情 報 保 護 學 會 誌
第 15 卷 第 4 號, 2005. 8

CC 3.0의 변화 내용 분석*

강 연 희**, 김 정 대**, 최 상 수**, 이 강 수**

요 약

ISO/IEC ISO/IEC 15408인 CC(Common Criteria)는 전세계 정보보호 학계와 산업 분야에서 정보보호 기능과 보증에 대한 표준적 개념, 평가 및 분류체계로 자리매김했다. 현재의 공식적인 버전은 CC 2.2이지만, 2004년 3월에 CC 2.4(초안)가 발표된후, 공식버전이 되기도 전인 2005년 7월에 CC 3.0 (초안, 수정 2판)이 발표되었다. CC 3.0은 50%이상 대폭 변화되었으며 지난 10년간 경험했던 CC의 문제점을 해결하려한 흔적이 역역하다. 본 논문에서는 CC 3.0의 변화를 CC 2.4와 함께 조사 및 분석하였고 변화에 따른 문제점과 대책을 제시한다.

Ⅰ. 서 론

ISO/IEC 15408로 표준화되어 있는 CC (Common Criteria)는 각종 정보보호 제품/시스템 (TOE: target of evaluation)에서 공통적으로 사용할 수 있는 보안기능 및 보증요구사항의 계층적 집합이다. CC는 미국의 TCSEC, 유럽의 ITSEC, 카나다의 CTCPEC과 같은 정보보호제품 평가기준을 통합한 것이며, 우리나라를 포함한 선진 각국은 CC를 평가기준으로 하는 정보보호제품 평가인증체계(scheme)를 운영하고, CCRA 협정을 통해 CC에 따라 평가된 제품의 평가보증 결과를 국가간에 상호인정하고 있다.[1] 우리나라도 2004년 하반기에 CCRA에 가입을 신청한바 있다.

CC 1.0은 1996년 1월에 발표되었으며, 2005년 7월의 공식버전은 CC 2.2(2004년 1월)이며 이는 1999년 8월에 발표된 CC 2.1의 오류를 수정한 것이다.[2] 우리나라의 "정보보호시스템 공통평가기준"(2002년 8월, 정보통신부장관고시)은 CC 2.1을 번역한 것이다[2-14].

카나다, 프랑스, 네덜란드, 영국, 미국으로 구성된 'CC 프로젝트지원조직'은 2004년 3월에 CC 2.4를 발표하였고[15,16], 2005년 7월에는 호주/뉴질랜드, 독일, 일본 및 스페인이 추가로 가담하여 CC 3.0(초안)을 발표하였다.[17-19] CC 3.0은 아직 초안 수준이지만 향후의 정보보호 제품의 개발방법론 및 평가기준의 골격이 될 것이다. 특히, 우리나라는 CCRA에 가입을 목전에 두고 있고 정보보호 제품의 개발 및 평가기술을 제고하기 위해 CC의 변화 동향을 주시해야한다. 이와 같은 배경에서, 본 논문에서는 CC 3.0의 변화내용을 분석한다. CC 2.4의 변화내용은 문헌 [20]을 참고하며 CC 3.0의 대략적인 변화내용은 문헌 [27]을 참조한다. CC 3.0은 아직 초안수준이므로, 각 보안기능 및 보증 컴포넌트간의 종속관계의 변화에 대해서는 분석하지 않았다.

본 논문의 2장에서는 CC 3.0의 변경의 목적과 보안개념과 평가에 대한 패러다임의 변화를 보였으며, 3장과 4장에서는 보안기능요구사항과 보증요구사항의 변화를 각각 분석하였다. 5장에서는 CC의 평가지침인 CEM과 PP와 ST의 변화를 보였고, 6장에서는 CC의 변화에 대한 분석결과를 결론과 함께 제시한다.

Ⅱ. 변경의 목적과 패러다임의 변화

2.1 CC 3.0 변경의 목적

CC의 문제점과 CC 3.0에서의 해결전략은 다음과 같다.[27]

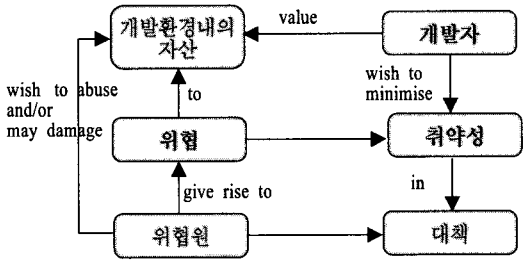
- 복잡성, 애매성 및 비일관성 문제 : 2부 기능요구사항은 354쪽에서 130쪽으로 간략화하였고 클래스와 패밀리의 수를 대폭 줄였으며, 사용한 용어의 개수도 줄이고 명확히 정의하여 일관성을 제고했다.

* 본 논문은 산업자원부 지역협력연구사업(R12-2003-004-01001-0) 지원으로 수행되었음.
** 한남대학교 컴퓨터공학과 ({dusi82, jdcom, gcss09}@se.hannam.ac.kr, gslee@eve.hannam.ac.kr)

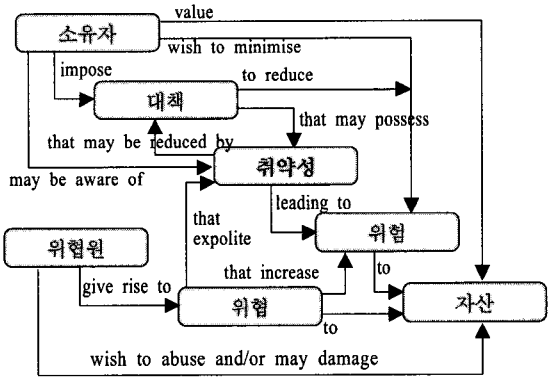
- 보증요구사항의 불합리성 및 중복 : 보증요구사항을 전면 재작성 및 재편집했다. 유사한 평가업무의 반복을 제거했고 보증에 필요한 분야를 강조했다.
- 개발자의 사용편이성 제고 : 개발자가 쉽게 볼 수 있도록 CC와 CEM을 재구성했다. 특히, 개발보증클래스(ADV)를 단순화하여 평가자가 아키텍쳐와 보안기능을 쉽게 이해할 수 있도록 했다.
- 합성형 보안제품(TOE)의 평가기준 추가 : 이미 평가된 "부품형 TOE" (component TOE)들을 통합하여 구축한 "합성형 TOE" (composed TOE)의 평가기준을 추가하였다.

2.2 패러다임의 변화

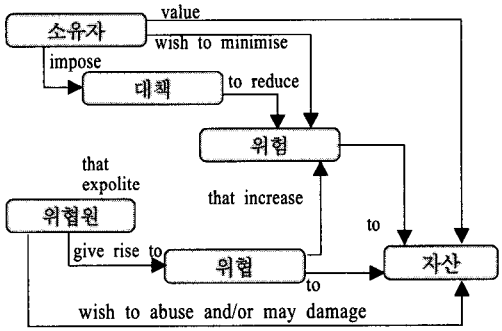
CC 2.4이후에서는 보안 및 평가의 개념과 관계에 대한 패러다임이 변하였다. 그림 1과 그림 2에서 보듯이 보안 개념에서 "취약성(vulnerability)"이라는 엔티티가 삭제되었고, 평가개념에서 "취약성"과 "충분함(sufficient)"이라는 엔티티가 추가되었다. 즉, 취약성 엔티티는 보안개념이 아니라 평가개념에서만 사용되는 개념이 되었다. 취약성과 위협 엔티티간의 관계는 이를 사용하는 문맥이나 접근방법에 따라, 매우 다양하게 정의된다.[21] 따라서, CC 2.4에서는 '보안개념'과 '평가개념'에서 애매하고 중복된 엔티티인 위협과 취약성 중 한 가지 개념만 사용하고 있다. 그러나, 그림 3과 같이 '개발자 개념'에서는 위협과 취약성을 함께 사용하고 있다.



〔그림 3〕 CC 2.4에서 추가된 개발자 개념과 관계

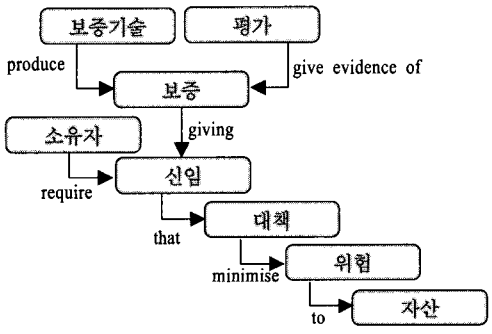


(a) CC 2.2

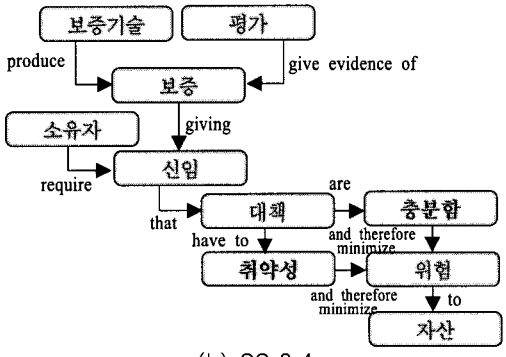


(b) CC 2.4, 3.0

〔그림 1〕 보안개념과 관계의 변화



(a) CC 2.2



(b) CC 2.4

〔그림 2〕 평가개념과 관계의 변화

Ⅲ. 보안기능요구사항의 변화

3.1 CC 2.2의 보안기능의 문제점

어떤 제품군(예: 스마트카드, VPN, OS, DBMS 등)의 PP(protection profile)는 CC내의 보안기능 요구사항 컴포넌트(패밀리, 클래스)중 일부를 선택하고, 보안기능문장내의 '할당'과 '선택' 부분을 적절한 값으로 '연산'(즉, 할당 및 선택)하여 실제의 '보안기능 문장'을 구성한다. PP를 개발해본 사람이라면, 어떤 보안기능요구사항 컴포넌트를 택할 것인가를 판단하는 일은 매우 어렵고 주관적인 일이다. 이는 기존의 CC 보안기능 컴포넌트(패밀리, 클래스)에 다음과 같은 문제가 있기 때문이다.

- 기능간의 중복성 문제 : 전체 보안기능을 독립된 부분적 보안기능으로 분할하는 것은 거의 불가능하므로, CC의 보안기능 컴포넌트 간에는 중복이 심하다. 예컨대, CC 2.2의 FDP_ACF (access control function)와 FDP_IFF (information flow control function)간에는 기능이 중복되며, FAT (TOE access)와 FTP (trusted path/channel) 클래스 전체도 다른 기능들과 매우 중복되어있다.
- 기능의 애매성 문제 : FPT_RPL (replay detection)이나 FPT_RVM (reference mediation)처럼 일반적인 보안기능으로 보기 어려운 것들도 있다. 예컨대, FAU (security audit) 클래스내의 FAU_SAR (security audit review), FAU_SEL (security audit selection), FAU_STG (security audit event storage)은 FAU_SAA (security audit analysis) 보안감사 자료 생성과 분석에 포함되는 기능이라고도 볼 수 있다.
- 기능의 크기 차이 문제 : 보안기능 컴포넌트들은 난이도나 기능의 '크기'(granularity)가 서로 다르다. 즉, 더욱 세분화 할 수 있는 컴포넌트(예: FAU_GEN.1 (audit data generation) 등)와 지나치게 세분화된 컴포넌트(예: FDP_ROL.1 (basic rollback) 및 FDP_ROL.2 (advanced rollback) 등)가 존재한다.
- 기능의 범위 문제 : CC는 원칙적으로 암호기능이나 물리적 보안관리 기능은 다루지 않지만, CC 2.2에는 FCS (cryptographic support) 클래스와 FMT (security management) 클래스는 CC와는 별도의 평가인증체계인 "암호모듈검증프로그램(CMVP)"와 "정보시스템보안관리 인증(ISMS)"에 각각 포함된다. 또한, FPR (privacy)내의 FPR_ANO (anonymity)와 FPR_PSE (pseudonymity) 패밀리는 관점에 따라 정보보호 기능으로 볼 수 없을 수 도 있다. 따라서, FCS와 FMT내의 컴포넌트를 선택할 때 망설여진다. CC 3.0에서는 이와 같은 문제점을 개선하여 대폭 변경하였다.

3.2 변화의 일반 내용

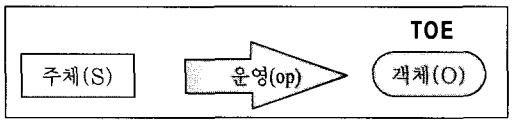
3.2.1 보안기능 분류의 단순화

TOE를 중심으로 다음과 같이 5가지로 보안기능을 분류하고 있다.

- TOE내부의 보안기능 : 접근통제 등 TOE내부 행동
- 외부 엔티티가 TOE에 접속 : 식별, 인증
- TOE와 접속된 외부 엔티티간의 통신보안 : 기밀성, 가용성, 무결성, 부인봉쇄
- 보안감사 : 로깅, 보안관련 사건의 대응
- 보안기능 자체의 보호 : 물리적 공격, 자원 고갈로 부터의 보안기능 자체의 보호

3.2.2 주체-객체-운영모델의 적용

CC 3.0에서는 그림 4와 같은 전통적인 보안모델인 주체(subject), 객체(object) 및 운영(operation) 모델("SOO모델")을 사용하여 각종 보안기능을 정의하고 분류하였다. 사용자는 주체의 한 인스턴스이다. 이 모델을 이용함으로서 보안기능의 정의와 분류가 용이하며 기능의 누락 및 중복을 최소화할 수 있다.



주체의 예: Name=Joe, Type=12, Id=2A1
객체 예: level=TOP, tag=now.md5
운영의 예: read, update, write

〔그림 4〕 SOO 모델

CC 3.0의 보안기능 패밀리인 FDP_OSD (object/subject destruction), FDP_UNL (unlinkability), FIA_UAU (subject authentication), FIA_USB (user-subject binding), FPT_PRI (priority), FPT_RSA (resource allocation)에서는

"SOO모델"을 통해 보안기능을 정의하고 있다.

3.2.3 기밀성-무결성-가용성모델 적용

CC 3.0에서는 보안성을 구성하는 3대 하위 속성인 기밀성(confidentiality), 무결성(integrity) 및 가용성(availability) 차원(이를 "CIA 모델"이라 하자)에서 통신 보안기능을 정의하고 분류하고 있다. CC 2.2에서는 CIA 모델을 전체 기능에서 일관성 없이 사용함으로써(예: FDP (user data protection), FPT (protection of the TSF)에서 CIA모델을 이용) 혼란이 있었으나, CC 3.0에서는 다음 표 1과 같이 FCO (communication) 클래스에서만 CIA 모델을 사용하고 있다. 즉, FCP이외의 보안기능 클래스에서는 무결성, 가용성 및 기밀성 용어를 사용하고 있지 않다.

3.2.4 전입-전출 모델 적용

TOE로의 전입(import)자료와 전출(export) 자료의 관점에서 보안기능을 정의 및 분류하고 있다(이를 'IE 모델'이라 하자). CC 2.2에서는 UDP (user data protection) 클래스에서 FDP_ETC, FDP_ITC (import/export from/to outside TSF control)에서는 전입과 전출기능을 모두 이용했지만, FPT (protection of the TSF)내의 FPT_ITA, FPT_ITC, FPT_ITI (availability/confidentiality/integrity of exported TSF data)에서는 전출만을 고려하였으므로, 전입과 전출기능의 사용에 있어서 일관성이 부족하다. CC 3.0에서는 표 1과 같이 FCO 클래스에서만 "IE모델"을 사용하여 기능상의 혼란을 줄이고 있다.

〔표 1〕 FCO 클래스에서 CIA모델 및 IE모델의 활용 예

방향 및 객체 / 기능 및 속성	전입(imported) 자료	전출(exported) 자료
기밀성	FCO_CID	FCO_CED
무결성	FCO_IID	FCO_IED
가용성	N/A	FCO_AED
변화(translation)	FCO_TID	FCO_TED
부인봉쇄 (non-repudiation)	FCO_NRI	FCO_NRE
비관찰가능성 (unobservability)	N/A	FCO_UNE (unobservability of export)
보안기능통제 (TSF control)	FCO_ETC (import from outside TSF control)	FCO_ETC (export to outside TSF control)

3.2.5 보안기능패밀리 구조의 변화

CC 2.2에서는 보안기능패밀리의 속성이 패밀리명, 패밀리행위, 컴포넌트 레이블링, 감사, 컴포넌트로 구성되었지만 CC 3.0에서는 '패밀리행위' 대신 '설명(narrative description)'으로 변경되었고 '관련연산'이 추가되었다.

3.3 세부 변경사항

부록 A에는 CC 2.2와 CC 3.0간의 보안기능요구사항의 차이를 보이며 주요 변경내용은 다음과 같다. CC 3.0에서 보안기능의 명칭은 변하지 않았어도 세부내용이 변한 것이 많음을 주의해야한다.

3.3.1 기능클래스

11개를 6개로 줄였고 중복된 클래스를 통합하였다.

- 완전 삭제 : FCS (cryptographic support)는 보안요구사항의 구현 수단(means)에 해당하므로 완전히 삭제함
- 다른 클래스로 통합 : FMT (security management)는 FDP (data protection and privacy)로 통합; FPR (privacy)는 FIA (identification, authentication and binding)와 FDP로 분산 통합; FRU (resource utilisation)은 FPT (protection of the TSF)로 통합; FTA (TOE access)는 FDP와 FIA로 분산 통합; 따라서 CC 3.0에서는 FDP와 FIA의 기능이 증대되었음; FDP (user data protection)은 FDP (data protection and privacy)로 세부 명칭이 변경됨
- 신규 클래스 : FMI (miscellanous)가 추가됨

3.3.2 기능패밀리

67개를 45개로 줄이고 소속 클래스를 변경하는 등 대폭 변경하였다.

- 완전 삭제 : FCS가 삭제되므로, FCS_CKM (cryptographic key mgmt.)과 FCS_COP (cryptographic operation); FTP가 삭제되므로, FTP_ITC (inter-TSF trusted channel)과 FTP_TRP (trusted path); FDP_DAU (data authentication)에서 자료의 소유권(authorization)은 보안기능이 아니라 속성에 해당하므로; FDP_ITT (internal TOE

transfer)는 분산 TOE에만 해당하므로; FMT_SMF (spec. of management function); FPT_RVM (reference mediation); FPT_SEP (domain separation); FPT_SSP(state synchrony protocol), FPT_ITT (internal TOE TSF data transfer), FPT_TDC (inter-TSF TSF data transfer) 및 FPT_TRC (internal TOE TSF data replication consistency)는 다른 기능에 포함된 보안기능이므로; FTA_SSL (session locking)과 FTA_TSE (TOE session establishment)는 통신기능에 해당하므로 삭제

- 다른 패밀리로 통합 : 보안감사 기능인 FAU_SAR (security audit event review), FAU_SEL (security audit event selection) 및 FAU_STG (security audit event storage)이 FDP_ACC (access control)로 통합됨; CC 3.0의 FDP_ACC (access control)은 CC 2.2의 12개의 패밀리의 내용을 수용하고 있음; (다른 패밀리로 통합된 패밀리는 전체 67개 패밀리중 39개에 이르며, 세부사항은 부록 A를 참고한다.)
- 신규 패밀리 : FCO_TID (translation of imported data); FCO_TED (translation of exported data); FDP_OSD (object/subject destruction); FIA_SAS (subject authentication); FMI_RND (random number gen.), FMI_CHO (choice)

Ⅳ. 보증의 변화

4.1 CC 2.2의 보증요구사항의 문제점

보증요구사항은 보안제품을 얼마나 안전한 곳에서(즉, 보안환경), 정형화되고 안전한 개발공정(프로세스)을 통해(즉, 생명주기), 안전한 보안제품을 개발, 시험 및 취약성분석을 하고 지침문서를 올바르게 개발했느냐에 대한 제삼자(즉, 인증자)로부터의 보증에 대한 요구사항이다.

CC에서 보증의 수준은 EAL1~EAL7이며 민간평가기관은 EAL1~EAL4까지만 평가한다. EAL이 높을수록 좀 더 구조적(structured), 방법론적(methodical) 및 정형적(formal)으로 TOE를 개발 및 시험했음을 보증해야한다.

보안기능요구사항과 달리, 보증요구사항은 보안제품별로 컴포넌트를 선택하는 것이 아니라, 목표 EAL만 정하면 이에 따른 보증요구사항 컴포넌트들이 미리 정해져 있으므로, PP나 ST 개발시 어려움은 없다. 그러나, 평가시에 다음과 같은 문제점이 보증요구사항에 존재한다.

- 보증간의 중복성 문제 : CM (configuration management) 클래스는 TOE의 개발 및 운영 전체기간 동안 실시되는 것이므로, ALC (life-cycle support) 클래스와 다소 중복된다. 특히, ACM_AUT (CM automation)는 ACM_CAP (CM capability)가 있으므로 중복된다.
- 불필요한 보증 문제 : ADO (installation, generation and set-up)는 CC의 범위라기보다는 보안관리(ISMS)의 범위라 볼 수도 있다. 특히, AVA_SOF (strength of TOE security function)은 3등급으로 보안기능의 강도를 보증하는 것이지만, CC에서의 보증수준인 7등급의 EAL이 있으므로 불필요한 보증요구사항이다. 또한, 그동안 3가지수준(즉, 비정형적, 반정형적 및 정형적)의 컴포넌트로 세분화되어있던 ADV_SPM (security policy modeling)은 1개의 컴포넌트(즉, '정형적' 보안정책모델)로 되었다. 이는 보안정책모델 자체는 무조건 정형적이어야 함을 의미한다.
- 보증 대상의 애매성 : AVA_MSU (Misuse)는 TOE의 '오용'에 대한 보증이지만 오용이라는 개념은 보증요구사항에 포함시키기가 애매하다.
- 합성된 제품의 보증 문제 : 기 평가된 기존의 제품(예: OS, DB 등)위에 새로운 보안기능을 추가한 합성 제품에 대한 평가기준이 없다.

4.2 변화의 일반 내용

4.2.1 ST평가 클래스의 추가

CC 2.4와 CC 3.0에서는 그동안 독립적인 클래스로 구성되어 있던 ASE (security target evaluation) 클래스가 각 EAL별로 추가되었으며 EAL1을 위한 ST(이를 '하위보증 ST'라 함)와 EAL2 이상을 위한 ST의 구조가 구분되었다. 또한, AVA_SOF (strength of TOE security function) 패밀리가 삭제되었으며, AVA_VLA (vulnerability analysis) 패밀리의 종속성이 삭제되었다.

4.2.2 ACO (composition) 클래스 추가

ACO 보증클래스는 2개 이상의 "컴포넌트형" TOE

가 안전하게 "합성형" TOE로 통합될 수 있는지에 대한 보증을 평가하기 위한 기준이며, 합성형 TOE의 평가에만 적용한다. 합성형 TOE에 대해, 다음 사항을 평가한다.

- "기반컴포넌트"(base component)[1]가 합성형 TOE의 요구된 보증을 제공하는지를 결정
- 기반 컴포넌트와 "종속컴포넌트"(dependent component)[2]가 호환적(compatible)인지 결정
- 기반컴포넌트와 종속컴포넌트가 하나의 합성형 TOE로 합성될때 삽입된 취약성을 조사

4.2.3 소프트웨어공학과 모듈분할원리의 적용

문제의 복잡성을 다소 해결할 수 있는 방법론인 "모듈분할(modular decomposition) 원리"를 적용하였고 복잡도를 줄이기 위해 모듈화와 계층화(layering)를 강조하고 있다. 특히 모듈내의 결속도(cohesion)와 모듈간의 결합도(coupling)를 분석 및 측정하여 개발의 보증수준을 평가하고 있다. 결속도와 결합도는 다음중 하나로 평가된다(앞에 있는 것이 우수한 것임).

- coupling : call, data, stamp, control, common content
- cohesion : functional, sequential, communicational, temporal, logical(procedural), coincidental

표 2는 원본문서 [19]의 내용을 보이며, 오류가 있음을 알 수 있다. 즉, ADV_INT.4에 'temporal' 이 추가되어있고, 매우 불량한 cohesion인 'common' 이 전체 패밀리에 포함되어있다. 표 3은 본연구팀이 제시한 바람직한 해결책을 보인다. 이 문제는 계속 연구가 되어야 한다.

[표 2] 결속도/결합도와 보증수준 간의 관계(오류가 존재함)

패밀리	ADV_INT.1	ADV_INT.2	ADV_INT.3	ADV_INT.4
EAL	N/A	EAL5	EAL6	EAL7
cohesion	functional, sequential, communicational, temporal		functional, sequential, comm.	functional, sequential, comm., temporal
coupling	call, common			

1) 스마트카드 OS나 윈도우즈 처럼 보안 서비스 제공자의 역할을 하는 TOE이다(dominant peer라함).

2) 서비스 요청자의 역할을 하는 TOE이다(minor peer라 함).

[표 3] 결속도/결합도와 보증수준 간의 관계[3]

패밀리	ADV_INT.1	ADV_INT.2	ADV_INT.3	ADV_INT.4
EAL	EAL 4	EAL5	EAL 6	EAL 7
cohesion	functional, sequential, comm., temporal	functional, sequential, comm.	functional, sequential	functional
coupling	call, data, stamp, control	call, data, stamp	call, data	call

4.2.4 모듈의 분류

정보시스템 내에는 다수의 서브시스템들로 구성되어 있다. 예컨대, DB가 정보보호시스템(즉, TSF: TOE security function)일 경우, OS나 외부의 서버들은 'IT 환경'(즉, Non-TSF)이라 한다. CC 3.0에서는 정보시스템내의 기능을 다음과 같이 분류하고 있다.[3]

- SFR-enforcing 모듈 = Assigned SFR-enforcing 모듈 + non-Assigned SFR-enforcing 모듈
- Non-SFR-enforcing 모듈 = SFR-supporting 모듈 + SFR-non-interfacing 모듈

여기서, non-Assigned SFR-enforcing 모듈은 Assigned-SFR-enforcing 모듈과 대화하는 모듈에 대해서만 '정당화(justification)'할 것을 요구하고 있다.

4.2.5 용어 변경

CC 3.0에서는 이전에 사용하던 용어를 다음과 같이 현실에 맞게 변경했다. 이에 따라서, 클래스나 패밀리 이름도 변경되었다.

- 보안환경 ⇒ 보안문제 정의
- 관리자(administrator) 지침 ⇒ 운영적사용자(operational user) 지침
- 사용자(user) 지침 ⇒ 준비적사용자 (preparative user) 지침
- 상위수준(high-level) 설계 ⇒ 구조(architectural) 설계
- 하위수준(low-level) 설계 ⇒ 평가대상물(TOE) 설계

4.3 세부 변경사항

부록 B에는 CC 2.2와 CC 3.0간의 보증요구사항의 차이를 보인다.

3) CC 3.0의 오류를 수정하여 본 연구팀이 제안한 것

4.3.1 보증 클래스

- 삭제 및 다른 클래스로 통합 : 형상관리는 개발의 전체에 걸친 업무이므로, ACM (configuration management)이 ALC (life-cycle support)로 통합; ADO (delivery and operation)가 ALC와 AGD (guidance document)로 분산통합;
- 신규 : 합성형 TOE의 보증을 위한 ACO (composition) 추가

4.3.2 보증 패밀리

- 완전삭제 : 자동화는 기능의 구현에 관한 사항이므로 ACM_AUT (CM automation) 삭제; ADV클래스 전체에 해당되는 내용이므로 ADV_RCR (representation correspondence) 삭제; 보안강도는 보증과 중복되고 취약성분석에서 평가하므로 AVA_SOF (strength of TOE security function) 삭제;
- 다른 패밀리로 이동 및 통합 : ACM_CAP (CM capabilities)가 ALC_CMC (CM capabilities)로 이동; ADO_DEL (delivery)는 개발자 사이트는 ALC_DEL (delivery)로, 사용자사이트는 AGD_PRE (preparative user guidance)로 각각 분리되는 등 변경내용 많음 (세부내용은 부록B 참조)
- 신규 : ACO 클래스내의 신규 패밀리와 컴포넌트는 다음과 같다.

- ACO_COR (composition rationale) : 합성시에 기반컴포넌트가 적절한 수분의 보증을 제공함을 보임
  - ACO_COR.1 (composition rationale) : 기반컴포넌트에 대한 합성정보 제공 (독립컴포넌트의 보증수준보다 높을것)
- ACO_DEV (development evidence) : 기반컴포넌트의 명세 요구사항
  - ACO_DEV.1 (functional description) : 인터페이스의 목적 식별
  - ACO_DEV.2 (basic evidence of design) : 인터페이스의 목적과 메소드, 파라메터, 연산자, 오류메시지 식별
  - ACO_DEV.3 (detailed evidence of design) : 인터페이스의 목적과 메소드, 파라메터, 연산자, 오류메시지, 기반컴포넌트의 구조, 아키텍쳐 식별
- ACO_REL (reliance of dependent component) : 종속컴포넌트가 기반컴포넌트를 신뢰(reliance)한다는 서술에 대한 근거를 제공함
  - ACO_REL.1 (basic reliance information) : 기반컴포넌트의 기능설명, 인터페이스 목적 및 메소드 서술
  - ACO_REL.2 (reliance information) : 기반컴포넌트의 기능설명, 인터페이스 목적 및 메소드, 파라메터, 예상 연산 및 결과, 일부 인터페이스에 대해 오류처리 서술
  - ACO_REL.3 (detailed reliance information) 기반컴포넌트의 기능설명, 인터페이스 목적 및 메소드, 파라메터, 예상 연산 및 결과, 모든 인터페이스에 대해 오류처리 서술
- ACO_TBT (base TOE testing) : 기반컴포넌트의 시험 실시
  - ACO_TBT.1 (interface testing) : 종속컴포넌트가 의존하고 있는 기반컴포넌트가 시험되었음을 보증함(시험문서, 결과, 시험계획, 시험절차 설명, 예산시험결과와 실제 결과 등 제고)
- ACO_VUL (composition vulnerability analysis) : 합성의 결과로 삽입된 취약성을 분석함
  - ACO_VUL.1 (composition vulnerability review) : 잔여 취약성이 악용되지 않음을 보임; ST내의 가정과 목적이 충족됨을 보임; 취약성에대한 공개소스 조사; 침투시험 실시
  - FCO_VUL.2 (composition vulnerability analysis) : 잔여 취약성이 악용되지 않음을 보임; ST내의 가정과 목적이 충족됨을 보임; 취약성에대한 공개소스 조사; 침투시험 실시; 독립적 취약성분석 실시
  - ACO_VUL.3 (extended-basic Composition vulnerability analysis) : 잔여 취약성이 악용되지 않음을 보임; ST내의 가정과 목적이 충족됨을 보임; 취약성에대한 공개소스 조사; 침투시험 실시; 독립적 취약성분석 실시(extended-basic 공격잠재성).

4.4 보증수준의 변화

4.4.1 EAL의 변화

EAL 기준에 ASE (Security Target evalua-

Assurance class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Configuration management	ACM_AUT				1	1	2	2
	ACM_CAP	1	2	3	4	4	5	5
	ACM_SCP			1	2	3	3	3
Delivery and operation	ADO_DEL		1	1	2	2	2	3
	ADO_IGS	1	1	1	1	1	1	1
Development	ADV_FSP	1	1	1	2	3	3	4
	ADV_HLD		1	2	2	3	4	5
	ADV_IMP				1	2	3	3
	ADV_INT					1	2	3
	ADV_LLD				1	1	2	2
	ADV_RCR	1	1	1	1	2	2	3
	ADV_SPM				1	3	3	3
Guidance documents	AGD_ADM	1	1	1	1	1	1	1
	AGD_USR	1	1	1	1	1	1	1
Life cycle support	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD				1	2	2	3
	ALC_TAT				1	2	3	3
Tests	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	2	2	3
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Vulnerability assessment	AVA_CCA					1	2	2
	AVA_MSU			1	2	2	3	3
	AVA_SOF		1	1	1	1	1	1
	AVA_VLA		1	1	2	3	4	4

Table 2 Evaluation assurance level summary

(a) CC-2.2

Assurance class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Development	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	4
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
Guidance documents	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
Life-cycle support	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD				1	2	2	3
	ALC_TAT				1	2	3	3
Security Target evaluation	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASE_REQ	1	2	2	2	2	2	2
	ASE_SPD			1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
Tests	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	2	2	3
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Vulnerability assessment	AVA_VAN	1	2	2	3	4	5	5

(b) CC-3.0

〔그림 5〕 EAL 정의의 변화

tion) 클래스가 추가되고 ACM, ADO, AVA_MSU (misuse) 및 AVA_SOF (strength of TOE security function) 가 삭제되었다. CC 3.0 에서는 각 컴포넌트의 평가기준도 대폭 변경되었으므로, 단순 비교는 무의미하지만 그림 5는 CC 3.0의 보증수준의 정의의 변화를 보인다.

4.4.2 CAP 보증수준 추가

'컴포넌트형' TOE는 7등급 EAL로 보증수준을 평가하지만, '합성형' TOE는 표 4와 같이 3등급으로 Composed Assurance Packages (CAP)를 평가한다. CAP에 대한 정의는 다음과 같다.

- CAP-A (Structurally composed) : 합성형 TOE가 통합되고 결과물의 올바른 보안 운영에 대한 신임이 필요할 때 적용한다. 기반 컴포넌트의 개발자의 참여를 요구하지 않고, 독립 컴포넌트의 개발자의 협조가 필요하다(예: 설계정보와 시험결과 제공). 완전한 개발기록이 없을때, 개발자나 사용자가 하급부터 중급수준의 "독립적으로 보증된 보안"을 요구하는 환경에서 적용한다.
- CAP-B (Methodically composed) : 합성된 TOE내의 통합된 컴포넌트들 간의 대화의 영향을 이해함으로서, 개발자가 서브시스템 수준에서 최대 보증수준을 얻고자할 때 적용한다. 이때, 기반 컴포넌트의 개발자의 참여를 최소화한다. 개발자나 사용자가 중간수준의 "독립적으로 보증된 보안"을 요구하고, 합성된 TOE와 그 개발 환경을, 대폭적인 리엔지니어링 없이, 철저히 조사할 것을 요구하는 환경에서 적용한다.
- CAP-C (Methodically composed, tested and reviewed) : 합성된 TOE내 컴포넌트간의 대화를 엄격하게 분석하여 개발자가 최대보증을 얻고자할 때 적용한다. 기반 컴포넌트의 모든 평가근거에 접근할 것을 요구하지는 않는다. 전통적인 상품(conventional commodity)형태의 합성된 TOE에서, 개발자나 사용자가 중간부터 높은 수준의 "독립적으로 보증된 보안"을 요구하는 환경에서 적용하며, 개발자나 사용자는 부가적인 보안관련 엔지니어링 비용을 부담할 준비가 되어 있다.

V. CEM과 PP/ST의 변화

5.1 CEM의 변화

5.1.1 CEM 2.4

CEM 2.2에서 Part 1과 Part 2 및 독립적인 "결함 교정" 부록으로 분류되어 있는 문서를 CEM 2.4에서는 하나로 통합하였다. ASE (ST evaluation) 클래스가 EAL에 관여함에 따라, ST 평가의 부활동이 평가보증등급에 관계하여 포함하였다. 또한, CEM 2.2에서는 PP와 ST, EAL1~EAL4의 부활동이 명시

〔표 4〕 CAP의 수준 정의

보증클래스	보증 패밀리	합성보증수준에 의한 보증컴포넌트		
		CAP-A	CAP-B	CAP-C
Composition	ACO_COR	1	1	1
	ACO_DEV	1	2	3
	ACO_REL	1	2	3
	ACO_TBT	1	1	1
	ACO_VUL	1	2	3
Life-cycle support	ALC_CMC	1	1	1
	ALC_CMS	2	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
Security Target evaluation	ASE_CCL	1	1	1
	ASE_ECD	1	1	1
	ASE_INT	1	1	1
	ASE_OBJ	1	2	2
	ASE_REQ	1	2	2
	ASE_SPD		1	1
	ASE_TSS	1	1	1

되어 있으나, CEM 2.4에서는 PP와 EAL1, EAL4 부분의 부활동만이 존재하며 EAL2와 EAL3 등급에 관한 사항은 특별한 평가활동으로 분류해놓고 있다. 또한, 평가기술보고서(ETR)와 관련하여 평가기술보고서에는 입력작업과 출력작업에 관련된 식별자 및 관련된 사항을 보고서 자체나 평가프로젝트 정보에 포함하도록 명시하고 있다. 그리고, 평가참여자의 역할 중 "감독자(overseer)"가 "평가 권위자(evaluation authority)"로 명칭이 바뀌며 증명/검증보고서를 발행하는 업무가 부가되었다.

5.1.2 CEM 3.0

CEM 2.2와 2.4에서는 민간평가기관이 행할 수 있는 평가수준인 EAL4이하에 대한 평가지침만을 제공하고 있으며 내용상의 중복이 심하다.[22~25] 표 5와 같이 CEM 3.0에서는 이 문제를 해결하여 모든 보증클래스별로 평가지침을 제시함으로써, 모든 보증수준에 대한 평가지침을 제공하고 있다.[26]

5.2 PP와 ST의 변화

5.2.1 PP와 ST의 구조 변경

CC 2.4와 CC 3.0에서는 표 6, 표 7과 같이 대폭 변경되었으며 주요 변경사항은 다음과 같다.

〔표 5〕 CEM의 구조변경

CEM 2.1, 2.2	CEM 2.4	CEM 3.0
1. 서론 2. 일반 평가 업무 3. PP 평가 4. ST 평가 5. EAL1 평가 6. EAL2 평가 7. EAL3 평가 8. EAL4 평가 9. 결함치유 부활동 A. 용어 B. 일반 평가 지침	1. 서론 2. 평가 프로세스와 관련 업무 3. PP 평가 4. EAL1 평가 5. EAL4 평가 6. 결함치유 부활동 A. 용어 B. 일반 평가 지침	1. 범위 2. 선언적 참조 3. 용어와 정의 4. 심볼과 약자 5. 개요 6. 문서지침 7. 평가프로스와 관련 업무 8. ADV: 개발 9. AGD: 지침문서 10. ALC: 생명주기 지원 11. APE: PP 평가 12. ASE: ST 평가 13. ATE: 시험 14. AVA: 취약성 평가 부록 A. (비공식적)일반평가 지침 부록 B. (비공식적) 취약성 평가(AVA)

- '준수요구(conformance claim)' 장이 추가되었다.
- '보안환경' 용어가 '보안문제 정의'로 변경되었다.
- 보안목적을 2가지(TOE, 환경)에서 3가지(TOE, 개발환경, 운영환경)로 세분화 하였다.
- CC에서 정의된 보안기능 이외 것을 사용할 경우를 위해 '확장된 컴포넌트 정의' 장이 추가되었다.
- 'PP응용노트'와 '근거'가 삭제되었다.
- EAL1 수준 경우, PP와 ST에서 '보안문제 정의'와 '보안목적'을 생략한다.

5.2.2 PP/ST의 보증 평가요구사항 변경

APE_DES (TOE description)이 APE_INT (PP introduction)와 APE_CCL (conformance claims)에 통합되었고, APE_ENV (security environment)가 APE_SPD (security problem definition)으로 명칭이 변했다. APE_INT (PP introduction), APE_OBJ (security objectives) 및 APE_REQ (IT security requirement)은 내용만 5~10%정도 변했고, APE_SRE (explicitly stated IT security requirements)는 APE_REQ (security requirement) 변경되었다. PP 평가시에는 ST와 TOE 평가와 달리 평가요약보고서(ESR)를 작성하지 않으며, EAL1수준의 PP에 대해서 PP 소개, 준거 요구, 확장된 컴포넌트 정의, 보안 요구사항으로 구성되며 보안문제 정의와 보안목적은 생략한다. ASE (ST evaluation) 클래스도 APE와 유사하게 변경되었다. 세부사항은 부록 B를 참고한다.

Ⅵ. 분석 및 결론

6.1 CC 3.0의 변화내용 분석

지난 10년간 CC를 이용하여 PP/ST를 작성하거나 TOE를 평가할 때 발생했던 문제점(예: 보안기능 컴포넌트의 중복성, 애매성, 평가 범위문제 등)을 파악하여 다음과 같이 해결하였다.

- 간단하고 일반적인 모델의 이용: 주체-연산-객체 모델과 전입-전출모델을 적용하여 보안기능을 효과적으로 분류하였다.
- 문제가 되었던 보안 및 보증요구사항 조정 : 해당 컴포넌트, 패밀리, 클래스를 변경, 삭제 또는 추가하였다. 보안기능은 50%가 변경되었고 보증은 30%가 변경되었으며, CEM도 70%정도가 변경되었다.
- CC 범위의 명확화 : 암호기능 및 보안관리 기능을 제외하였다. 이로서, 기존의 CMVP나 ISMS와의 중복성이 줄어들었다.
- 평가된 기반컴포넌트(예: 스마트카드 OS)위에 종속컴포넌트(예: 스마트카드를 이용한 인증제품)를 통합하여 개발한 '합성형' TOE를 평가하기 위해 ACO (Composition) 보증클래스가 추가되고 3등급으로된 컴포넌트보증패키지(즉, CAP-A, CAP-B 및 CAP-C)가 추가되었다. 기반컴포넌트 또는 컴포넌트형 TOE는 7등급의 EAL로 평가하지만, 합성형 TOE는 3등급으로 평가한다.
- PP/ST의 변화: 현실에 맞게 EAL 1수준의 PP/ST는 매우 간략히 작성할 수 있게 하였고, 보증평가에 ST 평가를 포함하였다.
- CEM 3.0의 변화 : 모든 EAL에 대한 평가지침을 지침 제공하고 있다.
- 소프트웨어공학 원리의 적용 : 복잡성을 극복하는 설계원리인 '모듈화'와 '계층화' 설계원리를 적용하고 있다. 특히, 모듈간의 '결합도'와 모듈내의 '결속도'를 측정하여 설계의 구조성을 평가한다.

6.2 CC 3.0의 변화에 따른 문제점

- CC 2.2를 이용해 작성 및 평가된 PP나 ST는 다시 작성 및 평가되어야한다. 특히, CC 3.0은 대폭 변경되었으므로 CC 2.2의 기능이나 보증컴포넌트들은 CC 3.0의 그것과 1:1 대응이 어렵다. 따라서, 기존의 PP나 ST 들을 CC 3.0으로 변경할 때 어려움이 예상된다.
- 기존 CC 2.2기반 PP로 평가 및 인증된 TOE를 CC 3.0에 따라 재평가해야 한다.

〔표 6〕 PP의 변화

<table>
<tr><th colspan="3" rowspan="2">CC 2.1, 2.2의 PP</th><th colspan="4">CC 2.4 및 CC 3.0의 PP</th></tr>
<tr><th colspan="2">일반 PP</th><th colspan="2">하위 보증 PP (EAL1용)</th></tr>
<tr><td>1. PP 서론</td><td colspan="2">• PP 식별(id.)
• PP 개요(overview)</td><td>1. PP 서론</td><td>• PP 참조(ref.)
• TOE 개요(overview)</td><td>1. 일반 PP와 동일</td><td>일반 PP와 동일</td></tr>
<tr><td>2. TOE 설명</td><td colspan="2">-</td><td>2. 준수요구</td><td>• CC 준수요구
• PP 클레임
• 패키지 클레임</td><td>2. 일반 PP와 동일</td><td>일반 PP와 동일</td></tr>
<tr><td>3. TOE 보안환경</td><td colspan="2">• 가정
• 위협
• 조직의 보안정책</td><td>3. 보안 문제 정의</td><td>• 가정
• 위협
• 조직의 보안정책</td><td>없음</td><td>없음</td></tr>
<tr><td rowspan="2">4. 보안목적</td><td colspan="2" rowspan="2">• TOE에 대한 보안목적
• 환경에 대한 보안목적</td><td>4. 보안목적</td><td>• TOE에 대한 보안목적
• 개발환경에 대한 보안목적
• 운영환경에 대한 보안목적
• 보안목적의 근거</td><td>없음</td><td>없음</td></tr>
<tr><td>5. 확장된 컴포넌트 정의</td><td>• 확장된 컴포넌트의 정의</td><td>3. 일반 PP와 동일</td><td>일반 PP와 동일</td></tr>
<tr><td rowspan="2">5. IT 보안요구사항</td><td>• TOE 보안 요구사항</td><td>• TOE 보안기능 요구사항
• TOE 보안보증 요구사항</td><td>6. 보안요구사항</td><td>• 보안기능 요구사항
• 보안보증 요구사항
• 보안요구사항의 근거</td><td>4. 보안 요구사항</td><td>• 보안기능 요구사항
• 보안보증 요구사항</td></tr>
<tr><td colspan="2">• IT환경에 대한 보안요구사항</td><td colspan="3"></td></tr>
<tr><td>6. PP응용노트</td><td colspan="2">-</td><td>없음</td><td>없음</td><td>없음</td><td>없음</td></tr>
<tr><td>7. 근거</td><td colspan="2">• 보안목적의 근거
• 보안요구사항의 근거</td><td>없음</td><td>없음</td><td>없음</td><td>없음</td></tr>
</table>

〔표 7〕 ST의 변화

CC 2.1, 2.2의 ST		CC 2.4, 3.0의 ST			
		일반 ST		하위보증 ST (EAL1용)	
1. ST 서론	• ST 식별(id.) • ST 개요(overview) • CC 준수(conformance)	1. ST 서론	• ST 참조(ref.) • TOE 참조(ref.) • TOE 개요(overview) • TOE 설명(descr.)	1. 일반 ST와 동일	일반ST와 동일
2. TOE 설명	-	2. 준수요구	• CC 준수클레임 • PP 클레임 • 패키지 클레임	2. 일반 ST와 동일	일반 ST와 동일
3. TOE 보안환경	• 가정 • 위협 • 조직의 보안정책	3. 보안문제 정의	• 가정 • 위협 • 조직의 보안정책	없음	없음
4. 보안목적	• TOE에 대한 보안목적 • 환경에 대한 보안목적	4. 보안목적	• TOE에 대한 보안목적 • 개발환경에 대한 보안목적 • 운영환경에 대한 보안목적 • 보안목적의 근거	없음	없음
		5. 확장된 컴포넌트 정의	• 확장된 컴포넌트의 정의	3. 일반 ST와 동일	일반 ST와 동일
5. IT 보안 요구사항	• TOE 보안요구사항: • TOE 보안기능요구사항, • TOE 보안보증요구사항 • IT환경에 대한 보안요구사항	6. 보안요구사항	• 보안기능 요구사항 • 보안보증 요구사항 • 보안요구사항의 근거	4. 보안요구사항	• 보안기능 요구사항 • 보안보증 요구사항
6. TOE 요약명세	• TOE 보안기능 • 보증대책	7. TOE 요약명세	• TOE 요약명세	5. 일반 ST와 동일	일반 ST와 동일
7. PP클레임	• PP 참조 • PP 수정(tailoring) • PP 추가	없음	없음	없음	없음
8. 근거	• 보안목적의 근거 • 보안요구사항의 근거 • TOE 요약명세의 근거 • PP 클레임의 근거	N/A	N/A	N/A	N/A

- CC 3.0은 아직 초안이며 오류나 미완성 부분이 많다(예: ADV_INT, FMI (miscellaneous) 등). 따라서, 공식버전이 되기까지 충분히 개선되야 한다.
- 현재의 정보보호제품은 시스템 인프라(OS, DB 등)위에서 작동되므로 보안기능과 비보안기능간의 구분이 모호하다. CC 3.0에서도 이 문제를 좀 더 해결해야한다.

6.3 대책 및 향후 연구과제

문제를 좀 더 자세히 알수록, 그 문제의 해결 및 진화속도는 빨라진다. 예컨대, 웹을 잘 알수록 웹의 변화는 빨라진다. 마찬가지로, CC에 대해 경험이 쌓임에 따라, CC는 매우 빠른 속도로 진화(변화)되고 있다. 우리나라도 CC의 변화에 대비한 평가인증체계를 구축할 필요가 있다. 지금까지는 끌려 왔지만, 이제부터는 CC의 변화를 끌고 갈 수 있도록 해야 할 것이다.

CC 3.0은 초안중의 초안일 뿐이며 보안평가의 문제점에 대한 하나의 해결방안을 제시한 것일 뿐이며, 그 해결방안이 정당한지를 계속하여 검증해야한다. 또한, CC 2.2와의 대응 지침을 개발하여 기존의 PP나 ST를 CC 3.0에 맞게 변경할 때 사용해야한다.

참 고 문 헌

〔1〕 KISA 보안성평가센터내 수록자료, http://www.kisa.or.kr/

〔2〕 Common Criteria for Information Technology Security Evaluation (CC), Part 1: Introduction and general model, CCEB-96/011, Version 1.0, Jan. 1996.

〔3〕 Common Criteria for Information Technology Security Evaluation (CC), Part 2: Security functional requirements, CCEB-96/012, **Version 1.0**, Jan. 1996.

〔4〕 Common Criteria for Information Tech-

부록 A. CC2.2와 CC3.0간의 보안기능요구사항 비교(패밀리)

CC 2.2 및 CC 2.4 보안기능요구사항	CC 3.0 보안기능요구사항
FAU (Security Audit) FAU_ARP (security audit automatic response) ➡ FAU_ARP FAU_GEN (security audit data generation) ➡ FAU_GEN FAU_SAA (security audit analysis) ➡ FAU_SAA FAU_SAR (security audit review) ➡ FDP_ACC FAU_SEL (security audit event selection) ➡ FDP_ACC FAU_STG (security audit event storage) ➡ FDP_ACC	**FAU (Security Audit)** FAU_ARP (security audit automatic response) ⬅ =FAU_ARP FAU_GEN (security audit data generation) ⬅ *FAU_GEN FAU_SAA (security audit analysis) ⬅ =FAU_SAA
FCO (Communication) FCO_NRO (non-repudiation of origin) ➡ [FCO_NRE, FCO_NRI] FCO_NRR (non-repudiation of receipt) ➡ [FCO_NRE, FCO_NRI]	**FCO (Communication)** FCO_NRE (non-repudiation of exported data) ⬅ [FCO_NRR, FCO_NRO] FCO_NRI (non-repudiation of imported data) ⬅ [FCO_NRR, FCO_NRO] FCO_CID (confidentiality of imported data) ⬅ *[FDP_ITC, FDP_UCT] FCO_CED (confidentiality of exported data) ⬅ *[FDP_UCT, FDP_ETC, FPT_ITC] FCO_IID (integrity of imported data) ⬅ *[FDP_UIT, FDP_ITC, FPT_RPL] FCO_IED (integrity of exported data) ⬅ *[FDP_UIT, FDP_ETC, FPT_ITI, FPT_RPL] FCO_AED (availability of exported data) ⬅ *FPT_ITA FCO_ITC (import from outside TSF control) ⬅ *FDP_ITC FCO_ETC (export to outside TSF control) ⬅ *FDP_ETC FCO_TID (translation of imported data) ⬅ 신규 FCO_TED (translation of exported data) ⬅ 신규 FCO_UNE (unobservability of export) ⬅ [FPR_UNO, FDP_ETC]
FCS (Cryptographic Support) ➡ 삭제 FCS_CKM (cryptographic key management) FCS_COP (cryptographic operation)	없음
FDP (User Data Protection) FDP_ACC (access control policy) ➡ FDP_ACC FDP_ACF (access control functions) ➡ FDP_ACC FDP_DAU (data authentication) ➡ [삭제; 자료의소유권은 보안속성에 해당] FDP_IFC (information flow control policy) ➡ FDP_ACC FDP_IFF (information flow control functions) ➡ FDP_ACC FDP_ETC (export to outside TSF control) ➡ [FCO_CED, FCO_IED, FCO_ETC, FCO_UNE] FDP_ITC (import from outside TSF control) ➡ [FCO_CID, FCO_IID, FCO_ITC] FDP_ITT (internal TOE transfer)➡ [삭제, 분산 TOE에만 해당] FDP_RIP (residual information protection) ➡ FPT_RIP FDP_ROL (rollback) ➡ FDP_ROL FDP_SDI (stored data integrity) ➡ FDP_ACC FDP_UCT (inter-TSF user data confidentiality transfer protection) ➡ [FCO_CID, FCO_CED] FDP_UIT (inter-TSF user data integrity transfer protection) ➡ [FCO_IID, FCO_IED]	**FDP (Data Protection and Privacy)** FDP_ACC (access control) ⬅ [FDP_ACC, FDP_ACF, FDP_IFC, FDP_IFF, FDP_SDI, FAU_SAR, FAU_SEL, FAU_STG, FMT_MOF, FMT_REV, FMT_MTD, FMT_SMR] FDP_ROL (rollback) ⬅ -FDP_ROL FDP_ISA (initialisation of security attributes) ⬅ [FMT_MSA, FIA_ATD, FTA_LSA] FDP_MSA (management of security attributes) ⬅ [FMT_MSA, FMT_SAE, FMY_REV] FDP_OSD (object/subject destruction) ⬅ [신규; FIA_USB의 반대개념] FDP_UNL (unlinkability) ⬅ +FPR_UNL FDP_UNO (unobservablity) ⬅ -FPR_UNO
FIA (Identification and Authentication) FIA_AFL (authentication failures) ➡ FIA_AFL FIA_UID (user identification)➡ FIA_UID FIA_ATD (user attribute definition) ➡ [FDP_ISA; user개념 단순화] FIA_UAU (user authentication)➡ FIA_UAU FIA_USB (user-subject binding) ➡ [FIA_USB; 반대개념(destruction)으로 사용 및 변형] FIA_SOS (specification of secrets) ➡ FIA_QAD	**FIA (Identification, Authentication and Binding)** FIA_AFL (authentication failures) ⬅ = FIA_AFL FIA_UID (user identification) ⬅ [FIA_UID, FPR_ANO; FPR_PSE] FIA_URE (user registration) ⬅ [FPR_ANO, FPR_PSE; user개념을 단순화함] FIA_UAU (user authentication) ⬅ *FIA_UAU FIA_SUA (subject authentication) ⬅ [신규; 사용자가 웹서버를 인증] FIA_QAD (quality of authentication data) ⬅ = FIA_SOS FIA_USB (user-subject binding) ⬅ *[FIA_USB, FPR_ANO, FPR_PSE] FIA_TBR (TSF binding rules) ⬅ [FIA_USB, FTA_MCS] FIA_LOB (lock-out of bindings) ⬅ *FTA_SSL FIA_TOB (termination of bindings) ⬅ + FTA_SSL.3 FIA_TIN (TSF Information) ⬅ [FTA_TAB.1, FTA_TAH.1]

CC 2.2 및 CC 2.4 보안기능요구사항	CC 3.0 보안기능요구사항
FIA (Identification and Authentication) FIA_AFL (authentication failures) ➡ FIA_AFL FIA_UID (user identification)➡ FIA_UID FIA_ATD (user attribute definition) ➡ [FDP_ISA; user개념 단순화] FIA_UAU (user authentication)➡ FIA_UAU FIA_USB (user-subject binding) ➡ [FIA_USB; 반대개념(destruction)으로 사용 및 변형] FIA_SOS (specification of secrets) ➡ FIA_QAD	FIA (Identification, Authentication and Binding) FIA_AFL (authentication failures) ⬅ = FIA_AFL FIA_UID (user identification) ⬅ [FIA_UID, FPR_ANO; FPR_PSE] FIA_URE (user registration) ⬅ [FPR_ANO, FPR_PSE; user개념을 단순화함] FIA_UAU (user authentication) ⬅ *FIA_UAU FIA_SUA (subject authentication) ⬅ [신규; 사용자가 웹서버를 인증] FIA_QAD (quality of authentication data) ⬅ = FIA_SOS FIA_USB (user-subject binding) ⬅ *[FIA_USB, FPR_ANO, FPR_PSE] FIA_TBR (TSF binding rules) ⬅ [FIA_USB, FTA_MCS] FIA_LOB (lock-out of bindings) ⬅ *FTA_SSL FIA_TOB (termination of bindings) ⬅ + FTA_SSL.3 FIA_TIN (TSF Information) ⬅ [FTA_TAB.1, FTA_TAH.1]
FMT (Security Management) ➡ 삭제 FMT_MOF (management of functions in TSF) ➡ FDP_ACC FMT_MSA (management of security attributes)➡ FDP_MSA FMT_MTD (management of TSF data) ➡ FDP_ACC FMY_REV (revocation) ➡ [FDP_ACC, FDP_MSA] FMT_SAE (security attribute expiration) ➡ FDP_MSA FMT_SMF (specification of management functions)➡ 삭제 FMT_SMR (security management roles) ➡ FDP_ACC	FDP로 통합
FPR (Privacy) ➡ 삭제 FPR_ANO (anonymity) ➡ [FIA_URE, FIA_UID, FIA_USB] FPR_PSE (pseudonymity) ➡ [FIA_URE, FIA_UID, FIA_USB] FPR_UNL (unlinkability) ➡ FDP_UNL FPR_UNO (unobservability) ➡ [FDP_UNO, FCO_UNE]	FIA와 FDP로 통합
FPT (Protection of the TSF) FPT_FLS (fail secure) ➡ FPT_FLS FPT_RCV (trusted recovery)➡ FPT_RCV FPT_PHP (TSF physical protection) ➡ FPT_PHP FPT_TST (TSF self test) ➡ FPT_TST FPT_RPL (replay detection) ➡ [FCO_IED, FCO_IID] FPT_RVM (reference mediation) ➡ [삭제; ADV_ARC에서 커버] FPT_SEP (domain separation) ➡ 삭제 FPT_SSP (state synchrony protocol) ➡ [삭제; 함축된 요구사항이므로] FPT_STM (time stamps) ➡ FMI_TIM FPT_AMT (underlying abstract machine test) ➡ FPT_TOU FPT_ITA (availability of exported TSF data)➡ FCO_AED FPT_ITC (confidentiality of exported TSF data)➡ FCO_CED FPT_ITI (integrity of exported TSF data) ➡ FCO_IED FPT_ITT (internal TOE TSF data transfer) ➡ [삭제; 함축된 요구사항이므로] FPT_TDC (inter-TSF TSF data consistency) ➡ [삭제; 함축된 요구사항이므로] FPT_TRC (internal TOE TSF data replication consistency) ➡ [삭제; 함축된 요구사항이므로]	FPT (Protection of the TSF) FPT_FLS (fail secure) ⬅ + FPT_FLS FPT_RCV (trusted recovery) ⬅ FPT_RCV FPT_PHP (TSF physical protection) ⬅ FPT_PHP FPT_TST (TSF self test) ⬅ + FPT_TST FPT_TOU (testing of users) ⬅ [FPT_AMT; 추상기계 시험과 동일] FPT_RIP (residual information protection) ⬅ *FDP_RIP FPT_FLT (fault tolerance) ⬅ - FRU_FLT FPT_PRI (priority) ⬅ FRU_PRS FPT_RSA (resource allocation) ⬅ = FRU_RSA
FRU (Resource Utilisation) ➡ 삭제 FRU_FLT (fault tolerance) ➡ FPT_FLT FRU_PRS (priority of service) ➡ FPT_PRI FRU_RSA (resource allocation) ➡ FPT_RSA	FPT로 통합
FTA (TOE Access) ➡ 삭제 FTA_LSA (limitation on scope of selectable attributes) ➡ FDP_ISA FTA_MCS (limitation on multiple concurrent sessions) ➡ FIA_TBR FTA_SSL (session locking) ➡ 삭제 FTA_TAB (TOE access banners) ➡ [FIA_TIN, FIA_TBR] FTA_TAH (TOE access history) ➡ [FIA_TIN, FIA_TBR] FTA_TSE (TOE session establishment) ➡ 삭제	FDP와 FIA로 통합
FTP (Trusted Path/Channels) ➡ 삭제 FTP_ITC (inter-TSF trusted channel) FTP_TRP (trusted path)	FCO에서 커버
	FMI (Miscellaneous) ⬅ 신규 FMI_RND (random number generation) ⬅ 신규 FMI_TIM (time stamps) ⬅ = FPT_STM FMI_CHO (choice) ⬅ 신규

부록 B. CC-2.2와 CC-3.0의 보증요구사항 비교(패밀리)

CC 2.2 보증요구사항	CC 3.0 보증요구사항
APE (Protection Profile Evaluation) APE_DES (TOE description) ➡ [APE_INT, APE_CCL] APE_ENV (security environment) ➡ [APE_SPD; 'security problem definition'으로 명칭변경] APE_INT (PP introduction) ➡ APE_INT APE_OBJ (security objectives) ➡ APE_OBJ, APE_ECD APE_REQ (IT security requirements) ➡ APE_REQ APE_SRE (explicitly stated IT security requirements) ➡ APE_REQ	APE (Protection Profile Evaluation) (CC-2.4과 동일) APE_CCL (conformance claims)⬅ APE_DES APE_ECD (extended components definition) ⬅ APE_OBJ APE_INT (PP introduction) ⬅ [APE_INT, APE_DES] APE_OBJ (security objectives) ⬅ APE_OBJ APE_REQ (security requirements) ⬅ APE_REQ APE_SPD (security problem definition) ⬅ APE_ENV
ASE (Security Target Evaluation) ASE_DES (TOE description) ➡ [ASE_INT, ASE_CCL] ASE_ENV (security environment) ➡ [ASE_SPD; '보안문제정의'로 명칭 변경] ASE_INT (ST introduction) ➡ ASE_INT ASE_OBJ (security objectives)➡ [ASE_OBJ, ASE_ECD] ASE_PPC (PP claims) ➡ [ASE_CCL] ASE_REQ (IT security requirements)➡ ASE_REQ] ASE_SRE (explicitly stated IT security requirements)➡ ASE_REQ ASE_TSS (TOE summary specification)➡ ASE_TSS	ASE (Security Target Evaluation) (CC-2.4와 동일) ASE_CCL (conformance claims) ⬅ ASE_DES ASE_ECD (extended components definition) ⬅ ASE_OBJ ASE_INT (ST introduction) ⬅ [ASE_INT, ASE_DES] ASE_OBJ (security objectives) ⬅ ASE_OBJ ASE_REQ (security requirements) ⬅ ASE_REQ ASE_SPD (security problem definition) ⬅ ASE_ENV ASE_TSS (TOE summary specification) ⬅ ASE_TSS
ACM (Configuration Management) ➡ 삭제 ACM_AUT (CM automation) ➡ 삭제 ACM_CAP (CM capabilities)➡ ALC_CMC ACM_SCP (CM scope)➡ [ALC_CMS; 5개 컴포넌트로 세분화]	ALC로 통합
ADO (Delivery and Operation) ➡ 삭제 ADO_DEL (delivery)➡ [개발자사이트는 ALC_DEL; 사용자사이트는 AGD_PRE] ADO_IGS (installation, generation and start-up) ➡ [AGD_PRE; 개발자와 사용자 공통]	ALC, AGD로 통합
ADV (Development) <TSF의 분할에 관한 패밀리> ADV_FSP (functional specification)➡ ADV_FSP ADV_HLD (high-level design) ➡ ADV_TDS; 'high-level' 용어를 'architecture'로 명칭변경; 2개 컴포넌트로 단순화] ADV_IMP (implementation representation) ➡ ADV_IMP ADV_LLD (low-level design) ➡ ADV_TDS ADV_RCR (representation correspondence) ➡ [삭제; ADV전체로 분산]	ADV (Development) <TSF의 분할에 관한 패밀리> ADV_FSP (functional specification) ⬅ ADV_FSP ADV_IMP (implementation representation) ⬅ [ADV_IMP, ADV_RCR] ADV_TDS (TOE design) ⬅ [ADV_HLD, ADV_LLD; 설계는 semiformal 까지, 표현은 formal까지; 모듈의 알고리즘 명세 추가(.3이상)]
<TSF의 이해성과 견고성에 관한 패밀리> ADV_INT (TSF internals) ➡ ADV_INT ADV_SPM (security policy modeling) ➡ ADV_SPM	<TSF의 이해성과 견고성에 관한 패밀리> ADV_INT (TSF internals) ⬅ [ADV_INT; modular- decomposition 추가; coupling, cohesion 분석 실시] ADV_SPM (security policy modeling) ⬅ [ADV_SPM; 정형적 보안정책 모델만 사용함] ADV_ARC (architectural design) ⬅ [신규; 다른 근거 내에서 제공된 세부내용의 차원에서 아키텍쳐적 견고성을 서술함]
AGD (Guidance Documents) AGD_ADM (administrator guidance) ➡ [AGD_OPE; 'administrator'는 'preparative user'로 명칭변경; 2개 컴포넌트로 단순화; acceptance와 installation 만 강조함(준비 개념)] AGD_USR (user guidance) ➡ [AGD_PRE; 'user' 는'operative user'로 명칭 변경; "각 역할에 대해 ---"로 구체화 함]	AGD (Guidance Documents) AGD_OPE (operational user guidance) ⬅ [AGD_ADM, AVA_MSU; TOE 운영지침; 인간 상호작용을 목적으로함; 오용분석의 대상임] AGD_PRE (preparative user guidance) ⬅ [ADO_DEL, ADO_IGS, AVA_MSU; 사용자사이트에서 실시; 수신 및 시동절차 포함; 오용분석 대상임]
ALC (Life Cycle Support) ALC_DVS (development security) ➡ ALC_DVS ALC_FLR (flaw remediation) ➡ ALC_FLR ALC_LCD (life cycle definition) ➡ ALC_LCD ALC_TAT (tools and techniques) ➡ ALC_TAT	ALC (Life-Cycle Support) ALC_DVS (development security) ⬅ = ALC_DVS ALC_FLR (flaw remediation) ⬅ = ALC_FLR ALC_LCD (life-cycle definition) ⬅ = ALC_LCD ALC_TAT (tools and techniques) ⬅ = ALC_TAT ALC_CMC (CM capabilities) ⬅ *ACM_CAP ALC_CMS (CM scope) ⬅ *ACM_SCP 개발자측 시동절차 ALC_DEL (delivery) ⬅ [ADO_DEL; 개발자사이트에서]

CC 2.2 보증요구사항	CC 3.0 보증요구사항
ATE (Tests) ATE_COV (coverage) ➡ ATE_COV *ATE_DPT (depth)* ➡ *[ATE_DPT; 문장 간략화]* ATE_FUN (functional tests)➡ ATE_FUN ATE_IND (independent testing)➡ ATE_IND	ATE (Tests) ⬅ 95% 동일 ATE_COV (coverage) ⬅ = ATE_COV ATE_DPT (*depth*) ⬅ = ATE_DPT ATE_FUN (functional tests) ⬅ = ATE_FUN ATE_IND (independent testing) ⬅ = ATE_IND
AVA (Vulnerability Assessment) AVA_VLA (vulnerability analysis) ➡ AVA_VAN AVA_CCA (covert channel analysis) ➡ AVA_VAN AVA_MSU (misuse) ➡ [AGD_OPE, AGD_PRE] AVA_SOF (strength of TOE security functions) ➡ [삭제; AVA_VAN에 개념 포함]	AVA (Vulnerability Assessment) AVA_VAN (vulnerability analysis) ⬅ [AVA_VLA, AVA_CCA, AVA_SOF; 코버트채널 분석 포함; 취약성관련 공개정보 조사; 보안강도 요구사항 삭제됨]
없음	*ACO (Composition)* ⬅ 신규 ACO_COR (composition rationale) ACO_DEV (development evidence) ACO_REL (reliance of dependent component) ACO_TBT (base TOE testing) ACO_VUL (*composition vulnerability analysis*)

nology Security Evaluation (CC), Part 2: Annexs, CCEB-96/012_A, **Version 1.0**, Jan. 1996.

[5] Common Criteria for Information Technology Security Evaluation (CC), Part 3: Security assurance requirements, CCEB-96/013, **Version 1.0**, Jan. 1996.

[6] Common Criteria for Information Technology Security Evaluation (CC), Part 4: Predefined Protection Profiles, CCEB-96/014, **Version 1.0**, 96/01/31.

[7] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, **Version 2.1**, CCIMB-99-031, August 1999.

[8] Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements, **Version 2.1**, CCIMB-99-032, Aug. 1999.

[9] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, **Version 2.1**, CCIMB-99-033, August 1999.

[10] 국제공통평가기준, CC 2.0, 한국정보보호센터, 1998.11.

[11] 정보보호시스템 공통평가기준, 정보통신부 한국정보보호진흥원, 2002.8.

[12] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, **Version 2.2**, Revision 256, CCIMB-2004-01-001, Jan. 2004.

[13] Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements, **Version 2.2**, CCIMB-2004-01-002, Jan. 2004.

[14] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, **Version 2.2**, Revision 256, CCIMB-2004-01-003, Jan. 2004.

[15] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, CCIMB-*2004-03-001*, **Version 2.4**, Revision 256, Mar. 2004.

[16] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, CCIMB-2004-03-003, **Version 2.4**, Revision 256, Mar. 2004.

[17] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, CCIMB-2005-07-001, **Version 3.0**, Revision 2, June 2005.

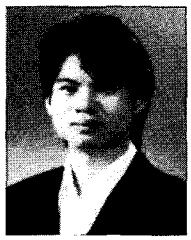
[18] Common Criteria for Information Technology Security Evaluation, Part 2: Se-

curity functional requirements, **Version 3.0**, Revision 2, CCIMB-2005-07-002, July 2005.
[19] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements, **Version 3.0**, Revision 2, CCIMB-2005-07-003, July 2005.
[20] 강연희, 김정대, 방영환, 최성자, 이강수, "공통평가기준(CC)과 공통평가방법론(CEM)의 변경 내용 분석", 한국정보보호학회지, 제 13권 제 4호, 2004. 8.
[21] 최상수, 방영환, 최성자, 이강수, "보안관리 및 위험분석을 위한 분류체계, 평가기준 및 평가스케일의 조사연구", 한국정보보호학회지, 13권 제 3호, pp. 38-49, 2003.6.
[22] Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology, **Version 1.0**, CEM-99/045, Aug. 1999.
[23] Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology Supplement: ALC_FLR - Flaw Remediation, **Ver 1.1**, CEM-2001/0015R, Feb. 2002.
[24] Common Methodology for Information Technology Security Evaluation Methodology, **Version 2.2**, Revision 256, CCIMB-2004-01-004, Jan. 2004.
[25] Common Methodology for Information Technology Security Evaluation Methodology, **Version 2.4**, Revision 256, CCIMB-2004-03-004, Mar. 2004.
[26] Common Criteria for Information Technology Security Evaluation - Evaluation Methodology, **Version 3.0**, Revision 2, CCIMB-2005-07-004, July 2005.
[27] Common Criteria Version 3.0 Update, http://www.commoncriteriaportal.org/public/files/CCv3.0%20transition.pdf, June 2005.

〈著 者 紹 介〉

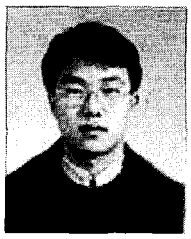
강 연 희 (Yeon-Hee Kang)
학생회원

2003년 : 한남대학교 컴퓨터멀티미디어공학과 졸업(학사)
2005년 : 한남대학교 대학원 컴퓨터공학과 졸업
〈관심분야〉 소프트웨어공학, 정보보호시스템 평가, 보안공학



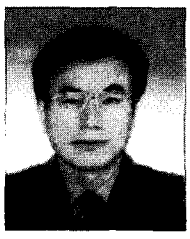
김 정 대 (Jung-Dae Kim)
학생회원

2003년 : 한남대학교 컴퓨터공학과 졸업(학사)
2004년~현재 : 한남대학교 컴퓨터공학과 석사과정
〈관심분야〉 소프트웨어 품질 평가 및 보증, 소프트웨어 표준화, 보안공학



최 상 수 (Sang-Soo Choi)
학생회원

2001년 2월 : 한남대학교 컴퓨터공학과 졸업 (학사)
2003년 2월 : 한남대학교 대학원 컴퓨터공학과 졸업 (석사)
2003년 3월~현재 : 한남대학교 대학원 컴퓨터공학과 박사과정
〈관심분야〉 소프트웨어공학, 웹공학, 보안공학, 정보보호 컨설팅 및 위험분석



이 강 수 (Gang-Soo Lee)
종신회원

1981년 : 홍익대학교 전자계산학과 학사
1983년 : 서울대학교 대학원 전산학과 석사
1989년 : 서울대학교 대학원 전산학과 박사

1985년~1987년 : 국립한밭대학교 전자계산학과 전임강사
1992년~1993년 : 미국일리노이대학교 객원교수
1995년 : 한국전자통신연구원 초빙연구원
1998년~1999년 : 한남대학교 멀티미디어학부장
1987년~현재 : 한남대학교 컴퓨터공학과 정교수
〈관심분야〉 소프트웨어공학, 병행시스템 모델링 및 분석, 보안공학, 정보보호시스템 평가, 멀티미디어교육 커리큘럼