분산시스템 환경에서 관리 객체에 대한 역할기반 접근제어 모델

A Role-Based Access Control Model of Managed Objects in Distributed System Environments

최 은 복*
Eun-Bok Choi

요 약

본 논문에서는 역할기반 접근제어 정책을 지원할 수 있도록 관리객체 클래스 계층구조를 확장하였다. 또한, 관리 프로세스에 의해 설정되어지는 접근제어 규칙을 정적인 시간속성 외에 관리 정보에 대한 접근 권한을 수행하는 동적인 시간지원 기능을 갖는 선행조건에 관련된 제약사항 그리고 제약사항에 위배되었을 경우 관리자에게 보고되어야 할 위반 통지에 대해 기술하였다. 그리고 망관리 정보베이스에 대해 동적 시간지원 제약사항과 역할기반 접근제어 정책을 지원하는 시스템 구조를 제시하였다. 이 시스템에서는 BDL에 기반하여 접근제어의 집행과 결정함수에 관련된 모듈과 제약사항 그리고 각 역할들의 활성화 전이과정을 기술하므로써 동적 특성을 체계적이고 명확하게 표현하였다.

Abstract

In this paper, we extended hierarchial structure of managed object class to support Role-Based Access Control, and described constraint conditions that have support dynamic temporal function as well as statical temporal function established by management process. And we defined about violation notifications should report to manager when rules violate constraint conditions. Also we presented system architecture that support RBAC with MIB(Management Information Base) of ITU-T recommendation. By access control enforcement and decision function, constraint conditions and activated translation procedure of each roles are described, our system presents dynamic temporal property systematically.

키워드 : RBAC, Managed Objects, Network Management

1. 서 론

망 관리 시스템의 여러 가지 구성 요소들 중 가장 핵심적인 요소 중의 하나는 망관리에 필요한 정보 즉, 관리 객체들의 저장소인 관리 정보 베이스이다. 관리 정보베이스에 저장된 관리 객체들은 망 관리에 필수적이며 중요한 모든 정보들을 유지하고 있기 때문에 안전하게 유지가 되어야 한다. 망 자원에 접근을 원하는 사용자가 자신의 신원을 제시하고 인증 시스템으로부터 신원 인증을 받은 후, 확인된 사용자에 대한 망 자원을 접근하는 권한을 확인하는 과정을 접근 제어라고 한다. 이러한 접근 제어를 효과적으로 수행하기 위해서는 접근 권한의 불법 취득을 방지하고, 접근 권한에 관한

* 정 회 원 : 전주대학교 정보기술컴퓨터공학부 전임강사
ebchoi@jeonju.ac.kr (제 1저자)

불법 변조가 일어나지 않도록 하여야 한다.

관리 정보베이스의 보안유지와 관련된 대표적인 접근 제어정책은 크게 임의적 접근제어정책, 강제적 접근제어정책 그리고 역할기반 접근제어정책 등이 있다. 임의적 접근제어 정책은 정보객체에 대한 사용을 요청하는 사용자의 신원(identification)에 근거를 두고 접근허가를 결정하는 정책이다[7].

강제적 접근제어 정책은 정부 및 국방분야 등의 매우 제한적인 응용분야에서 매우 제한된 수의 보안 관리자들에 의해 정의된 일정한 규칙에 따라 정보에 대한 사용자의 접근 여부를 결정한다.

임의적 접근제어는 각 정보의 소유자들의 임의적인 판단에 따라 그 정보에 대한 접근권한을 다른 사용자에게 위임하거나 취소할 수 있는 권한을 가지고 있어, 강제적 접근제어 정책보다 정보에 대한 훨씬 유연하고 분산된 접근제어 기능을 수행할 수 있는 특징을 가지고 있다. 그러나 이 두 정책 모두 실제 기업환경에 적용되기에는 부적합한 특성들이 있다[11].

역할기반 접근제어 정책은 기업 환경뿐만 아니라 데이터베이스, 운영체제 등에 적용될 수 있는 매우 유연한 접근제어 정책으로, 임의적 또는 강제적 접근제어 정책보다 정보에 대한 추상적인 접근제어와 효율적인 접근권한 관리를 수행할 수 있는 장점을 가지고 있다. 또한 역할기반 접근제어 모델의 구성요소를 변경함으로써 이미 정의된 정책만을 지원하는 대신, 주어진 기업 환경의 특성과 필요에 따라 유연한 접근권한 정책을 구현할 수 있어 보안정책에 독립적인 특성을 제공한다[10,12].

OSI에서는 객체지향 개념을 이용하여 관리객체들을 정의하고 전체적인 정보를 모형화하고 있으며, 이러한 관리객체 클래스를 정의하기 위하여 GDMO(Guidelines for the Definition of Managed Objects)라고 하는 아홉 개의 템플릿을 제공하고 있다[15]. ITU-T X.741과 X812 표준안에서는 접근제어에 관한 전반적인 관리객체와 그들의 속성들에 대해 표현하고 있다. 여기에는 관리객체 클래스의 상속계층구조와 관리객체 상호관계를 표현하고 있으며 임의적 접근제어 모델에 해당하는 접근제어리스트와 능력리스트에 대한 관리객체를 표현하고 있고 강제적 접근제어 모델에 해당하는 보안등급에 관한 관리객체와 속성들을 표현하고 있다[16,17].

그러나 이러한 관리객체를 기술하기위한 가이드라인인 GDMO는 관리 객체의 구조와 속성 등을 포함한 대부분의 정적특성을 적절히 표현하고 있는데 반해, 동적특성을 표현하는 방법으로는 자연어를 이용하고 있다. 특히 실세계에 적용 가능한 역할기반 접근제어 정책에 관한 관리객체의 기술이 없으며 관리객체의 접근권한을 집행하고 결정하는 접근제어 집행함수와 접근제어 결정함수에 대한 세부적인 속성과 행위 등이 기술되지 않다. 이 때문에 동작과정에 대한 절차 등에 대한 관리객체의 모든 특성을 완전하게 표현하지 못하고 있으며, 이는 결국 접근제어에서의 동작과정에 직접적인 영향을 미치며 규칙에 대한 타당성 및 보안검증은 물론 시스템 구현 시에도 많은 어려움이 있다.

2. 역할기반 접근제어 모델

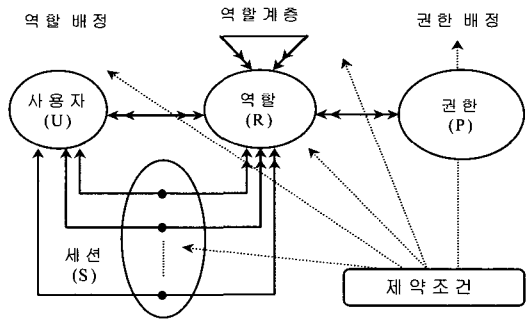
역할기반 접근제어 정책은 기업 환경뿐만 아니라 데이터베이스, 운영체제 등에 적용될 수 있는 매우 유연한 접근제어 정책으

로, 임의적 또는 강제적 접근제어 정책보다 정보에 대한 추상적인 접근제어와 효율적인 접근권한 관리를 수행할 수 있는 장점을 가지고 있다. 역할기반 접근제어 모델에서 가장 큰 특징은 정보에 대한 연산을 수행할 수 있는 권한들은 사용자에게 직접 할당되지 않고, 주어진 기업 환경에서 정의된 역할에 대해서만 배정한다는 점이다. 따라서 사용자가 원하는 정보에 대한 연산을 수행하기 위해서는 먼저 해당 정보에 대한 연산을 실행할 수 있는 권한을 가진 역할의 소속원(Member)이 되어야 한다. 사용자를 특정 역할의 소속원으로 배정하는 권한은 미리 정해진 보안 관리자들에 의해 수행될 수 있다. 그리고, 전체 시스템의 보안 관리를 위한 통제가 몇 명의 보안관리자에 의해 이루어지게 되므로, 임의적 접근제어 정책에서 발생할 수 있는 접근권한 통제의 어려움을 해결할 수 있다.

이처럼 역할기반 접근제어 모델에서는 권한의 관리를 사용자와 정보 객체간의 관계로 인식하는 대신 기업 환경에서의 역할과 정보 객체간의 관계로 설정, 관리함으로써 사용자와 정보 객체의 수가 대단히 많은 실제의 기업 환경에 매우 적합한 특성을 제공한다. 또한 최소권한 원칙(least privilege principle), 임무분리(SOD: separation of duty), 자료 추상화(data abstraction)와 같은 주요 보안 원칙들 역시 지원하고 있다[11].

▶ 최소권한 원칙

최소권한 원칙은 어떤 역할에 배정된 사용자가 실행하는 프로세스에게 그 프로세스의 수행에 필요한 권한만을 그 역할에게 배정하는 규칙을 의미한다.



(그림 1) 역할기반 접근제어 모델의 구성요소

▶ 임무분리 성질

보안 시스템에 의해 관리되는 정보의 무결성 보장을 위하여 정보의 무결성에 영향을 미치는 역할들에 배정되는 사용자들을 통제함으로써 보안시스템에 의해 관리되는 보안 특성을 유지한다.

▶ 자료 추상화

자원에 대해 허용되는 접근모드가 다른 접근제어 정책에서 사용되는 '읽기', '쓰기', '실행' 등 운영체제에서 지원하는 저수준의 권한대신 보다 '입금', '출금' 등 트랜잭션 수준의 보다 의미있고 추상적인 권한을 허용하여, 상업환경에서의 보안 정책의 설계와 개발과정을 용이하게 한다. 그림 1은 역할기반 접근제어 모델의 주요 구성요소와 구성요소간 관계를 나타내고 있다[11].

▶ 역할(role)

역할은 역할기반 접근제어 모델의 핵심요소로서, 주어진 기업 환경에서 정의된 업무의 기능을 바탕으로 정의된 의미적 구조체로 역할의 모든 구성원에게 부여된 권한과 책임으로 구성된다. 각 역할은 해당 역할에서 수행 가능한 권한의 집합으로 구성된다. 역할에 배정된 권한은 조직의 규정이나 규칙에 의해 정의되며, 역할에 소속된 사용자

들에게 동일하게 제공된다.

▶ 역할 계층(role hierarchy)

역할계층은 역할에 배정된 권한들 사이에 포함관계가 있는 역할들간의 부분순서(partial order) 관계로서 기업의 권한과 책임 체계와 매우 유사하여 기업의 권한체계를 모델링하는데 매우 유용하다. 역할계층 관계는 반사적(reflexive), 비대칭적(asymmetric), 이행적(transitive) 관계이며 도식화된 역할 계층에서는 사이클이 존재하지 않는다.

▶ 사용자(user)

사용자는 컴퓨터 시스템을 통하여 시스템 내의 정보를 사용하는 객체로서 한 사용자는 한 명의 사람에 대응된다. 그 이유는 역할기반 접근제어 정책이 지원하는 특성중 하나인 임무분리 성질을 만족시키기 위해서는 한 사람이 여러 개의 사용자 식별자를 소유하지 않아야 하기 때문이다.

▶ 권한(permission)

권한은 정보객체에 대해 수행 가능한 접근모드들의 집합으로 구성된다. 허가정보, 접근권한, 특권 등이 권한과 같은 의미로 사용된다. 일반적으로 권한은 권한이 적용되는 자원의 특성에 따라 달라진다. 예를 들어 운영체제 환경에서는 파일, 디렉토리, 입출력 디바이스, 포트 등의 자원에 적용되는 권한은 '읽기', '쓰기', '실행' 등이지만, 데이터베이스 환경에서는 테이블, 투플, 속성, 뷰 등의 자원에 적용되는 권한은 'select', 'update', 'insert', 'delete' 등으로 두 환경사이에 사용되는 권한의 종류와 의미에 차이가 있다.

▶ 세션(session)

세션은 한 사용자와 여러 개의 역할들로 구성된 집합으로 표현될 수 있으며, 사용자는 세션을 통하여 자신에게 배정된 역할들 중의 일부 또는 전체를 수행할 수 있다. 특히, 세션과 결합된 역할들 중에서 사용자에 의해 현재 실행중인 역할들을 활성 역할(active role)이라 하며, 활성 역할은 세션의 사용자에 배정된 역할들의 집합에 포함되는 특성을 만족해야 한다. 따라서 세션을 수행하는 사용자는 세션에 의해 활성화된 역할들에 포함된 권한들만을 실행할 수 있으므로 역할기반 접근제어 정책이 지원하는 최소권한 원칙을 지원하는 실질적인 기능을 제공하게 된다.

▶ 사용자 배정(user assignment)과 권한 배정(permission assignment)

사용자 배정, 권한 배정은 다대다 관계이며 역할기반 접근제어 모델에서 매우 중요한 구성요소이다. 역할기반 접근제어 모델의 가장 특징중의 하나는 사용자가 정보 객체들에 대해서 실행할 수 있는 연산들을 직접 사용자에게 부여하는 대신 조직의 업무 수행에 필요한 역할과 각 역할에 대한 권한을 배정하고(권한배정), 사용자는 해당 역할에 구성원이 됨으로써(사용자 배정) 정보객체에 대한 원하는 연산을 수행하도록 하는 것이다. 이러한 방법은 사용자와 정보 객체의 수가 많은 일반 기업 환경에서 권한의 관리를 매우 용이하게 수행할 수 있는 장점을 제공한다.

▶ 제약조건(constraints)

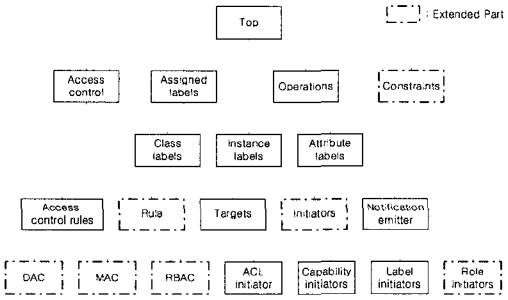
제약조건은 위에서 정의된 모든 구성요소들에 대하여 적용될 수 있으며, 각 구성요소가 가지는 특성을 제한사항이나 조건 등

을 기술한다. 제약조건의 예로서는 임무분리, 한 역할에 할당될 수 있는 최대 사용자 수(cardinality), 선수 역할(prerequisite roles), 시간제약사항(temporal constraint) 등이 있다[3].

3. 망관리 객체의 역할기반 접근제어 모델

ITU-T X.741과 X.812에서는 접근 제어 관리 기능에 대한 정의와 함께 관리 정보 및 연산에 대한 접근을 제어하기 위한 모델에 대하여 기술하고 있다. 그리고 접근 제어 정책에 따라서 접근을 허용하거나 제한을 하는데 사용되는 관리 객체 및 속성들을 정의하고 있다[16,17]. 그러나 이러한 관리 객체를 기술하기 위한 가이드라인인 GDMO는 관리 객체의 구조와 속성 등을 포함한 대부분의 정적특성을 적절히 표현하고 있는데 반해, 동적 특성을 표현하는 방법으로는 자연어를 이용하고 있다. 특히 관리객체의 접근권한을 집행하고 결정하는 접근제어 집행함수와 접근제어 결정함수에 대한 세부적인 속성과 행위 등이 기술되지 않다. 이 때문에 동작과정에 대한 절차 등에 대한 관리객체의 모든 특성을 완전하게 표현하지 못하고 있으며, 이는 결국 접근제어에서의 동작과정에 직접적인 영향을 미치며 규칙에 대한 타당성 및 보안검증은 물론 시스템 구현시에도 많은 어려움이 있다.

본 논문에서는 역할기반 접근제어 정책을 지원할 수 있도록 관리객체 클래스 계층구조에 'Role initiators 관리객체를 확장하였다(그림2 참조). 또한, 관리자에 의해 설정되어 지는 정적인 시간속성 외에 관리 정보에 대한 접근 권한을 수행하는 동적인 시간지원 기능을 갖는 선행조건(PRECOND)에 관련된 제약사항 그리고 제약사항에 위배되었을 시 관리자에게 보고되어야 할 통지에 대해 기술하였다. 그리고, 망관리 정보베이스에 대해 동적 시간지원 제약사항과 역할기반 접근제어 정책을 지원하는 시스템 구조를 제시하였다. 이 시스템에서는 BDL에 기반 하여 접근제어의 집행과 결정함수에 관련된 모듈과 제약사항 그리고 각 역할들의 활성화 전이과정을 기술하므로써 동적 특성을 체계적이고 명확하게 표현하였다.



(그림 2) 확장된 관리객체 클래스 계층구조

3.1 role initiators

'initiator' 관리 객체 클래스는 관리 연산에 대한 허용 가능한 initiators에 대해서 정의한다. 'role initiator' 는 역할기반 접근제어 정책을 지원하기 위해 추가된 것으로써, role initiator가 관리 객체들에게 역할을 부여하고 주어진 역할에 따른 접근 제어를 할 수 있도록 함으로써 효율적 권한관리, 역할의 계층구조에 의한 상속관계, 역할에 따른 최소 권리 부여, 역할에 의한 임무분리, 그리고 역할 클래스에 의한 권한관리 등 기존의 자율적 접근 제어보다는 안전한 정보 흐름을 보장하고 강제적 접근제어 보다는 융통성 있는 접근제어를 보장할 수 있다. 표 1에서는 통신망 관리 연산을

역할기반 접근제어 정책의 역할에 맞추어 일반 연산과 관리 연산 역할로 세분하였다.

(표 1) 관리연산 역할

연산 도메인	역 할	권한 집합
일반 연산	Read Role(R-Role)	Get, Filter..
	Write Role(W-Role)	Replace, Replace With Default, Add Member..
관리 연산	No Read/Write(M-Role)	Create, Action..

3.2 동적 시간지원 제약사항

ITU-T X.746 표준 권고안에는 시간관리 객체의 활동을 제어하는 스케줄링 관리객체가 정의되어있다. 본 논문에서는 운영관리 및 구현의 일관성을 기하고자 스케줄링 관리객체 권고안 내에 동적인 시간지원 제약사항을 추가한다. 다시 말해, 스케줄러 관리객체 클래스에 동적인 시간지원 속성을 지원할 수 있는 Dynamic Scheduler관리 객체 클래스와 Dynamic Operation Scheduler 관리 객체 클래스를 정의한다. 그리고 Dynamic Scheduler 관리 객체의 필수 package로, 특정관리자에 대한 관리객체의 접근 모드에 따라 해당 관리자의 접근 여부가 결정되는 dependencyScheduling Package와 관리자가 관리객체에 접근요청을 시작하는 시점을 기준으로 종료시점을 부여하는 timestampSchedulingPackage를 정의하였다.

▶ dependencySchedulingPackage

특정 관리자에 대한 관리객체의 접근 모드에 따라 해당 관리자의 접근여부가 결정되는 'dependencySchedulingPackage'를 정의하였다. 여기서 적용되는 'dependencyMode'에는 'UNLESS',' WHENEVERNOT',' WHENEVER',' ASLONGAS' 모드가 있다. 이에 해당하는 규칙을 정의하면 다음과 같다.

[규칙 1] 관리자 mi와 관리자 mj의 실제 관리객체 moi를 관리하는 대리자 ai의 접근모드 p의 연산은 dependencyMode에 따라 결정된다.

⇨ (mi, ai, moi, p) <dependencyMode> (m$_j$, ai, moi, p)

단, mi, mj ∈ M : 관리자 (i ≠ j), ai ∈ A : 대리자,
moi ⊆ MO : 관리객체, p ⊆ P : ACCESS_MODE

▶ timestampSchedulingPackage

관리자가 관리객체에 접근 요청을 시작하는 시점을 기준으로 하여 종료시점을 부여하는 timestamp 개념을 도입하여 정보의 안전성을 도모하는 'timestampSchduling Package" 를 정의하였다.

[규칙 2] 관리자 mi와 실제 관리객체 moi를 관리하는 대리자 ai의 접근모드 p의 연산은 타임스템프 Ti 동안에만 수행된다.

⇨ (mi, ai, moi, p, Ti)

단, mi ∈ M : 관리자, ai ∈ A : 대리자,
moi ⊆ MO : 관리객체,
p ⊆ P : ACCESS_MODE,
Ti ∈ N ∪ ∞ : EXPIREDTIME

3.3 위반 통지 타입 (Violation Notification Type)

권고안에서는 다양한 관리객체 클래스에서 이용될 수 있는 통지 타입을 정의하고 있는데, 각각의 통지 타입에서는 관리 프로토콜상에서 전송되어지는 통지 데이터의 구조와 통지에 따른 행위, 그리고 통지결과

데이터의 구조 그리고 객체 식별자 값의 할당 등을 정의하고 있다. 본 논문에서 제시한 동적 시간지원 제약사항에 대한 규칙을 수행한 후 실행되어지는 위반사항에 대한 통지를 integrity vioaltion, operational violation, security service or mechanism violation으로 나누어 분류하였다.

integrity violation 통지 타입의 의미는 정보가 부정적으로 수정되었거나 삽입, 삭제되는 등의 잠재적인 정보방해와 같은 정보의 흐름이 발생할 경우 보고되어지며 Operational Violation 통지타입의 의미는 요청된 서비스의 항목이 이용불가능, 기계 이상 또는 부적절한 서비스 호출 등으로 서비스가 불가능할 경우 보고되어진다. 그리고 security service or mechanism violation은 보안 서비스나 메커니즘에 의해 감지된 보안 공격에 의해 생성된 보고에 이용되는 통지타입이다. 이들 violation 통지에 관련된 속성에는 securityAlarmCause, securityAlarmSeverity, SecurityAlarmDetector, ServiceUser 등의 공통적인 속성 정보를 갖는다[18].

```
SecurityAlarmCause ::= OBJECT IDENTIFIER
SecurityAlarmSeverity ::=
    PerceivedSeverity(indeterminate | critical |
                      major | minor | warning)
SecurityAlarmDetector ::=
    CHOICEE {mechanism [0] OBJECT
        IDENTIFIER, object[1] ObjectInstance,
        application[2] AE-title }
ServiceUser ::= SEQUENCE {
        identifier  OBJECT IDENTIFIER,
        details ANY DEFINED BY indentifier }
```

(그림 3) 통지 관련 속성 타입

3.4. 접근 제어 시스템의 설계

3.4.1 BDL개요

본 논문에서는 관리 객체의 행위 특성을 체계적이고 정형적으로 표현하기 위하여 프로그래밍 언어 형태의 관리객체 기술언어인 BDL(Behavior Description Language)을 이용한다[19]. BDL은 다음과 같은 구성요소를 사용하여 표현한다.

▶ EVENT

관리객체의 동작 절차의 결정요소로서 통신망 관리자로부터 전달되는 관리객체에 대한 관리연산의 실행요청과 다른 관리객체로부터 전달되는 통지 또는 사건 보고 등이 있다.

▶ PRECOND, INVARIANTS

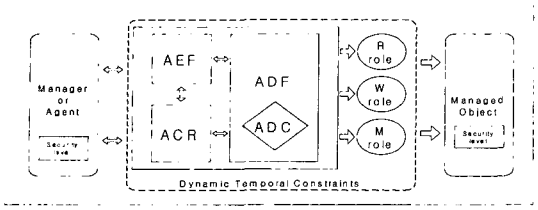
선행조건은 관리연산의 정상적인 수행이나 통지가 발생되기 위한 논리 조건을 표현하고 불변조건은 관리 객체의 동작 전체 과정에서 지켜져야 하는 조건을 표현한다. 선행조건과 불변조건은 참, 거짓을 결정할 수 있는 논리식 형태로 표현된다.

▶ PROCEDURE

관리객체로 전달되는 외부요인이나 관리객체 내부의 속성값, 조건부 패키지의 선행조건 등을 이용한 ECA 규칙을 기반으로 작성된다. 관리객체에 대한 사건(EVENT)이 발생하면, 발생된 사건의 처리여부를 결정(PRECOND)하고 사건에 대한 적절한 조치(ACTION)를 수행하는 구조를 가진다.

3.4.2 시스템 구조

그림 4는 망관리에 필요한 모든 정보들을 저장하고 있는 개념적인 정보 저장소인 망관리 정보베이스에 대해 동적 시간지원 제약사항을 가진 역할기반 접근제어 정책을 지원하는 시스템 아키텍쳐이다.



(그림 4) 시스템 구조

먼저 AEF(Access control Enforcement Function) 모듈은 관리자로부터 호출된 관리 연산을 받아서 관리 객체에 접근하여 관리 연산을 수행하는 모듈로서, 관리 객체에 접근하기 위하여 먼저 ADF(Access control Decision Funtion)모듈에 접근 허용 여부를 의뢰한다. 또한 관리 객체에서 발생한 사건 보고에 대하여도 마찬가지의 기능을 수행한다.

ADF 모듈은 접근 제어 수행 모듈로부터 넘겨받은 접근 제어 정보를 ACR(Access control Rule) 모듈에 의해 접근 허용 여부를 결정하여 접근 제어 수행 모듈에게 통보하여 주는 역할을 수행한다. 그리고 접근 제어 결정을 위하여 필요한 모든 정보를 제공하고 변경된 접근제어 정보들을 첨가, 삭제 그리고 수정하는 역할을 수행하므로서 관리자나 에이전트가 요청한 네트워크 관리 역할을 수행할 수 있는지 없는지에 대한 판단을 결정한다. 또한 이 모든 모듈은 3.2에서 기술한 동적 시간지원 제약사항을 지원함으로써, 연관된 제약사항과 속성이 만족하는 한 관리객체 정보를 보다 안전하게 유지할 수 있게 한다.

본 논문에서는 R-role, W-role, M-role 등의 망 관리 역할의 허용여부를 판단하기 위해 4가지의 점검 규칙을 이용하여 접근제어를 시행하므로서 보다 망 관리객체의 무결성을 보장하도록 하였다.

[규칙 3] 주체의 집합 S에 해당 주체 S가 속하고, 역할의 집합 R에 읽기 전용 역할 R_r이 속할 때, 주체 S가 읽기 전용 역할(R_r)에 배정되기 위해서는 주체의 인가등급이 읽기 전용 역할(R_r)의 최소 보안등급에 지배되어야 한다.

⇨ $\forall S \in S, \forall R_r \in R$

RoleAssign(S, R_r) ⇒ $\lambda(S) \leq$ r-level(R_r)

[규칙 4] 주체의 집합 S에 해당 주체 S가 속하고, 역할의 집합 R에 쓰기 전용 역할 R_w이 속할 때, 주체 S가 쓰기 전용 역할(R_w)에 배정되기 위해서는 주체의 인가등급이 쓰기 전용 역할(R_w)의 최대 보안등급을 지배하여야 한다.

⇨ $\forall S \in S, \forall R_w \in R$

RoleAssign(S, R_w) ⇒ $\lambda(S) \geq$ w-level(R_w)

[규칙 5] 주체의 집합 S에 해당 주체 S가 속하고, 역할의 집합 R에 읽기쓰기 역할 R_{rw}이 속할 때, 주체 S가 읽기쓰기 역할(R_{rw})에 배정되기 위해서는 읽기 전용 역할(R_r)의 최소 보안등급이 쓰기 전용 역할(R_w)의 최대보안등급을 지배하여야 하고, 주체의 인가등급이 읽기 전용 역할(R_r)의 최소 보안등급은 지배하고 쓰기 전용 역할(R_w)의 최대 보안등급에는 지배되어야 한다.

⇨ $\forall S \in S, \forall R_{rw} \in R$

RoleAssign(S, R_{rw}) ⇒ r-level(R_r) ≥ w-level(R_w) AND $\lambda(S) \leq$ r-level(R_r) AND $\lambda(S) \geq$ w-level(R_w)

[규칙 6] 주체의 집합 S에 해당 주체 S가 속하고, 역할의 집합 R에 관리 역할 R_m이 속할 때, 주체 S가 관리 역할(R_m)에 배정되기 위해서는 읽기 전용 역할(R_r)의 최소 보안등급과 쓰기 전용 역할(R_w)의

최대보안등급이 같아야 한다. w-level(R_w)

⇨ $\forall S \in S, \forall R_m \in R$

RoleAssign(S, R_m) ⇒ r-level(R_r) =

```
rule MANAGED OBJECT CLASS
DERIVED FROM accessControl;
CHARACTERIZED BY rulePackage PACKAGE
BEHAVIOUR ruleBehaviour BEHAVIOUR
DEFINED AS
EVENT : AccessControlEnforcementEvent
accessControlObject : accessControlObjectName;
INVARIANTS :
            startTime <= stopTime;
PRECOND:
        administrativeState == unlocked AND
        operationalState == enabled AND
        availabilityStatus != Offduty;
PROCEDURE:
if ( NOT (EXISTS durationPackage AND
currentTime VALID IN [startTime, stopTime] )
OR (EXISTS dailySchedulingPackage AND
currentDay VALID IN [startDay, stopDay] )
OR (EXISTS weeklySchedulingPackage AND
currentWeek VALID IN [startWeek, stopWeek] )   )
then
emit timeDomainViolation notification;
abort;
if ( (NOT( (delegatedManagedObject
UNLESS sourceManagedObject)
 OR (delegatedManagedObject WHENEVERNOT
sourceManagedObject)
 OR (delegatedManagedObject ASLONGAS
sourceManagedObject)
 OR (delegatedManagedObject WHENEVER
sourceManagedObject) ) ) AND
( ( timestampedExpiredTime != NULL ) AND
( OVER timestampedExpiredTime ) ) ) then
emit AccessControlDecisionEvent notification;
else emit integrity Violation notification;
abort;
endif;
if (enforcementAction == allow) then
"access is permitted";
validAccessAttempts = validAccessAttempts + 1;
"send to a security audit trail log";
emit usageReport notification;
if (enforcementAction == deny with response) then
"access is denied";
invalidAccessAttempts = invalidAccessAttempts + 1;
"send to a security audit trail log";
emit usageReport notification;
emit operational violation notification;
endif;
endif;
END;
```

a. 동적 시간지원 제약사항과 통지 타입

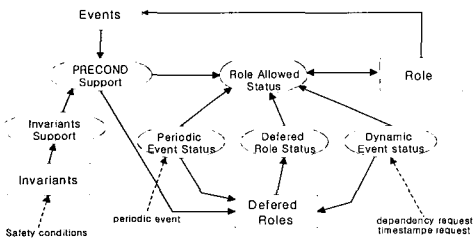
```
EVENT : AccessControlDecisionEvent
 accessControlObject : accessControlObjectName;
PRECOND:
   administrativeState == unlocked AND
   operationalState == enabled AND
   availabilityStatus != Offduty;
PROCEDURE:
switch(Role) {
case R-Role : if r-level(Rr) DOMINATES
initiator-level then     enforcementAction = allow
else    enforcementAction = deny with response ;
endif;
case W-Role : if initiator-level DOMINATES
w-level(Rw) then   enforcementAction = allow
else    enforcementAction = deny with response ;
endif;
case RW-Role : if initiator-level(Rr) DOMINATES
target-level(Rw) then
if (r-level(Rr) >= w-level(Rw) ) AND
(initiator-level <= r-level(Rr) ) AND
(initiator-level >= w-level(Rw) ) then
 enforcementAction = allow
else    enforcementAction = deny with response ;
endif;
endif;
case  M-Role : if initiator-level(Rr) EQUALS
target-level(Rw) then
 enforcementAction = allow
else    enforcementAction = deny with response ;
endif;
END;
```

b. ADC(Access Decision Check) 부분

(그림 6) BDL을 이용한 관리객체 클래스의 동작특성표현

3.4.3 역할 활성화 전이 절차

그림 5는 동적 시간지원 제약사항을 갖는 역할기반 접근제어 시스템에서 역할의 활성화 과정을 설명한 역할 활성화 전이 절차를 보여준 것이다. 제안된 역할 접근제어 정책은 역할의 활성화 과정이 즉시 될 수도 있고 유한적인 시간 간격에 의해 연기될 수 있다. 더욱이 주기적인 사건을 발생시키는 정적 시간속성이나 타임스탬프에 의한 동적 시간지원 제약사항에 의해 주어진 시간 조건을 만족한다면 언제든지 역할이 활성화될 수 있다. 그림 5에서 다양한 개체간의 상호작용은 화살표에 의해 표현되어지며, 사각형은 역할의 수행에 의해 관리 객체의 상태가 변할 수 있는 부분을 나타내고 원형은 역할의 상태를 검사하는 부분을 나타내고 있다.



(그림 5) 역할 활성화 전이 절차

3.4.4 동작특성 표현

그림 6은 BDL을 이용하여 역할기반 접근제어 관리 객체의 동적 특성 중 접근 제어의 역할영역과 동적 시간지원 제약사항과 위반 통지에 근거한 관리 객체 접근을 제어하는 과정과 수행절차를 정형적으로 표현하고 있다. 그림 6의 a는 동적 시간지원 제약사항과 위반 통지 타입을 기술한 것이며, b는 ADC 모듈을 BDL로 기술한 부분이다. 아래 그림에서 이태릭체로 표현된 부분은 동적 시간지원 제약사항과 통지타입별 정의 부분과 연동한 접근제어 기능을 나타낸다.

4. 결 론

상호 독립적으로 운영되는 통신망들이 상호연동 됨에 따라 전체적인 통신망의 규모가 점점 커지고 복잡해지고 있으며 다양한 사용자들로 인해 관리객체를 저장 관리하는 관리정보베이스에 대한 보안이 필수적인 요소가 되었다. ITU-T X.741과 X.812 표준안에서는 접근제어에 관한 전반적인 관리객체 클래스의 상속계층구조와 관리객체 상호관계를 표현하고 있으며 임의적 접근제어 모델에 해당하는 접근제어리스트와 능력리스트에 대한 관리객체 그리고 강제적 접근제어 모델에 해당하는 보안등급에 관한 관리객체와 속성들을 표현하고 있다. ITU-T 표준안에서는 아직까지 실세계 환경에서 다양하게 적용될 수 있는 역할기반 접근제어에 대한 관리객체와 관련된 속성 특히, 접근제어모델내의 함수인 접근제어 결정함수와 접근제어 집행함수에 대한 세부적인 동작과정이 기술되어 있지 않다. 또한, 관리객체를 기술하는 GDMO는 관리 객체의 구조와 속성 등을 포함한 대부분의 정적특성을 적절히 표현하고 있는데 반해, 동적특성을 표현하는 방법으로는 자연어를 이용하고 있다. 이 때문에 동작과정에 대한 절차 등에 대한 관리객체의 모든 특성을 완전하게 표현하지 못하고 있으며, 이는 결국 접근제어에서의 동작과정에 직접적인 영향을 미치며 규칙에 대한 타당성 및 보안검증은 물론 시스템 구현 시에도 많은 어려움이 있다.

본 논문에서는 역할기반 접근제어 정책을 지원할 수 있도록 관리객체 클래스 계층구조에 'Role initiators 관리객체를 확장하

였다. 또한, 관리 프로세스에 의해 설정되어지는 정적인 시간속성 외에 실질적으로 관리 정보에 대한 접근 권한을 수행하는 동적인 시간지원 기능을 갖는 선행 조건에 관련된 제약사항 그리고 제약사항에 위배되었을 경우 관리자에게 보고되어야 할 위반 통지 타입들에 대해 기술하였다. 그리고 망관리 정보베이스에 대해 동적 시간지원 제약사항과 역할기반 접근제어 정책을 지원하는 시스템 구조를 제시하였으며 BDL에 기반하여 접근제어의 집행과 결정함수에 관련된 모듈과 제약사항 그리고 각 역할들의 활성화 전이과정을 기술하므로써 동작 특성을 보다 체계적이고 명확하게 표현하였다.

참 고 문 헌

[1] Elisa Bertino, Claudio Bettini, Pierangela Samarati, "A Discretionary Access Control Model with Temporal Authorizations," IEEE New Security Paradigms Workshop, 8, 1994.

[2] Jonathon E. Tidswell, Trent Jaeger, "Integrated Constraints and Inheritance in DTAC", ACM Workshop on Role-Base Access Control, July 2000

[3] Jonathon E. Tidswell, Trent Jaeger, "An Access Control Model for Simplifying Constraint Expression", ACM Conference on Computer and Communications Security, 2000

[4] Jonathon E. Tidswell, Jonh M. Potter, "A Dynamically Typed Access Control Model", In Proceedings of the Third Australasian Confernce on Information Security and Privacy, July 1998.

[5] Masum X. Hasan, "An Active Temporal Model for Network Management Databases," Proceedings of the 4th International Symposium on Integrated Network Management, 1995

[6] Oliver Festor, Georg Zorntlein, "Formal Description of Managed Object Behavior - A Rule Based Approach," Proceedings of the IFIP TC6/WG6.6 3rd International Symposium on Integrated Network Management, 1993

[7] Silvana Castano et al., Database Security, Addison-Wesley, 1994, pp18-34.

[8] John Barkley, "Comparing Simple Role Based Access Control Models and Access Control Lists," Proceedings of 2nd Workshop on Role-Based Access Control, August, 1997, pp.127-132.

[9] David F. Ferraiolo, Janet A. Cugini, D. Richard Kuhn, "Role-Based Access Control(RBAC): Features and Motivations," Proceedings of the 11th Annual Computer Security Applications Conferences, December 1995, pp. 241-248.

[10] David Ferraiolo, Richard Kuhn, "Role-Based Access Controls," Proceedings of the 15th NIST-NCSC National Computer Security Conference, October 1996, pp.554-563.

[11] Ravi S. Sandhu, Edward J. Coyne, "Role-Based Access Control Models," IEEE Computer, February 1996, pp.38-47.

[12] Ravi S. Sandhu, "Role Hierarchies and Constraints for Lattice-Based Access Controls," Proceedings of the 4th

European Symposium on Research in Computer Security, September 1996.

[13] Oliver Festor, Georg Zorntlein, "Formal Description of Managed Object Behavior - A Rule Based Approach," Proceedings of the IFIP TC6/WG6.6 3rd International Symposium on Integrated Network Management, 1993

[14] Masum X. Hasan, "An Active Temporal Model for Network Management Databases," Proceedings of the 4th International Symposium on Integrated Network Management, 1995

[15] ITU-T X.722 : "Information Processing Systems - Open Systems Interconnection - Management Information Services - Structure of Management Information Part 4: Guidelines for the Description of Managed Objects", Geneva

[16] ITU-T X.741 : "Information Technology - Open Systems Interconnection - System Management - Objects and Attributes for Access Control "[X741] ITU-T X.741 : "Information Technology - Open Systems Interconnection - System Management - Objects and Attributes for Access Control "

[17] ITU-T X.812 : "Information Technology - Open Systems Interconnection - Security Frameworks For Open System - Access Control Framework "

[18] ITU-T X.721 : "Information Processing Systems - Open Systems Interconnection - Management Information Services - Structure of Management Information Part 2: Definition of Management Information", Geneva

[19] 최은복, 이형효, 노봉남," 망관리 객체의 시간지원 능동특성에 대한 정형적 모델링" , 정보처리학회 논문지, 제 6권 제 9호, 1999.

◐ 저 자 소 개 ◑

최 은 복

1992년 전남대학교 전산학과 (이학사)
1996년 전남대학교 전산학과 (이학석사)
2000년 전남대학교 전산학과 (이학박사)
2001년 순천제일대학 인터넷정보학부 전임강사
2002년 ~ 현재 : 전주대학교 정보기술컴퓨터공학부 전임강사
관심분야 : 통신망관리, 정보보안, 엑티브 네트워크 등
E-mail : ebchoi@jeonju.ac.kr