情 報 保 護 學 會 誌
第 13 券 第 6 號, 2003. 12

정보보호시스템 시험과정 보증요구사항 작성 기법에 관한 고찰

백 남 균*, 최 용 준*, 이 강 수**

요 약

본고에서는 개발자에게 단위, 통합 및 서비스 시험에 대한 시험과정 보증문서 작성 기법에 대한 참고자료를 제공하기 위하여, 국내에서 개발된 침입차단시스템과 침입탐지시스템 평가기준에 의한 시험과정 보증요구사항을 분석하여 소프트웨어 공학에 기반 한 전통적인 구조적 방법론을 토대로 한 시험과정 보증문서 작성 경험을 기술하고자 한다. 따라서, 개발자는 사용자 요구사항에 의한 객관적이고 체계적인 시험과정을 이해하여 시스템 개발에서 발생할 수 있는 오류를 줄일 수 있으며 또한, 정확한 보안 기능명세 및 시험보증 설계·개발·구현을 통하여 제품의 안전성 및 신뢰성 향상에 기여할 것으로 기대된다.

Ⅰ. 서 론

인터넷 환경의 급속한 확장 및 보급으로 각종 통신망은 시간과 장소에 제약을 받지 않고 다양한 정보를 공유할 수 있도록 네트워크를 형성하여 전세계가 하나로 상호 연결되어 있다. 이러한 정보화의 가속화는 정보시스템을 통한 정보제공 서비스 등으로 필요한 정보를 공유할 수 있는 혜택을 주는 반면 각종 통신망을 통하여 개인 및 사회의 중요한 정보에 대한 불법적인 침입 및 공격 등의 역기능에 처해 있다.

체계적이고 총체적인 정보보호 대책은 이러한 정보화 역기능으로부터 정보시스템과 통신망을 보호하고 안전한 운영을 가능하게 한다. 즉, 인터넷 사용이 보편화되고 전자상거래의 규모가 커져 갈수록 정보보호 시스템에 대한 신뢰성 및 안전성에 대한 입증 노력 또한 더 강하게 요구되기에 제도적 노력의 일환으로 정보통신 환경에 적합하게 개발되어진 평가기준에 근거한 정보보호시스템 평가제도가 필요하다.

한국정보보호진흥원에서는 1995년 10월에 제정된 정보화촉진기본법과 동법 시행령에 의거, 국내 평가·인증 제도가 구축되어 시행되어 왔다. 현재, 정보보호진흥원에서는 2000년 2월에 개정한 정보통신망 침입차단시스템 평가기준과 2000년 7월에 고시된 침입탐지시스템 평가기준에 근거하여 두 개 제품군에 대한 평가를 실시하고 있다. 또한, 2002년 8월에 고시된 정보보호시스템 공통평가기준에 의하여 침입차단시스템, 침입탐지시스템 및 가상사설망 평가를 수행하고 있으며 2004년도에는 지문인식제품, 운영체제보안시스템 및 스마트카드에 대한 공통평가기준을 적용한 평가가 추가적으로 확대 수행될 예정이다.

국내에서 개발되어진 침입차단시스템 및 침입탐지시스템 평가기준은 등급별 보안기능요구사항과 보증요구사항으로 이루어져 있으며 보증요구사항 중 시험과정은 K1 에서 K7 까지 모든 등급에서 요구하는 필수 보증요구사항으로 반드시 문서화된 자료를 제출토록 요구되어 진다. 하지만, 정보보호시스템 개발업체의 영세성 및 시장 선점을 위한 과도한 출시 경쟁 등 여러 요인으로 인하여 제품 개발수행방법(절차 및 산출물)과 과정 중에 필요한 각종 기법 및 도구를 체계 있게 정리한 표준화 지표가 부재한 실정이며 몇몇 평가제품은 역공학 방식에 의해 개발이 완료된 정보보호시스템의 구성요소 및 관계를 식별·분석하는 과정을 수행하므로 평가과정 중에 요구하는 보안기능에 대한 보증요구사항을 충족시키지 못하는 어려움이 있을 수 있

* 한국정보보호진흥원 산업지원단 평가2팀(namkyun, yjchoi@kisa.or.kr)
** 한남대학교 컴퓨터공학과(gslee@eve.hannam.ac.kr)

다. 또한, 국내에서는 평가 시험과정 보증문서에 대한 전문적인 연구가 미비하여 평가 신청업체의 보증요구사항문서 작업의 참고 자료가 부족한 실정이다.

본고에서는 이러한 문제에 대한 이해를 도와 개발자에게 단위, 통합 및 서비스 시험에 대한 시험과정 보증문서 작성 기법에 대한 참고자료를 제공하기 위하여, 국내에서 개발된 침입차단시스템과 침입탐지시스템 평가기준에 의한 시험과정 보증요구사항을 분석하여 소프트웨어 공학에 기반 한 전통적인 구조적 방법론을 토대로 한 시험과정 보증문서 작성 경험을 기술하고자 한다. 서술된 시험 과정 보증문서 작성 기법을 참고로 하여, 개발자는 사용자 요구사항에 의한 객관적이고 체계적인 시험과정을 이해하여 시스템 개발에서 발생할 수 있는 오류를 줄일 수 있으며 또한, 정확한 보안 기능명세 및 시험보증 설계·개발·구현을 통하여 제품의 안전성 및 신뢰성 향상에 기여할 것으로 기대된다.

본고의 구성은 다음과 같다. 제 2장에서는 정보보호시스템 국내 평가 기준 및 시험과정에서 요구되는 보증 요구사항을 먼저 기술한다. 제 3장에서는 구조적 소프트웨어 개발 방법론에 의한 단위, 통합 및 서비스 시험의 정의와 접근방식을 서술하며 제 4장에서 구조적 방법론에 의해 시험될 단위, 통합 및 서비스 시험에 대한 시험과정 보증문서 작성 예제를 소개한다. 그리고 제 5장에서 마지막으로 결론을 정리한다.

Ⅱ. 정보보호시스템 국내 평가기준의 시험과정 보증 요구사항

1. 정보보호시스템 국내 평가 기준

보안 기능에 대한 신뢰성 및 안전성이 증명되지 않은 정보보호시스템의 사용은 보안 취약성이 내재되어 있을 가능성이 있을 수 있다. 정보보호시스템 평가제도는 안전성과 신뢰성을 지닌 우수한 보안제품 개발을 유도함으로써 정보보호산업 육성에 기여하고, 안전한 정보보호시스템 사용으로 '건강한 정보사회' 구축에 일조하기 위하여 정보화 촉진 기본법 제15조, 동법 시행령 제15조 및 제16조를 근거로 정보통신망 침입차단시스템과 침입탐지시스템의 평가기준과 지침서를 고시하였으며 침입차단시스템에 대한 평가는 1998년부터, 침입탐지시스템에 대한 평가는 2000년부터 한국정보보호진흥원에서 시행하고 있다. 〔표 1〕은 정보보호제품 국내 평가기준에 의한 평가·인증 체계를 요약한 것이다.

〔표 1〕 평가·인증 기준 및 지침

평가기준	· 정보통신망 침입차단시스템 평가 기준 (2000. 2, 정보통신부 고시 2000-14호) · 정보통신망 침입탐지시스템 평가 기준 (2000. 7, 정보통신부 고시 2000-62호)
평가·인증지침	· 정보보호시스템 평가·인증 지침 (2002. 8, 정보통신부 고시 2000-41호)
평가기관	· 한국정보보호진흥원
인증기관	· 국가정보원

2. 시험과정 보증요구 사항

정보보호시스템의 개발·평가·사용을 염두에 두고 개발된 국내 표준 및 관련 가이드라인은 실제로 제품을 구현하고 평가한 기술과 경험을 바탕으로 작성되었다. 국내 평가기준의 내용을 살펴보면 〔표 2〕에서와 같이 등급별로 필요로 하는 보안기능요구사항은 침입차단시스템 6개/침입탐지시스템 8개와 제품을 사용하는 환경 및 제품의 신뢰성 및 안전성을 보증하기 위한 보증요구사항은 침입차단시스템 6개/침입탐지시스템 6개로 이루어져 있다.

〔표 2〕 평가기준별 요구사항

평가기준	보안기능요구사항	보증요구사항
침입차단시스템	· 신분확인 · 접근통제 · 무결성 · 비밀성 · 감시기록 및 추적 · 보안관리	· 개발과정 · 시험과정 · 형상관리 · 운영환경 · 설명서 · 취약성
침입탐지시스템	· 축약감사데이터 생성 · 보안위반 분석 · 보안감사 대응 · 신분확인 · 데이터 보호 · 보안감사 · 보안관리 · 보안기능의 보호	· 개발과정 · 시험과정 · 형상관리 · 운영환경 · 설명서 · 취약성

국내 기준에 명시하고 있는 시험과정 보증요구사항은 K1에서 K7까지 모든 등급에서 요구하는 필수 보증요구사항으로 반드시 문서화된 자료를 제출토록 요구되어 지며 등급에 상관없이 제품의 기능과 운영 보증에 관한 최소한의 기본적인 원칙을 전체적인 보안기술 관점에서 다음과 같이 규정하고 있다.

- 시험과정의 평가에는 시험서, 개발자가 시험에 사용한 시험프로그램과 시험도구 일체가 있어야한다.

- 시험서에는 각 보안기능이 보안목표명세서에 명시된바와 같이 동작하며, 각각의 보안기능이 통합되어 보안목적에 부합하도록 동작하는지 확인할 수 있도록 시험계획 및 시험 결과를 서술하여야 한다.
- 시험계획에는 개발과정의 각 단계별로 수행하여야 할 시험항목과 각 시험항목별로 시험목적 및 시험절차를 서술하여야 한다.
- 시험결과에는 각 시험의 예상결과 및 실제결과를 명시하여야 한다.

이와 같이 시험과정 보증요구사항에서 살피고자 하는 바는, 정보보호시스템의 보안기능이 보안목표명세서에 서술된 보안 목적에 부합하도록 동작하는지를 시험 계획 및 시험 결과를 통하여 확인하고자 함이다.

Ⅲ. 구조적 방법론에 기반 한 시험과정

1. 소프트웨어 개발 방법론

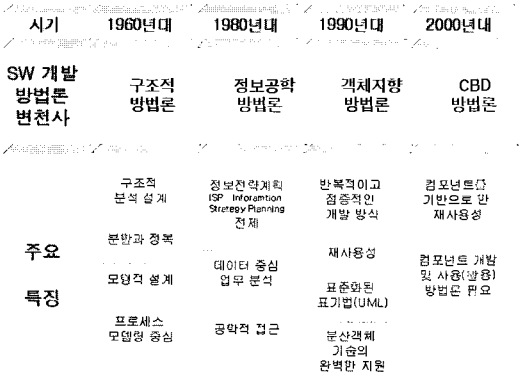
1968년 소프트웨어 위기에 따른 공학의 발전과 소프트웨어 이용범위 확대, 프로젝트 개념 등장, 프로젝트 대형화 및 수행 경험의 재활용 필요로 인하여 소프트웨어 개발 방법론이 출현한 이후 개발 생산성의 향상, 품질의 제고를 통한 고객만족의 실현, 개발조직의 의사소통 활성화, 시스템 개발 경험지식 전수 및 개발조직의 문화 형성의 요구를 충족시키면서 소프트웨어 개발 방법론은 발전하였다.

소프트웨어 개발 방법론은 시스템을 구축하는데 필요한 여러 가지 일들의 수행방법(절차 및 산출물)과 이러한 일들을 효율적으로 수행하는 과정에서 소프트웨어 공학원리에 입각한 각종 기법 및 도구를 기업의 문화를 바탕으로 체계 있게 정리하여 표준화한 것이라고 정의 할 수 있다. [그림 1]은 소프트웨어 개발 방법론의 간략한 변천사를 나타낸다.

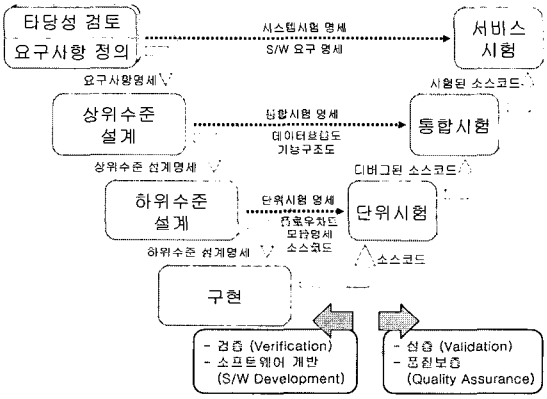
본고에서는 위의 그림에 나타낸 여러 소프트웨어 개발 방법론 중 지금까지 가장 많이 사용되어 지는 전통적인 SDLC(Software Development Life Cycle) 방법론에 근거한 순차적 모델인 구조적 방법론을 토대로 하여 시험과정 보증문서 작성을 설명하고자 한다.

2. 구조적 방법론에 대한 시험 전략

전통적인 SDLC(System Development Life Cycle)에 근거한 구조적 방법론은 [그림 2]와 같이



[그림 1] 개발방법론 변천사



[그림 2] 단계별 시험 전략

다단계로 구분되며 각 계층마다 설계 검증 및 실증을 위한 시험을 수행하게 된다.

구조적 분석, 설계 및 개발 기법을 활용하는 구조적 방법론은 프로그래머가 코딩을 할 수 있을 정도의 수준에 이르기까지 시스템의 물리적인 요소(데이터와 프로세스)에 대하여 구체화의 수준에 따라 다양한 도식적인 표현이 가능하며 세부사양에 대하여 모듈화가 쉽게 이루어 질 수 있다.

- 타당성 검토 : 사용자 요구사항에 대한 명확한 정의와 요구사항을 만족시키기 위한 분석 작업으로 시스템 구현에 따른 전략적 이익을 결정한다.
- 요구사항 정의 : 개발하고자 하는 소프트웨어에 초점을 맞추어 사용자의 요구사항을 수집·정리하여 정의한다. 소프트웨어 개발자는 구축하고자 하는 서비스의 특성을 이해하기 위해 요구되는 기능, 환경 및 인터페이스 등을 이해하여야 하며 정의된 요구사항들은 서비스 시험으로 보증되어 진다.
- 상위수준 설계 : 요구사항을 각 서비스로 분배하

는 과정으로 시스템의 구조를 결정하게 되며 나누어진 서비스 기능은 통합 시험으로 보증되어 진다.
- 하위수준 설계 : 상위수준에서 나누어진 서비스 기능을 세부적인 기능 사항에 맞추어 상세화 하는 과정으로 세부 기능은 단위 시험으로 보증되어 진다.
- 구현 : 설계의 각 부분을 실제로 프로그램밍 언어를 이용하여 구현하는 코드화 단계이다.

3. 구조적 방법론에 의한 단계별 시험

일반적으로 제품 개발과정에서의 시험은 제품의 품질을 보증하는 중요한 요소로서, 개발자가 수행한 타당성 검토, 요구사항 정의, 상위수준 설계, 하위수준 설계 및 구현에 대한 최종 검토를 수행하는 단계이다. 또한, 시험은 제품 개발과정에서 소요되는 전체 개발자 노력의 30~40%를 차지하며, 성공적인 시험은 좋은 품질의 제품을 보증하고, 사용자 요구사항에 적합한 제품임을 보증한다.

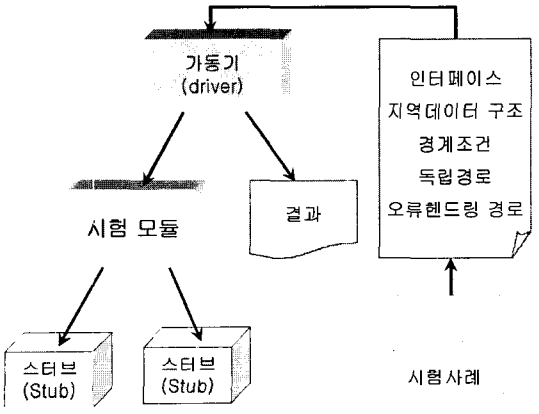
개발과정과 마찬가지로 제품 평가에서도 시험은 제품을 보증하기 위한 중요한 요소로서, 개발자가 수행한 각 단계별 시험을 확인하는 과정으로 구조적 방법론이 적용된 시험 보증 문서에 대해서 세 가지 단계의 시험 즉, 단위 시험, 통합 시험, 서비스 시험을 수행한다. 개발자는 개발된 제품에 대해 각 단계별 시험에 대한 문서를 작성하여 평가자에게 제출하고, 평가자는 제출된 문서를 기반으로 각 시험의 적절성과 완전성을 평가한다.

3.1. 단위 시험(Unit Test)

단위 시험은 모듈시험을 의미하고, 제품 설계의 기본 단위인 모듈의 내부적인 오류를 발견하기 위해 수행하는 시험을 의미한다. 모듈은 응집도(기능의 크기) 및 결합도(모듈간의 연관성)에 따라 개발자가 적절하게 나눌 수 있다.단위 시험은 블랙박스 시험 방식과 화이트박스 시험 방식을 사용할 수 있지만, 시험을 수행하는 개발자 및 평가자가 개발 소프트웨어의 내부 처리 로직을 상세하게 검토하여 모듈의 정확한 동작을 검증할 수 있는 화이트박스 시험방식을 일반적으로 채택하여 수행한다.

모듈은 독립적으로 수행되는 프로그램이 아니기 때문에 모듈 시험을 수행하기 위해서는 드라이버(Driver)와 스텁(Stub)과 같은 시험도구를 필요로 한다. 시험을 수행하기 위해 필요한 드라이버는 단위 시험을 위한 제어프로그램으로 시험사례 데이터를 받아서 모듈에 전달해 주고, 모듈이 수행한 결과를 출력하는 역할을 담당한다. 스텁은 시험할 모듈에 종속된 모듈을 대체하는 역할을 담당하고, 최소한의 데이터 처리를 수행하고, 호출을 확인하는 출력을 수행한 후에 시험 모듈에 제어를 반환하는 간단한 역할을 수행한다.

단위 시험은 [그림 3]과 같은 환경에서 드라이버와 스텁을 통해 모듈인터페이스, 지역 데이터 구조, 경계조건, 독립 경로, 오류 처리 경로에 대한 시험을 수행함으로써 모듈 내부에 논리적 오류가 존재하는 지를 시험한다.

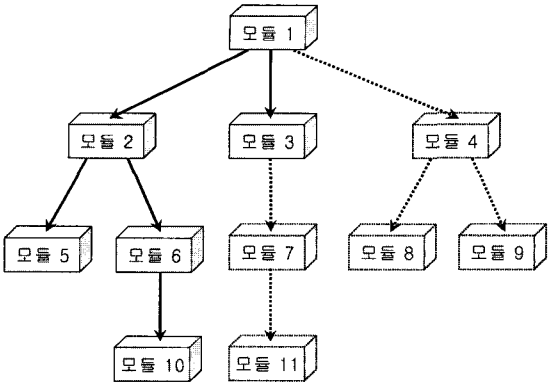


[그림 3] 단위 시험 환경

- 모듈 인터페이스에 대한 시험은 다른 시험을 수행하기 전에 수행하는 시험으로, 모듈의 입출력 값이 적절하게 설계되어 있는지를 확인하는 시험이다. 예를 들어 입력되는 매개변수의 수와 모듈 인자값 수가 일치하는지를 시험한다.
- 지역 데이터 구조에 대한 시험은 모듈 내부에서 일시적으로 저장되는 데이터가 모듈 수행동안에 무결성을 유지하는가를 확인하는 시험이다. 이는 모듈을 구현한 알고리즘 구현상의 오류를 찾아내는 시험으로 오류 발생시에 코딩상의 문제점을 찾아내어 모듈 내부의 내재적인 오류를 제거하여야 한다. 예를 들어 오류가 발생하기 쉬운 초기화나 기본값 또는 잘못 쓰여진 변수 이름 등에 대해 시험한다.
- 경계 조건에 대한 시험은 모듈이 수행하기 위해 설정된 데이터의 경계값에서 모듈이 적절하게 동작하는지를 확인하는 시험이다. 이는 모듈이 설정된 데이터의 경계값에서 오류를 발생시킬 확률이 높기 때문에 설정된 데이터의 최대값 및 최소

값을 입력함으로써 시험할 수 있다.

- 독립 경로에 대한 시험은 모듈에 포함되는 모든 문장이 제어 흐름에 따라 적어도 한번은 실행되는지를 확인하는 시험이다. 이는 모듈에 대한 입력에서부터 출력까지 내부의 제어 흐름에 한번도 실행되지 않는 문장이 있는지를 시험한다.
- 오류 처리 경로에 대한 시험은 모듈에 오류가 발생 시에 적절한 처리 경로를 제공하는지 확인하는 시험이다. 이는 모듈에서 언급된 오류와 발생한 오류가 일치하지 않는지와 언급하지 않는 오류가 발생하는지 등을 시험한다.



〔그림 4〕 하향식 통합 시험

3.2. 통합 시험(Integration Test)

통합 시험은 단위 시험을 수행한 결과에서 모듈의 내부적 오류가 없음을 확인한 후에 수행하는 시험으로, 모듈간의 인터페이싱과 관련된 오류를 발견하기 위해 수행하는 시험을 의미한다. 또한 통합 시험을 통해 체계적으로 제품의 구조를 구성할 수 있다. 시험절차는 하나의 모듈에서 시작하여 해당 모듈과 관련된 다른 모듈간의 인터페이스에 오류가 없음을 확인해 가며, 점증적으로 모듈들을 통합하면서 시험을 수행한다. 통합 시험이 종료된 후에는 통합된 모듈들은 사용가능한 부분적인 제품의 형태로 구성된다. 점증적으로 모듈을 통합하는 방식으로는 상향식 통합 방식과 하향식 통합 방식이 있다.

3.2.1 하향식 통합(Top-Down Integration)

〔그림 4〕에서와 같이 하향식 통합은 제품의 주 모듈로부터 하위 세부 모듈들로 점차적으로 통합해 나가는 방식으로 하향식(Top-down) 프로그래밍에 적합한 방식이다.

하향식 통합 방식을 사용한 통합 시험은 제품의 주 모듈인 모듈 1에서 시작하여 아래 방향으로 모듈 1과 관련된 하부 세부 모듈들 간의 인터페이스를 시험하여 통합해 가는 방식으로 깊이-우선(Depth-First) 방식과 너비-우선(Breadth-First) 방식이 있다.

깊이-우선 방식은 하위 세부 모듈 중에서 통합 시에 많은 인터페이스 문제를 유발할 수 있는 모듈이나, 기술적으로 많은 위험요소들을 지닌 모듈을 가지고 있는 경우에 사용하는 접근 방식으로, 주 모듈에서 우선적으로 이러한 모듈들을 통합시켜 추가적으로 통합되는 모듈간의 인터페이스 문제를 해결하는 체계적이고 수정이 쉬운 접근 방식이다.

너비-우선 방식은 위험 요소를 지닌 모듈이 거의 없는 경우에 사용하는 접근 방식이다.

〔그림 4〕에서 깊이-우선 방식을 통해 통합 시험을 수행한다면 모듈 1로부터 모듈 2, 모듈 5, 모듈 6, 모듈 10, 모듈 3순으로 시험할 수 있고, 너비-우선 방식을 통해 통합시험을 수행한다면, 모듈 1로부터 모듈 2, 모듈 3, 모듈 4 순으로 시험을 수행한다.

하향식 통합 방식에 의한 통합 시험은 하위 세부 모듈 위치에 스텁(Stub)을 도입하여 시험한 후에 스텁의 위치에 실제 모듈을 대체함으로써 시험을 수행한다.

하향식 통합 시험은 다음과 같은 단계를 통해 수행된다.

첫째, 주 모듈에 직접 종속되는 모든 모듈을 스텁으로 대체

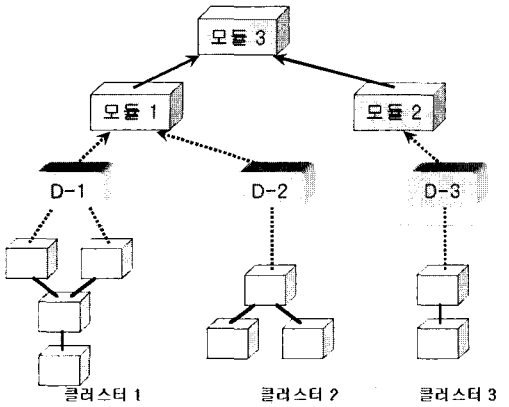
둘째, 선택한 통합 방식(깊이-우선, 너비-우선)에 따라 종속된 스텁들을 한번에 하나씩 실제 모듈로 대체

셋째, 각 모듈이 통합된 후에 시험을 수행하여 통합 시에 발생하는 인터페이스 상의 오류를 찾아 오류를 제거

넷째, 각 시험이 끝난 후에 통합된 모듈의 하위 세부 모듈을 스텁으로 대체

다섯째, 회귀 시험을 통해 새로운 오류가 발생하지 않는지를 확인

통합 시험은 주 모듈에서 제품의 구조가 구축될 때까지(모든 모듈이 통합될 때까지) 두 번째 단계부터 다섯 번째 단계까지를 연속적으로 반복한다. 다섯 번째 단계에서 수행하는 회귀 시험은 새로이 추가된 모듈에 의해서 기존에 통합된 부분에 새로운 문제가 발생하지 않는 가를 확인하는 시험으로 이미 수행했던 시험의 부분집합을 다시 수행한다.



〔그림 5〕 상향식 통합 시험

3.2.2. 상향식 통합

상향식 통합은 제품 구조에서 최하위 레벨인 모듈, 즉 원자 모듈에서 상위 모듈을 통합해 나가는 방식이다. 제품 구현 시 위험 요소를 많이 가진 부분으로부터 구현하고 통합하는 방식을 지원할 수 있으므로, 위험 요소를 많이 가진 제품 개발 시에 유용한 통합 시험 방식이다.

〔그림 5〕에서와 같이 상향식 통합 방식을 사용한 통합 시험은 원자 모듈들이 클러스터 1, 클러스터 2, 클러스터 3을 형성하기 위해서 결합되며 각 클러스터는 드라이버 D-1, D-2, D-3를 사용해서 시험되고, 시험 후에 드라이버 D-1, D-2는 제거되고 클러스터 1과 클러스터 2는 모듈 1에 직접 통합된다. 또한 드라이버 D-3도 제거되고 클러스터 3은 모듈 2에 직접 통합되는 순으로 시험하는 방식이다. 여기서, 클러스터는 동일한 업무를 수행하는 실제 모듈들의 집합을 의미한다.

상향식 통합 시험은 다음과 같은 단계를 통해 수행된다.

첫째, 원자 모듈들은 특정한 제품 기능을 수행하는 클러스터(Cluster)에 결합
둘째, 드라이버는 시험 사례의 입력과 출력을 조정하기 위해 개발
셋째, 개발된 드라이버를 통해 클러스터를 시험
넷째, 드라이버를 실제 모듈로 대체
다섯째, 클러스터가 상위 모듈로 이동하면서 결합

3.3 서비스 시험(Service Test)

서비스 시험은 검증 시험을 의미하며 개발된 제품을 가지고 블랙박스 시험을 통해 개발 초기에 분석된 사용자 요구사항을 충족하는지를 검증하는 시험이다. 순서상 통합 시험이 완료된 후에 수행하며 개발된 제품을 시험 도구로 사용하기 때문에 제품 시험이라고도 한다.

서비스 시험 시에 발생하는 오류는 모든 서비스 시험을 수행한 후에 수정하여야 하고 또한 최종 사용자만이 찾을 수 있는 오류를 발견하기 위해 개발된 제품에 대한 설명문서를 참고하여 모든 기능에 대한 시험을 수행한다.

Ⅳ. 구조적 방법론에 기반 한 시험과정 보증요구사항 서술

정보보호시스템의 국내 평가기준 및 가이드라인은 실제로 제품을 구현하고 평가한 기술과 경험을 바탕으로 작성되었다, 하지만 개략적인 기준 서술, 실증된 참고자료의 부재 및 전문적인 연구 미비로 인하여 평가신청을 원하는 정보보호시스템 개발자는 평가 제출문서 작성 시 보증요구사항을 충족시키지 못하는 어려움이 있을 수 있다. 따라서, 본 장에서는 시험과정 보증문서 작성 경험을 기술하여 개발자에게 시험과정 보증요구사항의 이해를 돕고자한다.

시험서는 평가기준에 기술된 바와 같이 각 보안기능이 보안목표명세서에 명시된 요구행위를 정확하게 수행하며, 보안목적에 부합하도록 동작하는지를 확인할 수 있도록 시험계획 및 시험결과를 기술하여야 한다. 시험계획에는 시험항목, 각 시험항목별 시험 목적 및 시험절차를 서술하고, 시험결과에는 각 시험의 예상결과 및 실제결과를 명시하여야 한다.

시험서는 보안기능에 대하여 단위시험, 통합시험 및 서비스시험을 모두 기술하며 일반적으로 단위시험은 제출물의 범위에 의해서 시험범위가 결정되며 원시코드를 이용한 시험일 경우는 원시코드에 대한 논리적 분석을 하여야 한다. 통합시험은 여러 개의 모듈을 통합하여 시험하는 것을 의미하며 통합된 시험프로그램을 사용하여 모듈간의 정상적인 상호동작에 관한 시험내용을 기술하여야 한다. 서비스 시험은 통합된 모듈인 제품에 대한 사용자 요구사항 측면에서의 정상 기능 동작을 검증할 수 있도록 기술하며 네트워크 환경 구성에 따라서 시험방법 및 결과에 차이가 있을 경우에는 각 네트워크 환경 구성별로 구분하여야 한다.

시험서는 1)개요, 2)시험목록, 3)단위시험, 4)통합시험 및 5)서비스시험 등으로 구성될 수 있으며 다음에 작성한 형태를 제시한다. 지면상 단위·통합· 서비스

시험에 대한 제출물 작성 및 예제에서는 유사한 서술내용에서는 단위 시험을 기준으로 서술하고 소프트웨어 공학에 기반 한 전통적인 구조적 방법론을 토대로 한 시험간 차이·특이점은 주석을 두어 설명하고자 한다.

정보보호시스템 V1.0 시험서

1. 개요

개요에서는 목적, 구성, 용어 설명 등을 기술한다.

1.1. 목적

목적에서는 시험서 작성의 목적을 명시하여야 한다.

작성예)

시험서의 목적은 시스템을 구성하는 모든 보안기능이 보안목표명세서에 명시된 대로 동작하는지 확인하고, 보안기능들이 전체적으로 효용성 있는 하나의 보안 시스템으로서 동작하는지의 여부를 확인하는 것이다.

...

{ 개발자 작성 }

1.2. 구성

구성에서는 시험서의 구성에 대하여 기술한다.

작성예)

본 문서는 아래와 같이 구성되어 있다.
제1장에서는 개요, 구성, 용어 설명 등을 기술
제2장에서는 단위 시험, 통합시험 및 서비스시험 전체목록 등을 기술
제3장에서는 단위 시험환경 및 시험별 시험절차 등을 기술
제4장에서는 통합 시험환경 및 시험별 시험절차 등을 기술
제5장에서는 서비스 시험환경 및 시험별 시험절차 등을 기술

...

{ 개발자 작성 }

1.3. 용어 설명

용어 설명에서는 시험서에 사용된 전문용어나 특정용어에 대하여 간략하게 기술한다.

작성예)

신분확인 : 정보보호시스템에 접근하는 관리자 및 사용자의 신분을 증명하기 위한 기능이다.

...

{ 개발자 작성 }

2. 시험목록

시험항목에서는 상세설계서에 기술된 보안모듈에 대하여 시험한 단위 시험목록, 통합된 단위 시험 보안기능에 대하여 시험한 통합 시험목록 및 보안목표명세서 및 관리자(사용자) 설명서에서 서술된 보안기능 및 동작메뉴에 대하여 시험한 서비스 시험목록을 기술한다.

2.1 단위 시험목록

단위 시험목록에서는 상세설계서에 명시된 각 보안모듈이 시험되었음을 보여주는 단위 시험목록에 대하여 기술한다.

작성예)

상세설계서에 명시된 보안모듈이 시험되었는지 확인하기 위한 단위 시험목록을 〈표 2-1〉과 같이 제공한다.

...

{ 개발자 작성 }

※ 단위 시험은 독자적인 시험 프로그램에 의해 수행되므로 보안모듈자체에 의한 독립 시험 프로그램이 반드시 제공되어야 한다.

2.2 통합시험목록

통합시험목록은 여러 개의 모듈을 통합한 보안기능이 시험되었음을 보여주는 통합시험목록에 대하여 기술한다.

작성예)

통합된 보안기능이 시험되었는지 확인하기 위한 통합시험목록을 〈표 2-2〉와 같이 제공한다.

...

{ 개발자 작성 }

※ 통합 시험은 독자적인 시험 프로그램에 의해 수행되므로 통합된 보안기능에 의한 독립 시험 프로그램이 반드시 제공되어야 한다.

2.3 서비스 시험목록

〈표 2-2〉 통합시험목록

시험항목	시험도구	보안기능	시험목적	비고
F_IDENTY	F_IDENTY_program	IA_1	사용자신분확인 동작여부 확인	
...				

서비스 시험목록은 통합된 모듈인 제품에 대한 사용자 요구사항 측면에서의 기능동작에 기술한다.

작성예)

보안기능 제품에 정확히 구현되었는지를 확인하기 위한 통합시험목록을 〈표 2-3〉과 같이 제공한다.

〈표 2-3〉 서비스 시험목록

시험항목	시험도구	보안기능	시험목적	비고
I&A	평가제품	F_IDENTY	사용자신분확인 동작여부 확인	
...				

...

{ 개발자 작성 }

※ 서비스 시험은 개발된 제품을 기반으로 수행되므로 독립 시험 프로그램이 필요하지 않다. 하지만, 시험환경에 설정에 필요하거나 지원하는 도구 및 서비스는 시험도구에 포함될 수 도 있다.

3. 단위 시험(통합 및 서비스 시험 포함)

단위 시험은 일반적으로 정보보호시스템에서 제공되는 보안기능별로 보안모듈시험 프로그램을 작성하여 보안모듈별 세부 모듈로 구분하여 모듈시험을 수행하게 된다.

3.1. 단위 시험을 위한 환경

단위 시험을 위한 물리적인 환경 및 네트워크 환경 등에 대하여 기술한다.

작성예)

시험을 위한 시스템은 개발에 사용되는 네트워크와 분리되어야 하며, 인터넷 등 외부망과 단절된 상태에서 진행되어야 한다. 테스트에 사용되는 시스템은 테스트를 위한 유틸리티를 제외한 다른 불필요한 프로그램이 설치되어 있지 않도록 한다.

...

{ 개발자 작성 }

※ 단위 및 통합시험은 독자적인 테스트 프로그램에 의해 수행되므로 실제환경이 아닌 시스템상의 시뮬레이션 또는 에뮬레이션 환경에서 시험되어 질 수도 있다. 하지만 서비스 시험의 경우, 평가 신청된 실제 제품에 대하여 기능 검증을 수행하게 되므로 가능한 실제운영환경과 유사한 환경을 설정하여 시험되어야 한다.

3.2. 개별 단위 시험

작성예)

I&A_Identification_input(식별정보 확인 시험)

3.2.1. 시험계획

시험 계획에는 단위 시험별로 시험대상 단위 모듈명, 시험도구명, 해당보안기능, 기능 분류, 시험목적, 시험환경 및 시험절차 등을 기술한다.

작성예)

단위 시험 대상 모듈명 : I&A_Identification_input
시험도구명 : I&A_Identification_input_test
보안기능 : 식별정보 확인
기능분류 : 신분확인의 식별 및 인증 기능

...

{ 개발자 작성 }

3.2.1.1 시험목적

시험목적에서는 각 단위 시험의 목표를 기술하는 것으로 개발과정에서 단계별로 수행되는 시험항목을 명확하게 기술한다.

작성예)

정보보호시스템의 보안모듈 중 I&A_Identification_input 단위 모듈에 대한 시험이다. 사용자가 로그인을 위해 관리자가 지정한 정확한 식별정보를 입력할 경우에만 로그인할 수 있으며 비인가자의 불법적인 접근을 차단하는 기능을 제대로 수행하는지 확인한다.

...

{ 개발자 작성 }

3.2.1.2. 시험환경

시험환경에서는 보안모듈을 시험하기 위한 구성 설정, 제한사항, 물리적 환경 및 사용된 시스템에 대한 사양 등을 기술한다.

작성예)

I&A_Identification_input 단위 모듈을 시험하기 위해서는 시험프로그램인 I&A_Identification_input_program이 필요하며 이 시험 프로그램은 자체적으로 제작되었다. 시험에 사용될 설정값을 미리 정의하여야 한다.

- 시험에 사용될 설정 값
① 사용자 계정-1 : test123
② 사용자 계정-2 : unittest

...

{ 개발자 작성 }

※ 시험도구를 사용한다면 도구의 도구명 및 버전번호, 도구 사용의 목적, 도구의 사용법에 대하여 기술한다.

작성예)

I&A_Identification_input_program V1.0은 사용자 식별정보를 입력받아 인가된 사용자인지를 식별하는 시험 프로그램으로 정보를 입력받았을 때 사용자의 입력 계정과 미리 입력되어진 계정을 비교하여 동일한 값인지 판단한다. I&A_Identification_input_program V1.0에 대한 정보는 다음과 같다.
· 시험프로그램 : I&A_Identification_input_program V1.0
· 사용목적 : 사용자 식별정보를 입력받아 입력받은 값이 정확한지 검증하여 식별결과를 확인
· 설치 및 사용방법 : I&A_Identification_input_program V1.0의 설치방법은

...

{ 개발자 작성 }

3.2.1.3. 시험절차

시험절차에서는 시험동안에 수행하는 단계를 상세히 기술하여야 하며 이러한 절차를 이용하여 시험을 재수행할 수 있어야 한다. 시험 과정 중에 입력에 사용하는 데이터는 시험과 연관되며 일반적으로 추정되는 데이터를 사용한다.

작성예)

① 테스트 프로그램 설치
② 시험프로그램을 실행시키면 다음과 같은 화면이 출력된다.
③ 메뉴 중 설정 버튼을 클릭하면 다음과 같은 화면이 출력된다. 시험할 사용자 계정 정보를 입력하여 저장한다.
④ 가능한 상세(GUI 출력 등)하게 절차 누락없이 작성

...

{ 개발자 작성 }

3.2.2. 시험결과

시험결과에서는 예상결과와 실제결과를 기술하여야 한다.

3.2.2.1. 예상 결과

예상결과에서는 시험 절차가 올바르게 수행되었음을 나타내는 예상결과를 기술한다.

작성예)

시험을 수행하는 시험자가 정확한 사용자 계정을 정확히 입력하였을 경우에만 식별확인이 가능하며, 잘못된 계정을 입력하였을 경우에는 각각의 경우에 대한 오류메시지를 나타낸다.

① 정확한 계정 및 패스워드 입력시 정상적인 식별확인이 이루어짐.

② 잘못된 계정 및 패스워드

- 계정을 정확하게 입력하지 않은 경우

오류 : 정확한 계정을 입력하여 주십시오.

...

{ 개발자 작성 }

3.2.2.2. 실제결과

실제결과에서는 수행된 시험에서 나온 실제결과를 기술한다. 실제결과는 테스트한 실제결과화면을 캡쳐하여 삽입하고 설명추가가 필요한 경우 설명을 추가 기술한다.

4. 참고자료

참고자료에서는 시험서를 작성하기 위하여 참고한 자료를 자료명, 발행일, 발행자를 포함하여 기술한다.

작성예)

〔1〕 보안목표명세서, 2002.12, 개발자 회사(주)

〔2〕 기능명세서, 2002.12, 개발자 회사(주)

〔3〕 기본설계서, 2002.12, 개발자 회사(주)

〔4〕 상세설계서, 2002.12, 개발자 회사(주)

〔5〕 검증명세서, 2002.12, 개발자 회사(주)

〔6〕 정보통신망 침입차단시스템 평가기준, 2000. 2, 한국정보보호진흥원

〔7〕 정보보호시스템 평가인증지침, 2002. 8, 한국정보보호진흥원

...

{ 개발자 작성 }

V. 결론

지금까지 국내 정보보호시스템 평가기준에 의한 시험과정 보증 요구사항을 분석하여 소프트웨어공학의 구조적 접근 방법론에 의한 보증문서 작성에 대하여 살펴보았다. 이는 정보보호시스템 보증요구사항에 대한 서술이 하나의 단일화된 명제로 정의되어 있는 것이 아니라 제품의 여러 특징(개발방법론 등)에 달라질 수 있으므로 구조적 접근 방법론을 기준으로 시험과정 보증 요구사항에 따른 제출물 작성의 한 예를 설명하였다.

앞에서 서술한 구조적 방법론을 비롯한 각종 소프트웨어공학기반 시험과정 작성 기법을 이해하여 평가 제출물 작성에 적용함으로써, 사용자 요구사항으로부터 시작된 보안 시스템이 객관적이고 체계적인 시험과정을 통하여 시스템 개발에서 발생할 수 있는 오류를

줄일 수 있으며, 나아가서 정보보안시스템에 정확한 명세가 구현할 수 있어 정보보호에 신뢰성 향상에 기여할 수 있다.

또한, 본고의 내용은 정보보호시스템 개발자의 보증 문서 작성에 대한 이해 수준의 지침으로 널리 활용될 것으로 기대되며 향후, 객체 지향 방법론 및 기타 가능한 방법론에 대한 작성법 개발도 현재 진행 중에 있다.

참고문헌

[1] 한국정보보호진흥원, 산업지원, http://www.kisa.or.kr
[2] "정보통신망 침입차단시스템 평가기준," 한국정보보호진흥원, 2000. 2.
[3] "정보통신망 침입탐지시스템 평가기준," 한국정보보호진흥원, 2000. 7.
[4] "정보보호시스템 평가·인증 지침," 정보통신부, 2002. 8.
[5] "정보보호시스템 평가·인증 가이드," 한국정보보호진흥원, 2002. 12.
[6] "개발자를 위한 침입차단·탐지시스템 평가제출물 작성 가이드," 한국정보보호진흥원, 2003. 10.
[7] "CobiT (3rd Edition) control objective," IT Governance Institure, 2000.
[8] "소프트웨어 테스트 전문기술," 한국정보통신기술협회, 2003. 11.
[9] "정보보호학회지," 한국통신정보보호학회, 2000. 9.
[10] Cem kaner, James bach and Bret pettichord, "Lessons learned in SOFTWARE TESTING," John Wiley & Sons, Inc., 2000.
[11] William E. Perry, "Effective Methods for Software Testing - 2nd edition", John Wiley & Sons, Inc., 2000
[12] 우치수 외 3인 공역, "소프트웨어 공학 실무적 접근, 한국맥그로힐(주), 2001.

〈著 者 紹 介〉

백 남 균 (Nam-Kyun Baik)
정 회 원

1998년 2월 : 숭실대학교 전자공학과(공학사)
1998년~1999년 : IntSecu Corp 연구개발부
2001년 2월 : 숭실대학교 정보통신전자공학부(공학석사)
2000년~현재 : 한국정보보호진흥원 산업지원단 평가2팀
〈관심분야〉 네트워크 보안, 네트워크 QoS, 정보통신 시스템 감사

최 용 준 (Yong-Joon Choi)
정 회 원

2000년 2월 : 한남대학교 컴퓨터공학과(공학사)
2002년 2월 : 한남대학교 컴퓨터공학과(공학석사)
2002년 1월~현재 : 한국정보보호진흥원 산업지원단 평가2팀
〈관심분야〉 네트워크 보안, 소프트웨어 공학, PKI, 스마트카드 보안

이 강 수 (Gang-Soo Lee)
종신회원

1981년 : 홍익대학교 컴퓨터공학과 졸업 (학사)
1983년 : 서울대학교 대학원 전산학과 졸업 (이학석사)
1989년 : 서울대학교 대학원 전산학과 졸업 (이학박사)
1985년-1987년 : 국립대전산업대학교 전자계산학과 전임강사
1992년-1993년 : 미국일리노이대학교 객원교수
1995년 : 한국전자통신연구원 초빙연구원
1998년-1999년 : 한남대학교 멀티미디어학부장
1987년~현재 : 한남대학교 컴퓨터공학과 정교수
〈관심분야〉 소프트웨어공학, 병행시스템 모형화 및 분석, 보안공학, 정보보호시스템 평가, 멀티미디어교육 커리큘럼