Journal of Korea Multimedia Society Vol. 7, No. 11, November 2004(pp. 1620-1629)

무결성이 강화된 역할 그래프 모델의 정형적 명세

최은복†, 이형옥††

요 약

접근제어의 목적은 컴퓨팅 자원 및 통신 정보자원 등을 부당한 사용자로부터 사용되거나, 수정, 노출, 파괴와 같은 비합법적인 행위로부터 보호하는데 있다. 대표적인 보안 정책 중에서 Biba 모델은 정보의 무결성을 보장하지만 상업적인 환경에 적용되기는 다소 미흡하며, 역할기반접근제어 정책은 상업적인 측면의 보안정책에 적용이 가능하지만 접근되는 객체의 중요도에 따른 등급이 고려되어 있지 않아 정보의 무결성을 해칠 우려가 있다. 본 논문에서는 기존의 역할 그래프 모델에 무결성 등급을 갖는 사용자와 객체를 배정하여 주체의 무결성 등급과 역할에 관련된 객체의 무결성 등급에 따라 권한을 부여하므로 수많은 접근권한을 관리하는데 융통성을 제공할 뿐 아니라 정보의 무결성을 보장한다. 또한, 세분화된 역할들의 제약조건들을 명세 언어인 Z를 이용해 정형화된 구조로 명확하게 표현함으로써 접근제어정책에 대한 설계 및 구현시 시간과 비용 절감효과를 기대할 수 있다.

A Formal Specification of Role Graph Model Increasing Integrity

EunBok, Choi†, HyeongOk, Lee††

ABSTRACT

The objectives of access control are to protect computing and communication resources from illegal use, alteration, disclosure and destruction by unauthorized users. Although Biba security model is well suited for protecting the integrity of information, it is considered too restrictive to be an access control model for commercial environments. And, Role-Based Access Control(RBAC) model, a flexible and policy-neutral security model that is being widely accepted in commercial areas, has a possibility for compromising integrity of information. In this paper, We present the role graph model which enhanced flexibility and integrity to management of many access permission. Also, In order to represent those rule and constraints clearly, formal descriptions of role assignment rule and constraints in Z language are also given.

Key words: Acess Control(접근제어), Role-Based Access Control(역할기반접근제어), Role Graph(역할 그래프), Z Language(Z언어)

※ 교신저자(Corresponding Author) : 이형옥, 주소 : 전라남도 순천시 매곡동(540-742), 전화 : 061)750-3345, FAX : 061)750-3308, E-mail : oklee@sunchon.ac.kr
접수일 : 2004년 3월 2일, 완료일 : 2004년 6월 17일
† 정회원, 전주대학교 정보기술공학부 조교수
(E-mail : ebchoi@jj.ac.kr)
†† 순천대학교 컴퓨터교육과 조교수
(E-mail : oklee@sunchon.ac.kr)

1. 서 론

대부분의 기업과 정부 기관들이 관련 업무를 정보처리 시스템에 의존함에 따라 인가된 주체나 사용자들에게 자원들이 불법 노출되면 조직 운영이 어렵게 될 수 있다. 따라서 권한이 있는 사용자들에게만 특정 데이터 또는 자원들이 제공되는 것을 보장하기

위해 서로 다른 종류의 접근제어 정책을 구현하기 위해 노력하였다.

접근제어 정책은 1985년 미국 국방성에서 작성된 TCSEC에 의해 크게 강제적 접근제어 정책과 임의적 접근제어 정책의 두가지로 분류된다[11]. 강제적 접근제어 정책은 미리 정의된 엄격한 규칙에 의해 정보가 저장된 객체에 대한 사용자의 접근허가 여부가 결정되는 방식이며, 임의적 접근제어 정책에서는 정보 객체 소유자가 임의적으로 자신이 소유하고 있는 정보객체에 대한 접근허가 여부를 판단하는 특징을 가진다. 또한, Sandhu는 1996년 이 두 정책 모두 실제 기업환경에 적용되기에는 부적합한 특성이 있다고 언급하고 역할기반 접근제어 정책을 제안했다[7].

정보의 비밀성에 기반을 둔 BLP 모델에 의거한 기존의 역할 그래프 모델[8]은 정보의 비밀성보다는 무결성이 강조되는 상업적인 환경에 적용하기에는 미흡하다고 본다. 그래서 본 논문에서는 기존의 역할 그래프 모델[7]에 무결성 등급을 갖는 사용자와 객체를 배정하여 주체의 무결성 등급과 역할에 관련된 객체의 무결성 등급에 따라 권한을 부여하므로 수많은 접근권한을 관리하는데 융통성을 제공할 뿐 아니라 정보의 무결성을 보장한다. 또한, 시스템 개발단계에서 비정형적인 요구사항 기술로 인한 부정확성, 모호성, 불완전성에 대한 문제를 해결하기 위해 세분화된 역할들의 제약조건들을 명세 언어인 Z를 이용해 정형화된 구조로 명확하게 표현함으로써 접근제어정책에 대한 설계 및 구현시 시간과 비용 절감효과를 기대할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 접근제어 정책과 정형화된 정책을 기술하기 위한 명세언어인 Z 언어를 기술하고 3장에서는 역할에 배정되는 사용자와 역할에 의해 수행되는 정보객체의 권한에 무결성 등급을 배정하므로서 정보의 무결성을 보장하는 역할그래프 모델을 제시하였으며 이들에 대한 제약조건들을 Z언어를 사용하여 정형화된 표기법으로 명세하였다.

2. 관련연구

접근제어 정책은 크게 자율적 접근제어(DAC : Discretionary Access Control)정책, 강제적 접근제어(MAC : Mandatory Access Control)정책, 그리고 역할기반 접근제어(RBAC : Role Base Access Control)정책 등이 있다.

2.1 자율적 접근제어 정책

자율적 접근제어 정책은 접근을 요청한 주체가 객체에 대한 권한을 자율적으로 다른 주체에게 권한을 부여하거나 철회할 수 있음을 의미한다. 여기에 해당하는 접근제어 정책에는 접근제어 행렬(Access Control Matrix), 접근제어 리스트(ACL), 능력리스트(Capability List) 등이 있다.[8]

접근제어행렬은 주체에 해당하는 행과 객체에 해당하는 열, 그리고 행과 열이 교차하는 곳은 연산을 나타낸다. 그러나 대규모 시스템에서는 수많은 주체와 객체가 존재하기 때문에 이들간의 연산을 행렬로 기술하기에는 시스템 공간의 낭비를 초래한다. 따라서 이러한 단점을 해결할 수 있는 접근제어 리스트나 능력리스트를 이용한다.

2.2 강제적 접근제어 정책

강제적 접근제어정책은 시스템 관리자에 의해 보안등급이 결정되는 정책이다. 주체가 객체에 대한 권한을 자율적으로 수행하는 자율적 접근제어와는 다른 정책이다. 특히, 강제적 접근제어 정책에서는 주체에 부여되는 등급을 인가등급(Clearance level)이라 하며 객체에 부여되는 등급을 보안등급(Classification level)이라 한다.

강제적 접근 제어 정책을 이용한 대표적인 모델로는 비밀성을 중요시하는 BLP(Bell-LaPadula) 모델과 정보의 무결성을 강조하는 Biba 모델이 있다.

□ BLP model

BLP model은 강제적인 정책에 기반을 둔 데이터 보호를 위한 참조 모델이다. 각 등급은 두 가지 구성요소에 의해 정의되어지는데, 하나는 보안등급이고 다른 하나는 범주의 집합이다. 보안등급은 TS, S, C, U의 4가지 요소로 구성되고 이들은 TS>S>C>U의 관계를 갖는다. 범주의 집합은 요소들의 비계층 구조를 가지는 부분집합으로 정보가 포함되는 조직의 환경에 의해 명명되어진다[7].

□ Biba model

BLP모델은 권한을 갖지 않는 사용자에게 정보가 흘러가는 것을 예방하는 비밀성에 기반을 둔 모델이다. 이 모델은 정보의 비밀성은 보장하지만 등급이

낮은 주체가 등급이 높은 객체의 정보를 변경할 수가 있어 정보의 무결성을 보장하지는 못한다. 이러한 단점을 보완하기 위해 Biba 모델이 제안되었다. 이 모델에서도 주체와 객체의 보안등급에 의해 정책이 수행되는데 특히 보안등급을 무결성 등급이라 한다. 이 무결성 등급은 크게 두가지로 분류한다. 하나는 Crucial(C), Very Important (VI), Important(I)로 구분되는 무결성 등급이고 다른 하나는 범주의 집합이다. 무결성 등급은 C>VI>I의 관계를 형성하며 범주의 집합은 BLP모델과 마찬가지로 비계층 구조 관계를 갖는다. 사용자를 대신하여 수행하는 프로세스나 접근이 수행되는 객체에게 무결성 등급이 부여된다[7].

이 두가지 모델은 등급과 범주가 각각 $C_1 \geq C_2$ 이고 범주 $S_1 \supseteq S_2$의 관계를 가지면 등급 $L_1=(C_1,S_1)$은 $L_2=(C_2,S_2)$를 지배한다. 만약 등급이 $L_i \geq L_2$ 나 $L_2 \geq L_1$ 의 관계가 모두 아니면 이 두 등급은 비교불가능하다고 말한다.

2.3 역할기반 접근제어 정책

자율적 접근제어 정책은 주체와 객체에 대한 접근제어가 접근제어 행렬이나 접근제어리스트에 등록되어 있어 해당되는 주체에게만 객체의 정보를 제공한다. 하지만 주체가 객체에 대한 권한을 자율적으로 다른 주체에게 권한을 부여하거나 철회할 수 있어 악의적인 목적에 이용될 수 있는 보안상의 취약점이 있다.

강제적 접근제어 정책은 시스템 관리자에 의해 등급이 결정되는 정책으로 모든 주체들이 서로 다른 인가등급(clearence level)을 할당받으며, 모든 객체에게도 다양한 보안등급(classification level)을 부여하여 인가등급이 객체의 보안등급을 지배할 때 접근을 허용한다. 이 정책에는 BLP모델과 Biba모델이 있는데, BLP모델은 정보의 비밀성을 중요시하는 군사분야의 응용에 적합하도록 만들어져서 객체에 대해 엄격하게 제한하기 때문에 일반적인 응용에 적용하기에는 문제점이 있다[2,6]. 이러한 단점을 보완하고 정보의 비밀성보다는 무결성을 보장하며, 상업적인 환경에 적용하도록 만든 모델이 Biba 모델이다. 하지만 이 두 모델들은 한 주체가 어느 한 객체를 접근하지 못하면 자신의 인가등급을 변경하지 않는 한 그 객체와 동일한 보안등급을 갖는 모든 객체에 접근이 허락되지 않는다. 또한 공통적인 기능을 수행하는 다중 사용자들이 객체를 접근할 수 있는 보안 요구사항을 표현하는데는 부적절하다.

역할기반 접근제어 정책은 상업적인 측면의 보안정책을 강화시킬 수 있는 정책으로, 사용자들이 수행하는 공통적인 기능들에 기반을 둔 그룹들인 역할로 구성되며 조직이나 환경에 따라 역할이 자연스럽게 생성되고 재구성될 수 있는 유연성을 갖는다.

이 정책은 역할과 권한, 사용자와 역할, 그리고 역할과 역할의 관계와 같은 역할기반 접근제어 정책 구성요소를 포함하는데 이들 구성요소는 시스템관리자에 의해 직접적으로 구성된다.

많은 기업들이 사용자에게 정보의 소유권을 부여하지 않고 회사나 대리점과 같은 공통적인 역할 수행기관에게 정보의 변경이나 삭제, 첨가 등 연산의 소유권을 부여함으로써 수많은 접근권한을 관리하는데 효율성과 융통성을 제공한다.

역할기반 접근제어 정책의 개념은 다음과 같이 잘 알려진 세 가지 보안 원리를 뒷받침하는데, 첫째, 역할 계층성을 이용하여 작업에 꼭 필요한 최소한의 권한만을 역할에 배정하는 최소권한원칙(least privilege principle), 둘째, 정보의 무결성을 침해하는 사기행위나 부정수단을 유발할 수 있는 작업은 상호배타적인 역할로 지정하여 임무를 분리시켜 수행하는 임무 분리(separation of duty), 마지막으로, 전형적인 운영체제나 시스템에서 사용되어졌던 데이터를 처리하는 read, write, execute 등의 권한 대신에, 다양한 기능을 수행할 수 있고 명령어를 추상화시키는 상업적인 처리 명령어 credit, debit, transfer, create account, delete account 등을 사용하는 데이터 추상화(data abstraction)이다[4].

2.4 Z 명세 언어

시스템을 개발하는데 있어서 개발을 의뢰한 고객 또는 실제 사용할 사용자의 요구사항과 다르게 구성되는 오류중 하나는, 시스템의 개발단계에서 비정형적인 요구사항 기술로 인한 부정확성, 모호성, 불완전성, 이해오류 등에 기인한다. 이러한 문제에 대한 방안으로 제시된 Z언어와 같은 명세 언어는 의미가 명확한 수학적 기호를 이용하고, 집합, 관계, 함수 등을 가지는 집합론에 기초를 두고 있다.

정형적 명세 언어는 시스템의 특성을 자세하게 정

의하고 명확하게 기술하는데 사용되는 언어로써 명세에서 기술하는 것은 시스템이 어떻게 수행되는가(How)가 아니고 시스템이 무엇을 하는가(What)을 나타내고 있다. 이러한 명세 언어는 시스템의 오퍼레이션 뿐만 아니라 시스템의 다양한 상태를 나타내는 스키마 구조를 포함한다. 따라서 이러한 정형적 명세 언어를 이용하여 시스템을 정의하고 기술하면 비정형적 언어로 작성한 것에 비해서 많은 이점들을 얻을 수 있는데 이러한 이유들 때문에 정형적 명세 언어인 Z, VDM 등이 개발되었다. Z 명세 언어는 집합론에 기초한 스키마 구조를 이용하여 시스템을 명세화한다. Z 명세언어는 일반적으로 상태스키마(State Schema)와 오퍼레이션 스키마(Operation Schema)를 포함하고 있으며 이들 스키마는 다른 스키마에 의해 참조될 수 있도록 이름이 주어진다[9,10].

3. 무결성을 보장하는 역할 그래프 모델의 정형적 명세

강제적 접근제어 정책은 접근제어 판단의 기준이 되는 인가등급과 보안 등급을 사용자와 정보객체에 부여하고, 접근제어 규칙을 정의하는 기능이 제한된 수의 보안관리자에 의해 관리되므로 시스템의 보안관리가 중앙집중적으로 이루어진다. 그러나 BLP, Biba 모델이 적용되기 위해서는 시스템을 구성하는 사용자와 정보객체에게 계층적 구조를 가지는 보안등급이 일관성있게 부여될 수 있는 환경이여야 하며, 사용자와 정보객체의 수가 많아지고 다양한 보안 특성을 가지게 되는 환경에서는 적용이 용이하지 않는 문제점을 가지고 있다.

역할기반 접근제어 정책은 역할을 기반으로 접근제어 서비스를 제공하는 모델로서 권한이 역할에 부여되고, 사용자는 조직내에서 책임과 자격에 맞는 역할에 할당됨으로써, 기업의 조직구조에 적합하고 정보자원을 효율적으로 관리할 수 있다. 하지만 이 정책은 상위 역할에 배정된 사용자는 하위 역할에 배정된 사용자의 권한을 상속받으므로 권한 남용의 위험성을 갖고 있다. 또한, 역할에 의해 접근되는 객체에 대한 중요도에 따른 등급이 기술되어 있지 않아 해당 역할에 배정된 모든 사용자들에게 객체에 대한 권한이 주어지므로 정보의 비밀성과 무결성을 해칠 우려가 있다.

정보의 비밀성에 기반을 둔 BLP 모델에 의거한 기존의 역할 그래프 모델[3,5]은 정보의 비밀성보다는 무결성이 강조되는 상업적인 환경에 적용하기에는 미흡하다고 본다. 그래서 본 논문에서는 무결성 등급을 갖는 사용자와 객체를 배정하여 주체의 무결성 등급과 역할에 관련된 객체의 무결성 등급에 따라 권한을 부여하므로 상업적인 환경에 적용가능하며 수많은 접근권한을 관리하는데 융통성을 제공할 뿐 아니라 정보의 무결성을 보장한다. 또한, 시스템 개발단계에서 비정형적인 요구사항 기술로 인한 부정확성, 모호성, 불완전성에 대한 문제를 해결하기 위해 세분화된 역할들의 제약조건들을 명세 언어인 Z를 이용해 정형화된 구조로 명확하게 표현함으로써 접근제어정책에 대한 설계 및 구현시 시간과 비용 절감효과를 기대할 수 있다.

3.1 역할그래프 모델

역할 그래프 모델은 사용자, 권한, 그리고 역할의 개념을 사용한다. 권한은 객체에 대한 연산의 집합인 (x, m)으로 구성되는데, x는 객체, m은 x에 대한 접근 모드를 나타낸다. 역할은 권한과 관련되어 명명된 집합인 (r.name, r.pset)으로 구성되며 r.name은 역할 이름, r.pset은 역할에 대한 권한집합을 의미한다. 역할은 역할 그래프의 한 노드를 구성하는데, 만약 R_1.pset $\subseteq R_2$.pset 이면 R_1이 R_2의 하위역할이 된다. 역할 그래프에서는 한 주체가 역할에 배정되면 그 주체는 해당 역할의 모든 권한을 수행할 수 있을 뿐만 아니라 자신의 하위 역할의 모든 권한을 수행할 수 있게 된다[1].

논문에 필요한 기본타입과 이들을 Z 언어를 이용하여 정형적으로 기술한 스키마 명세 내용은 다음과 같다.

□ 기본타입의 정의

[SUBJECT, OBJECT, ROLE, SEC_LEVEL, ACCESS_MODE]

- SUBJECT : 사용자(주체)의 집합
- OBJECT : 객체의 집합
- ROLE : 역할의 집합
- SEC_LEVEL == { **i, vi, c**} <무결성 등급의 집합>
- ACCESS_MODE == {**read, write**} <접근모드의 집합>

□ 기본 스키마 명세 정의

READ_PRIV_SET
object : P OBJECT
access_mode : ACCESS_MODE
---
access_mode = **read**

<읽기 권한 집합>

WRITE_PRIV_SET
object : P OBJECT
access_mode : ACCESS_MODE
---
access_mode = **write**

<쓰기 권한 집합>

ROLE
r-level : SEC_LEVEL
r-scope : READ_PRIV_SET
w-level : SEC_LEVEL
w-scope : WRITE_PRIV_SET

<역할>

LEVEL_TO_INT
level_to_int : SEC_LEVEL → N

∀s : SEC_LEVEL, l : N •
(s = i ⇒ l = 1) ∨
(s = vi ⇒ l = 2) ∨
(s = c ⇒ l = 3)

<등급의 정수화>

INT_TO_LEVEL
int_to_level : N → SEC_LEVEL

∀s : SEC_LEVEL, l : N •
(l = 1 ⇒ s = i) ∨
(l = 2 ⇒ s = vi) ∨
(l = 3 ⇒ s = c)

<정수의 등급화>

INIT_R_LEVEL
r-level : SEC_LEVEL
r-scope : P READ_PRIV_SET
r-temp : N
level_to_int : SEC_LEVEL → N
int_to_level : N → SEC_LEVEL

∀obj ∈ r-scope.object
r-temp = min[level_to_int(obj)]
r-level = int_to_level(r-temp)

<읽기 등급 초기화>

INIT_W_LEVEL
w-level : SEC_LEVEL
w-scope : P WRITE_PRIV_SET
w-temp : N
level_to_int : SEC_LEVEL → N
int_to_level : N → SEC_LEVEL

∀obj ∈ w-scope.object
w-temp = max[level_to_int(obj)]
w-level = int_to_level(w-temp)

<쓰기 등급 초기화>

3.1.1 read only 역할

역할 R의 권한집합에서 read 권한을 갖는 모든 객체의 집합을 r-scope(R)라고 write 권한을 갖는 모든 객체의 집합을 w-scope(R)이라고 하자. 또한, r-scope(R)에 존재하는 모든 객체 중에서 최소의 무결성 등급을 갖는 등급을 해당 역할 R의 r-level(R)이라 정의하고 w-scope(R)에 존재하는 모든 객체 중에서 최대의 무결성 등급을 갖는 등급을 해당 역할 R의 w-level(R)이라 정의한다.

read-only 역할(R_r)의 경우 w-scope의 값이 공집합이다. 만약 r-scope의 모든 객체의 등급이 모두 같은 무결성 등급을 갖는 경우, 이 역할은 Biba 모델의 simple integrity 정책이 적용된다. 만약, r-scope의 모든 객체들이 서로 다른 무결성 등급을 가질 경우, 주체의 무결성 등급이 해당 역할의 최소 무결성 등급에 해당하는 r-level(R)에 지배되어야 한다. 만약 그렇지 않으면 해당 주체는 자신보다 등급이 낮은 객체를 읽게 되어 정보의 무결성을 보장받지 못하게 된다. 왜냐면, 자신보다 낮은 객체를 읽어 자신의 등급에 쓸 경우 낮은 정보가 상위 등급으로 흐를 수 있기 때문이다.

이의 내용을 제약조건과 정형적 명세로 표현하면 다음과 같다.

[제약조건 1] $\forall S \in S, \forall R_r \in R$
$RoleAssign(S, R_r) \Rightarrow \lambda(S) \leq r\text{-}level(R_r)$

READ_ONLY_ROLE_ASSIGN
s : SEC_LEVEL
level_to_int : SEC_LEVEL → N
r-level : SEC_LEVEL
---
level_to_int(s) ≤ level_to_int(r-level)

3.1.2 write only 역할

다음으로 write only 역할(R_w)의 경우 r-scope의 값이 공집합이므로, 만약, w-scope의 객체 등급이 모두 같은 무결성 등급을 갖는 경우, 이 역할은 Biba 모델의 integrity *-property 정책이 적용된다. 만약, w-scope의 모든 객체들이 서로 다른 무결성 등급을 가질 경우, 주체의 무결성 등급이 해당 역할의 최대 무결성 등급에 해당하는 w-level(R)을 지배하여야 한다. 만약 그렇지 않으면 주체가 자신보다 높은 등급이 객체를 쓰게 되어 정보의 무결성을 보장받지 못하게 된다.

이의 내용을 제약조건과 정형적 명세로 표현하면 다음과 같다.

[제약조건 2] $\forall S \in S, \forall R_w \in R$
RoleAssign(S, R_w) $\Rightarrow \lambda(S) \geq$ w-level(R_w)

WRITE_ONLY_ROLE_ASSIGN
s : SEC_LEVEL
level_to_int : SEC_LEVEL → N
w-level : SEC_LEVEL
---
level_to_int(s) ≥ level_to_int(w-level)

3.1.3 read-write 역할

다음으로 read-write 역할(R_{rw})을 고려하자. 이 경우는 여러 가지 경우의 가능성이 존재한다. 먼저 r-scope와 w-scope가 하나의 등급에 모두 존재하는 경우, 기존의 Biba 정책인 simple integrity 정책과 Integrity *-property를 적용한다.

다음은 w-scope와 r-scope가 하나이상의 무결성 등급을 포함하면서 정확히 하나의 등급에 겹쳐있다면 이 역할은 겹쳐있는 등급을 갖는 객체에만 배정이 가능하다. 그러나 둘 이상의 등급이 겹쳐 있다면 정보의 무결성을 해치므로 이 역할에는 배정이 불가능하다. 만약, r-scope와 w-scope가 겹치는 부분이 존재하지 않는다면 두 scope 사이의 등급에 해당하는 주체에 역할을 배정할 수 있다.

r-scope가 w-scope보다 상위에 있는 경우에는 신뢰하는 주체나 비신뢰하는 주체든 상관하지 않고 해당 정책에 부합되는 주체에만 역할을 배정할 수 있다. 하지만 w-scope가 r-scope보다 상위에 존재하는 경우에 신뢰할 수 없는 주체는 Biba 정책을 위반하는 read-down, write-up 경우가 발생하므로 배정할 수 없다. 단, 신뢰주체에는 Biba 정책의 Integrity *- propety를 적용하여, 만약 주체의 등급이 w-level(R)을 지배하면 해당 역할에 배정할 수 있다.

이와 같은 내용을 기반으로 제약조건3과 제약조건 3′, 그리고 정형적 명세를 정의하였다.

[제약조건 3] $\forall S \in S, \forall R_{rw} \in R$
RoleAssign(S, R_{rw}) $\Rightarrow$ r-level(R_r) $\geq$ w-level(R_w) AND $\lambda(S) \leq$ r-level(R_r)
AND $\lambda(S) \geq$ w-level(R_w)

[제약조건 3′] $\forall S \in St \ \forall R_{rw} \in R$
RoleAssign(S, R_{rw}) $\Rightarrow \lambda(S) \geq$ w-level(R_w)

READ_WRITE_ROLE_ASSIGN
s : SEC_LEVEL
level_to_int : SEC_LEVEL → N
r-level, w-level : SEC_LEVEL
---
level_to_int(r-level) ≥ level_to_int(w-level)
∧
level_to_int(r-level) ≥ level_to_int(s)
≥ level_to_int(w-level)

READ_WRITE_ROLE_ASSIGN(TRUST)
s : SEC_LEVEL
level_to_int : SEC_LEVEL → N
w-level : SEC_LEVEL
---
level_to_int(s) ≥ level_to_int(w-level)

이들에 대한 역할배정규칙을 정의하면 다음과 같다.

□ 역할 배정 규칙

[read-only 역할의 경우]
RoleAssign[S, R_r] =
True : if Dominate[r-level(R_r), λ(S)]
False : Otherwise

[write-only(append) 역할의 경우]
RoleAssign[S, R_w] =
True : if Dominate[λ(S), w-level(R_w)]
False : Otherwise

[read-write 역할의 경우]
RoleAssign[S, R_{rw}] =
True : if [Dominate(r-level(R_r), w-level(R_w)]

{
if
Equal[r-level(R_r), w-level(R_w)] == λ(S) OR
[{r-scope(R_r) ∩ w-scope(R_w)} == ∅
AND r-level(R_r) ≥ λ(S) ≥ w-level(R_w)]
}
else if [S ∈ St] AND Dominate[St, w-level(R_w)]
False : Otherwise

3.2 역할 그래프 도식화

먼저 역할의 r-level과 w-level이 비교 가능한 무결성 등급을 가진 경우의 역할을 고려하자. 이때의 r-level은 r-scope 객체중에서 최소의 등급을 취하고 w-level은 w-scope 중에서 최대의 등급을 취한다. 역할 그래프의 정의에 의해 R_1의 r-scope는 R_2의 r-scope에 모두 포함이 된다.

read 역할의 경우, R_2의 r-level은 R_1의 r-level과 같은 등급을 가져야 한다. 그렇지 않으면 역할 그래프 정의의 포함 개념에 의해 read up 정책이 위배된다. 그림 1은 비교가능한 read 역할을 보였다.

[제약조건 4] $R_1 \rightarrow R_2$ 일 때, read 역할의 경우(비교가능)

r-level(R_2) = r-level(R_1)

READ_ROLEGRAPH_ASSIGN(COMP.)
senior_role : ROLE
junior_role : ROLE
---
senior_role.r-level = junior_role.r-level

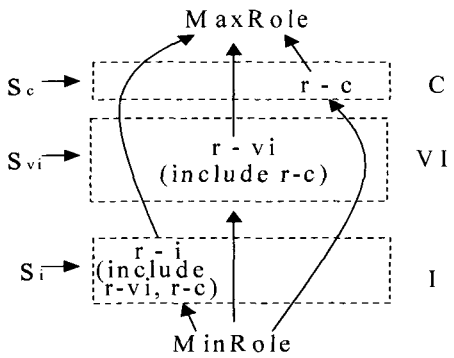

그림 1. read 역할(비교가능)

R_2의 w-level이 R_1의 w-level을 지배하여야 하는 write 역할을 제약조건 5와 그림 2에서 기술하고 도식화하였다.

[제약조건 5] $R_1 \rightarrow R_2$ 일 때, write 역할의 경우(비교가능)

w-level(R_2) ≥ w-level(R_1)

WRITE_ROLEGRAPH_ASSIGN(COMP.)
senior_role : ROLE
junior_role : ROLE
levle_to_int : SEC_LEVEL → N
---
level_to_int(senior_role.w-level) ≥
level_to_int(junior_role.w-level)

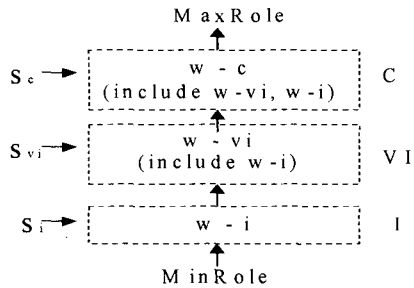

그림 2. write 역할(비교가능)

비교 가능한 등급의 경우 이들간의 관계와 제약조건을 기준으로 그림 3과 같이 read-write 역할을 그래프로 도식화하였다. 특히 read 권한의 경우 특정 등급에 할당이 되면 다른 등급에서는 권한이 부여되지 않는다. 왜냐하면 만약, 다른 등급에서 read 권한의 등급이 부여될 경우 해당 주체의 역할에 하위 등급의 read 권한이 포함되므로 객체를 읽을 수 있게 되어 Biba 모델의 read up 정책을 위반하게 되기 때문이다.

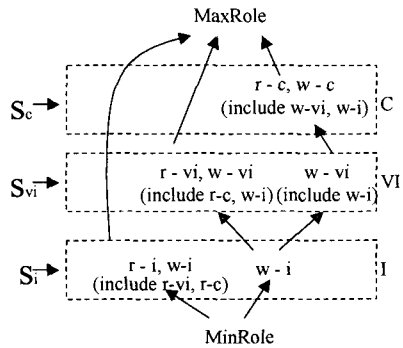

그림 3. read-write 역할(비교가능)

또한 그림 4와 같이 read-write 역할의 그래프에서 두 개의 등급에 동시에 양립하는 권한이 존재한다. read 권한의 경우 Biba 모델의 read-up 정책으로 중첩되는 두 개의 등급 중에서 상위 등급이 read 등급으로 부여되고 write 권한의 경우 write-down 정책으로 중첩되는 등급 중에서 하위 등급이 부여된다.

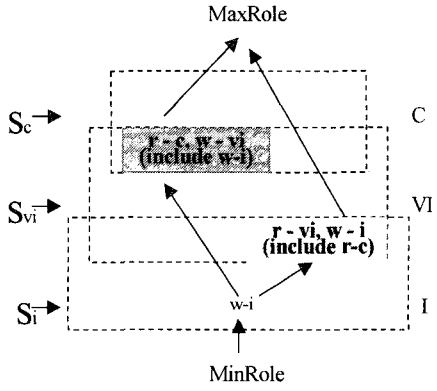

그림 4. 양립하는 역할(비교가능)

다음으로 r-level과 w-level의 등급을 서로 비교할 수 없는 경우의 역할을 고려하자.

먼저 read 역할의 경우, r-level(R_2)는 자신의 하위 역할 R_1과 $R_{1'}$의 r-level중에서 최대하한의 등급을 취한다. 이렇게 함으로써 read up 정책을 유지할 수 있다.

[제약조건 6] $R_1 \rightarrow R_2$, $R_{1'} \rightarrow R_2$일 때, read 역할의 경우

r-level(R_2) = GLB[r-level(R_1), r-level($R_{1'}$)]

```
READ_ROLEGRAPH_ASSIGN(NONCOMP.)
senior_role : ROLE
junior_role1 : ROLE
junior_role1' : ROLE
---
senior_role.r-level =
GLB[junior_role1.r-level, junior_role1'.r-level]
```

이들간의 역할 그래프를 도식화하면 다음 그림 5와 같다. r-vi와 r-vi' 등급이 비교 불가능한 등급일 경우 C 등급에 해당하는 주체는 vi 등급의 최대 하한 등급인 i 등급을 부여함으로써 read up 정책을 유지할 수 있다.

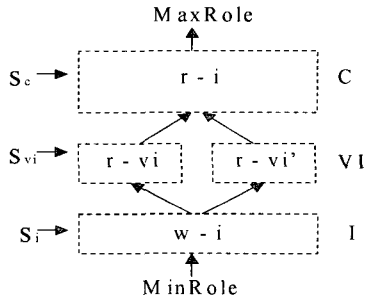

그림 5. read 역할(비교불가능)

다음으로 write 역할의 경우, w-level(R_2)는 자신의 하위역할 R_1과 $R_{1'}$의 w-level중에서 최소상한의 등급을 취한다. 따라서 write down 정책을 유지할 수 있다.

[제약조건 7] R1→R2, R1' →R2일 때, write 역할의 경우

w-level(R_2) = LUB[w-level(R_1), w-level($R_{1'}$)]

```
WRITE_ROLEGRAPH_ASSIGN(NONCOMP.)
senior_role : ROLE
junior_role1 : ROLE
junior_role1' : ROLE
---
senior_role.w-level =
LUB[junior_role1.w-level, junior_role1'.w-level]
```

w-vi와 w-vi' 등급이 비교 불가능한 등급일 경우 그림 6와 같이 C 등급에 해당하는 주체는 vi 등급의 최대 상한 등급인 c 등급을 부여함으로써 write down 정책을 유지할 수 있다.

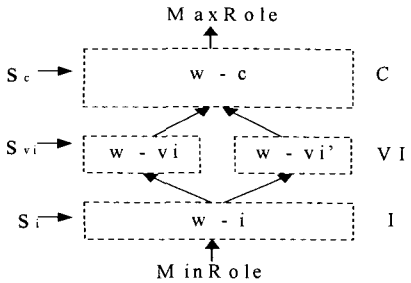

그림 6. write 역할(비교불가능)

4. 결 론

기업은 기업 활동 과정에서 발생하는 다양한 형태

의 기업정보를 컴퓨터로 관리하고 있는데, 기업의 규모가 커지고 업무 처리과정이 복잡해짐에 따라 정보에 대한 보안 관리에 어려움을 겪고 있다. 지금까지 많이 연구된 접근제어 정책에는 임의적 접근제어정책과 강제적 접근제어 정책 그리고 역할기반 접근제어 정책이 있다.

강제적 접근제어 정책은 접근제어 판단의 기준이 되는 인가등급과 보안 등급을 사용자와 정보객체에 부여하고, 접근제어 규칙을 정의하는 기능이 제한된 수의 보안관리자에 의해 관리되므로 시스템의 보안관리가 중앙집중적으로 이루어진다. 그러나 BLP, Biba 모델이 적용되기 위해서는 시스템을 구성하는 사용자와 정보객체에게 계층적 구조를 가지는 보안등급이 일관성있게 부여될 수 있는 환경이여야 하며, 사용자와 정보객체의 수가 많아지고 다양한 보안 특성을 가지게 되는 환경에서는 적용이 용이하지 않는 문제점을 가지고 있다.

역할기반 접근제어 정책은 역할을 기반으로 접근제어 서비스를 제공하는 모델로서 권한이 역할에 부여되고, 사용자는 조직내에서 책임과 자격에 맞는 역할에 할당됨으로써, 기업의 조직구조에 적합하고 정보자원을 효율적으로 관리할 수 있다. 하지만 이 정책은 상위 역할에 배정된 사용자는 하위 역할에 배정된 사용자의 권한을 상속받으므로 권한 남용의 위험성을 갖고 있다. 또한, 역할에 의해 접근되는 객체에 대한 중요도에 따른 등급이 기술되어 있지 않아 해당 역할에 배정된 모든 사용자들에게 객체에 대한 권한이 주어지므로 정보의 비밀성과 무결성을 해칠 우려가 있다.

정보의 비밀성에 기반을 둔 BLP 모델에 의거한 기존의 역할 그래프 모델[3]은 정보의 비밀성보다는 무결성이 강조되는 상업적인 환경에 적용하기에는 미흡하다고 본다. 그래서 본 논문에서는 기존의 역할 그래프 모델에 무결성 등급을 갖는 사용자와 객체를 배정하여 주체의 무결성 등급과 역할에 관련된 객체의 무결성 등급에 따라 권한을 부여하므로 수많은 접근권한을 관리하는데 융통성을 제공할 뿐 아니라 정보의 무결성을 보장한다. 또한, 시스템 개발단계에서 비정형적인 요구사항 기술로 인한 부정확성, 모호성, 불완전성에 대한 문제를 해결하기 위해 세분화된 역할들의 제약조건들을 명세 언어인 Z를 이용해 정형화된 구조로 명확하게 표현함으로써 접근제어정책에 대한 설계 및 구현시 시간과 비용 절감효과를 기대할 수 있다.

향후 연구방향으로는 역할기반접근정책의 특징 중 하나인 데이터 추상화의 개념으로 확장시키는 것으로, 운영체제나 시스템에서 사용되어졌던 데이터 처리 명령어인 read, write 대신에, 다양한 기능을 수행할 수 있는 상업적인 처리 명령어에 적용시키고자 한다.

참 고 문 헌

[1] Matunda Nyanchama, Sylvia Osborn, "Modeling Mandatory Access Control in Role-Based Security Systems, Database Security IX status and prospects, pp. 129-144, 8, 1995.

[2] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, Charles E. Youman, "Role-Based Access Control Models", COMPUTER SOCIETY, IEEE, FEB. pp.38-47, 1996.

[3] Sylvia Osborn, "Mandatory Access Control and Role-Based Access Control Revisited", Second ACM Workshop on RBAC, pp. 31-40, 11. 1997.

[4] Ravi S. Sandhu and Pierangela Samarati, "Access Control : Principles and Practice", IEEE Communications Magazine, pp. 40-48, 9, 1994.

[5] Matunda Nyachama and Sylvia Osborn, "The Role Graph Model and Conflict of Interest", ACM Transactions on Information and System Security, VOL.2, NO.1, pp. 3-33. 1999

[6] David d. Clark, David R. Wilson, "A Comparison of commercial and Military computer policies", IEEE, 1987.

[7] Silvana Castano, DATABASE SECURITY, ADDISON-WESLEY

[8] Warwick Ford, Computer Communications Security, Prentice Hall

[9] David Rann, John Turner and Jenny Whitworth, "Z : A Beginner's Guide", School of Computing Staffordshire University, 1994

[10] Bryan Ratcliff, "INTRODUCING SPECIFICATION USING Z : A Practical Case Study Approach", McGRAW-HILL BOOK COM-

PANY, 1994.

[11] U.S. Department of Defense, Department of Defence Trusted Computer System Evaluation Criteria, DOD 5200.28-STD, Natioal Computer Security Center, Dec. 1985.

최 은 복

1992년 전남대학교 전산학과 졸업(이학사)
1996년 전남대학교 전산통계학과 졸업(이학석사)
2000년 전남대학교 전산통계학과 졸업(이학박사)
2001년 순천제일대학 인터넷정보학부 전임강사
2002년~현재 전주대학교 정보기술공학부 조교수
관심분야 : 통신망관리, 정보보안, 엑티브 네트워크 등

이 형 옥

1994년 2월 순천대학교 전산학과 졸업(이학사)
1996년 2월 전남대학교 전산통계학과 졸업(이학석사)
1999년 2월 전남대학교 전산통계학과 졸업(이학박사)
1999년 10월~2002년 2월 한국전산원(선임연구원)
2002년 3월~현재 순천대학교 컴퓨터교육과 조교수
관심분야 : 그래프이론, 알고리즘, 병렬처리