보안 운영체계의 개발 과정 중 보안성 검증방법

강호석[0*] 김남창* 심영철* 이진석** 장인숙**
홍익대학교*, 국가보안기술 연구소
*{hskang[0], nckim, shim}@cs.hongik.ac.kr **{jinslee, jis}@etri.re.kr

Security Evaluation Method for Secure Operating Systems during Development Process

Hoseok Kang[0*] Namchang Kim* Youngchul Shim* Jinseok Lee** Insook Jang**
*Dept. of Computer Engineering, Hongik University
**National Security Research Institute

요 약

표준으로 정의된 IT제품의 보안성 평가 방안들은 CC나 ITSEC, TCSEC과 같이 많이 나와있다. 그러나 이러한 표준은 너무 추상적이고 소프트웨어의 각 단계에 따른 보안성을 측정하는데 불충분하다. 또 SSE-CMM같은 프로세스를 평가하는 방법의 경우 역시 너무 추상적이고 포괄적인 면만을 설명하였고 프로세스 자체에 대한 평가에 중점을 두었을 뿐 각 단계의 보안강도를 평가하는데는 부족하다. 이 논문에서는 이러한 단점을 보안하고자 각 공정단계에서, 특히 제품을 계획하는 단계인 요구분석단계와 디자인 단계에서 그래프 모델링과 모듈 다이어그램 통한 검증방법을 제시하였다. 여기서는 보안 운영체계라고 특정 제품을 명시하였지만 실제로 다양한 IT제품에 적용될 수 있다.

1. 서 론

IT 제품의 보안성을 평가하는 방법으로는 ITSEC(Information Technology Security and Evaluation Criteria) TCSEC(Trusted Computer System Evaluation Criteria)을 비롯하여 표준으로 정의된 CC(Common Criteria)등이 있다. 이러한 표준문서의 경우 제품이 완성되면 그 제품을 만든 과정의 문서와 제품을 대상으로 평가를 하게 된다. 또 다른 관점으로 CMM(Capability Maturity Model)과 같이 개발공정 자체를 평가하는 방법도 있다. 이러한 방법들을 바탕으로 하여 운영체계라는 특정한 분야에서 각 개발 단계별로 보안등급을 평가할 수 있는 방법을 제시하고자 한다.

보안 운영체계의 신뢰성 및 안정성을 제공하려면 보안 운영체계의 개발을 진행할 때에 각 소프트웨어 공학의 공정 단계별로 발생되는 중간 결과물과 최종적으로 발생되는 보안 운영체계에 대한 평가가 필요하다. 또한, 각 공정 단계를 진행함에 있어서 개발 공정에 대한 효율성과 안전성을 입증할 만한 근거와 평가도 필요하다.

이러한 각 공정단계별 평가를 위하여 요구분석, 디자인, 구현, 그리고 시험의 4단계로 분류를 하고 각 단계에 대하여 검증 작업을 수행하게 된다. 요구분석과 디자인은 구현에 의한 결과가 나오기 전 단계로 이를 검증하기 위하여 모듈 다이어그램을 이용한 분석방법을 제시하였고 구현과 검사의 단계에서는 결과물(TOE : Target of Evaluation)를 이용한 검증을 수행한다.

여기서 검증이라는 용어를 사용한 것은 표준으로 정의된 문서가 아니고 아직 구체적으로 평가에 적용될 만큼은 아니지만 개발이나 평가를 위한 방안으로 제시하기 위하여 검증이라 칭한다.

이 논문의 구성은 2장에서 관련연구로 대표적인 평가방법인 CC와 SSE-CMM에 대하여 알아보고 3장에서 우리가 제시하고자 하는 공정단계별 평가방법에 대하여 설명하고 4장에서 결론을 맺는다.

2. 관련연구

이 장에서는 국제적인 표준 평가 방안으로 사용되고 있는 CC와 보안제품을 단계별로 평가한 SSE-CMM에 대하여 알아본다.

2.1 CC

CC는 국제사회에서 널리 이용할 수 있는 IT 보안성 평가를 위한 기준 마련하기 위해 개발되었다. 그 전에는 미국에서 TCSEC이 사용되고, 유럽에서는 ITSEC이, 캐나다에서는 CTCPEC(Canadian Trusted Computer Product Evaluation)이 사용되는 등 여러나라가 자기만의 표준을 가지고 있었고 표준 평가방법의 필요성에 의하여 ISO(International Standard Organization)에서 CC를 개발하였다.

CC는 제품의 모든 결과물인 TOE가 만들어 진 후 제품 및 시스템을 개발하는 중간 산출물과 결과물을 평가기관에 보안성을 검사를 요구하고 평가기관은 이 결과물에 대한 보안 기능을 보증하는 방법을 취한다.

CC문서는 CC에 대한 소개와 구성, 그리고 일반적인 구성을 담은 Part 1과 보안기능요구사항을 담고 있는 Part 2가 있고, Part 3에는 보안보증요구사항의 내용의 3부분으로 구성되어있다.

CC를 이용하여 평가를 하기 위해서는 PP(Protection Profile)이라고 불리는 보호 프로파일을 만들어야 한다. 이 파일의 구성은 여러 보안 기능요구사항과 보증요구사항을 포함하고 있는 문서이다. ST는 PP를 만들고 난 후에 이 PP들을 이용하여 개발자가 만드는 문서이다. 이런 PP와 ST는 클래스와, 패밀리, 컴퍼넌트로 구성되어있고 CC 문서의 Part2와 Part3는 기능, 보증 요구사항을 이러한 클래스의 형태로 설명하고 있다.

평가 결과로 나오는 CC의 보증은 EAL(Evaluation Assurance Level)로 표시되며 총 7단계로 등급이 나누어져 있다.

2.2 SSE-CMM

SSE-CMM은 CMM에서 파생된 SE-CMM을 기반으로 하여 보안 제품을 개발할 때에 시스템 공학적인 측면에서 보안 제품을 어떻게 개발할 것인가에 대한 지침을 제공한다. SSE-CMM은 보안 제품의 요구사항, 개발 과정에서 필요한 요구사항 및 고려하여야 할 사항들을 포함하고 있고, 보안 제품을 개발하기 위한 프로젝트를 진행할 때에 필요한 프로젝트 조직의 구성까지도 포함한다. 이러한 사항들의 보증을 통하여 보안 제품 개발 과정의 신뢰성 및 개선 방안을 제공할 수 있고, 올바르고 효과적인 개발을 통하여 보안 제품에 대한 신뢰성을 제공할 수 있다.

SSE-CMM의 소프트웨어 공학의 개념에 보안 공학의 개념을 추가하여 보안 소프트웨어 개발 전반에 걸쳐서 필요한 내용을 학문적으로 연구하였다. SSE-CMM에서는 보안 공학의 진행 단계를 3가지로 나누어 설명하고 있다.

- 위험 분석 단계
- 공정 단계
- 검증 및 시험 단계

위험 분석 단계에서는 위험을 분석하기 위해서 취약점 및 위협에 대한 분석을 하여 위험 목록을 만들게 되고, 이 목록은 공정 단계에서 시스템에 적용된다. 공정 단계는 디자인 및 구현 단계를 포함한다. 마지막인 검증 및 시험 단계에서는 개발된 제품의 보안성, 성능 등을 평가하고, 검증한다.

SSE-CMM은 공정 및 프로젝트 조직을 나타내기 위한 영역(Domain)과 공정 및 프로젝트 조직의 성숙도를 나타내기 위한 능력(Capability)의 측면으로 나누어서 보안 공정을 분석한다. 영역은 크게 22개의 공정 분야(Process Area; PA)로 이루어져 있고, 공정 분야는 다수의 기반 실무(Base Practice)로 구성되어 있다. 능력은 성숙도 단계로 이루어져 있고, 각 단계는 보편 실무(Géneral Practice; GA)로 구성된다.

SSE-CMM은 이러한 영역과 능력의 측면을 이용하여 각 공정 분야의 기반 실무에 대한 조직의 능력을 나타내기 위해서 기반 실무와 보편 실무를 매트릭스 구조를 이용하여 조직 능력의 프로파일을 생성할 수 있고, 이러한 프로파일을 이용하여 조직의 보안 공정을 평가한다.

2.3 그 외의 보안 평가 방법론

CC와 SSE-CMM외에 가장 먼저 적용된 ITSEC, TCSEC이 있고 미국에서 보안제품을 평가하기 위하여 만든 IATF(Information Assurance Technical Framework)가 있다.

IATF는 NSA(National Security Agency)에서 주관하는 포럼으로 보호대상의 중요도에 따른 위협의 단계별로 요구사항을 제시하고 보안 구조들의 등급을 정의해 놓은 문서이다.

3. 공정 단계별 검증 방법

소프트웨어 공학의 공정단계는 요구분석, 디자인, 구현, 시험의 단계로 구분을 한다. 보안 운영체계의 개발에 있어서 각 공정 단계에서 발생되는 중간 결과물에 대한 보안성 검증 방법이 필요하다. 이 장에서는 이러한 단계별 중간 결과물의 보안성 검증 방법에 대하여 알아본다.

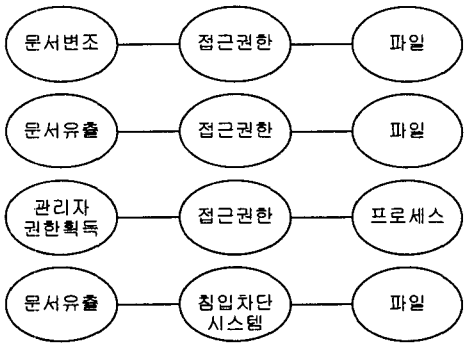
3.1 요구분석 단계

요구분석 단계에서는 위험, 취약점에 대한 정의를 하고 이러한 위험, 취약점에 대하여 보호하고자 하는 대상을 정의한다. 그리고 마지막으로 이런 위험과 취약점으로부터 대상을 보호하기 위한 보안구조를 정의하는 단계이다. 즉 위험, 보호대상, 보안구조를 구성요소로 하고 있다.

이런 세 가지 구성요소를 가진 요구분석 단계를 검증하기 위하여 요구분석 단계에서 작성되는 PP나 ST같은 문서의 내용을 아래 (그림 1)과 같이 위험 및 취약점, 보호대상, 보안구조로 연결하여 모든 위험으로부터 보호대상이 보안구조에 의하여 보호될 수 있는지 연결을 함으로써 요구분석 단계가 잘 수행되어 안전하게 계획되었는지를 평가한다. 여기서 보안구조에는 보안 알고리즘, 메커니즘, 프로토콜 그리고 정책 등을 포함하고있다.

(그림 1) 요구분석단계의 평가를 위한 모델

이러한 원리를 이용하여 (그림 2)에서는 우리가 목표로 하는 운영체계에 대하여 위 구조를 정의한 예를 몇 가지 제시하였다.



(그림 2) 운영체계에서의 예

(그림 2)에서 (문서유출)-(접근권한)-(파일)의 경우 파일을 접근하는 것을 통제하는 것이고 (문서유출)-(침입차단시스템)-(파일)의 경우는 네트워크를 통한 외부 유출을 막는 것이다. 이러한 경우는 파일을 보호하는 관점이 틀리기 때문에 발생하는 현상이다. 이러한 관계를 잘 연결시켜 주어야 한다.

3.2 디자인 단계

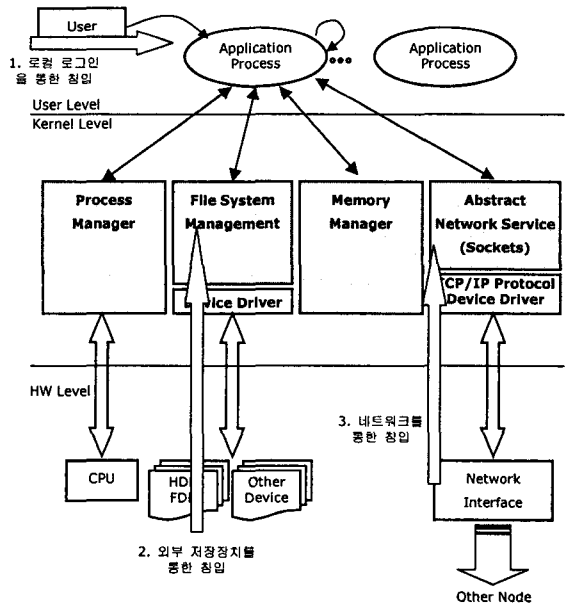
디자인 단계에서는 요구분석 단계에서 고려되었던 보호 대상, 위험, 보안기능 요구사항, 보안 알고리즘 및 메커니즘을 이용하여 보안운영체계를 설계한다. 우선, 운영체계의 모듈과 보안 모듈을 정의하고 각 모듈의 데이터 구조나 기능을 상세히 디자인한다. 마지막으로 각 모듈의 관계를 정의하여 보안운영체계의 설계를 마친다.

디자인 단계의 결과물에 대한 검증은 다음과 같은 항목으로 수행될 수 있다.

- 보안기능 요구사항에 대한 디자인 표현의 검증
- 디자인 표현간의 일치성 검증

보안기능 요구사항에 대한 디자인 표현의 검증은 요구분석 단계에서 고려되었던 보안기능 요구사항을 각 보안 모듈과 모듈간의 관계에 정확히 디자인되었는지를 확인하는 것이고, 디자인 표현간의 일치성 검증은 보안 모듈과 운영체계 모듈간의 일치성에 대한 검증을 통하여서 구현 단계에서 발생할 수 있는 모듈간의 충돌을 최소화하여야 한다.

위에서 설명된 검증 항목에 대한 검증을 실시하기 위해서는 실제로 수행할 수 있는 검증 방법이 필요하다. 우선, 디자인 표현간의 일치성 검증을 위해서 필요한 검증 방법은 기능적으로 중복된 보안 모듈, 보안 모듈간의 관계 및 보안 모듈과 운영체계 모듈간의 관계를 정의한 서로 공유하고 있는 데이터 구조 및 상호 전달되는 데이터 구조의 일치성을 분석하여야 한다. 다음으로는 보안기능 요구사항에 대한 디자인 표현의 검증을 위해서 필요한 검증 방법은 요구분석 단계에서 생성한 위험 목록과 위험을 해결하기 위한 보안기능을 디자인에 고려하였는지 검증을 하여야 한다. 이러한 검증을 위해서 모듈 다이어그램을 이용한 방법을 사용할 수 있다. 이 방법에서는 기본적으로 디자인 단계에서 고려된 보안 모듈과 운영체계 모듈, 각 모듈간의 관계를 모듈 다이어그램을 이용하여 도식화하고, 위험에 따른 공격 및 침입 경로를 나타낸다. 이때 공격 및 침입 경로의 중간에 놓이게 되는 각 보안 모듈들이 이러한 공격 및 침입을 방지할 수 있는 기능들이 디자인에 고려되어 있는지를 검증할 수 있다.



(그림 3) 모듈 다이어그램과 공격 및 침입 경로

(그림 3)은 운영체계의 중요 모듈의 관계를 나타내는 모듈 다이어그램이다. 또한, 운영체계에 대한 공격 및 침입 경로를 크게 3가지로 분류하였다. (그림 3)에서 침입 경로 1은 시스템 해킹을 위해 로컬 로그인을 통하여 시스템에 침입을 나타내고 있고, 침입 경로 2는 외부 저장장치를 통하여 악의적인 코드 및 바이러스의 침입을 나타낸다. 마지막으로 침입 경로 3은 네트워크를 통한 침입을 통해 시스템 내부의 공격을 가할 수 있게 하는 침입을 나타내고 있다. 이러한 침입에 대하여 중간에 거치게 되는 보안 모듈이 각 공격 및 침입을 막을 수 있는 기능이 디자인되어 있는지 검증을 할 수 있다. 실제적으로 보안 운영 체계를 개발할 때에는 공격 및 침입에 대하여 더 자세히 분류할 수 있고, 보안 운영 체계의 모듈 또한 더 세분화될 수 있다.

3.3 구현 및 시험 단계

지금까지 본 요구분석 단계와 디자인 단계는 제품이 나오기 전에 제품의 보안성을 검증하는 부분이고 여기 구현 및 시험 단계는 실제 우리가 요구하는 제품이 만들어지고 이 제품을 대상으로 검증을 한다. 이 단계에서는 구현과 시험 단계가 동일한 선상에서 검증을 할 수 있다고 생각하고 한 단계로 묶어서 평가를 하였다.

검증방법은 앞에서 분석된 공격 및 침입 시나리오를 이용하여 실험을 하게 되고, 또 앞 단계에서 정의된 보안기능 요구사항에 따른 각 모듈의 동작 시나리오와 모듈간의 상호 동작 시나리오를 바탕으로 분석과 설계 단계에서 정의된 내용이 구현에 얼마나 충실하게 반영되었는지를 평가할 수 있다.

운영체계와 관련된 검증을 위한 관점에는 크게 성능측면과 보안측면으로 나눌 수 있고 검증을 하기 위한 방법으로는 아래와 같이 세 가지로 나눌 수 있다.

- Application 검사
- OS(제품) 내구도 검사
- 배포와 업데이트 보안성 검사

이중 마지막 배포와 업데이트 보안성 검사는 이 단계에서 고려되는 것은 아니지만 따로 공정단계를 추가하지 않고 이 단계에 함께 검사를 수행한다.

먼저 Application검사는 운영체계의 가장 중요한 Application이 제대로 동작하는지를 검사하는 방법이다. 우선 정상적으로 안전하다고 판단되는 프로그램을 수행시키고 공격을 행하고, 다음으로 버그가 있는 프로그램과 웜, 백도어가 설치되고 바이러스에 감염된 비정상적인 프로그램을 동작시켜서 이러한 경우에도 운영체계가 안전하게 동작하는지 검사한다. 마지막으로 여러 비정상적인 프로그램의 시스템 자원 획득에 대한 실험도 하게 된다.

OS 내구도 검사는 위의 요구분석단계와 디자인 단계에서 계획했던 사항이 제대로 구현되었는지를 실제 소스코드를 통하여 검사를 하는 방법이다.

4. 결론

표준으로 정의된 IT제품의 보안성 평가 방안들은 CC나 ITSEC, TCSEC과 같이 많이 나와있다. 그러나 이러한 표준은 너무 추상적이고 소프트웨어의 각 단계에 따른 보안성을 측정하는데 불충분하다. 또 SSE-CMM같은 프로세스를 평가하는 방법의 경우 역시 너무 추상적이고 포괄적인 면만을 설명하였고 프로세스 자체에 대한 평가에 중점을 두었을 뿐 각 단계의 보안강도를 평가하는데는 부족하다. 이 논문에서는 이러한 단점을 보안하고자 각 공정단계에서, 특히 제품을 계획하는 단계인 요구분석단계와 디자인 단계에서 그래프와 모듈 다이어그램을 통한 검증방법을 제시하였다. 여기서는 운영체계라고 특정 제품을 명시하였지만 실제로 다양한 IT제품에 적용될 수 있다.

<참고문헌>

[1] ISO/ICE, "Common Criteria v2.1," August, 1999
[2] Carnegie Mellon Univ. "SSE-CMM,"
[3] IATFF, "Information Assurance Technical Framework documents v3.1," september 2002