그래픽 기반의 객체지향 표기법을 이용한 정형적인 보안정책 기술 언어[1]

강철범*, 김상현*, 김지영*, 장희진*, 이우진*, 김상욱*, 유동영**, 김형종**

*경북대학교 컴퓨터과학과, **한국정보보호진흥원

A Security Policy Description Language Based on Graphical Object Oriented Notations

Chulbum Kang*, Sanghyun Kim*, Jeeyoung Kim*, Heejin Jang*, Woojin Lee*, Sangwook Kim*, Dongyoung Yoo**, and Hyungjong Kim**

*Department of Computer Science Kyungpook National Univ.
**Korea Information Security Agency

요 약

인터넷의 급속한 확대로 인해 컴퓨터 시스템 또는 네트워크 상에서 보안 문제는 점차 중요한 이슈로 다루어지고 있다. 시스템 또는 네트워크에서 보안 문제를 어떻게 다루느냐는 어떠한 보안정책을 구현하고 있느냐와 밀접하게 연결된다. 이 논문에서는 시스템 또는 네트워크에서 다루어지고 있는 기존 보안정책들을 표현할 수 있는, 그래픽 기반의 새로운 보안정책 언어를 제시한다. 이 언어는 객체 지향 기법에 기반을 두고 있는 그래픽 보안정책 기술 언어이다. 보안정책은 객체 내부 상황 및 객체들 사이 연관성을 표현하여 보안정책이 적용되는 상황을 기술하고 보안정책에서 만족되어야 하는 조건 및 행위 또한 객체 내부와 객체간의 연관성으로 표현된다. 객체지향 보안정책 언어는 그래픽 언어이기 때문에 명확하고 이해하기 용이하다는 장점이 있으며 UML의 클래스 다이어그램과 상태도로 대상 시스템을 표현할 수 있으므로 명시된 보안정책들에 대한 분석을 수행할 수 있다는 장점이 있다.

I. 서론

컴퓨터 시스템 보안을 위하여 다양한 하드웨어 보안 장비 및 보안 소프트웨어를 사용한다. 컴퓨터 시스템과 보안정책들이 통합되고 관리자들은 시스템에 따라 다른 형식의 보안정책을 적용하고 있다. 또한 같은 종류의 보안정책도 시스템에 따라 다르게 표현된다. 이런 보안정책들을 정형화하고 명확하게 분류하는 것이 필요하다. 또한 보안정책의 재사용, 일괄적인 보안정책의 관리를 위하여 보안정책을 독립적으로 표현하는 일반적인 형식 언어가 정의될 필요가 있다.

현재까지 제안된 보안정책 언어에는 분산 시스템의 보안 및 관리를 정책을 정의하는 Ponder[1], 시스템에 발생한 이벤트를 기반으로 정의한 PDL[2], 그래픽 기반의 보안정책을 기술하는 LaSCO[3], 네트워크 트래픽 및 경로 제어 정책을 표현하는 PPL[4] 등이 있다. 기존의 보안정책 형식 언어는 네트워크 혹은 호스트를 대상으로 표현한 언어로서 도메인의 국한을 받는다. Ponder는 호스트에 대한 보안정책을 주로 기술하고 네트워크 환경에서 경로 설정과 같은 경우를 표현하기 어렵다. 또한 PPL은 주로 네트워크 트래픽의 경로설정을 위주로 하는 언어로서 호스트의 의무정책이나 위임정책을 표현하기 어렵다. LaSCO는 그래픽 기반으로 정책을 표현하지만 주로 객체간의 이벤트에 따른 연관성을 표시하고 객체 내부의 다양한 상태 변화를 나타내지 못한다.

본 논문에서 제안하는 보안정책 형식 언어는 객체 지향 기반의 언어이다. 언어의 표현면에서 그래픽기반으로 표현함으로서 이해하기 쉽다. 또한 다양한 객체간의 연관성 관계를 표현할 수 있으므로 네트워크환경과 호스트 등 다양한 도메인에 적용될 수 있다. 시스템 객체의 내부상태 및 객체간의 연관성을 객체의 애트리뷰트와 객체간의 오퍼레이션 호출관계로 명확히 표현할 수 있다. 이 언어는 정책이 적용되는 시스템의 모델 표현, 정책이 적용되는 상황을 나타내는 정책조건, 그리고 정책에서 수행하고자 하는 행위를 나타내는 정책행위로 구성된다.

1. 본 연구는 한국정보보호진흥원 지원으로 수행되었습니다.

본 논문의 2장에서는 보안정책을 분류와 기존의 정책언어에 대해서 분석하고 3장에서는 보안정책의 언어 구성을 살펴보고 이 논문에서 제시하는 보안정책 언어가 가져야 할 구조적인 특성을 정의한다. 4장에서는 새로운 객체지향 보안정책 언어를 제시하고 5장에서 결론을 맺는다.

II. 보안정책 분류 및 기존 정책 언어 분석

2.1 보안정책의 분류

기존의 보안정책언어에서는 다양한 보안정책의 분류 기준이 제시되어 있지만 그 중에서 가장 세분화하여 분류하고 있는 Ponder[1]에서는 인가(authorization) 정책, 의무(obligation) 정책, 제한(refrain) 정책, 위임(delegation) 정책, 복합(composite) 정책의 5가지 정책으로 보안정책을 분류하고 있다[1].

- 인가정책 : 인가정책은 보안을 위한 접근 제어를 명세 한다. 즉 인증되지 않은 행위로부터 대상 객체를 보호하는 정책이다. 이는 주체가 대상에 대해 어떤 행동을 수행하는 허락여부를 정의한다. 예를 들면 "일반 사용자는 패스워드 파일을 읽을 수만 있다."는 인가정책에 속한다.

- 의무정책 : 의무정책은 주체가 대상에 대해 수행해야 하는 행동에 대해서 정의한다. 예를 들면 "시스템 관리자는 매주 토요일 저녁 7시에 사용자 데이터를 백업해야 한다."는 의무정책에 속한다.

- 제한정책 : 제한정책은 주체가 대상에 어떤 행동을 수행할 수 있는 권한을 제한하는 정책이다. 예를 들면 "일반 사용자는 /lib 디렉토리에 파일을 저장하면 안된다."는 제한정책에 속한다.

- 위임정책 : 위임정책은 주체가 다른 사람에게 권한을 위임을 하는 정책이다. 위임정책은 특정 목적을 위해 서버나 제삼자에 권한을 위임하는데 사용된다. 예를 들면 "시스템 관리자는 일요일에 루트 권한을 당직자에게 위임한다."는 위임정책에 속한다.

- 복합정책 : 복합정책은 위에서 설명된 기본 정책들을 복합적으로 사용하기 위한 정책이다. 큰 분산 시스템의 정책을 쉽고 간단하게 관리하기 위하여 서로의 관계가 있는 기본 정책들을 그룹화 한 것이다. 로그인 복합정책은 "사용자가 대상 컴퓨터에 접속하기 위해서는 아이디와 패스워드를 입력하여 로그인하는 인가정책과 로그인 시에는 성공 유무와 무관하게 로그인 에이전트가 사용자 아이디와 컴퓨터 아이디를 기록해야 하는 의무정책"으로 구성된다.

2.2 기존 정책 언어 분석

Ponder는 분산 시스템의 보안 및 관리를 위한 정책을 정의하는데 적합한 언어이다. Ponder는 접근 제어(access control)를 위한 인가, 제한, 위임정책을 지원하고 관리를 위해 의무정책도 지원한다. Ponder는 네트워크, 시스템, 어플리케이션 등과 같은 넓은 범위의 관리 어플리케이션들과 관련된 정책을 정의하는데 적합하다. 뿐만 아니라 대규모의 단체 내에 적용하기에 용이한 혼합형태의 정책들도 제공한다. 이러한 형태의 정책들은 유동성 뿐 아니라 확장성도 제공한다.

Policy Description Language(PDL)은 Bell-Labs에서 개발된 이벤트 방식의 언어이다. 발생한 이벤트에 대해 그에 적합한 행위로 바꾸기 위한 함수로 정책을 정의하기 위해서 데이터베이스의 이벤트-조건-행위 패러다임을 사용한다.

Language for Security Constraints on Objects(LaSCO)는 그래픽 기반의 보안정책을 기술하는 언어이다. 보안정책은 시스템에 대한 적용 상황을 기술하는 도메인(domain) 부분과 실제 제약 사항을 나타내는 요구사항(requirement) 부분으로 나뉘어진다. LaSCO는 대상 시스템의 상태나 이벤트에 제약을 가하는 유형의 보안정책만을 적용된다.

Security Policy Language(SPL)[5]은 이벤트 기반의 정책 언어로서 접근 제어, 히스토리 기반의 정책과 의무 기반의 정책 표현을 지원한다. SPL은 정책 위주의 제약 기반 언어이다. 이것은 엔터티, 집합, 규칙, 정책의 네 가지 기본요소로 구성된다.

Policy Core Information Model(PCIM)[6]은 Distributed Management Task Force(DMTF)에서 정의된 common information model(CIM)을 확장하여 설계하였다. PCIM은 객체 지향의 방법으로 정책을 표현하며 IETF 정보 모델을 정의하여 QoS를 보장하기 위한 네트워크 QoS 자원 관리, 접근을 제어하는 정책을 표현한다. 정책 규칙은 일련의 조건들과 이러한 조건이 만족될 때 일어날 수 있는 일련의 행동들로 구성 되였다.

Path-based Policy Language(PPL)은 네트워크환경에서 트래픽 경로를 지정하여 트래픽에 대한 제어를 표현하는 언어이다. 특정 대상(target)에 대한 트래픽 경로, 조건 및 행동을 명시할 수 있음으로서 다양하고 통합된 네트워크관련 정책을 표현한다.

기존의 정책 언어들은 정책 분류의 모든 정책을 다 표현하는 것은 아니다. 일부 언어는 특정 목적, 환경에 따라 거기에 맞는 정책을 구현하였기 때문이다. 아래의 표 1는 각각의 정책 언어가 어느 정책을 표현할 수 있는가를 보여준다.

표1: 보안정책 분류 비교

정책 언어	보안정책				
	인가	의무	제한	위임	복합
Ponder	O	O	O	O	O
PDL		O			
LaSCO	O		O		
SPL	O	O		O	
PCIM	O	O	O	O	O
PPL	O	O	O	O	

III. 보안정책 언어 구성

기존의 정책 언어들에 대해 보안정책조건을 표현하는 능력 측면에서 객체 내부 상태를 표현할 수 있는지 여부, 객체간의 연관 관계를 표현할 수 있는지 여부를 분석하며, 정책행위 측면에서 하나의 객체에 국한된 객체 행위와 객체간의 연관 관계의 허락/불허 등과 같이 객체 연관성에 관련된 행위를 표현하는지 여부를 분석한다. 표 2는 기존 정책언어의 구문에서 이러한 정책조건 및 정책행위의 표현 능력을 보여준다.

표에서 알 수 있듯이 Ponder와 PCIM, PPL 등은 모든 정책 적용 조건과 정책행위에서 모두 객체 내부의 상황과 객체간의 연관성을 표현하지만 나머지 정책 언어는 부분적으로만 지원이 이루어지고 있다.

본 논문에서 제안할 정책 언어는 정책조건과 정책행위 표현에서 정책조건과 정책행위에서 객체내부 상황과 객체간의 연관성을 모두 표현한다.

표 2: 기존 정책 언어의 구문 분석

정책언어	정책조건		정책행위	
	객체상태	객체연관성	객체상태	객체연관성
Ponder	O	O	O	O
PDL	O		O	
LaSCO	O	O		O
SPL	O	O	O	O
PCIM	O	O	O	O
PPL	O	O	O	O

IV. 객체지향 보안정책 언어

객체지향 보안정책 언어(OOSPL)는 대상 시스템이 객체지향 모델링 기법을 통해 기술되어 있다고 가정하고 시스템 모델을 참조하여 보안정책이 적용될 상황을 나타내는 정책조건과 보안정책에서 행해지는 정책행위를 기술한다.

4.1 시스템 모델

보안정책을 표현하기 위해서는 대상 시스템의 객체, 객체의 상태, 이벤트 등을 참조하게 된다.

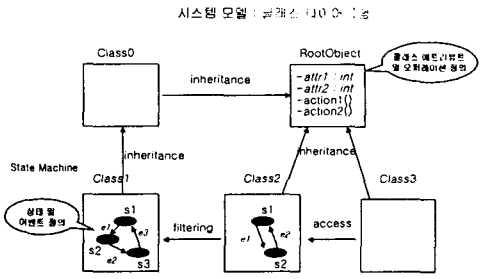

그림 1: 대상 시스템의 클래스 모델

보안정책 표현에 참조되는 객체, 상태, 이벤트 등은 시스템 모델에 미리 정의되어 있어야 한다. 객체 지향 시스템 모델링에는 주로 표준화된 표기법인 Unified Modeling Language (UML)[7]을 이용한다. 대상 시스템에 존재하는 객체와 객체간의 연관성은 클래스 다이어그램으로 정의된다. 그리고 객체의 상태와 이벤트는 상태도(State Diagram)에서 정의된다.

그림 1에서와 같이 객체는 클래스로 표현된다. 클래스는 클래스가 가지는 클래스 변수를 선언한다. 그리고 객체의 행위를 클래스 오퍼레이션으로 정의한다. 객체의 상태는 상태도로 표현되며 객체가 가지는 상태들과 상태간 전이를 명확하게 정의한다. 클래스 다이어그램과 상태도에 정의된 객체, 객체의 애트리뷰트, 객체의 오퍼레이션, 객체의 상태 등은 보안정책 기술 시에 참조된다.

4.2 보안정책의 표현

보안정책은 보안정책을 적용하는 상황을 나타내는 정책조건 부분과 실제 정책의 요구사항을 나타내는 정책행위 부분으로 구성된다. 정책조건은 객체의 내부상태, 객체간의 연관성으로 정책이 적용되는 상황을 표현한다. 정책조건은 두 개의 객체로 국한하지 않고 여러 객체간의 연관성을 나타낼 수 있다. 그리고 정책행위는 시스템의 기능을 제한하는 제약행위와 가능하지 않았던 혹은 존재하지 않던 기능을 추가적으로 가능하게 하는 첨가행위로 나뉠 수 있다. 제약행위는 Ponder의 정책 분류에서 부정적 인가정책과 제한정책에 해당하며 첨가행위는 긍정적 인가정책, 의무정책, 위임정책이 해당된다. 그림 2는 제약행위를 나타내고 있다.

그림 2에서 시스템의 행위는 Student는 ScoreDB를 검색할 수 있다는 상황을 보여준다. 여기에서 적용되는 보안정책은 "해당 과목을 수강하는 학생만 해당 과목 성적을 검색해야 한다"라는 정책이다. 이것을 OOSPL로 나타내려면 우선 관련된 객체의 인스턴스를 원으로 나타내고 참조되는 애트리뷰트를 명시한다. 그림에서는 두 객체 모두 class 라는 애트리뷰트가 $D, SS 라는 변수값으로 바인딩되어 있다. 그리고 이러한 애트

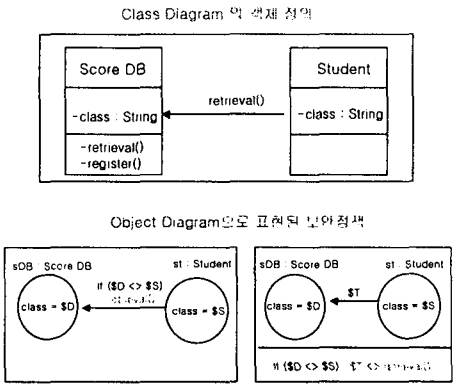

그림 2: OOSPL의 보안정책의 예

리뷰트값에 따라 retrieve()라는 메소드가 사용을 제한하고 있다. 보안정책을 나타내는 방법은 그림의 왼쪽처럼 객체호출 아크 위에 간단히 제약조건을 표현할 수도 있고 오른쪽과 같이 독립적인 공간에 제약조건을 표현하기도 한다. 제약조건이 복잡하거나 복합적인 조건문이 있는 경우는 독립적인 공간으로 표현하는 것이 편리하다. 보안조건 및 보안정책 부분에서 표현되는 제약사항들은 UML의 OCL(Object Constraints Language)로 표현된다. OCL의 클래스 다이어그램에서 클래스가 가지는 불변사항(invariant), 오퍼레이션의 선,후조건 등을 표현하는데 널리 이용되고 있다.

4.3 OOSPL 보안정책 표현 예제

이 절에서는 보안정책을 OOSPL로 나타내는 예제를 보인다. 로그인에 관련된 정책으로 "사용자는 특정 호스트를 접근하기 위해서는 로그인 과정을 거쳐야 하며 로그인 시 사용된 사용자 ID와 패스워드는 로그로 남겨야 한다."라는 인가정책 및 의무정책이 같이 있는 복합정책이 있다고 가정하자. 이러한 로그인 정책을 정책에서 OOSPL로 나타내면 그림 3과 같다. 사용자 P는 컴퓨터 C를 접근하기 위해 C.login() 이라는 메소드를 이용하여야 하며 C.login(id, passwd) 행위가 발생하면 C.log(id, passwd) 가 수행되어야 함을 나타낸다.

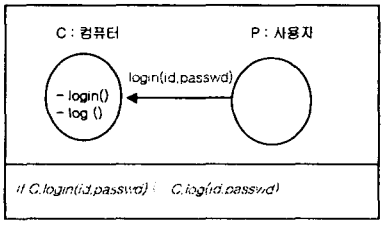

그림 3: 로그인 정책

다른 예제로, "외부 머신에서 특정 호스트 머신을 이용하기 위해서는 반드시 FireWall을 통해야만 하며 FireWall에서는 미리 정의된 IP들은 통과시키지 않아야 한다." 라는 네트워크 정책이 있다고 가정하자. 이러한 정책을 OOSPL로 표현하면 그림 4와 같다. 그림 4에서는 외부머신을 나타내는 E에서는 H로 곧바로 access()를 수행할 수 없으며, 항상 F.access()를 통하여 접근하여야 함을 나타내고 F.filter() 함수가 만족되어야만 접근이 허용됨을 나타낸다. F.filter() 함수는 접근이 허용되는 IP는 true를 불허하는 IP는 false를 리턴하는 함수라고 가정한다.

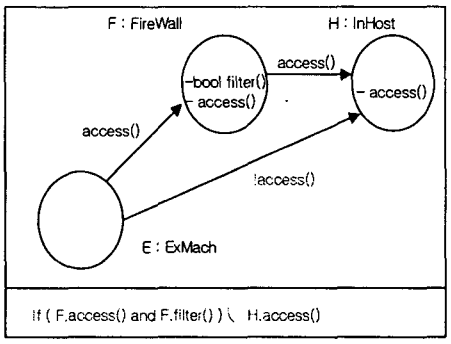

그림 4: Firewall을 통한 접근 제한정책

앞서 설명한 예제에서는 주로 객체간의 연관성 위주로 설명되었지만 상태도에서의 상태 또는 객체의 애트리뷰트 값 등의 객체 상태값으로 보다 복잡한 정책조건 및 정책행위를 표현할 수 있다.

V. 결론 및 향후 연구

시스템 또는 네트워크 보안정책들을 정형적으로 표현할 수 있는 그래픽 기반의 객체지향 보안정책 언어를 소개하였다. 객체지향 보안정책 언어는 다양한 도메인의 특성을 나타내고 객체의 상태, 객체간의 연관성 등을 통해 정책조건 및 정책행위를 표현한다. 본 논문에서는 제안한 OOSPL은 그래픽 기반의 객체지향 모델링 표기법인 UML의 클래스 다이어그램과 OCL 등의 표준화된 표기법을 이용하여 간결하고 명확하게 보안정책들을 나타낸다.

현재까지의 연구는 보안정책들을 표현하는데 중점을 두고 있으며 향후 과제로는 이러한 보안정책을 수행하는 프레임워크 환경과 보안정책들을 분석할 수 있는 다양한 분석 방법 및 시뮬레이션 환경 등에 관한 연구 등이 있다. 그리고 보안정책을 확장 및 응용하여 공격 시나리오를 표현하고 이를 바탕으로 시스템의 생존성을 평가할 수 있는 방법에 대해 연구할 예정이다.

참고문헌

[1] Nicodemos Damianou, "Ponder: A Language for Specifying Security and Management Policies for Distributed Systems", *Workshop on Policies for Distributed Systems and Networks(LNCS #1995)*, Jan. 2001, pp 18-39.

[2] Lobo,J., R.Bhatia, "A Policy Description Language," AAAI, 1999.

[3] Hoagland, J.A., Pandey, "Security Policy Specification Using a Graphical Approach," *Technical Report CSE-98-3,* UC, 1998.

[4] Stone,G.N., Lundy, ET AL. "Network Policy Languages: A Survey and a New Approach" IEEE Network, 2001, pp. 10-20.

[5] Ribeiro,C., A Zuquete, "SPL: An access control language for security policies with complex constraints," NDSS, 2001.

[6] Moore,B., E.Ellesson, ET AL "Policy Core Information Model-Version 1 Specification." http://www.ietf.org/rfc/rfc3060.txt.

[7] OMG, "UML specification 1.4", http://www.omg.org/cgi-bin/doc?formal/01-09-67, Sept. 2001.