通信情報保護學會論文誌
第4卷 第2號, 1994. 12

동적 객체지향 데이타베이스의 다단계 보안 모델링

김 영 균*, 노 봉 남*

Modeling the Multilevel Security of Active Object-Oriented Databases

Young-Kyun Kim*, Bong-Nam Noh**

요 약

본 논문은 데이타베이스의 동적 기능을 모형화하기 위해서 동적 규칙을 사건과 동적 규칙 객체로 취급하여 개념적 스키마에 표현하는 동적 객체지향 데이타 모델을 제안한다. 제안된 모델에서 정적 구조와 동적 구조에 대한 개념들을 정의하고, 모형화 과정에서 사용자의 이해도를 증진시키는 모델의 구성요소에 대한 그래픽 다이어그램을 제시하였다. 그리고 동적 규칙이 데이타베이스 구조에 포함되므로써 발생가능한 정보의 불법적인 노출 또는 변경을 방지하기 위해서 BLP 모델의 보안 정책을 확장하여 제안된 모델에 대한 11가지 종류의 다단계 보안 성질들을 정의하였다. 또한, 정의된 다단계 보안 성질들이 타당한가를 조사하기 위해서 패트리네트 확장하여, 보안 성질의 검증작업을 수행하였다.

Abstract

To model dynamic behaviors of databases we propose an active data model, which represents active rules in the conceptual schema by events and rules. In the proposed model we defined database constructs of static structures as well as dynamic structures and presented graphic diagrams for users' good understandability. And in order to protect illegal disclosures and modifications of data which can occur by adding active rules into databases we extended the security policy of BLP model and defined 11 kinds of multilevel security properties in our model. Also to show that defined security properties are certainly correct we performed validation processes of the security properties using the extended petri-net.

* 전남대학교 전산학과

+ 이 연구는 1994년도 학술진흥재단 연구비 지원으로 이루어진 것임

1. 서 론

컴퓨터 이용 설계 및 제조, 컴퓨터 통합 제조, 분산 네트워크 관리, 항공 교통 제어 시스템, 핵처리 시스템 등의 실시간 응용 분야들은 대용량의 공유 데이타와 지식 저장소를 접근하고, 특정한 사건이 발생할 때 적절히 반응할 수 있는 기능을 포함한다. 따라서, 실시간 응용을 지원하기 위해서는 데이타베이스 상태를 감독하고, 정의된 조건을 만족시키는 사건이 발생할 때 시간적 제약사항을 충족시키면서 적절한 행위를 수행하는 기능이 기존의 데이타베이스 관리 시스템에 포함되어야 한다[15, 18, 21, 22].

기존의 데이타베이스 시스템은 사용자의 응용 프로그램에 의해서 외부에서 요구될 때만 이에 대한 반응으로 데이타를 조작하는 연산들은 실행시키기 때문에 수동적인 시스템으로 고려할 수 있다. 반면에, 사용자의 간섭없이 특정 사건이 발생할 때 정의된 연산을 데이타베이스 시스템에 의해서 자동적으로 수행되는 데이타베이스 시스템을 동적 데이타베이스 시스템으로 서술한다. 그러므로, 동적 데이타베이스 시스템은 기존의 데이타베이스 관리 시스템의 모든 기능들(무결성 제어, 접근 제어, 유도 데이타 처리, 뷰 제공, 추론 지원)을 지원할 뿐만 아니라 데이타베이스 상태를 감독하고, 특정한 사건이 탐지될 때 미리 정의된 행위를 실행하는 부가적인 기능을 갖는 데이타베이스 시스템이다[2, 15, 22].

이제까지의 연구들에서는 동적규칙을 응용 데이타베이스의 논리적 또는 물리적 구현 단계에서 모형화하여 처리하고 있다. 그러나 물리적 구현 단계에서 동적 규칙을 정의하는 방법은 초기설계 단계에서 고려해야 하는 데이타베이스의 동적 행위가 설계의 마지막 단계에서 수행하기 때문에, 구현 단계의 복잡도로 인하여 실세계의 동적 행위에 대한 의미가 모호해지기 쉽고 그리고 절차적이며 비구조적인 규칙기반 프로그래밍 때문에 동적 규칙의 중복성이나 비일관성 상태를 야기시킬 수 있다[1].

한편으로, 데이타베이스 시스템에 동적 규칙을 추가하므로써, 데이타베이스 보안 분야는 한층 더 복잡해 진다. 동적 규칙이 사용자의 역할을 대신하므로써, 사용자와 데이타 객체들간에 정의되는 다단계 보안 제약조건들이 동적 규칙 객체와 객체 사이의 범위로 확장되어 정의되어야 한다. 즉, 데이타베이스내의 데이타를 검색 또는 변경시키는 동적 규칙을 적절한 보안 정책에 따라서 제어해야만 불법적인 정보의 흐름을 방지할 수 있다. 따라서 다단계 보안을 지원하는 동적 데이타베이스는 아주 중요한 문제이지만, 동적 데이타베이스에서 다단계 보안성질에 관련된 연구는 아주 미흡한 실정이다.

본 논문에서는 기존의 동적 규칙 설계시 문제점을 극복하기 위해서 동적 규칙을 초기 데이타베이스 설계 단계에서 모형화할 수 있도록 동적 객체지향 데이타 모델을 제안하고, 제안된 모델에서 서술되는 동적 규칙이 갖는 11가지의 다단계 보안 성질들을 정의한다. 그리고 제시된 다단계 보안 성질들에 대한 타당성 검증을 수행한다.

제안된 모델은 정적 구조와 동적 구조에 대한 여러가지 개념들을 표현하고, 모형화 과정에서 사용자의 이해도를 증진시키는 모델의 구성요소에 대한 그래픽 다이어그램을 제공한다. 그리고 동적 규칙이 데이타베이스 구조에 포함되므로써 발생가능한 정보의 불법적인 노출 또는 변경을 방지하기 위해서 Bell LaPadula 모델의 보안 정책을 확장하고, 보안 정책을 만족시키는 다단계 보안 성질들을 정의한다. 또한 본 논문에서 정의된 동적 규칙에 대한 다단계 보안 성질들의 타당성을 제시하기 위해서 데이타베이스 설계의 일관성 분석 도구로 많이 이용되는 패트리네트를 이용하여 동적 규칙의 다단계 보안 성질들을 검증하였다.

본 논문은 2장에서 관련된 연구들을 간략히 살펴보고, 제안된 동적 객체지향 모델을 3장에서 설명한다. 그리고 4장에서는 확장된 다단계 보안 정책과 보안 정책을 만족시키는 다단계 보안 성질을 정의하였고, 5장에서 패트리네트를 이용하여 제안

된 보안 성질들의 타당성을 검증하였다. 마지막으로 결론과 추후 연구방향을 6장에서 언급하였다.

2. 관련연구

풍부한 기능을 제공하는 동적 규칙을 데이타베이스 시스템에 적용하기 위해서 많은 연구들이 관계형 데이타베이스 시스템을 기반으로 수행되었고, 현재는 동적 기능을 포함하는 상용화된 관계형 데이타베이스 시스템 즉, Ingres, Sybase, Oracle 등이 이미 개발되었다. 그리고 객체지향 데이타베이스 시스템에서도 동적 규칙 메카니즘을 제공하기 위해서 ODE, HiPAC, ADAM, O_2 등에서 최근에 많은 연구들이 수행되고 있다[2, 12, 13, 15, 22].

대부분의 연구들에서 능동적 기능은 프로그래밍 언어 수준에서 모형화하기 때문에 동적 규칙의 현실적인 의미가 불확실해지고, 동적 규칙들의 중복과 비일관성 문제가 발생한다. 이러한 문제는 데이타베이스의 개념적 설계 단계에서 동적 규칙을 모형화하므로써 해결될 수 있고, 또한 특정한 시스템에 종속되지 않는 동적 규칙의 모형화 방법을 이용하여 특정한 시스템에 쉽게 이식될 수 있는 장점을 갖는다.

개념적 설계 단계에서 동적 규칙을 모형화하는 연구는 Tanaka에 의해 처음으로 시도되었다[11]. 이 연구에서 관계형 데이타베이스 시스템을 기반으로 하여, 동적 규칙에 대한 그래픽 표기법과 동적 규칙에 부합하는 데이타 연산 언어인 SQL을 확장하여 정의하였다. 또한 관계형 시스템으로 변환하는 과정에서 데이타베이스 스키마에 표현되는 무결성 제약조건들을 자동적으로 유지시키기 위해 무결성을 동적 규칙으로 변환시키는 알고리즘을 제안하였다. 현실 세계에는 하나이상의 사건들이 결합하여 하나의 동적 규칙을 구성하는 복합 사건이 많이 존재하기 때문에 응용에 내포되어 있는 복합 사건들을 모형화할 수 있는 구조가 반드시 제공되어야만 완전한 동적인 시스템을 구성할 수 있다. 그러나 이 연구에서는 단순 사건과 동적 규칙만을 고려하였고, 복합 사건(composite event)은 지원하지 못했다.

보안 정책의 개념은 광범위하고 여러 분야들에서 다른 방법들로서 많이 이용되고 있다. 보안정책은 접근 제어 정책으로도 정의될 수 있는데, 접근 제어 정책의 목적은 컴퓨터, 통신 그리고 정보 자원에 대한 권한이 부여되지 않는 접근을 방지하는 것이다. 객체지향 데이타베이스 시스템에서 접근 제어 정책은 BLP 모델의 정책을 많이 수용하고 있으나, 동적 규칙이 추가되므로써 주체와 객체간의 접근 제어가 아니라 객체와 객체간의 접근 권한을 정의하여 정보의 흐름을 제어해야 한다. 따라서 본 논문은 BLP 모델의 정책을 확장한 보안 정책을 정의하고, 보안 정책에 부응하는 데이타베이스의 동적인 측면에 대한 다단계 보안 성질들을 제안한다.

전통적인 객체지향 데이타베이스에서 다단계 보안성을 고려한 연구가 Thuraisinghum에 수행되었다[9, 10]. 이 연구에서는 객체지향 패러다임의 기본 개념들인 객체, 클래스, 그리고 일반화에 대한 보안 제약조건들 뿐만 아니라 복합 객체와 버전에서의 다단계 보안 성질을 규명하여, 객체지향 데이타베이스에서 다단계 보안 특성들을 명확하게 정의하였다. 또한 관계형 데이타베이스에서 제기된 다중인스턴스화(polyinstanciation) 개념에 대해서도 언급하였다.

본 논문에서 제안된 보안성을 갖는 동적 객체지향 데이타 모델도 Thuraisnghum의 연구에서 정의된 기본적인 보안성 개념이 적용됐으며, 보안 정책도 또한 동일하다. 그러나 Thuraisinghum의 연구는 단지 데이타 모델의 정적인 측면 즉, 데이타 구조에 대한 다단계 보안 특성들을 정의할 뿐 데이타베이스의 능동적인 측면은 고려하지 않고 있다. 1장에서도 언급된 바와 같이 데이타베이스 분야에서 실시간 요구사항들을 지원하기 위해서 동적 데이타 모델에 대한 연구들이 많이 수행되고 있는 실정이다. 따라서 객체지향 데이타 모델의 구성요소가 기존의 정적 구조 뿐만 아니라

동적구조를 포함할 때는 동적 구조에 대한 다단계 보안 특성들도 정의되어야 보다 완벽한 데이타베이스의 보안을 유지할 수 있다.

그러므로 제안된 동적 데이타 모델에서는 기존의 정적인 측면들에 대한 다단계 보안 특성은 Thuraisinghum의 보안성 제약조건들을 이용하지만, 동적인 구조인 사건과 규칙 객체에 대한 다단계 보안 성질과 사건과 규칙 사이에서 정의되는 다단계 보안 성질들이 본 논문에서 제안되고 있다. 사건과 규칙 객체의 보안 성질은 Thuraisinghum 연구에서의 제시된 보안 객체의 특성을 확장하여 정의하고, 사건과 규칙 객체사이의 다단계 보안 성질은 제시된 보안 정책을 만족시키도록 검증된 방법으로 정의한다. 그리고 Thuraisinghum이 언급한 객체의 다중인스턴스화 개념은 본 논문의 사건과 규칙 객체에 대해서는 고려하지 못했다.

다단계 보안 성질을 갖는 동적 데이타베이스에 관한 연구는 미흡한 실정이고, 단지 Smith가 다단계 보안 관계형 시스템에서 동적 규칙의 보안 요구사항을 파악하여 연구하였으며, 다단계 보안 규칙을 다단계 보안 객체로 고려한 확장된 관계형 모델을 제안하였다[7]. 그러나 관계형 시스템과 객체지향 시스템은 기본 패러다임이 서로 다르기 때문에 관계형 데이타베이스에서의 보안요구사항을 객체지향 데이타베이스로 적용하기 어렵다. 본 논문에서는 객체지향 데이타베이스의 특성을 고려하여, 객체지향 환경에 적합한 동적 규칙에 대한 다단계 보안 성질들을 파악하여, 11가지 종류의 다단계 보안 성질들을 정의하고, 제시된 보안 성질들의 타당성을 검증하는 작업을 수행한다.

3. 동적 객체지향 데이타 모델

응용 데이타베이스를 설계하는 작업은 복잡한 처리과정이며, 특히 그 응용이 다단계 보안 데이타베이스인 경우는 설계 작업이 한층 더 어렵게 된다. 그러므로 응용 영역을 개념적 또는 추상적 수준에서 고려하기 위해서는 응용의 구조적 특징과 의미를 모형화하는 데이타 모델이 필요하다. 따라서 응용의 정적 구조와 동적인 행위를 모형화는 구조를 제공하며, 다단계 보안 성질을 갖는 동적 객체지향 데이타 모델을 제안한다.

제안된 모델은 정적 구조와 동적인 행위 구조로 구성된다. 정적 구조는 객체, 클래스 그리고 관련성으로 구성되며, 여러 연구들에서 제안된 보편적인 공통된 개념들인 클래스, 상속성, 집단화 등을 지원한다. 그리고 동적 행위 구조는 동적 데이타베이스의 특성을 정의한 HiPAC 시스템에서 제안된 사건-조건-행위 규칙(event condition action : ECA rule)을 기본으로 하여, 동적 기능을 독립적인 사건과 동적 규칙을 이용하여 모형화한다.

3.1 정적 구조

객체지향 모델에서 정적 구조의 기본 개념은 객체이다. 즉, 실세계에 존재하는 추상적인 개념 또는 물리적인 사물 등을 객체로 모형화한다. 객체는 상태와 행위를 함께 캡슐화하고, 객체의 메소드를 통해서만 객체의 상태에 접근한다. 또한 모든 객체는 고유한 객체 식별자를 갖는다. 그리고 의미와 형태가 동일한 객체들은 하나의 클래스로 추상화되며, 클래스는 자신의 고유한 이름과 객체들을 가르키는 포인터를 갖는다.

■ 정의 3.1 클래스(class)
클래스를 C로 표현할 때, C는 다음과 같이 구성된다.

$C = \{N_C, A, M\}$, N_C : 클래스 이름
A : 속성
M : 메소드

클래스 속성은 객체의 상태를 표현하며 속성의 이름, 속성의 타입 그리고 속성 대응수로 구성된다. 속성 대응수는 특정한 속성이 소속되는 클래스에서 그 속성이 갖을 수 있는 인스턴스 수를 제

한한다. 예를 들면, 사람 클래스가 정의될 때, 사람은 하나 이상의 거주지와 근무처를 갖을 수 있다. 따라서 이러한 의미를 표현할 수 있도록 속성 대응수를 정의한다.

■ 정의 3.2 속성(attribute)
속성을 A로 표현할 때, A는 다음과 같이 구성된다.

$A = \{N_a, T, C_a\}$, N_a : 속성 이름
T : 속성 타입
C_a : 속성 대응수

객체지향 데이타베이스에서 연산은 클래스 내부에 구현되는 메소드로 정의된다. 즉, 메소드는 객체의 행위에 대한 의미를 나타낸다. 메소드는 메소드 이름, 매개변수, 그리고 객체의 상태를 조작하는 실제 연산에 관한 코드로 구성된다.

■ 정의 3.3 메소드(method)
메소드를 M으로 표현할 때, M은 다음과 같이 구성된다.

$M = \{N_m, P, I\}$, N_m : 메소드 이름
P : 매개변수
I : 실제 구현 코드

관련성은 여러 클래스들 사이에서 논리적인 연결을 제공한다. 현실 세계의 다양한 객체들의 관계를 데이타베이스 스키마상에 적절하게 표현하기 위해서는 이러한 관계를 추상화하여 표현하는 구조를 데이타 모델이 제공해야 한다. 대부분의 객체지향 모델들은 상속을 이용한 일반화 계층구조와 복합 객체를 표현하는 집단화 계층구조에 의해 일반화와 집단화 관련성을 표현하고 있으나, 대부분의 의미 데이타 모델(semantic data model)에서 제공하는 연관성(association)은 객체 내부 포인터로 표현하기 때문에 연관성의 의미가 직관적으로 제공되지 못하고 있다. 따라서 제안 모델은 연관성을 관련성으로 포함시켜서 설계자가 모형화할 수 있도록 한다.

■ 정의 3.4 관련성(relationship)
클래스 C_i, C_j사이의 관련성을 R로 표현할 때, 다음과 같이 관련성을 갖는다.

$R = \{C_i$ r_type $C_j\}$, r_type
= IS_A or IS_PART_OF
or IS_RELATED_TO.

정의 3.4에서 r_type은 관련성의 종류를 나타내며, IS_A는 일반화 관련성, IS_PART_OF는 집단화 관련성, 그리고 IS_RELATED_TO는 연관성을 의미한다. 그리고 정의된 관련성들에 대한 그래픽 표기법은 아래의 그림 1과 같이 표현한다.

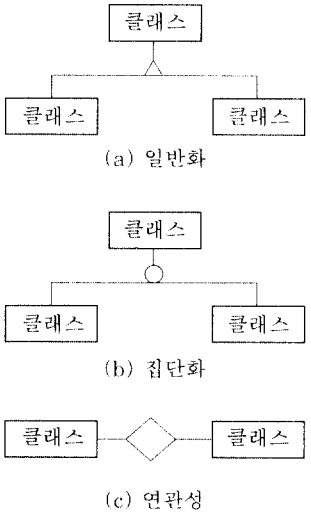


그림 1 제안 모델의 관련성 표기법

3.2 동적 규칙의 구조

제안 모델에서 동적 규칙의 모형화 방법은 사건과 동적 규칙을 분리하여, 각기 독립적 객체로 취급한다. 개념적인 데이타베이스 스키마 설계 과정에서 사건과 동적 규칙의 모형화는 다음의 설계 기준들을 기반으로 한다.

◖ 설 계 기 준 ◗

1) 사건, 동적 규칙을 독립적인 객체로 정의한다.
2) 기본 사건 뿐만 아니라 복합 사건을 지원한다.
3) 동적 규칙의 결합 모드를 지원한다.
4) 대체 행위를 정의한다.
5) 사건과 규칙의 그래픽 표기법을 지원한다.

제시된 설계 기준을 기존의 정적 구조와 통합시킨 구조는 그림 2와 같은 메타 스키마와 같은 형태를 갖는다. 기존 모델들의 공통적인 개념인 객체 뿐만 아니라 사건 객체와 동적 규칙 객체가 제안 모델의 전체적인 객체의 구성요소로 간주한다.

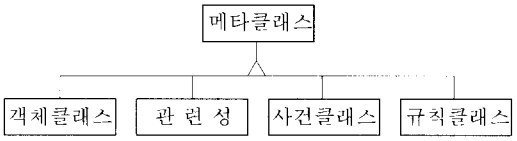


그림 2 제안 모델의 메타 스키마

3.2.1 사 건

사건은 데이타베이스의 상태를 변경시키는 갱신 연산, 시간적 제약사항을 갖는 시간 사건(temporal event) 그리고 외부 응용 프로그램에 의해 발생하는 외부 사건으로 구분할 수 있으나, 본 논문에서는 객체의 상태를 변경시키는 데이타베이스의 갱신 연산만을 사건으로 고려한다. 관계형 데이타베이스 시스템에서 갱신 연산은 삽입, 삭제, 변경 연산으로 정의될 수 있으나, 객체 지향 데이타베이스에서 객체의 갱신 연산을 객체 내부의 메소드에 의해 이루어지기 때문에 제안모델에서 데이타베이스 사건은 메소드의 실행을 의미한다.

▩ 정의 3.5 사건(event)
데이타베이스 사건을 E_{db}로 표현할 때, E_{db}는 다음과 같이 구성된다.

$E_{db} = \{N_e, \text{oid}, [\text{composite\_op } E_{db}']\}$

N_e : 사건 이름
oid : 사건에 영향을 주는 객체 이름
composite_op : 복합 사건 연산자

사건은 기본 사건과 복합 사건으로 구분된다. 기본 사건은 객체에서 수행된 단일 메소드에 의해 정의되고, 반면에 복합 사건은 하나의 객체에서 둘 이상의 메소드 또는 둘 이상의 객체에서 각각의 단일 메소드들의 수행으로 발생한다. 따라서, 제안된 모델에서는 기본 사건외에도 복합사건을 지원하기 위해서 다음과 같은 복합 사건을 모형화할 수 있는 구조를 제공한다.

▩ 정의 3.6 복합 사건
순서 사건을 E_{seq}, 동시 사건을 E_{and}, 분할 사건을 E_{or}, 그리고 배타 분할 사건을 E_{xor}라 할 때

- 순서 사건 $E_{seq} = (E_i \ll E_j)$
- 동시 사건 $E_{and} = (E_i \parallel E_j)$
- 분할 사건 $E_{or} = (E_i \vee E_j)$
- 배타 분할 사건 $E_{xor} = (E_i \otimes E_j)$

순서 사건은 둘 이상의 사건들이 차례로 발생한 사건을 의미하고, 동시 사건은 두 개의 사건들이 어느 시점에 동시에 발생할 때를 나타낸다. 반면에 분할 사건은 두개의 사건들이 동시에 발생하거나 또는 둘 중에서 하나만 발생할 수 있는 복합 사건을 의미한다. 마지막으로 배타 분할 사건은 정의된 두가지 사건들 중에서 오직 한 사건만이 발생할 때만 의미를 갖는 복합 사건이다.

3.2.2 동적 규칙

동적 규칙은 기본 사건 또는 복합 사건이 발생했음을 탐지하고, 사건이 정의된 조건을 만족하면, 데이타베이스 연산을 실행한다. 동적 규칙도 사건과 마찬가지로 독립적인 객체로 간주하여, 일반 객체와 동일하게 취급한다. 따라서, 동적 규칙은 사건부, 조건부, 실행부, 결합형태부 그리고 대

체행위부로 구성된다. 조건부는 발생한 사건에 대해서 평가를 수행하고, 결합형태부는 사건과 조건 사이의 결합 형태 즉, 사건이 발생한 후 곧바로 조건을 평가할 것인지 아니면 사건을 발생시킨 트랜잭션이 완료되기 직전에 평가할 것인지를 나타낸다. 일반적으로 결합형태부는 동적규칙의 실행에 관련하여 트랜잭션과의 관계를 나타낸다.

■ 정의 3.7 동적 규칙

동적 규칙을 R로 표현할 때, R은 다음과 같이 구성된다.

R == {E, C, A, CM, EH}

E : 사건의 이름, C : 조건 술어

A : 정의된 연산, CM : 결합형태

EH : 대체 연산

대체 연산은 발생된 사건의 조건을 평가하여, 조건을 만족하였음에도 불구하고 정의된 연산이 실행되지 않을 때, 행해지는 연산을 의미한다. 예를 들면, 조건의 평가 3분내에 연산이 실행되지 않으며, 사건에 해당하는 메소드의 실행을 취소시키는 것에 해당한다.

3.3 동적 규칙의 그래픽 표기법

동적 규칙을 그래픽 다이어그램으로 모형화하는 것은 기존의 의미 데이타 모델과 같이 사용자가 동적 규칙을 이해하기 쉽다는 장점을 갖는다. 본 논문에서 동적 규칙의 그래픽 모형화 구조는 Tanaka가 제시한 그래픽 표기법을 확장하여 이용한다.

단순한 형태의 동적 규칙의 그래픽 다이어그램이 그림 3에 제시되어 있다. 객체 클래스는 객체-관련성 모델에서와 같이 사각형으로 표시하고, 사건 객체는 원으로 표현한다. 그리고 동적 규칙 객체는 평행사변형으로서 다이어그램상에 표현된다. 사건 객체와 동적 규칙이 방향성이 있는 화살표로 연결되고, 이 연결 화살표는 화살표 뒤쪽에 위치한 사건이 화살표 머리쪽에 위치한 동적 규칙을 활성화시킨다는 의미를 갖는다. 사건과 규칙 객체의 내부에는 고유한 구별자가 서술되고, 실제 사건과 동적 규칙의 이름은 사건 클래스와 규칙 클래스 정의 구조에 서술한다.

그림 3 동적 규칙의 그래픽 표기법

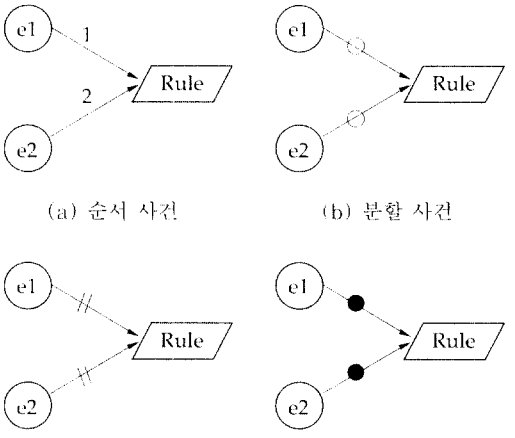


그림 4 복합 사건의 그래픽 표기법

제안 모델은 단순 사건이외에도 그래픽 다이어그램상에서 복합 사건들을 모형화할 수 있도록 그림 4에서와 같이 순서 사건, 동시 사건, 분할 사건 그리고 배타 분할 사건에 대한 그래픽 표기법을 제공한다. 순서 복합 사건은 발생 순서에 따라 명시적으로 양의 정수로서 우선순위를 정의하여 나타내고, 분할 사건은 작은 원으로, 동시 사건은 두 개의 수직선으로, 그리고 배타 분할 사건은 원내부에 검은색으로서 표현한다.

4. 동적 규칙의 다단계 보안 성질

다단계 보안을 유지시키기 위한 보안 정책으로

서 BLP 모델의 정책이 널리 이용되고 있는데, 이 정책에서는 객체에 대해서는 보안등급(classification)을 할당하고, 주체 즉 사용자에 대해서는 인가등급(clearance)을 부여한다. 그리고 주체가 객체에 대해서 읽기 연산과 쓰기 연산을 수행할 때는 주체의 인가등급과 객체의 보안등급을 비교하여 수행의 여부를 결정한다.

기존의 다단계 보안 정책은 데이타에만 관심을 두었으나, 객체지향 데이타베이스에서는 데이타와 프로시쥬어가 객체에 함께 캡슐화되어 있고, 속성뿐만 아니라 메소드에도 다단계 보안등급이 존재하기 때문에 프로시쥬어 또는 프로세스 실행에 관련하여 정보의 흐름을 고려해야 한다. 그러므로, 본 논문에서는 BLP의 정책을 기반으로 하여, 다단계 보안 정책이 프로세스에 대한 정책까지 포함시킨 확장된 보안 정책을 제시한다.

▩ 정의 4.1 확장된 다단계 보안 정책
사용자를 S, 객체 또는 데이타를 O, 사용자의 인가등급을 $L(S)$ 그리고 보안등급을 $L(O)$로 프로세스의 실행등급을 $L(\mathrm{Exec}(O))$로 표현할 때, 다단계 보안 정책은 다음과 같다.
(1) 읽기 연산 : $L(S) \geq L(O)$,
(2) 쓰기 연산 : $L(S) = L(O)$,
(3) 실행 연산 : (1)의 조건을 만족하고,
$L(S) = L(\mathrm{Exec}(O))$,
$\geq$ 기호는 등급들간의 지배 관계(dominance relation)를 표시한다.

정의 4.1에서 (3) 정책은 객체내부의 메소드가 실행되는 보안등급을 나타내며, 메소드 실행은 사용자가 최소한 메소드의 존재여부를 파악할 수 있어야 하며, 실제적인 메소드 실행의 보안등급은 메소드 자체의 보안등급이 아니고, 그 메소드를 실행시킨 사용자의 인가등급이어야만 낮은 수준으로의 정보 흐름을 방지할 수 있다.

데이타베이스에서 동적 규칙이 갖는 다단계 보안성은 세가지 범주로 구분하여 의미를 부여한다. 즉, 개개의 동적 규칙이 갖는 다단계 보안 성질, 권한이 없는 사용자에 의해 실행된 사건과 동적 규칙사이의 지배관계 그리고 동적 규칙은 데이타베이스에 대해서 읽기 연산과 쓰기 연산을 수행하기 때문에, 동적 규칙의 행위에 대한 다단계 보안 성질이다. 따라서 제시된 보안 정책을 만족시키며 세가지 범주에 해당하는 다단계 보안 성질들을 정의한다.

4.1 동적 규칙의 다단계 보안 성질

제안 모델에서, 동적 기능을 제공하는 동적 규칙은 사건과 규칙으로 분리되어 독립된 객체로 모형화되기 때문에 일반 객체 또는 클래스와 마찬가지로 각기 고유한 보안등급을 갖는다.

[보안 성질 1] 사건과 규칙 객체의 보안등급

사건 객체와 동적 규칙 객체에는 반드시 보안등급이 존재한다. 즉, 사건 객체를 e, 규칙을 r, 그리고 보안등급을 l로 표현할 때, 보안등급은 다음과 같다.

$$L(e) \leftarrow l,\ L(r) \leftarrow l,\ l \in \{U, C, S, TS\}$$

다음으로 고려할 보안 성질은 클래스에 대한 보안 특성이다. 사건과 규칙 클래스의 보안등급은 사건 객체와 규칙 객체의 보안등급에 지배를 받는다. 즉, 보안 성질 2에서와 같이 사건과 규칙 클래스의 보안등급은 객체의 보안등급들 중에서 가장 낮은 보안등급을 갖는다.

[보안 성질 2] 사건, 동적 규칙 클래스의 보안등급

사건 클래스를 E, 동적 규칙 클래스를 R로 표현할 때, 클래스 보안등급은 객체 보안등급에 지배를 받는다.

$$L(E) \leftarrow GLB(L(e_1), L(e_2), \ldots, L(e_i)),$$
$$L(R) \leftarrow GLB(L(r_1), L(r_2), \ldots, L(r_j)),$$

$1 \leq i, j \leq n$, GLB : 가장 낮은 보안등급을 찾는 최대하계 함수

4.2 사건과 동적 규칙의 다단계 보안 성질

동적 데이타베이스 시스템에서, 사건이 발생하고, 발생된 사건에 의해서 동적 규칙이 실행되어서 데이타베이스 상태에 영향을 미친다. 따라서, 다단계 보안 환경에서는 사건이 발생하는 보안 성질과 발생된 사건에 의한 동적 규칙의 실행, 동적 규칙의 발화 보안등급 그리고 연쇄적으로 발생하는 사건과 동적 규칙에 대한 보안 성질들이 정의되어야 한다. 객체지향 데이타베이스에서 사건의 발생은 메소드의 실행으로 고려하기 때문에, 제시된 보안 정책에 따라서 다음과 같은 사건 발생에 관한 보안 성질을 정의할 수 있다.

[보안 성질 3] 사건 발생 보안등급

사건이 발생되기 위해서는 명시적으로 정의된 사건의 보안등급이 사용자에 의해 실행되는 사건에 해당하는 메소드의 보안등급을 지배해야 한다.

$$\{\forall E_i \mid L(\mathrm{EXEC}(M)) \le L(E_i)\} \Rightarrow \mathrm{Raise}(E_i)$$

발생된 사건에 대해 동적 규칙이 반응하기 위해서는, 사건을 동적 규칙이 탐지해야 한다. 다단계 보안 환경에서, 높은 보안등급의 사건을 낮은 보안등급의 동적 규칙이 탐지하게 되면, 고수준의 정보가 낮은 수준의 규칙 객체의 쓰기 연산으로 인하여 불법적인 정보의 흐름이 발생되며, 이것은 제시된 보안 정책(2)를 위반하게 된다.

[보안 성질 4] 동적 규칙의 발화 보안등급

동적 규칙이 실행되기 위해서는, 동적 규칙의 보안등급이 발생된 사건의 보안등급을 지배해야 한다. 즉, 동적 규칙이 사건에 대한 읽기 연산이 가능해야 한다.

$$\{\forall R_i \mid L(E) \le L(R_i)\} \Rightarrow \mathrm{Fire}(R_i)$$

동적 규칙의 실행으로 인하여 데이타베이스의 상태는 변경되므로, 동적 규칙의 행위가 쓰기 연산인지 또는 읽기 연산인가에 따라서 제시된 보안 정책을 만족시키면 된다. 동적 규칙은 주체 즉, 사용자를 대변하는 역할을 수행하므로, 동적 규칙의 보안등급이 실제적인 사용자의 인가등급과 동일하다고 고려할 수 있다.

[보안 성질 5] 동적 규칙의 인가등급

동적 규칙은 사용자의 대변인이기 때문에, 동적 규칙의 행위에 대한 보안등급은 사용자의 인가등급으로 간주한다. 따라서, 동적 규칙의 인가등급은 발생된 사건에 반응하는 규칙의 보안등급이 된다.

$$\{\forall R_i \mid \mathrm{Fired}\ R_i\} \Rightarrow L(S^R) = L(R)$$

지금까지는 단일 사건과 단일 동적 규칙을 고려한 다단계 보안 성질을 정의하였다. 그러나 현실세계에서는 하나의 동적 규칙이 다른 동적 규칙을 실행시키는 연쇄 체인(cascade chain) 동적 규칙과 연쇄사건이 존재한다. 즉, 동적 규칙이 데이타베이스에 대한 연산을 수행하는 대신에 다시 새로운 사건을 발생한 것으로 고려한다. 따라서 연쇄적으로 발생되는 사건과 동적 규칙에서의 다단계 보안 성질들을 정의한다.

먼저, 연쇄적으로 발생하는 사건들에 대한 다단계 보안 특성으로서, 사건들 사이의 보안등급의 관계는 동적 규칙의 행위에 의해 생성된 사건의 보안등급은 최소한 이전에 발생한 사건의 보안등급보다는 같거나 높아야 한다. 즉, 보안 성질4와 보안 성질 5를 고려하면, 생성되는 새로운 사건의 보안등급은 이전 사건의 보안등급보다는 높게 정의되어야 한다.

[보안 성질 6] 연쇄 사건 보안등급

연쇄 체인에 관련된 두개의 사건을 각각 e_i, e_j, 그리고 e_i가 e_j보다 먼저 발생한 사건이라 할 때, 다음과 같은 다단계 보안 특성을 갖는다.

$$\{\forall E_i, E_j \mid (E_i \ll E_j) \wedge L(E_i) \leq L(E_j)\}$$
$$\Rightarrow E_i \text{ Cascade } E_j$$

연쇄 체인에 관련된 동적 규칙들에 있어서도, 먼저 수행된 동적 규칙의 보안등급이 나중에 수행되는 동적 규칙의 보안등급보다 낮다고 가정하면, 보안 성질 5에 의해서 높은 인가등급의 사용자가 자신의 권한을 낮은 인가등급을 갖는 사용자에 위임하는 결과가 된다. 이러한 경우는 보안정책을 위배하고, 높은 보안등급의 정보가 낮은 보안등급으로 흐르게 되는 상황을 발생시킨다. 따라서, 연쇄 체인 규칙에서도 연쇄 사건과 마찬가지로 뒤에 수행되는 동적 규칙의 보안등급이 이미 수행된 동적 규칙의 보안등급을 지배해야 한다.

【 보안 성질 7 】 연쇄 동적 규칙 보안등급

연쇄 체인에 관련된 두개의 규칙을 각각 r_i, r_j, 그리고 r_i가 r_j보다 먼저 수행된 규칙이라 할 때, 다음과 같은 다단계 보안특성을 갖는다.

$$\{\forall r_i, r_j \mid (R_i \rightarrow R_j) \wedge L(R_i) \leq L(R_j)\}$$
$$\Rightarrow R_i \text{ Cascade } R_j$$

4.3 복합 사건의 다단계 보안성질

제안된 모델은 동적 규칙의 모형화 과정에서 단일 사건뿐만 아니라 복합 사건을 지원할 수 있는 구조를 제공하고 있다. 일반적으로 복합사건은 하나 이상의 단일 사건들이 여러가지의 의미적 관련성에 따라서 결합되어 하나의 사건을 구성되며, 복합사건이 다단계 보안 환경에 적용될 때는 고려해야 하는 문제점들이 발생한다. 즉, 각각의 보안등급을 갖는 사건들이 하나의 복합사건을 구성하기 때문에, 결과적으로 복합사건을 구성하는 사건 객체의 보안등급을 어떻게 결정해야 하는 것이 중요한 문제이다. 따라서, 이 절에서는 3장에서 제안된 4가지의 복합 사건 구조에 대한 다단계 보안 의미를 파악하고, 보안 정책을 유지하는 다단계 보안 성질을 정의한다.

복합 사건의 첫번째 구조인 순서 사건은 사건들이 순서적으로 모두 발생해야만 동적 규칙이 실행되는 사건이다. 만일 순서 복합 사건을 구성하는 사건들이 서로 다른 보안등급을 갖으면, 동적 규칙은 어떤 보안등급으로서 실행되어야 하는지를 결정하지 못한다. 만일, 동적 규칙이 낮은 보안등급의 사건과 같은 보안등급으로 실행되면, 높은 보안등급을 갖는 사건의 정보가 낮은 수준의 정보로 노출되어 보안 성질 4를 위반하게 되며, 반대로 높은 보안등급을 갖는 사건의 수준에서 실행되면 낮은 수준의 정보가 보다 높은 보안수준을 갖는 정보로 반영되기 때문에 보안정책(2)를 위반하게 된다. 따라서 순서 복합 사건을 구성하는 단일 사건들의 보안등급이 모두 동일해야만 정보가 안전하게 유지될 수 있다.

【 보안 성질 8 】 순서 복합 사건 보안 특성

순서 복합 사건을 구성하는 단일 사건을 각각 e_i, e_j 라 할 때, 다음의 조건을 만족해야 한다.

$$\{\forall E_i, E_j \mid (E_i \ll E_j) \wedge L(E_i) = L(E_j)\}$$
$$\Rightarrow L(E_i \ll E_j) \leftarrow L(E_i) \mid L(E_j)$$

동시 사건은 순서 복합 사건과는 의미적 차이가 있으나, 보안성 측면에서는 두개의 사건들이 모두 발생해야 하는 복합 사건이기 때문에 순서 사건과 마찬가지로 보안 의미를 부여해야 한다. 따라서, 동시 사건의 보안등급도 보안성질 8에서와 같이 다단계 보안특성을 갖는다.

【 보안 성질 9 】 동시 복합 사건 보안 특성

동시 복합 사건을 구성하는 단일 사건을 각각 e_i, e_j 라 할 때, 다음의 조건을 만족해야 한다.

$$\{\forall E_i, E_j \mid (E_i \parallel E_j) \wedge L(E_i) = L(E_j)\}$$
$$\Rightarrow L(E_i \parallel E_j) \leftarrow L(E_i) \mid L(E_j)$$

분할 사건은 복합 사건을 구성하는 단일 사건들이 모두 발생하거나 또는 그 중에서 하나의 사건

만이라도 발생하면, 동적 규칙이 수행되는 복합 사건이다. 따라서 분할 복합 사건의 경우는 두가지의 경우로 구분하여 단일 사건의 경우와 복합 사건의 경우에 대해서 모두 다단계 보안 성질을 정의한다. 분할 복합 사건에서 하나의 사건만 발생할 경우는 그 사건의 보안등급을 복합 사건의 보안등급으로 결정하고, 둘 이상의 사건일 경우는 보안 성질 8과 보안 성질 9에서와 마찬가지로 두 개의 사건의 보안등급은 서로 동일해야 한다.

【 보안 성질 10 】 분할 복합 사건 보안 특성

(1) 하나의 사건인 e_i만 발생할 때,

$$\{\forall E_i, E_j \mid (E_i \vee E_j) \wedge E_i \text{ only occurs}\} \Rightarrow L(E_i \vee E_j) \geq L(E_i)$$

(2) e_i, e_j 모두 발생할 때는 다음의 조건을 만족해야 한다.

$$\{\forall E_i, E_j \mid (E_i \vee E_j) \wedge L(E_i) = L(E_j) \Rightarrow L(E_i \vee E_j) \geq L(E_i) \mid L(E_j)$$

마지막으로 배타 분할 사건인 경우는, 사건들 중에서 오직 한 사건만이 발생한다는 의미를 갖기 때문에 모형화 단계에서는 의미적으로는 복합 사건이지만, 보안성 측면에서는 단일 사건으로 고려해야 한다. 따라서, 배타 분할 사건의 보안등급은 사건들 중에서 발생한 사건의 보안등급으로 결정된다.

【 보안 성질 11 】 배타 분할 복합 사건 보안 특성

(1) 사건 e_i 발생하고, e_j 발생하지 않으면,

$$\{\forall E_i, E_j \mid (E_i \otimes E_j) \wedge E_i \text{ only occurs}\} \Rightarrow L(E_i \otimes E_j) \geq L(E_i)$$

(2) 사건 e_j발생하고, e_i 발생하지 않으면,

$$\{\forall E_i, E_j \mid (E_i \otimes E_j) \wedge E_j \text{ only occurs}\} \Rightarrow L(E_i \otimes E_j) \geq L(E_j)$$

5. 보안 성질의 타당성 검증

앞장에서는 동적 규칙이 데이타베이스의 상태를 변경시킬 때 보안정책을 위반하는 정보의 불법적인 흐름을 발생시킬 수 있기 때문에, 이를 방지하기 위한 11가지의 다단계 보안성질들을 정의하였다. 본 논문에서 제안된 보안 성질들이 정보의 흐름을 안전하게 관리하는가를 검증하기 위해서 이 장에서는 기존의 패트리네트(petri-net)을 이용하여 타당성 검증을 수행한다. 지금까지 연구된 패트리네트는 여러 분야들에서 이용되고 있으며, 특히 데이타베이스에서는 데이타베이스 설계의 일관성을 분석하기 위한 도구로 많이 활용되고 있다. 본 논문에서는 기존의 상태/변이 패트리네트를 확장 변경하여 이용한다.

상태/변이 패트리네트(place/transition petri_net)는 데이타의 현재 상태를 표시하는 상태와 상태에 대한 동적인 행위를 나타내는 변이 그리고 토큰(token)으로 구성되며, 상태는 원으로, 변이는 상태와 상태사이의 단방향의 아크(arc)로 그리고 토큰은 점으로서 그래픽 표기를 한다. 그러나 본 논문에서는 동적 규칙을 패트리네트에 적용하기 위해서 기존의 상태/변이 패트리네트의 상태를 사건으로 고려하고, 변이를 동적 규칙으로 대체하여 이용한다. 따라서 토큰이 사건을 표현하는 원에 포함될 때 사건이 발생하고, 동적 규칙의 실행은 전달된 토큰에 의해 동적 규칙에서 정의된 연산이 실행되어 객체 클래스 상태를 변경시키거나 또는 토큰이 다른 사건으로 전달되는 경우를 의미하게 된다.

앞장에서 제시된 여러가지의 보안 성질들이 적용되는 동적 규칙의 안전성 검사는 변형된 패트리네트에서 정보 즉, 토큰이 보안 정책을 위반하지 않고 사건에서 동적 규칙으로 안전하게 전달되거나, 또는 동적 규칙에서 사건으로 안전하게 전달될 수 있는가를 결정하므로써 수행된다. 따라서 패트리네트에서 토큰이 전달될 때의 안전성을 증명하므로써 제시된 보안 성질들의 타당성을 검증할 수 있다. 그러므로 패트리네트에서 정보의 흐름의 안전성을 다음과 같이 정리한다.

✣ 정리 5.1 변형된 패트리네트의 안전성

사건에 대한 조건과 토큰이 사건에 안전하게 전달되고, 동적 규칙에 대한 조건이 안전하면 패트리네트에서 정보의 흐름은 전체적으로 안전하다.

증명 : 패트리네트의 안전성에 대한 증명은 다음과 같은 네가지 과정들을 통해서 수행된다.

단계1 : 사건에 대한 조건의 안전성 검사

단계2 : 동적 규칙의 발화 조건의 안전성 검사

단계3 : 동적 규칙의 실행 조건의 안전성 검사

단계4 : 동적 규칙의 실행 결과의 안전성 검사

그리고 각 단계의 안전성은 다음의 네가지 종류의 소정리들로 증명된다. □

✣ 소정리 5.1 사건 발화 안전성

토큰이 사건을 표시하는 상태에 포함되는 동시에 토큰 보안등급이 사건 보안등급의 영역에 포함되면, 사건은 안전하게 발생한다.

증명 : 토큰 보안등급이 객체 메소드를 실행시킨 사용자의 인가등급과 동일한 수준이라고 가정할 때, 소정리 5.1을 논리적인 형식으로 표현하면 다음과 같이 기술된다.

$$P : \text{SecureInvoke}(e_i) \rightarrow \text{tok} \in e_i \wedge L(\text{tok}) \in L(e_i)$$

여기서 P가 갖는 의미는 토큰이 사건에 포함되고 또한 토큰 보안등급이 사건의 보안등급 영역에 포함되는 조건이 참이어야 안전하게 사건이 발생할 수 있음을 의미한다. 따라서 P의 대우명제를 이용하여, 대우명제가 참인 것을 보이면 명제 P에 대한 증명이 이루어진다.

대우명제 : $\neg(\text{tok} \in e_i \wedge L(\text{tok}) \in L(e_i))$

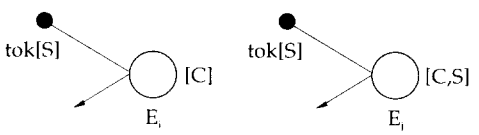
$$\rightarrow \neg\text{SecureInvoke}(e_i)$$

$$\equiv \text{tok} \notin e_i \vee L(\text{tok}) \notin L(e_i)$$

$$\rightarrow \neg\text{SecureInvoke}(e_i)$$

즉, 토큰이 사건상태에 포함되지 않거나 또는 토큰 보안등급이 사건 보안등급에 포함되지 않으면 사건은 발생하지 않기 때문에, 대우명제도 또한 참이 된다. □

따라서 사건 발생 보안 성질(보안 성질3)에서, 실행된 메소드의 보안등급은 사용자의 보안등급을 의미하기 때문에 토큰의 보안등급이 사건 보안등급보다 높은 보안등급이면 사건은 발생하지 않는다. 즉, 그림 5에서와 같이 사건의 상태에 토큰이 포함되지 않으므로 보안등급이 높은 정보를 낮은 보안등급의 사건이 실행할 수 없기 때문에 보안성질3으로 인하여 정보의 보안성은 유지된다.



(a) 사건 미발생 (b) 사건 발생

그림 5 사건 안전성 검증

✣ 소정리 5.2 동적 규칙 발화 안전성

사건에서 생산되는 토큰 보안등급이 동적 규칙의 보안등급에 포함되면, 동적 규칙은 안전하게 발화한다.

증명 : 소정리 5.1의 증명에서와 동일한 방법으로 증명될 수 있다. 즉, 동적 규칙 발화 안전성을 다음과 같이 논리적인 형식으로 표현한다.

$P : \text{SecureFire}(r_i) \rightarrow L(e_i) \in L(r_i)$.

토큰 보안등급이 규칙의 보안등급에 포함된다는 의미는 사건의 발생으로 전달된 토큰의 보안등급이 동적 규칙의 보안등급에 포함되어야 동적 규칙이 안전하게 실행됨을 의미한다. 따라서 이 명제의 대우명제를 취해보면,

대우명제 : $\neg(L(e_i) \in L(r_i))$
$\rightarrow \neg \mathrm{SecureFire}(r_i)$
$\equiv (L(e_i) \notin L(r_i))$
$\rightarrow \neg \mathrm{SecureFire}(r_i)$
$\equiv (L(\mathrm{tok}) \notin L(r_i))$
$\rightarrow \neg \mathrm{SecureFire}(r_i)$.

즉, 토큰 보안등급이 동적 규칙의 보안등급에 포함되지 않으면 규칙은 안전성을 보장하는 발화는 이루어지지 않기 때문에 대우명제도 또한 참이 된다. □

따라서 동적 규칙의 발화 보안등급(보안 성질 4)에서 만약 동적 규칙의 보안등급보다 더 높은 토큰이 규칙에 전달되어 쓰기 연산을 실행한다면, 보안수준이 높은 데이타가 낮은 보안등급의 데이타로 처리되어 정보의 노출이 발생할 뿐만 아니라 데이타 내용의 변경이 이루어지므로써 보안정책을 위반하게 된다. 따라서 보안 성질 4와 같이 사건의 보안등급이 규칙의 보안등급에 포함되어야 그림 6과 같이 규칙에 토큰이 전달될 수 있기 때문에 정보가 안전하게 처리될 수 있다.

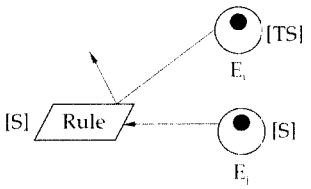


그림 6 동적 규칙의 발화 안전성 예

복합 사건에 대한 다단계 보안 성질들(8, 9, 10, 11)에서도 복합 사건은 하나 이상의 단위 사건들로 구성되어 있기 때문에, 복합 사건의 의미에 따라서 각각의 단위 사건에 각각의 토큰들이 전달되어야만 하나의 복합 사건이 실행될 수 있다. 따라서 복합 사건 보안 성질들은 보안 성질 3과 같이 소정리 5.1을 만족시켜야 정보의 노출을 방지할 수 있고, 두 사건의 보안등급들이 동일해야만 소정리 5.2의 조건을 만족시켜 동적 규칙이 안전하게 발화할 수 있다. 그림 7은 두개의 사건에 각각 토큰이 전달되어야 정의된 사건이 발생하고, 각각의 사건들의 보안등급이 같아야 토큰이 동적 규칙으로 안전하게 전달되는 의미를 나타내고 있다.

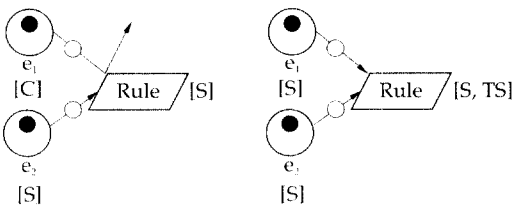


그림 7 동시 복합 사건의 안전성

✣ 소정리 5.3 동적 규칙 실행 안전성

동적 규칙의 실행으로 생성되는 토큰 보안등급은 최소한 동적 규칙의 보안등급과 동일해야만 규칙의 실행은 정보의 노출없이 안전하게 처리된다.

증명 : 동적 규칙에서 처리된 토큰을 p_tok라 가정할 때, 위의 소정리는 다음과 같은 논리적인 형식으로 서술할 수 있다.

P : $\mathrm{SecureFire}(\mathrm{tok}) \rightarrow L(r_i) \in L(\mathrm{p\_tok})$.

명제 P의 대우명제를 취해보면,

대우명제 : $\neg(L(r_i) = L(\mathrm{p\_tok}))$
$\rightarrow \neg \mathrm{SecureTrans}(\mathrm{tok})$
$\equiv L(r_i) \neq L(\mathrm{p\_tok})$
$\rightarrow \neg \mathrm{SecureTrans}(\mathrm{tok})$

동적 규칙의 실행으로 생성된 토큰의 보안등급이 동적 규칙의 보안등급과 동일하지 않으면 정보를 노출 또는 변경시키는 불법적인 규칙의 실행이 된다. 이 의미는 동적규칙이 실행되어 낮은 보안등급을 갖는 토큰을 생성하면 보안수준이 높은 동적 규칙의 존재가 노출되고, 반면에 토큰 보안등급이 높다면 보안 정책(2)를 위반하게 된다. 동적 규칙의 보안등급과 생성된 토큰의 보안등급이 같지 않으면 이 규칙은 정보를 노출 또는 변경시키는 연산을 실행하게 된다. 따라서 대우명제도 역시 참이 됨을 알 수 있다. □

✣ 소정리 5.4 동적 규칙의 실행 결과 안전성

동적 규칙에서 처리된 토큰이 다음 사건 또는 객체 클래스의 보안등급에 포함되어야 규칙의 실행결과는 안전하다.

증명 : 위의 소정리는 다음과 같은 논리적인 형식으로 서술할 수 있다.

P : SecureReceive(p_tok) $\rightarrow L(\text{p\_tok}) \in L(0)$ or $L(\text{p\_tok}) \in L(e_i)$.

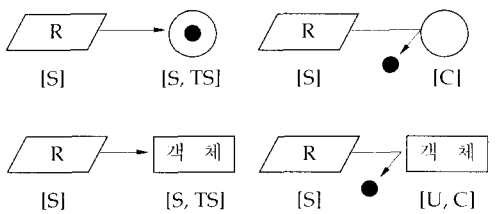
명제 P의 대우명제를 취해보면,

대우명제 : $\neg(L(\text{p\_tok}) \in L(0)$ or $L(\text{p_tok}) \in L(e_i) \rightarrow \neg$SecureReceive(p_tok)

$\equiv L(\text{p\_tok}) \notin L(0)$ or $L(\text{p_tok}) \notin L(e_i) \rightarrow \neg$SecureReceive(p_tok)

즉, 동적 규칙에서 생성된 토큰 보안등급이 객체 클래스 보안등급 또는 다음 사건 보안등급에 포함되지 않으면 동적 규칙이 높은 수준에서 처리한 데이터가 낮은 보안등급의 객체나 사건으로 전달되기 때문에 정보를 안전하게 처리하지 못한다. 따라서 이 대우명제 또한 참이 된다. □

그림 8에서와 같이 규칙에서 처리된 토큰의 보안등급은 객체 클래스 또는 사건 객체로 전달되면 정보의 처리는 안전하게 이루어진다. 따라서 동적 규칙의 인가등급 성질, 연쇄 사건 성질 그리고 연쇄 동적 규칙 성질은 소정리 5.4와 5.1을 만족시키는 조건이 된다.



(a) 규칙의 안전한 실행 (b) 정보 노출 방지

그림 8 동적 규칙의 실행 안전성 예

6. 결 론

개념적인 데이타베이스 스키마를 논리적인 스키마로 변환 후에 동적 기능을 정의하는 방법은 데이타베이스의 동적 행위에 대해서 설계과정에서 고려해야 하는 결정사항들을 설계의 마지막 단계인 구현과정에서 수행한다. 그러나 데이타베이스의 마지막 설계 과정에서는 구현에 대한 복잡도의 증가로 인하여 현실 세계의 동적 행위에 대한 의미들이 모호해지고, 복잡한 프로그래밍언어로 인하여 쉽게 모형화할 수 없다. 따라서 데이타베이스 관리 시스템 수준에서 동적 행위의 설계 복잡도로 인하여 사용자는 동적 규칙, 트리거 그리고 프로시쥬어의 기능을 적절하게 이용할 수 없게 된다. 따라서 데이타베이스 초기 설계 과정에서 동적 데이타베이스 행위를 모형화하므로써 이러한 문제점들을 해결할 수 있으나, 아직까지 새로운 응용의 동적 데이타베이스를 개념적으로 설계하는 도구에 관한 연구는 거의 이루어지고 있지 않다.

따라서 본 논문에서는 동적 데이타베이스를 개념적으로 설계하는 방법론을 제시하여 개념적 데이타베이스 설계 과정에서 쉽게 동적 규칙을 모형화할 수 있는 동적 객체지향 데이타 모델을 제안하였다. 제안된 모델은 단순한 동적 규칙 뿐만 아니라 연쇄적 체인 규칙과 여러가지의 현실적 의미를 갖는 복합 사건을 모형화하는 구조와 각각의 구조에 대한 그래픽 표기법을 제공한다.

최근에 데이타베이스의 보안에 대한 관심이 증대되는 시점에서, 불법적인 정보의 흐름을 방지하기 위해서 기존의 보안 정책을 확장하고, 동적 객체지향 데이타 모델의 모든 구성요소들에 대한 다단계 보안 성질들을 분석하여, 11가지의 다단계 보안 성질들을 정형적으로 정의하였다. 그리고 정의된 동적 규칙의 다단계 보안 성질의 타당성을 제시하기 위해서 패트리네트를 정의하여 동적 규칙의 다단계 보안 성질들이 정보의 불법적인 흐름을 방지함을 증명하였다.

마지막으로, 본 논문은 실세계의 완전한 모형화

방법론을 제시하기 위해서, 데이타의 정적 구조와 동적 규칙에 대해서 클래스 수준의 정의 구조와 이를 모형화하는 완벽한 그래픽 다이어그램을 연구중에 있으며 그리고 아직까지는 고려하지 않았던, 동적 규칙의 상속성 문제와 동적 규칙이 상속될 때 고려해야 하는 다단계 보안 성질들에 대한 연구를 계속해서 수행할 예정이다.

참 고 문 헌

[1] A.K. Tanaka, On Conceptual Design of Active Databases, Ph.D Thesis, Georgia Institute of Technology, 1992.

[2] D.R. McCarthy, U. Dayal, "The Architecture Of An Active Data Base Management System," Proc. the 1989 ACM SIGMOD Int. Conf. on Management of Data, 1989, pp 215-224.

[3] E. Anwar, L. Maugis, S. Chakravarty, "A New Perspective on Rule Support for Object-Oriented Databases," Proc.1993, ACM SIGMOD Int. Conf. on Management of Data, 1993, pp 99-108.

[4] G.Gajnak, "Some Results from the Entity-Relationship Multilevel Secure DBMS Project" Proc. IEEE Aerospace Conference, 1988, pp 66-71.

[5] G.Pernul, "Security Constraints Processing During Multilevel Secure Database Design" Proc. IEEE Computer Security Applications Conference, 1992, pp 75-84.

[6] G.W. Smith, "Modeling Security-Relevant Data Semantics," IEEE Transaction on Sortware Engineering, Vol.17, No.11, 1991, pp 1195-1203.

[7] K.Smith, M. Winslet, "Multilevel Secure Rules : Integrating the Multilevel Secure and Active Data Models," Database Security VI : Status and Prospects, Elsevier Science Publishers, 1993, pp 35-53.

[8] M.B. Thuraisinghum, "Security Query Processing in Intelligent Database Management Systems," Proc. IEEE Computer Security Applications Conference, 1990, pp 204-214.

[9] M.B. Thuraisinghum, "Multilevel secure object-oriented data model-issues on noncomposite objects, composite objects, and versioning," Journal of Object-Oriented Programming, 1990, pp 121-129.

[10] M.B. Thuraisinghum, "Mandatory Security in Object-Oriented Database Systems," Proc. of OOPSLA, 1989, pp 203-210.

[11] M.Morgenstern, " Security and Inference in Multilevel Database and Knowledge-based Systems," Proc. ACM SIGMOD Int. Conf. on Management of Data, 1987, pp 357-371.

[12] N.Gehani, H.V. Jagadish, "Ode an Active Database : Constraints and Triggers," Proc. the 17th International Conference VLDB, 1991, pp 327-336

[13] N.Gehani, H.V. Jagadish, "Active Database Facilities in Ode," Data Engineering, Vol.15, No.1-4, 1992, pp 19-22.

[14] N.H. Gehani, H.V. Jagadish, O. Shmueli, "Event Specification in an Active Object-Oriented Database," Proc. 1992, ACM SIGMOD International Conference on Management of Data 1992, pp 81-90.

[15] O.Diaz, N.Paton, Peter Gray, "Rule Management in Object-Oriented

Databases : A Uniform Approach," Proc. the 17th International Conference VLDB, 1991, pp 317-326.

[16] Oscar Diaz, "Deriving Active Rules for Constraint Maintenance in an Object-Oriented Database," Proc. 1992 ACM SIGMOD International Conference on Management of Data, 1992, pp 332-337.

[17] S.B. Navathe, A.K.Tanaka, S. Chakravarthy, "Active Database Modeling and Design Tools : Issues, Approach, and Architecture," Data Engineering, Vol.15, No.1-4, 1992, pp 6-9.

[18] S. Jajodia, B. Kogan, "Integrating An Object-Oriented Data Model with Multilevel Security," Proc. IEEE Symposium on Research in Security and Privacy, 1990, pp 76-85.

[19] T. Garvey, T.F. Lunt, "Multilevel Security For Knowledge Based Systems," Proc. IEEE Computer Security Applications Conference, 1990, pp 148-159.

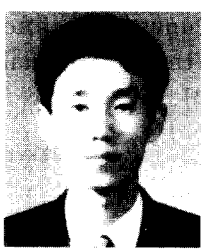
[20] T.F. Lunt, J.K. Millen, "Security for Object-Oriented Database Systems," Proc. IEEE Symposium on Research in Security and Privacy, 1992, pp 260-272.

[21] U.Dayal, A.P. Buchmann, D.R. McCarthy, "Rules Are Objects Too : A Knowledge Model For An Active, Object-Oriented Database System," Advanced Symposium in Object-Oriented Database Systems, 1988, pp 129-143.

[22] U.Dayal, B. Blaustein, U. Chakravarthy, "The HiPAC Project : Combining Active Database and Timing Constraints," ACM SIGMOD RECORD, Vol.17, No.1 1988, pp 51-70.

[23] Vijay Varadharajan, "Petri Net Modeling of Information Flow Security Requirements," Proc. of The Computer Security Foundations Workshop III, IEEE Computer Society Press, 1990, pp 51-61.

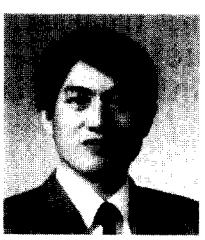
□ 著者紹介



김 영 균(정회원)

1991년 전남대학교 전산통계학과 이학사
1993년 전남대학교 대학원 전산통계학과 이학석사
1993년 ~ 현재 전남대학교 대학원 전산통계학과 박사과정

※ 주관심분야 : 객체지향 데이타베이스, 데이타베이스 보안, 능동 데이타베이스



노 봉 남(종신회원)

1978년 전남대학교 수학교육학과 이학사
1982년 한국과학기술원 전산학과 이학석사
1994년 전북대학교 대학원 전산통계학과 이학박사
1983년 ~ 현재 전남대학교 전산학과 교수

※ 주관심분야 : 객체지향 시스템, 통신망 관리, 정보 보안, 컴퓨터와 정보사회 등