 264-273, 2000

[7] J. B. D. Joshi and E. Bertino and A. Ghafoor "Temporal Hierarchies and Inheritance Semantics for GTRBAC", *Seventh ACM Symposium on Access Control Models and Technologies,* pp. 74-83, 2002

[8] J. B. D. Joshi and E. Bertino and A. Ghafoor "Hybrid Role Hierarchy for Generalized Temporal Role Based Access Control Model", *Proceedings of the 26 th Annual International Computer Software and Applications Conference*, 2002

[9] J. B. D. Joshi and E. Bertino and A. Ghafoor "Temporal Role Hierarchies in GTRBAC", *CERIAS*, 2002

[10] J. B. D. Joshi and E. Bertino "Fine-grained Role-based Delegation in Presence of the Hybrid Hierarchy", *SACMAT*, 2006

[11] He Wang, Sylvia L. Osborn, "Delegation in the Role Graph Model", *SACMAT,* 2006

[12] Anind K. Dey and Gregory Abowd, "Towards a Better Understanding of Contextand Context - Awareness", *Workshop on the what, who, where, when and how of context-awareness at CHI 2000*,April2000.

[13] 진희채, "지형,공간정보의 상호운영성을 보장하는 Open GIS", 1998.3 *정보과학회지* 제 16권 제 3호

[14] Ezedin Barka and Ravi Sandhu, "A Role-Based Delegation Model and Some Extensions", *Proc.Of 23rd National Information Systems Security Conference (NISSC 2000)*, pp.101-114, December, 2000.

[15] Xinwen Zhang, Sejong Oh and Ravi Sandhu, "PBDM:A Flexible Delegation Model in RBAC", *8th ACM Symposium on Access Control Modelsand Technologies,* pp.149-157, June, 2003.

[16] Ninghui Li, Mahesh V.Tripunitara. "Security Analysis in Role-Based Access Control", *Proceedings of the Ninth ACM Symposium on Access Control Models and Techniques,* pp.126-135, June,2004

[17] Elisa Bertino, Barbara Catania, Maria Luisa Damiani, Paolo Perlasca, "GEO-RBAC: a spatially aware RBAC", *CERIAS Tech Report* 2006-05

신 동 욱(申東旭)

2005년 2월 : 순천향대학교 정보기술공학부(공학사)
2007년 2월 : 순천향대학교 정보통신대학원(공학석사)
2006년 12월 ~ 현재 : 이지케어텍 SI개발팀
관심분야 : 네트워크 보안, 시스템 보안, 접근제어

황 유 동(黃有東)

1998년 2월 : 순천향대학교 제어계측 공학과 공학사
2000년 8월: 순천향대학교 전기전자공학과 석사
2003년~현재 : 순천향대학교 전기전자공학과 정보보호전공 박사과정
관심분야 : 네트워크 보안, 시스템 보안, 접근제어

박 동 규 (朴東圭)

1992년 : 한양대학교 대학원 전자공학과 공학박사
1999~2003년 : 순천향대학교 정보기술공학부 부교수
2004년~현재 : 순천향대학교 정보통신공학과 교수
관심분야 : 네트워크 보안, 유비쿼터스 컴퓨팅 보안