보안요구사항 도출 및 명세를 위한 CC기반 Misuse Case 모델

장세진[O]* 최상수* 이강수* 최희봉**
한남대학교 컴퓨터공학과, 국가보안기술연구소
{dar[O], gcss09}@se.hannam.ac.kr, gslee@mail.hannam.ac.kr, hbchoi@etri.re.kr

A CC-Based Misuse Case Model for Security Requirements Derivation and Specification

Sejin Jang[O]*, Sangsoo Choi*, Gangsoo Lee*, Heebong Choi**
*Dept of Computer Engineering, Han-Nam University
**National Security Research Institute

요 약

정보화가 가속화되면서 정보시스템은 우리의 일상생활에서 점점 더 중요한 요소로 자리 잡아 가고 있으며 정보시스템에 대한 의존도 커져가고 있다. 이에 따라 정보시스템은 우리의 생활에서 없어서는 안 될 보호되어야 할 주요 자산으로 여겨지고 있다. 본 논문에서는 시스템의 신뢰성과 안전성을 제고 하기 위하여 보안요구사항 도출 및 명세를 위한 CC 기반 Misuse Case 모델을 제시한다.

1. 서 론

정보화 사회가 급격히 발전하면서 다양한 정보 서비스를 통해 우리의 생활 전반에 걸쳐 편의성을 제공하고 있지만, 이러한 이면에는 인터넷 웜, 바이러스, 해커, 사이버 테러 등과 같은 위협요소가 발생하고 있어 정보화의 역기능에 노출되고 있다. 정보보호기술은 정보시스템을 이와 같은 위협요소로부터 보호하기 위한 것이며 정보보호 평가·인증 체제는 정보보호기술 및 정보보호 제품에 대한 신뢰성 및 안전성을 검증하기 위한 노력이다. 이러한 정보보호 평가·인증체계는 일부 선진국에서 전 세계적으로 확산되고 있으며 서로 상이한 평가·인증 체계를 국제적으로 표준화 한 것이 공통평가기준(Common Criteria: 이하 CC로 칭함)이다[1].

CC는 IT 제품 및 시스템의 보안성을 평가하기 위한 기초가 되는 기준을 정의하며 특정의 보안요구사항을 설정하기 위하여 클래스-패밀리-컴포넌트로 구성된 공통의 보안기능요구사항 및 보증요구사항을 제공한다. 보안요구사항이란 비 기능요구사항으로써 보호되어야할 자산 및 서비스와, 이러한 자산 및 서비스가 보호해야 하는 보안 위협에 대한 분석을 기초로 하여 분석된 보안위협을 완화시키기 위한 보안관련 요구사항들을 말한다[2]. 이러한 보안요구사항은 정보시스템의 의존도 및 중요도가 커져가면서 보다 중요시되고 있으며 이를 프로젝트 초기단계인 요구사항 분석단계에서 도출 및 명세 하려는 노력이 활발히 진행 중에 있다[2,6].

이러한 배경에서 본 논문에서는 CC체제 하에서 보안요구사항명세서 작성 시 보안요구사항 도출 과정을 보다 명확히 하고 보안요구사항을 분석할 수 있도록 UML(Unified Modeling Language)의 UC(Use Case)를 확장한 MUC(Misuse Case)를 이용한 보안요구사항 도출 및 명세 방법을 제시한다. 2장에서는 관련 연구로 CC의 개념 및 보안요구사항 도출과정과 MUC의 개념을 설명하고, 3장에서는 보안요구사항 도출 및 명세를 위한 CC기반 MUC 모델을 제안한다. 그리고 4장에서는 본 모델을 통한 적용 사례를 보여 주며, 5장에서 결론을 맺는다.

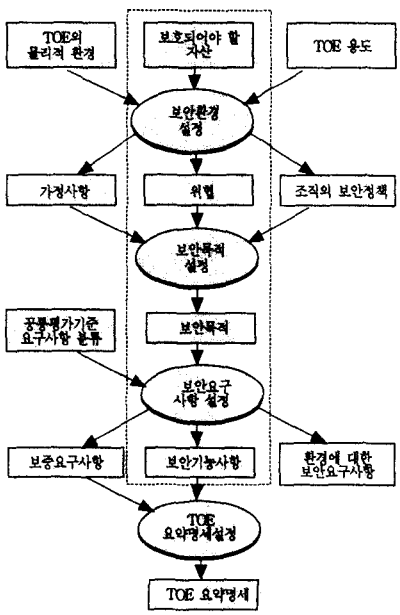
2. 관련연구

본 장에서는 CC의 개념과 CC에서의 보안요구사항 및 명세 유도과정에 대하여 기술하며, UC를 확장한 보안요구사항 도출 및 명세 방법을 소개한다.

2.1 CC의 개념 및 보안요구사항 도출과정

CC는 모든 정보보호시스템에서 필요로 하는 보안기능요구사항의 전체집합을 계층적으로 분류하고 있으며, 보안기능에 대한 구현의 정확성에 대한 보증요구사항의 전체집합을 계층적으로 분류하고 있다. CC에서 제시된 전체 보안기능 및 보증 요구사항으로부터 제품군별 공통보안요구사항인 보호 프로파일(PP: Protection Profile)과 특정 보안 제품별 보안요구사항인 보안목표명세서(ST: Security Target)를 개발하여 정보보호제품을 개발 및 평가하고 있다.

CC는 <그림 1>과 같이 여러 단계의 표현들로 계층화되어 있다. 이 그림은 보안요구사항 명세서 작성 시 보안요구사항과 보안명세를 도출하는 방법을 나타낸 것이다[1]. 이러한 보안요구사항은 항상 하향식(top-down)으로 개발되어야하며, 보안의 필요성을 정의하여야 한다. 그리고 필요성을 설명하기 위하여 보안목적을 파악하고 이를 통해 보안목적을 만족하기 위



<그림 1> CC에서의 요구사항 및 명세 유도과정

한 보안요구사항을 정의해야한다[7].

2.2 Misuse Case 모델

본 절에서는 기존에 제안된 MUC를 이용한 보안요구사항 분석 및 명세 모델들에 대해 조사·분석 한다.

(1) Sindre&Opdahl의 Misuse Case 모델

Sindre&Opdahl은 시스템이 허용해서는 안되는 기능이라 하더라도 여전히 기능에 해당하며 이것은 잠재적으로 UC에 의하여 다루어질 수 있다고 분석하였으며, 비기능 요구사항 중에서 보안요구사항에 초점을 맞추어 UCD를 확장한 MUCD의 개념을 최초로 제기하였다[2].

(2) McDermott의 Abuse Case 모델

McDermott는 그의 연구에서 기존의 수학적 보안모델의 문제점을 인식하고 보안에 대한 전문지식 없이도 간단하게 보안요구사항을 분석할 수 있도록 UCD 모델을 확장한 Abuse Case 모델을 제시하였다. Abuse Case란 시스템과 하나 또는 그 이상의 행위자들 사이에서 시스템 혹은 행위자에게 해로운 결과를 초래하는 상호작용의 유형에 대한 명세라 정의된다[3-5].

(3) Firesmith의 Security Use Case 모델

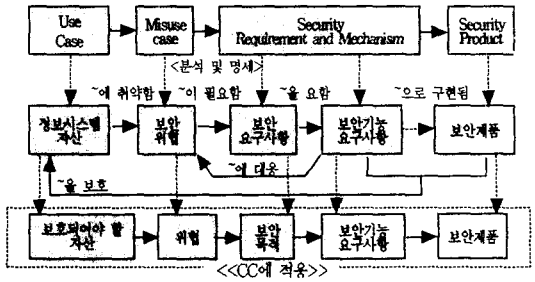
Firesmith는 그의 연구에서 기존의 연구들의 문제점 즉, MUCD 모델 및 Abuse Case 모델은 UCD를 이용하여 실제 보안요구사항 대신에 불필요한 보안 메커니즘에 대하여 명세하고 있다는 문제점을 해결하기 위하여 Security Use Case 모델을 제시하였다. 즉, 보안요구사항이란 보호되어야 할 자산(혹은 서비스)과 이러한 자산이 보호해야 하는 보안 위협에 대한 분석을 기초로 하여 도출되어야 한다. 따라서, 기존의 연구들은 보안 메커니즘에 초점을 두고 있으나 상대적으로 보안위협 및 보안요구사항에 대해서는 분석 및 명세 방법이 미비하다고 지적하였다. 특히, MUCD는 보안위협 분석 및 명세를 위한 모델이라 할 수 있으며, 이러한 MUCD를 다시 확장하여 분석된 보안위협을 막기 위한 보안요구사항을 분석하기 위하여 접근통제, 무결성 및 프라이버시 등의 Use Case를 사례로 하여 분석용 템플리트를 제시하였으며, 분석 가이드라인을 제시하였다[6].

그러나, Firesmith가 그의 논문에서 지적한 것과 같이 기존의 MUCD 기반의 보안요구사항 분석 및 명세 모델은 실제적인 보안요구사항 보다는 보안위협이나 보안 메커니즘에 대한 분석을 위한 방법이라 할 수 있으며, 구체적으로 보안위협 또는 보안위협에 대응하기 위한 보안요구사항, 그리고, 보안요구사항을 구현하기 위한 보안기능요구사항(보안메커니즘)의 분석 및 명세 방법에 대해서는 언급하지 않고 있다.

3. CC기반의 보안요구사항 명세 및 도출을 위한 프로세스 및 모델

3.1 보안요구사항 도출 및 명세 프로세스

<그림 2>는 Firesmith가 제안한 "보안 위협, 목적, 메커니즘"을 확장하여 CC기반의 MUC를 이용한 보안요구사항 도출 및 명세 프로세스를 도식화 한 것이다. Use Case는 보호되어야 할 정보시스템 자산에 해당하며 Misuse Case는 정보시스템의 취약점 또는 보안위협에 해당한다. 보안 요구사항과 보안메커니즘은 CC에서의 보안목적과 이를 구현하기 위한 보안요구사항과 대응된다. 본 프로세스는 <그림 1>의 점선으로 표기된 프로세스를 확장하여 MUC에 적용한 것이며 3.2절에서 제시할 모델에 대한 프로세스를 제공한다. 또한, <그림 2>에서는 CC에서의 각 단계와 MUC에서의 각 단계의 관계를 보여 주며 각각의 요소에 대한 관계를 표현하였다.



<그림 2> MUC를 이용한 보안요구사항 도출 및 명세 프로세스

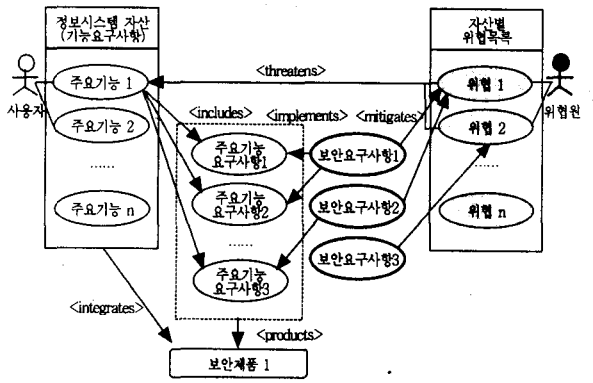
3.2 보안요구사항 도출 및 명세 모델

본 논문에서는 보안요구사항의 도출 및 명세를 위하여 기존의 UC와 MUC 모델을 확장하여 다음과 같이 확장된 MUC 모델을 정의한다. <그림 3>은 이와 같은 정의로 구현된 모델을 나타낸다.

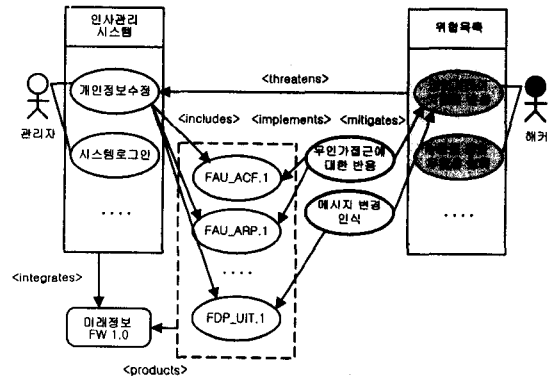
- Actor : 정보시스템 자산의 사용자들이 해당 정보시스템 자산과 상호작용 시 수행할 수 있는 사용자들의 역할의 집합(즉, 정보시스템 사용자)이다.
- Mis-Actor : 특수한 형태의 Actor로써 Misuse Case를 발생시키는 행위자(즉, 위협원)로써, Actor의 반전된(음영처리) 형태로 표시한다.
- Use Case : 정보시스템 자산의 소유자가 제공하기를 원하는 행위에 대한 설명(즉, 기능 요구사항) 또는, 분석된 보안위협을 완화시키기 위한 보안요구사항을 구현하는데 필요한 행위에 대한 설명(즉, 보안기능 요구사항)이다.
- Misuse Case : 정보시스템 자산의 소유자가 발생하기를 원하지 않는 행위에 대한 설명(즉, 보안위협)으로써, Use Case의 반전된(음영처리) 형태로 표시한다.
- Security Use Case : 정보시스템 자산의 소유자가 분석된 보안위협을 완화시키기 위하여 필요로 하는 행위에 대한 설명(즉, 보안요구사항)으로써, Use Case의 테두리를 진한 선으로 처리하여 표시한다.
- Security Product Case : 정보시스템 자산에 대하여 분석된 보안위협을 완화시키기 위하여 필요한 보안기능 요구사항들을 제품화(즉, 보안제품 또는 정보보호제품)한 것으로써, 끝이 둥근 사각형으로 표시한다.

제시한 확장된 MUC 모델의 6가지 구성요소 사이에 발생할 수 있는 관계(relation)는 다음과 같다.

- threatens : 해당 Use Case(기능 요구사항)는 해당 Misuse Case(위협)로부터 위협을 받음을 의미한다.
- mitigates : 해당 Security Use Case(보안요구사항)는 해당 Misus Case(위협)를 완화시킴을 의미한다.
- implements : 해당 Security Use Case는 해당 Use Case(보안기능 요구사항)로 구현될 수 있음을 의미한다.
- includes : 해당 Use Case(기능 요구사항) 또는 정보시스템 자산은 해당 Use Case(보안기능 요구사항)를 포함할 수 있음을 의미한다. 이 경우는, 정보시스템 개발 단계에서 적용할 수 있는 방법이다.
- products : 해당 Use Case(보안기능 요구사항)의 집합은 해당 Security Product Case(보안제품)으로 제품화 될 수 있음을 의미한다 즉, 해당보안기능 요구사항들이 포함된 보안제품을 제시하는 방법이다.



<그림 3> MUC를 이용한 보안요구사항 도출 및 명세 모델



<그림 4> 모델의 적용

정보시스템은 위협원에 의해 위협 대상이 되며 보안요구사항은 이러한 위협을 완화시켜 주는 역할을 제공한다. 그리고 이러한 보안요구사항은 CC의 보안기능요구사항으로 구현되어 보안기능을 명세하게 된다. 이러한 절차에 의해 시스템의 보안기능요구사항을 도출 및 명세 할 수 있으며 정보시스템에 보안기능을 만족시킬 수 있는 보안제품을 제시 할 수 있다.

4. 사례연구

<그림 4>는 본 논문에서 제시한 모델을 이용한 보안요구사항 도출과정을 특정시스템인 인사관리 시스템에 적용한 예이다. 인사관리 시스템의 관리자는 자신의 권한에 할당된 여러 가지 기능을 갖고 있으며, 이러한 기능들에 대하여 해커의 엿보기, 정보수정, 관리자 자신의 태만 등의 여러 가지 위협을 갖고 있다. 인사관리시스템과 위협목록 사이에는 <threatens> 관계연산자가 적용되어 위협목록은 인사관리 시스템의 오용사례 즉 Misuse case가 된다. 이러한 위협을 완화시키기 위하여 무인가 접근에 대한 반응, 메시지 변경 인식, 탬퍼링에 대한 저항 등의 여러 보안목적을 갖을 수 있으며 이와 같은 보안요구사항은 <그림 2>에서 나타난 것처럼 CC에서의 보안목적에 해당한다. 보안요구사항은 보안기능요구사항과 위협에 대하여 각 각 <implements>, <mitigates> 연산자가 적용된다. 즉 보안목적은 보안기능요구사항들로 구현되어 위협을 완화 시켜준다. 이때 사용되는 보안기능요구사항이 정형성과 일관성을 위해 제안된 CC의 보안기능요구사항이다. 그리고 보안기능요구사항을 만족하는 제품을 도출하기 위해 <product>연산자를 사용하였고 이러한 제품을 시스템에 통합하기 위하여 <integrates>연산자를 사용하였다. 이는 인사관리 시스템이 보안기능요구사항을 모두 포함하지 않을 때 고려되어 질 수 있다.

5. 결 론

전통적인 요구사항 분석 모델인 UML의 UC의 모델은 기능의 분석 및 명세에는 매우 효과적이지만 보안요구사항과 같은 비 기능요구사항의 분석 및 명세에는 적합하지 않다. 때문에 UC모델을 확장한 MUC 모델에 관한 연구가 활발히 진행되고 있으나 이들 모델은 보안위협의 분석 및 명세에 초점을 맞추고 있으며, 보안위협과 보안요구사항, 보안기능요구사항 사이에 대한 명확한 근거를 제시하지 못하고 있다. 이러한 배경에서 본 논문에서는 기 발표된 MUC 모델을 확장하여 CC기반의 보안요구사항 도출 및 명세를 위한 프로세스 및 모델을 제안하였으며, 이러한 프로세스 및 모델은 보안요구사항 작성시 보다 정형적이고 일관성 있는 명세 방법을 제공할 수 있을 것이다. 본 모델은 정보 시스템을 구축하기 위한 프로젝트 수행 시 요구사항 분석단계에서 사용되어 질 수 있으며, 또한 이미 구축되어진 정보 시스템을 재구성 할 때 본 모델이 적용 될 수 있다.

6. 관련연구

[1] CC, Common Criteria for Information Technology Security Evaluation, Version 2.1, CCIMB-99-031, August 199 www.commoncriteria.org

[2] G. sindre, A. L. OPdahl, "Capturing Security Requirements through Misuse Cases", Proc. 14th Norwegian Informatics Conference, Troms ϕ, Norway, PP.26-28, Nov, 2001

[3] J. McDermott, "Eliciting Security Requirements by Misuse Cases Proc. 37th Technology of Object-Oriented Languages an Systems(TOOLS-37 Pacific 2000), Sydney, Australia, pp.120-1 20-23, Nov 2000.

[4] J. McDermott, C. Fox, "Using Abuse Case Models for Securit Requirements Analysis," Proc. Annual Computer Security Applications Conference(ACSAC'99), Dec 1999.

[5] J. McDermott, "Abuse Case Based Assurance Arguments," Pro 17th Annual Computer Security Applications Conference(ACSAC'01), 2001.

[6] Donald G. Firesmith, "Security Use Cases," Journal of Objec Technology (JOT), 2(3), Swiss Federal Institute of Technolo (ETH), Zurich, Switzerland, pp.53-64, May/June 2003.

[7] ISO/IEC PDTR 15446, "Information technology - Securit techniques - Guide for the production of protection profiles and security targets," Draft, Apr 3, 2000.

[8] "정보보호시스템 평가/인증 가이드", 한국정보보호진흥원, 2002.12.

[9] CC, Common Evaluation Methodology, Version 1.0,CEM-99/045, August 1999, www.commoncriteria.org

[10] Final Interpretations, http://www.commoncriteria.org/ docs/PDF/CCPART1 V21.PDF.

[11] NIAP, List of Threat, Attack, Policy, Assumption, and Evnironment Statement Attribute, CC Profiling Knowledge base Report, 2002.