정보시스템 인증을 위한 정형기법의 활용

특집 06

목 차

서 동 수
(성신여자대학교)

1. 서 론

최근 들어 정보시스템에 대한 인증이 어느 때보다도 IT 업체들 사이에 중요한 이슈로 부각되고 있다. 잘 알려진 IT 부문의 인증으로는 ISO9001:2000 인증이 있으며 성격상 인증과는 다르지만 기관의 개발 능력의 성숙도를 평가하는 CMMI, SPICE 등은 인증에 준하는 용도로도 사용되고 있다.

정보시스템의 인증과 관련한 평가체계는 대상에 따라 크게 두 개의 범주로 구분한다. 먼저 IT 제품을 개발하는 개발 조직의 관리 수준과 개발 절차의 성숙도 측면에서 평가하는 범주가 있으며 ISO9001:2000 혹은 CMMI, SPICE와 같은 평가체계가 이에 속한다. 이와는 달리 제품을 대상으로 제품의 기능과 그 제품을 개발한 과정을 평가하는 분야가 있다. 이에 대한 예로는 정보보호 제품의 평가 기준으로 잘 알려진 공통평가기준, 혹은 ISO/IEC 15408)[1]과 같은 체계가 있다. 제품에 대한 평가는 주로 안전필수(safety-critical) 혹은 보안필수(security- critical) 시스템에 국한된 경우가 대부분이다. 영국에서 운영되는 국방관련 평가기준인 IDS(Interim Defence Standard) 00-55[2], 미국의 항공시스템 관련 기준인 RTCA DO-178B [3]등은 역시 안전과 국방관련 제품에 대해 활발히 적용되는 인증체계라 할 수 있다.

이들 평가체계는 제품이 제공하는 안전성 혹은 보안성의 강도에 따라 다양한 수준의 평가 등급을 제공한다. 공통평가기준의 경우 평가보증등급(Evaluation Assurance Level, EAL)으로서 EAL1~EAL7과 같은 7개 수준으로 구분하여 평가한다. 이 중 EAL1부터 EAL4 등급까지는 보증을 위해 특별한 보안기술 및 개발 기술을 적용시키지 않더라도 기존의 개발 절차를 체계화시키고 문서를 통해 일치성을 입증시킴으로서 요구하는 수준의 보증등급을 만족시킬 수 있다. 따라서 이들 보증등급은 일반적으로 기존에 있었던 제품과 시스템을 재정비하기 위한 관점에서 적용될 수 있다. 이에 비해 EAL5 이상의 등급은 정

도의 차이가 있기는 하지만 정형기법(formal methods)의 적용을 필수로 하고 있다.

정형기법의 적용은 개발자로 하여금 더욱 엄밀한 개발을 하도록 강요하며 이러한 이유로 EAL5 이상의 인증을 고등급 인증이라 부른다. 본 논문은 정보시스템의 인증에 있어 고등급 인증을 받고자 할 경우 고려해야 하는 기술적인 요소를 소개한다.

2. 관련분야 인증 현황

고등급 인증 현황은 그 나라의 IT 기술이 정교한 개발기법을 구사할 수 있는 수준으로 성숙되었는지를 말해줄 수 있는 지표이다. 정보보호 관련 제품의 경우 2003년 기준으로 보고된 현황에 따르면 미국, 영국, 호주 등 5개국의 123개의 인증된 제품 중 14개가 EAL5 이상의 인증을 받았으며 이는 전체 제품의 11%에 해당하는 것이다[4]. EAL5 이상의 등급을 획득한 제품들은 대부분 스마트카드 및 이와 관련된 운영체제로서 젬플러스, 필립스, 인피니언 등의 회사들이 개발한 바 있다[5]. 특히 슬럼버저는 2003년 7월 자사의 스마트카드와 터미널이 국제공통평가기준의 EAL7 등급을 획득했다고 발표했다. 이것은 세계 최초로 공통평가기준에 근거한 최고등급이며 평가는 프랑스의 평가기관인 DCSSI에서 수행하였다. 그러나 국내의 경우 2006년 현재 공통평가기준 및 K 등급 인증을 받은 45개의 제품 중 단 1건의 제품도 EAL5이상 수준의 고등급 인증을 받은 경험이 없다.

핵발전소의 통제 시스템은 어떤 시스템보다도 엄밀한 개발이 요구되는 분야로 손꼽아 왔다. 이 분야 고등급 인증의 첫 번째 경우로 언급되는 영국의 시즈웰 비(Sizewell B) 핵발전소는 안전필수 기준인 IEC880(International Electronic Commission)을 만족시킨 경우이다[6]. 이 시스템의 개발과정에 사용된 검증 기법으로는 동적인 테스팅과 정형기법에 의한 분석이 사용되었다.

항공분야에 있어서 록히드마틴에서 개발한 C130J 항공기에 탑재된 제어용 컴퓨터가 DO-178B의 레벨 A 기준을 만족하는 것으로 인증을 받았다. 이 시스템은 10만 줄의 코드로 구성되며 코드의 대부분은 SPARK 검증언어로 작성되었다.

3. 고등급 인증체계의 특성

고신뢰 시스템의 평가에 관련된 대표적 체계는 RTCA(Requirements and Technical Concepts for Aviation) DO-178B를 들 수 있다. DO-178B는 항공 시스템과 장비에 대한 산업계 지침서로서 미국과 유럽의 145개 이상의 기관이 참여하고 있다. 이 지침은 항공산업 장비 및 이와 관련된 소프트웨어의 인증 기준을 제공한다. 이 문서의 섹션 3에서는 소프트웨어 개발 생명주기를 다음의 3 영역으로 구분한다.

첫째, 계획 단계는 전체 개발 공정에 관한 계획을 수행하는 부분으로 개발 단계와 통합 단계를 포괄하여 수행된다. 둘째, 개발 단계는 소프트웨어 요구분석, 설계, 코딩, 통합 등의 세부 단계를 수행한다. 마지막으로 완전화 단계는 개발 영역의 일을 지원하는 절차로 개발영역에서 확인된 세부 산출물에 대한 정확성과 품질에 관한 확인을 하는 단계이며 소프트웨어 검증, 소프트웨어 형상관리, 소프트웨어 품질보증을 수행한다. 특히 DO-178B 섹션 6에서는 검증 과정에서 리뷰와 분석, 그리고 정형검증을 포함하여 테스트의 한계를 보완하길 권한다.

공통평가기준은 초기 제정되는 과정에서 RTCA DO-170B와 많은 관련을 가지고 출발하였다. 공통평가기준은 기능요구사항과 보증요구사항을 구분하여 평가한다. 기능요구사항은 평가대상이 되는 제품이 갖추어야 할 보안기능에 대한 서술이며 보증요구사항은 이러한 기능요구사항들이 얼마나 잘 구현되었는가하는 과정에 대한 서술이다. 요구사항은 엄밀성의 정도에 따라

EAL1에서 EAL7의 단계를 가지며 이 중 고등급에 해당하는 EAL5 이상의 보증요구사항을 요약하면 <표 1>과 같다.

EAL5는 개발자가 전문적인 보안공학 기법으로 인한 추가의 비용의 부담 없이 엄격한 개발방법론을 사용할 것을 요구하는 경우에 적용이 가능하다. EAL5는 기능명세, 완전한 인터페이스 명세, 기본설계 및 상세설계, 구현의 전부를 이용하여 보안기능을 분석함으로써 보증을 제공한다. 또한, 평가 제품의 보안정책에 관한 정형모델, 기능명세 및 기본설계의 준정형화된 표현, 그들 간의 준정형화된 일치성 입증을 통하여 보증을 제공한다.

EAL6은 개발자가 심각한 위험으로부터 높은 가치의 자산을 보호하기 위해 엄격한 개발환경에서 보안공학 기법을 적용해 얻을 수 있는 보증을 제공한다. 이 등급은 보호되는 자산의 가치가 추가적인 비용을 정당화할 수 있는 위험 상황에 사용하기 위한 제품을 개발할 경우에 적용이 가능하다.

<표 1> 개발 보증 클래스의 구분

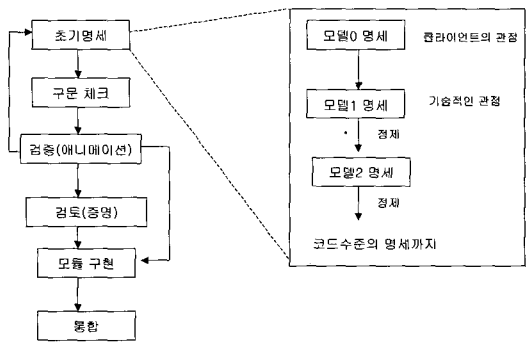
구분	EAL5	EAL6	EAL7
기능명세	준정형화된 기능명세	EAL5와 동일	정형화된 기능명세
기본설계	준정형화된 기본설계	EAL5와 동일	정형화된 기능설계
상세설계	서술적인 상세설계	준정형화된 상세설계	EAL6와 동일
구현	-보안기능에 대한 구현표현 -모듈화	-보안기능에 대한 구현 표현의 구조화 -복잡도 감소	-보안기능에 대한 구현 표현의 구조화 - 복잡도 최소화
일치성	준정형화된 일치성 입증	EAL5와 동일	정형화된 일치성 입증
보안모델	정형화된 평가대상의 보안정책모델	EAL5와 동일	EAL5와 동일

EAL7은 극도로 높은 위험 상황이나 자산의 가치가 높아서 많은 비용을 정당화할 수 있는 상황에서 사용하기 위한 제품 개발에 적용 가능하다. 현재 EAL7의 실제적인 적용은 광범위한 정형화된 분석을 필요로 하는 보안기능을 최대의 목표로 삼는 제품에 국한된다.

보안작용을 이해하기 위하여 평가대상의 기능과 인터페이스 명세, 서브시스템들의 기본설계, 평가대상 모듈의 상세설계 및 구현의 구조화된 표현을 이용하여 보안기능을 분석함으로써 보증을 제공하며 추가적인 보증은 정형모형과 정형기능명세, 기본설계서, 준정형 상세설계서 간의 일치성 검증을 수행한다.

4. 적용되는 정형기법 기술

시스템 개발 기법으로서의 정형기법의 핵심은 모호함과 불완전성, 모순의 문제를 가진 사용자 요구 사항을 대상으로 수학적으로 검증 가능한 수준의 명세를 획득하는데서 부터 시작한다. 이 과정은 (그림 1)의 개발 절차에 요약되어 있다.



(그림 1) 정형기법을 이용하는 시스템 개발 절차

첫 번째 단계는 정형명세를 획득하는 단계이다. 이 단계에서는 먼저 사용자와의 인터뷰를 통해 얻어진 비정형 문서를 바탕으로 명세를 구조화시키는 작업을 수행한다. 정형기법은 많은 경우 사용자 명세로부터 직접 정형명세를 얻어내는 직접적인 접근을 사용한다. 이 과정에서 사용자가 명세의 검증에 참여할 수 있는 기회는 제한될 수 밖에 없다.

획득 과정을 거쳐 작성된 정형명세에 대해 여러 가지 검증 기준이 적용될 수 있으나 가장 먼

저 적용 가능한 기준은 정확성이다. 정확성(correctness)이란 사용자가 원하는 모든 기능요구가 명세에 모두 반영되었음을 의미하는 개념이다. 정확성은 대상이 무엇인가에 따라 의미 정확성과 구문 정확성으로 나뉜다.

의미 정확성이란 주로 비정형적으로 서술된 요구내용이 정형명세에 표현되어 있는지 확인하는 개념이다. 의미 정확성을 가장 효과적으로 보여줄 수 있는 방법은 자연어 명세에 대해 구문 혹은 문단 단위로 구분하여 정형명세와 병치시켜 명세를 하는 것이다. 만일 단순한 병치가 충분히 그 의미를 전달해 줄 수 없는 경우 분석가는 정형명세가 왜 비정형 서술에 대한 올바른 정형적 해석인지를 설명하는 주석을 함께 제공하여 정확하게 의미가 포착되었음을 보여줄 수 있다. 예를 들어 "모든 보안 자산 요소는 유일하게 식별되어야 한다"라는 조건을 명세한다고 하자. 이 구문에서 나타난 유일성을 형식적으로 말하면 보안 자산 a1, a2에 대해 식별자 Id(a1)과 Id(a2)가 동일하다면 두 자산은 동일한 것이고 그 대우로 그 식별자들이 동일하지 않다면 보안 자산들 또한 동일하지 않다라고 할 수 있으며 이를 수식으로 표현하면 다음과 같다.

$$\forall a1,\ a2 \in Asset \bullet (Id(a1) = Id(a2) \Rightarrow a1 = a2) \wedge (a1 \neq a2 \Rightarrow Id(a1) \neq Id(a2))$$

구문 정확성은 명세가 문법적으로 올바른 구조를 가졌는지를 확인하는 개념이다. 대부분의 정형명세는 수학기호와 많은 괄호, 그리고 반복구조를 포함하고 있기 때문에 구문지향 편집기와 같은 도구를 사용하지 않고 구문에러가 없는 명세를 만드는 것은 어려운 일이다. 구문 정확성을 보이기 위해서는 도구에서 제공하는 타입검사기를 사용하여 문법이나 타입에 관한 오류가 없음을 확인시켜 줄 필요가 있다.

구문과 의미 정확성을 판단하는 과정 이외에 필요한 것이 모델 체킹이다. 모델 체킹은 개발자가 얻은 명세에 원하는 특성이 포함되었는지를 명세의 에니메이션 혹은 동치모형을 통해 확인하는 방법이다. 모델 체킹은 명세가 CCS나 CSP 스타일의 프로세스 대수적으로 표현되었을 때 유용하게 쓰이는 방법이다. 프로세스 대수적으로 표현된 명세는 쉽게 BDD(Binary Decision Diagram)으로 변환될 수 있으며 BDD를 통해 만족성(satisfiability)테스트나 동치성(equivalence)테스트를 쉽게 할 수 있다[8].

두 번째 검증은 초기 형태의 정형명세에 대해 명세 내부에 포함되어 있는 비일관성과 모순성이 제거되었는지를 확인하는 일관성에 관한 검증이다. 일관성(consistency)은 명세를 구성하는 요소 간에 논리적 모순이 존재하지 않음을 확인하는 개념이다. 만일 명세에 모순이 개입되면 그러한 명세로부터 유도되는 어떠한 결론도 참일 수가 없다. 따라서 특정 명세가 갖는 의미에 대해 진위 판단을 할 수 없는 문제가 발생한다. 일관성이 있는 명세가 되기 위해서는 명세들 사이에 모순이 존재하지 않음을 보여주어야 한다.

Z나 B와 같은 상태기반 기법에서는 명세의 무모순성을 보여주는 방법으로 상태증명 기법을 사용한다. 상태증명 기법이란 어떠한 명세이던 만일 그 명세가 모순적이지 않다면 상태가 존재할 수 있음을 보여주는 증명 방법이다. 상태기반 명세의 경우 개발자는 상태불변자, 즉 어떤 상태가 있다면 그 상태 내에서 항상 참으로 만족시켜야 할 성질을 모아 놓은 일종의 술어 명제집합을 명세 내에 포함하여 사용한다. 상태증명을 가장 간단하게 활용할 수 있는 방법은 초기상태가 존재함을 보이는 것이다. 초기상태 역시 상태 집합을 구성하는 많은 상태 중의 하나이나 종료상태를 존재하도록 하는 근거가 되는 정보이므로 초기상태가 존재함을 보여주는 것도 상당히 중요한 일이다. 초기상태가 존재함을 보여준다면 그 명세 내에는 모순이 없는 상태가 존재함을 보이

는 것이고, 결과적으로 일관성 있는 명세임을 보여주는 방법이 된다.

세 번째 단계에서는 완전성에 관한 검증을 수행한다. 완전성(completeness)에 관한 수학적 의미는 "참이라 알려진 모든 문장은 증명이 가능하다"라는 내용으로 요약할 수 있다. 앞서 설명한 일관성과 혼동이 될 여지가 있지만, 완전성을 다른 의미로 정의하면 "모델에서 정의된 규칙들만을 가지고 어떤 특성을 참인지 거짓인지를 증명할 수 있을 만큼 모델이 충분히 완전한가를 확인하는 작업이다"라고 말할 수 있다.

Z 기법에서 완전성은 어떠한 순서로 오퍼레이션이 작동하더라도 그 결과는 항상 정의된 도메인 내에 있다는 것을 보장하는 성질로 정의된다. 완전성은 오퍼레이션의 작동 결과로 오류가 일어나지 않음을 보장할 때 유용하게 사용된다. 이 때 사용되는 증명방식을 선조건 증명이라 부른다. 선조건 증명방식은 증명대상인 오퍼레이션이 갖는 후조건, 즉 오퍼레이션의 수행이 종료되었을 때 항상 참으로 만족시켜야 할 조건들로부터 명령어들을 역적용시켜 결국 얻는 최종 조건, 바꾸어 말하면 오퍼레이션의 초기조건은 다름 아닌 해당 오퍼레이션의 선조건임을 보이는 증명기법이다.

마지막 단계에서 고려되는 것은 일관성과 완전성, 무모순성이 검증된 명세로부터 구현 작업이 수행되었을 때 코드는 명세에 대한 올바른 반영이라는 일치성에 관한 증명이다. 일치성 증명은 코드와 코드가 만족시켜야 하는 조건으로서 선조건, 후조건, 루프 불변자와 같은 조건들이 만족되고 있는지를 보여 줌으로써 증명될 수 있다. JML[9]과 같은 주석 언어는 Java언어에 선조건, 후조건을 표현할 수 있도록 구문을 확장한 것으로 이것은 LOOP 컴파일러 불리는 컴파일러를 통해 얻은 중간 표현을 PVS(Prototype Verification System)을 통해 검증할 수 있도록 도와준다.

5. 결 론

정보시스템의 인증에 있어 정형기법의 성공적인 적용은 고등급 인증을 받기 위한 핵심요소로 작용하고 있다. 앞서 보아온 것처럼 스마트카드와 같은 분야에서는 EAL5 이상의 인증을 요구하는 것이 보편화되어가고 있다. 정형기법을 도입하는 과정에서 개발자들이 갖는 공통적인 고민은 두 가지이다. 첫째, 정형기법 자체가 산업체에서 일반적으로 받아들일 수 있는 수준으로 성숙되지는 못하다는 점이고 둘째, 적용과정에서 많은 시간과 비용이 든다는 점이다. 이러한 이유로 Z와 VDM이 개발되었던 영국에서조차 이미 IDS 00-55에서 정형기법의 사용을 의무화할 때부터 논란이 되었다. 그럼에도 불구하고 고등급 인증을 필요로 하는 시스템에 정형기법이 요구되는 이유는 아직까지 이 기술이 제공하는 견고성과 엄밀성을 능가하는 기술이 나오고 있지 않기 때문이다.

국내의 정보시스템 개발 현실에 비추어 볼 때 정형기법의 도입은 아직 시기상조라는 의견이 많다. 하지만 안전성과 보안성이 요구되는 분야에 있어 정형기법의 도입은 선택의 문제가 아닐 수가 있다. 특히 최근 몇 년 사이의 국내 IT 기술 동향이 점차 임베디드와 유비쿼터스 시스템 등 실시간, 안전성 필수 특성이 강조되는 추세로 가고 있다는 점을 고려하면 멀지 않아 국내에도 고등급 인증에 관한 수요가 증가하리라 예상된다.

정형기법은 도구를 구입한다거나 새로운 프로그래밍 환경을 도입함으로써 단기간에 효과 얻으려한다면 실패할 가능성이 많다. 물론 증명도구나 편집도구의 활용은 정형기법을 성공적으로 적용하기 위해 필수적인 사항이지만 그 이전에 평가자나 개발자 모두 기법에 대한 충분한 지식을 습득해야 한다. 이를 위해서는 장기적으로 인증 전문가 및 개발자 양성을 위한 교육 프로그램이 도입되어야 한다. 또한 이와 병행하여 고등급 인증을 위한 파일럿 프로젝트를 수행한다면 개발 지식의 축적에 큰 도움이 될 것으로 보인다.

참고문헌

[1] 정보시스템 공통평가기준, 정보통신부 2002
[2] MOD 00-55, The Procurement of Safety Critical Software in Defence Equipment, 1991
[3] RTCA DO178-B, Software Considerations in Airborne Systems and Equipment Certification, 1992
[4] 정보보호시스템 인증현황, KISA 세미나 자료, 2003
[5] http://www.commoncriteriaportal.org
[6] Bruns G., Anderson, S, Gaining Assurance with Formal Methods, Applications of Formal Methods, Prentice Hall, 1995, p33-54
[7] Woodcock J., Davis J., Using Z, Prentice Hall, 1996
[8] Dsoza, A, Bloom, B. Generating BDD Model from Process Algebra Terms, Computer Aided Verification LNCS 939, 1995 pp 16-30
[9] Leavens G., Poll E 외. JML Reference Manual, May 2006

저자약력

서 동 수

1987년 중앙대학교 컴퓨터공학과(학사)
1989년 중앙대학교 컴퓨터공학과(석사)
1994년 Univ. of Manchester, Dept. of Computation(박사)
1994년-1998년 전자통신연구원, 선임연구원
1998년-현재 성신여자대학교 컴퓨터정보학부 부교수
관심분야 : 소프트웨어공학, 정형기법, 정보보호기술
이 메 일 : dseo@sungshin.ac.kr