암호화된 정보의 복구를 위한 키복구 시스템 개발

(Development of a Key Recovery System for Recovery of Encrypted Data)

강 상 승 † 임 신 영 †† 고 정 호 ††† 전 은 아 †††† 이 강 수 †††††

(Sang-seung Kang) (Shin-young Lim) (Jeong-ho Ko) (Eun-ah Jun) (Gang-soo Lee)

요 약 CALS 및 전자상거래 시스템과 같은 정보보호 기능이 내장된 각종 정보시스템에서 정보의 안전한 저장 및 통신을 위해서는 정보의 암호화가 필수적이다. 또한, 손실된 키의 복구를 포함하여 수많은 암호키들의 안전한 관리가 중요하다. 본 논문에서는 암호화된 정보의 복구를 위한 암호키 복구 시스템의 설계 및 구현 결과를 논한다. 제안한 시스템은 키 캡슐화 방식이며 키복구 시스템의 국제표준에 해당하는 NIST의 RKRP(Requirements for Key Recovery Products)와 국제 공통 평가 기준인 CC 2.0의 스킴을 따르고 있다. 송신자(암호정보 생산자)는 비밀리에 키복구 에이전트 풀로부터 2개 이상의 에이전트를 선정하며, 키복구 정보는 복구할 키, 난수키 및 선택한 에이전트의 공개키를 이용하여 생성된다. 수신자(키복구 요청자)는 키복구 서비스를 위해 참여하는 키복구 에이전트들을 알 수 없다. 제안한 시스템은 C/Unix버전과 Java/NT버전으로 각각 개발하였다. 제안한 시스템은 암호화된 통신메시지 뿐만 아니라 저장메시지의 복구에 적용할 수 있으며, 전자상거래 서비스 기술 하부구조를 위한 새로운 보안서비스 솔루션으로서 사용될 수 있다.

Abstract Information systems, which support information security functions such as CALS and EC systems, should have cryptographic functions for information in order to storage and communicate securely. Additionally, including recovery of lost keys, lots of cryptographic keys should be securely managed. In this paper, we present some results of development of a key recovery system for recovery of encrypted data. The proposed system, in a type of key encapsulation approach, confirms to NIST's RKRP(Requirements for Key Recovery Products) that is a defecto international key recovery standard, as well as CC 2.0 that is a international security evaluation criteria. A message sender secretly choices two or more key recovery agents from a pool of key recovery agents. The key recovery information is generated by using the recovering key, random keys and public keys of the chosen agents. A message receiver can not know which key recovery agents are involved in his key recovery service. We have developed two versions of prototype of key recovery system such as C/Unix and Java/NT versions. Our systems can be used for recovery of communicating informations as well as storing informations, and as a new security service solution for electronic commerce service infrastructures.

† 정 회 원: 한국전자통신연구원 전자상거래연구부 전자지불연구팀 연구원
kss@etri.re.kr
†† 비 회 원: 한국전자통신연구원 전자상거래연구부 전자지불연구팀 연구원
limsy@etri.re.kr
††† 학생회원: 한남대학교 컴퓨터공학과
jhko@se.hannam.ac.kr
†††† 비 회 원: (주)퓨처시스템 정보통신연구소 보안팀 연구원
penguin@se.hannam.ac.kr
††††† 종신회원: 한남대학교 컴퓨터공학과 교수
gslee@eve.hannam.ac.kr
논문접수: 2001년 1월 22일
심사완료: 2001년 5월 17일

1. 서 론

정보통신을 기반으로 하는 정보시스템에서 정보보호 문제는 어떤 대수 시스템에서의 공리(axiom)에 해당한다. 따라서, 각종 정보시스템의 하부구조에는 각종 정보보호 제품 또는 시스템이 활용되고 있으며, 정보보호 제품이나 시스템들은 다양한 암호 알고리즘, 이들을 이용한 암호 프로토콜 및 보안 서비스들로 구성된다.

따라서, 정보시스템 내에는 많은 수의 암호키들이 존재하며 안전하고 효율적으로 관리(생성, 저장, 배포, 파기

등)되어야 한다. 암호키 관리시스템에는 분실한 암호키의 복구기능도 포함할 수 있다. 예컨대, 암호화된 정보를 수신했거나 암호화된 정보를 디스크로부터 읽었을 때 이를 복호화하기 위해서는 암호키가 필요하지만, 암호키가 파괴 또는 분실되었거나 알 수 없을 때는 암호화된 정보를 복호화 할 수 없게 된다. 따라서, 암호키의 안전한 생성 및 저장 못지 않게 복구가 필요하다.

키복구 시스템에 관한 연구 및 개발은 1993년 미국의 클리퍼 정책의 발표로부터 본격화되었으며, 일반적으로 암호화된 정보의 복구시스템인 키복구 방식들은 다음과 같이 분류할 수 있다[1,2]:

• 키 위탁(escrow) (예; 클리퍼 칩 또는 Escrowed encryption standard[3,4,5]): 사용자(암호화된 정보의 생산자)는 1개 이상의 키 위탁 기관에 자신의 암호키를 위탁하며, 키복구가 필요하면 위탁한 키를 위탁기관에 요청한다.

• Trusted Third Party(TTP) (예; Yaksha system [6,7], ANSI X9.17): 사용자는 TTP로부터 사용할 암호키(세션키)를 얻는다. 즉, TTP는 사용자에게 암호키를 제공하며, 키복구시 TTP는 사용자에게 제공했던 키를 제공한다.

• 상업적 키 백업 (예; AT&T Crypto-Backup [8]): 사용자는 자료를 대칭 암호화한 후 그 세션키를 자신의 공개키로 암호화하여 암호화된 자료와 함께 유지한다. 사용자의 비밀키는 키복구(백업) 기관에 저장된다. 키복구가 필요하면 키복구 기관은 저장된 사용자의 비밀키를 제공한다.

• 키 캡슐화(encapsulation) (예; TIS RecoveryKey [9], CyKey[10], SecretAgent[11], IBM CSKR 및 SecureWay[12], Binding Cryptography): 자료를 대칭 암호화한 세션키를 이용하여 키복구 정보를 생성한다. 키복구 정보는 키복구 에이전트만이 복호화할 수 있으며 암호화된 자료내에 캡슐화한다. 키복구가 필요하면, 키복구 정보를 키복구 에이전트에 보내어 세션키를 복구한다.

키 위탁 기관, TTP에서의 키분배 센터 및 키백업 기관은 전자상거래 시스템과 같은 트랜잭션이 빈번하고 저장할 정보들이 많은 정보시스템에서는 병목이 될 수 있다. 또한, 대부분의 사용자들은 자신의 비밀키 또는 세션키를 어떤 기관에도 위탁하거나 저장하지 않으려 한다. 이는 자신의 금고 열쇠를 남에게 위탁하는 것과 같기 때문이다. 특히, 키 위탁 방식은 개인의 프라이버시 문제와 바인딩 문제[5,8], 즉, 정부기관 모르게 사용자끼리 비밀통신 할 수 있는 문제에 취약하다.

키 캡슐화 방식은 일종의 분산 시스템을 이용한 키복구 방법에 해당한다. 즉, 각 사용자들은 키복구 정보를 생성 및 저장하며 키복구가 필요할 때에만 키복구 정보와 함께 키복구 센터에 요청한다. 따라서, 시스템의 병목이 줄어들며, 각 사용자가 키복구의 주도권을 가질 수 있으므로, 프라이버시가 보장될 수 있다. 그러나, 기존의 키 캡슐화 방법들은[9,10,11] 다음과 같은 단점이 있다.

• n개 키복구 에이전트를 활용하는 n-way 키복구 시스템에서 세션키 처리(즉, n개로 분할 및 병합 등)방법이 알려져 있지 않다.

• 송신자(키복구 정보 생성자)가 지정한 키복구 에이전트의 신원을 수신자(키복구 요청자)가 알게 되므로, 키복구의 투명성이 저하된다.

• 키복구 에이전트들이 정해져 있으므로, 에이전트들 간의 결탁 공격이 용이하다.

• 수신자는 다수의 키복구 에이전트들과 직접 통신해야 하며[9,11], 수신자마다 키복구 기능들이 설치되어야 한다.

• 정보보호시스템의 일종인 키복구 시스템에 대한 CC(Common Criteria) 2.0[13] 기반 평가에 관한 연구가 시도되고 있지 않다.

이러한 배경과 문제점들을 해결하기 위해 본 연구에서는 키 캡슐화 방식을 개선하였고, 키복구 시스템의 사실상의 국제 표준인 NIST의 RKRP(Requirements for Key Recovery Products)[14]를 수용하는 키복구 프로토콜을 개발하고 이를 이용한 2개 버전의 키복구 시스템을 개발 완료하였다. 또한, 제안 시스템은 정보보호시스템의 국제 공통 평가기준(Common Criteria 2.0, ISO/IEC 15408)[13]의 지침을 충실하게 따랐으며, 평가에 필요한 각종 산출물을 쉽게 확보할 수 있도록 소프트웨어 공학적으로 개발하였고 문서화에 역점을 두었다. 이들 자료는 향후의 보안평가 및 인증시의 노력을 줄이고 평가의 성공률을 높일 수 있을 것이다.

본 논문의 2장에서는 키복구 시스템의 요구사항 분석 결과를 논하며, 3장에서는 키복구 시스템의 프로토콜을 포함한 설계 과정과 결과를 기술한다. 4장에서는 두가지 버전에 대한 구현결과를 보이며, 5장에서는 개발한 키복구 시스템의 성능과 보안평가 결과를 기술한다. 6장에서는 개발한 시스템과 유사한 키 캡슐화 방식의 시스템들을 분석하고 개발한 시스템의 응용 방안들을 제시한다. 마지막으로 7장에서는 결론을 맺는다.

2. 키복구 시스템의 요구사항 분석

2.1 키복구 문제의 정의

공개키 기반구조 상에서 운영되는 정보시스템의 한

사용자 또는 송신자(즉, 암호화된 정보의 생산자)는 어떤 정보를 "세션키"(이를 "목표키"라 한다)로써 대칭 암호화하고, 그 세션키는 PKI를 통해 획득한 수신자(즉, 정보의 소비자)의 공개키로써 비대칭 암호화하여 수신자로 보낸다.

수신자는 우선 자신의 개인키로써 세션키를 복호화하고, 이를 이용하여 송신자가 보낸 정보를 복호화한다. 만일, (1) 수신자가 자신의 개인키를 분실함에 따라 세션키를 획득할 수 없을 때, (2) 송신자가 유효기간이 만료된 수신자의 공개키로 암호화했거나, (3) 수신자의 소재를 알 수 없거나, (4) 관할기관(정부 또는 기업)이 강제적인 법 집행을 해야할 때, 세션키를 복구함으로써 암호화된 정보를 복호할 수 있어야 한다.

위와 같이 암호통신 정보뿐만 아니라, 디스크에 저장된 암호화된 정보를 복구할 경우에도 위와 유사한 키복구 문제가 요구된다. 이 경우, 송신자는 디스크에 파일을 암호화하고 저장하는 "출력 프로그램"에 해당하며, 수신자는 디스크로부터 자료를 읽어 복호화하는 "입력 프로그램"에 해당한다.

2.2 보안관련 요구사항

키복구 서비스는 다양한 네트워크와 플랫폼상에서 제공되어야 하므로 호환성이 중요하다. 이를 위해 키복구 시스템 벤더들은 키복구 연맹(KRA)을 결성하여 시스템의 요구사항들을 공동 개발하고 있으며[1,15], 미국 NIST의 RKRP[14]는 키복구 시스템의 사실상의 국제 표준이라 할 수 있다. 본 연구에서는 RKRP의 요구사항과 CC 2.0의 스킴을 적용하여 키복구 시스템의 보안관련 요구사항을 도출하였다.

특히, 키복구 시스템도 정보보호시스템의 유형이므로, 본 연구에서는 정보보호시스템의 평가와 인증 맥락[16]에서 키복구 시스템의 요구사항들을 분석하였다. 이러한 본 결과들은 향후 CC 운영기관에 공식적으로 등록되어야 하는 키복구 시스템의 보호 프로파일(Protection Profile; PP)의 일부분에 해당한다. PP는 키복구 시스템 제품군이 가져야 할 공통 요구사항이 명세되고 CC 운영기관에서 평가 및 인증된 문서를 지칭한다.

(1) 보안위협, 보안목적 및 보안기능 요구사항

키복구 시스템의 보안목적 및 보안기능이란 시스템내의 "키복구 관련 정보"를 각종 "위협(threat)"으로부터 보호하기 위한 것이다. "보안목적(objective)"이란 각종 위협을 고려하여 키복구 시스템이 추구하는 보안관련 목적들을 의미한다. 하나의 보안목적은 1가지 이상의 다수의 위협을 대처하기 위한 것이다.

"보안기능 요구사항"은 보안목적을 달성하기 위한 정보보호관련 대책(countermeasure) 또는 기능이라 할 수 있다. CC 2.0[13]에는 모든 정보보호시스템들이 가질 수 있는 전체 기능 요구사항들이 포함되어 있으며 키복구 시스템의 기능 요구사항은 전체 요구사항들 중 특히 키복구와 관련된 것만을 선택하여 도출하였다.

표 1과 표 2는 본 논문에서 제안하는 키복구 시스템의 보안위협, 보안목적 및 보안기능 요구사항들을 나타낸다. 특히, 보안기능 요구사항들은 CC 2.0의 보안기능 요구사항들 중 키복구 시스템에서 필요로 하는 기능들을 선택한 것이다.

표 1 키복구 시스템에 대한 보안위협과 보안목적

보안위협	보안목적
T1 : 권한 가진 사용자에 의한 키복구 정보의 남용(abuse) T2 : 키복구 정보의 불법 접근 T3 : 키복구 정보를 권한 없는 사용자에게 전송 T4 : 키복구 정보의 송·수신 부인봉쇄(non-repudiation) T5 : 보안 위반 T6 : 키복구 정보의 무결성(integrity) 손상 T7 : 전송중인 키복구 정보의 가로채기 T8 : 합법적인 키복구 요청자에 대한 사칭(impersonation)공격 T9 : 키복구 정보의 수정이나 파괴 T10 : 권한 없이 키복구 정보를 표절(plagiaris)	O1 : 보안감사 O2 : 인가된 접근 O3 : 송·수신 부인봉쇄 O4 : 무결성 체크 O5 : 데이타의 암호화(encryption) O6 : 식별 및 인증

표 2 키복구 시스템의 보안기능 요구사항

보안기능 요구사항 (요구사항의 식별명은 CC 2.0의 내용을 따름)	
FAU_ARP.1 : 보안감사 자동응답 : 보안경보	FDP_DAU.1~2 : 자료인증 : 기본, 보증자의 식별과 함께 자료인증
FAU_GEN.1 : 보안감사 자료생성 : 감사자료 생성	FTP_TRP.1 : 신임된 경로
FAU_GEN.2 : 보안감사 자료생성 : 사용자 식별자 생성	FIA_AFL.1 : 인증 실패 : 기본적 인증 실패 처리
FDP_ACC.1 : 접근통제 : 부분적 객체 접근통제	FIA_UAU.1~5 : 사용자 인증: 기본적, 단일 사용자인증 메커니즘, 인증무결성, 다중 인증 메커니즘, 정책기반 인증 메커니즘
FDP_ACF.1 : 접근통제 : 단일 보안속성 접근통제	FIA_UID.1~2 : 사용자 식별 : 기본, 사용자의 유일한 식별
FCO_NRO.1 : 근원지 부인봉쇄 : 강요된 근원지증명	FIA_USB.1 : 사용자-주체 바인딩
FDP_UCT.1 : 보안강화 기능간 사용자 자료기밀성 전송보호 : 기본적 자료교환	
FDP_UIT.1 : 보안강화 기능간 사용자 자료전송 보호 : 자료교환 무결성	

(2) 부가적인 요구사항 및 가정

① 바인딩 문제: 키복구 시스템은 암호통신 당사자의 의지 또는 관할기관의 보안 정책하에서 수행되어야 한다. 특히, 키 캡슐화 방식은 키복구의 주도권을 사용자

가 가질 수 있으므로, 바인딩 문제[5,8]에 대한 부담이 줄어든다. 따라서, LEAF(law enforcement access field)는 캡슐화된 키복구 정보 내에서 사용하지 않는다. 바인딩 문제는 특히, 키 위탁 방식에서 중요한 문제가 된다.

② 복구의 대상: 원시자료를 비대칭 암호화한 세션키에 국한하며 원시자료 전체를 복구하지는 않는다. 따라서, 원시자료 전체보다는 세션키의 크기가 적으므로(블럭당 128비트) 키 캡슐화 방식을 사용한다.

③ 키복구 시스템은 정형적으로 명세 및 검증되어야 하고 구현결과에 대한 성능평가가 이루어져야 하며 시스템내의 각 기능들은 모듈화 되어야 한다. 이를 통해 시스템의 유지보수성이 증대되어야 한다.

④ 키복구 시스템 개발시 새로운 암호 알고리즘은 개발하지 않으며, 기존의 안전성이 검증되고 추천된 암호 알고리즘(RSA, DES, SHA-1, MD5, KCDSA 등)들을 구현해놓은 암호 라이브러리를 이용한다.

⑤ 키복구 시스템은 기존의 공개키 기반구조 상에서 수행된다. 따라서, 키복구 시스템 내의 모든 파트(다수의 사용자, 1개의 키복구 센터, 다수의 키복구 에이전트)들의 인증서와 공개키는 PKI의 인증기관에 의해 안전하게 분배되었다고 가정한다.

3. 키복구 시스템의 설계

3.1 시스템 설계

RKRP에서 요구하는 키복구 시스템의 기본적인 기능 그룹과 세부기능들은 다음과 같다.

- 키복구 정보생성 기능 그룹 : 키복구 정보생성 기능
- 키복구 정보관리 기능 그룹 : 키복구 정보검증 기능, 키복구 정보전달 기능
- 키복구 기능 그룹 : 키복구 정보요청 기능, 키복구 에이전트 기능

제안 시스템은 RKRP에서 요구하는 모든 기능들을 포함하고 있으며, 그림 1처럼 사용자 서브시스템, 키복구 센터 서브시스템 및 키복구 에이전트 서브시스템으로 구성되어 있으며 각 서브시스템들은 기능들을 공유한다. 각 서브시스템은 키복구 서비스에 참여하는 파트들이다.

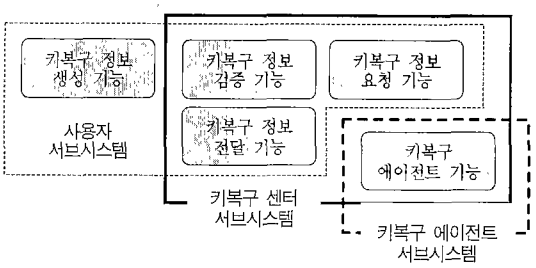
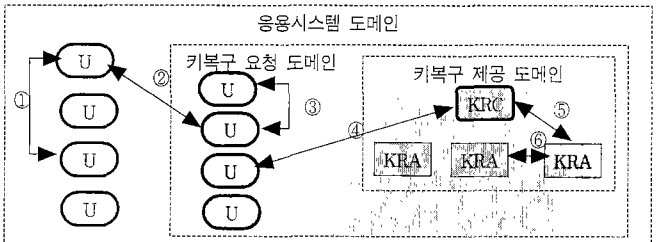


그림 1 제안 시스템의 서브시스템과 기능

키복구 시스템은 독자적인 시스템이 아니라, 기존의 응용시스템 내에 키복구 서비스를 제공하는 솔루션이며 도메인 구조는 그림 2와 같다. '응용시스템 도메인'(또는 스페이스)은 CALS, EC, EDI 시스템 등을 나타내며 인터넷/인트라넷 기반구조와 공개키 기반구조 위에 존재하며 일반 사용자들은 이 도메인 상에 존재한다. '키복구 도메인'은 응용 도메인 내부에 존재하며 키복구 서비스를 제공받을 수 있는 권한을 가진 사용자들은 이 도메인 상에 존재한다. 키복구 도메인 내에는 키복구 서비스를 제공하는 '키복구 제공 도메인'이 존재하며 1개의 키복구 센터와 2개 이상의 키복구 에이전트들로 구성된다. 키복구 제공 도메인과 같은 내포된 도메인은 상위 도메인의 기능(예; 인증서 발급)들을 상속받는다.

3.2 프레임워크와 파트-보안관계 구조

제안 시스템의 프레임워크는 그림 3과 같다. X.509, PKI, RKRP 및 CC 2.0상에서 운영되며, 응용 및 암호 API를 사용하고 사용자를 위한 서비스 제공 인터페이스



(U: 사용자, KRC: 키복구 센터, KRA: 키복구 에이전트)

그림 2 제안 시스템의 도메인 구조와 파트들간의 통신

가 존재한다[17].

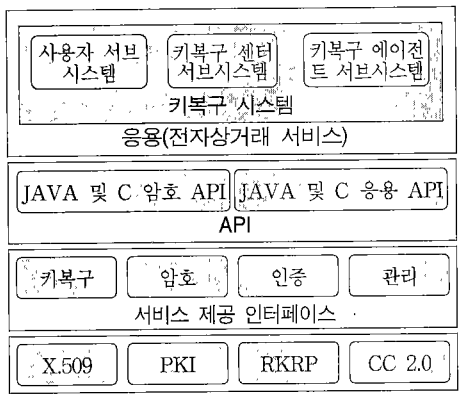


그림 3 제안 시스템의 프레임워크

"파트-보안관계"(part-security relation) 개발방법은 본 논문에서 새롭게 이용한 정보보호시스템의 분석 및 설계방법이며 제안 시스템의 개발시에 적용하였다. 기존의 데이타베이스 시스템 설계를 위한 객체-관계 접근방법론(ERA; entity-relation approach)과 Gomma의 실시간 시스템 설계방법론인 DART(design approach for real-time system)[18]들의 개념을 응용한 것이다. 정보보호 시스템에서 파트들간의 보안관계(security relation)는 다음과 같이 정의한다.

① 기밀성 관계(secrecy relation): 자료에 대한 비대칭 암호를 수행하므로 자료의 기밀성을 유지해야함

• 관계의 구현 예: 송신파트에서 M = DES(자료, 세션키) 송신, 수신파트에서 DES(M, 세션키) 수행

② 무결성 관계(integrity relation): 디지털 서명 및 검증을 통해 자료가 변경되지 않도록 해야함

• 관계의 구현 예: 송신파트에서 M = RSA(Hash(송신 원시자료), 수신파트의 공개키) 송신, 수신파트에서 Hash(RSA(M, 수신파트의 개인키)) = Hash(수신 원시자료) 여부를 체크

③ 상호 인증성 관계(mutual authentication relation): 파트들간에 상호 식별이 가능해야함

• 관계의 구현 예: 송신파트에서 M = RSA(RSA(자료, 송신파트의 개인키), 수신파트의 공개키)를 송신, 수신파트에서 RSA(RSA(M, 수신파트의 개인키), 송신파트의 공개키) 수행

④ 일방향 인증성 관계(one way authentication relation): 파트들간에 메시지의 수신자 또는 송신자가 식별되어야함(예컨대, 수신자의 공개키를 가진 파트들은 누구나 메시지를 암호화하여 보낼 수는 있지만 개인키를 가진 수신자만이 이를 복호화 할 수 있을 때)

• 관계의 구현 예: 송신파트에서 M=RSA(자료, 수신파트의 공개키)를 송신, 수신파트에서 RSA(M, 수신파트의 개인키) 수행

제안 시스템의 파트들간에 요구되는 보안관계들은 다이어그램으로 나타낼 수 있으며, 그림 4는 제안 시스템의 파트-보안관계 구조를 보인다. 파트-보안관계 구조도와 보안관계 정의 내의 관계 구현 예들을 이용하면 키복구 시스템의 프로토콜을 쉽게 설계 및 구현할 수 있다.

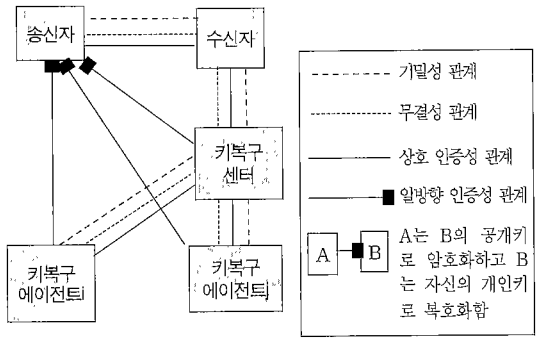


그림 4 키복구 시스템의 파트-보안관계 구조도

3.3 키복구 프로토콜

제안 시스템에서 사용하는 키복구 프로토콜은 그림 5처럼 UML(Unified Modeling Language)중 sequence diagram으로 표현하였다.

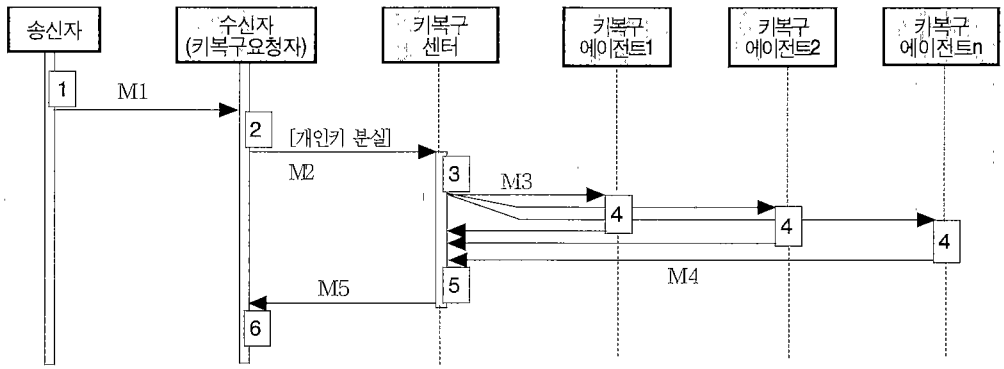
프로토콜에서 fork()함수는 세션키(ssk)와 난수키(rk)들을 이용하여 세션키를 키복구 에이전트의 개수만큼 분할하여 중간키(ik)들을 생성하는 것이며, join() 함수는 부분키들을 결합하는 것이다. 부분키는 키복구 에이전트당 1개씩 생성되며, 난수키는 선택한 키복구 에이전트 수보다 1개 적게 생성된다.

• fork(ssk, n) = ik_1, ... ik_i, ... ik_n, (여기서, $ik_1 = ssk \oplus rk_1$, $ik_2 = rk_1 \oplus rk_2$, ..., $ik_i = rk_{i-1} \oplus rk_i$, ..., $ik_n = rk_{n-1}$)

• join(ik_1, ... ik_i, ... ik_n) = $ik_1 \oplus ik_2 \oplus$ ..., $\oplus ik_i \oplus$ ... $\oplus$ ik_n

(여기서, ssk: 세션키, rk: 난수키, ik: 중간키, $\oplus$: 비트단위 exclusive-or 연산자).

그림 6은 선택한 키복구 에이전트가 2개일 경우와 3개일 경우의 중간키 생성과정을 예로 보인다.

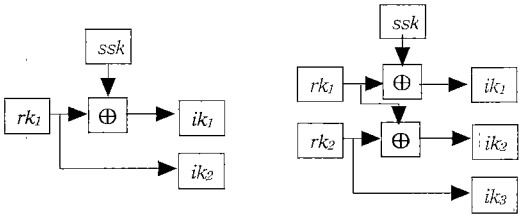


번호	처리 과정
1	*ssk* 생성; A=SCe(*data*, *ssk*); B=PCe(*ssk*, puk_B) ① 키복구 서비스 비선택시: M1 = A+B 송신 ② 키복구 서비스 선택시: KRA들 지정; 지정된 KRA들의 인증서(공개키) 획득; C = [KRI_1=PCe(fork(*ssk*, *1*), puk_{KRA1})] +[KRI_2=PCe(fork(*ssk*, *2*), puk_{KRA2})] + ... + [KRI_n=PCe(fork(*ssk*, *n*), puk_{KRAn})] + 지정된 KRA식별자; KRI= PCe(C, puk_{KRC}); M1(=A+B+KRI) 송신
2	① 키복구 서비스 비선택시 또는 prk_B 보존시: data = SCd(A, PCd(B, prk_B)) ② 키복구 서비스 선택시 및 (prk_B 분실 또는 법집행시): 새로운 키쌍(prk_B' 및 puk_B')생성; puk_B'를 CA에 등록; 키복구 요청(M2(=KRI)) 송신
3	KRI = PCe(M2, prk_{KRC}); KRI_1, KRI_2, ..., KRI_n, 지정된 KRA들의 식별자 분리; 지정된 KRA들에게 각각 KRI_1, KRI_2, ..., KRI_n 송신; M3(=KRA_i) 송신
4	RES_i = PCd(KRI_i, prk_{KRAi}); M4(=RES_i) 송신
5	*ssk* = join(RES_i); M5(=*ssk*) 송신
6	*data* = SCd(A, *ssk*)

<주> KRI에는 타임스템프, 파트들의 식별자 등과 같은 부가적 정보도 포함되어 있음. 각 파트간의 인증기능 및 보안감사를 위한 저장기능 등은 나타내지 않음

<약어 정의> *n*: 선택한 키복구 에이전트 수, +: 메시지의 결합, *ssk*: 복구할 세션키, puk_x: x의 공개키, prk_x: x의 개인키, KRI: 키복구 정보, KRA: 키복구 에이전트, SCe(x,y): 자료 x를 키 y로 대칭 암호화(예; DES), SCd(x,y): 자료 x를 키 y로 대칭 복호화(예; DES), PCe(x,y): 자료 x를 키 y로 비대칭 암호화(예; RSA), PCd(x,y): 자료 x를 키 y로 비대칭 복호화(예; RSA)

그림 5 제안한 키복구 시스템의 프로토콜 명세



(a) 2개일 경우 (b) 3개일 경우

그림 6 선택한 키복구 에이전트 수에 따른 중간키 생성 과정

송신자는 각 ik_i들을 해당하는 키복구 에이전트들의 공개키로 암호화하며, 각 키복구 에이전트들은 자신의 개인키로 복호화하여 키복구 센터로 보낸다. 키복구 센터에서는 join연산을 통해 세션키를 복구하게 된다. 한편, n개의 키복구 에이전트를 가질 때, $(\boldsymbol{ssk} \oplus rk_1) \oplus (rk_1 \oplus rk_2) \oplus ..., (rk_{i-1} \oplus rk_i), ... \oplus (rk_{n-2} \oplus rk_{n-1}) \oplus rk_{n-1} = \boldsymbol{ssk}$ 이므로 *ssk*는 보존된다.

송신자는 다수의 키복구 에이전트들을 임의로 선정하며 키복구 정보 내에 삽입하여 수신자가 키복구 에이전트를 알 수 없도록(또는, 알 필요가 없도록) 하고 있다. 수신자는 오직 1개의 키복구 센터의 공개키만을 알면 된다. 이를 통해 키복구 요청자에게 키복구의 투명성을 제공하고 파트들간의 공모공격 가능성을 줄일 수 있다.

3.4 모듈 구조

그림 7~9는 제안한 시스템의 모듈구조들을 보인다. 모듈구조에는 자료흐름, 제어흐름, 기능의 분류(보안/비보안, 송신자/수신자) 내용을 통합하여 나타내었다.

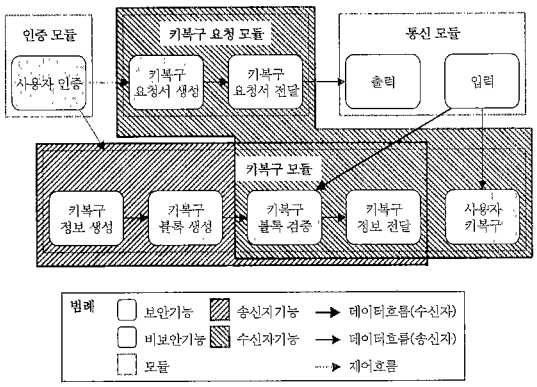


그림 7 사용자 서브시스템의 모듈구조

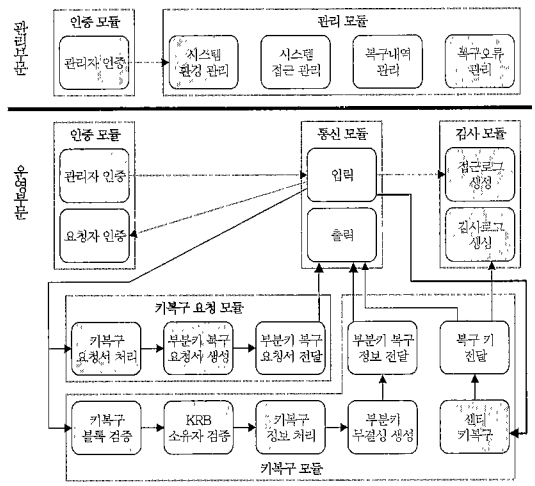


그림 8 키복구 센터 서브시스템의 모듈구조

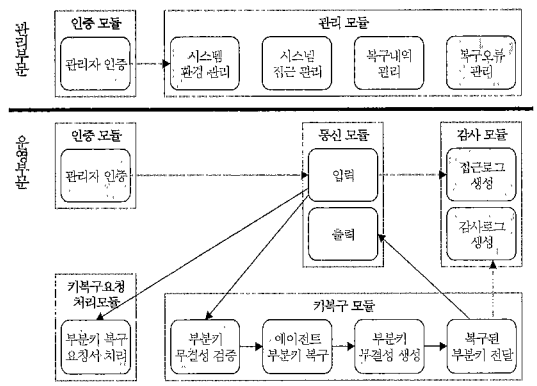


그림 9 키복구 에이전트 서브시스템의 모듈구조

4. 구 현

제안한 시스템은 C/Unix 버전과 Java/NT 버전으로 각각 구현하였다.

4.1 C/Unix 버전

(1) 구현 환경 : 제안한 시스템의 개발과 운용을 위한 하드웨어는 Sun Ultra Sparc Workstation을 사용하였으며, Solaris 2.7을 운영체제로 사용하였다. 구현언어는 C++이며, 암호 라이브러리로는 Ksafe(DES, RSA, KCDSA, SHA-1)를 이용하였고, 감사용 자료의 저장은 Unix 파일 시스템으로 처리하였다[17]. 또한, 키복구의 관리는 웹을 이용하였으며 이를 위한 관리 웹서버를 개발하였고 관리자는 웹을 통해 관리가 가능하다. CGI로 웹으로의 사용자 접근을 제어하였으며, 클라이언트와 서버간의 통신은 TCP/IP를 이용한 Unix Socket을 이용하여 구현하였다.

(2) 시스템 구성 : 개발한 시스템은 키복구 시스템의 기존 서브시스템(사용자, 키복구 센터 및 키복구 에이전트)과 관리를 위한 서브시스템(키복구 센터 관리, 키복구 에이전트 관리)으로 구성하였다. 키복구 에이전트는 2개로 두어(2개 이상 가능) 키복구 기관의 권한을 분산시켰다. 다음은 키복구 시스템을 구축한 관리 홈페이지의 예이다.

그림 10 C/Unix 버전의 키복구 센터 관리 홈페이지

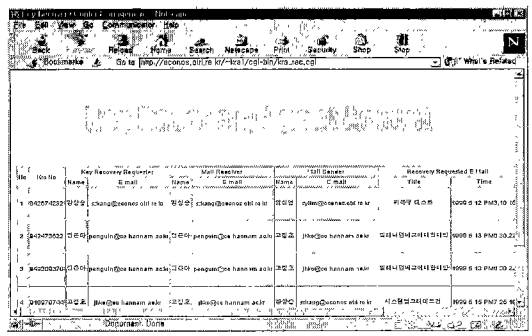


그림 11 C/Unix 버전의 키복구 에이전트 관리 홈페이지

4.2 Java/NT 버전

(1) 구현환경 : 시스템의 개발과 운용을 위한 하드웨어는 PC Server(서버)와 PC(클라이언트)를 사용하였으며, Windows NT 4.0과 Windows 98을 운영체제로 사용하였다. 구현언어는 자바언어로 JDK 1.2.2를 사용하였으며, 암호 라이브러리로는 ETRI Java Crypto Library를 이용하였고, DBMS는 Oracle 8.0.3을 이용하였으며, DB 접근을 위해 JDBC를 사용하여 Oracle 이외의 다른 RDB와 호환이 가능하도록 하였다. 또한, 클라이언트와 서버간의 통신을 위해 Java의 Socket/ServerSocket 클래스를 이용한 클라이언트/서버 Socket 프로그래밍을 구현하였다. 제안한 시스템은 Java환경과 ODBC를 이용하여 개발하였으므로, 시스템의 이식성이 향상되어 키복구 시스템들의 공통 요구사항인 호환성 문제를 다소 해결하였다.

(2) 시스템 구성 : 개발된 시스템은 기본 서브시스템(사용자, 키복구 센터, 키복구 에이전트)과 관리시스템(키복구 센터 및 키복구 에이전트 관리)으로 구성된다. 키복구 에이전트는 최대 10개까지 사용할 수 있다.

그림 12는 각 서브시스템에 접근하기 위해 사용하는 로그인 다이얼로그와 운용시스템, 관리시스템의 인터페이스를 보인다. 표 3은 두 가지 버전의 소스코드 크기를 보인다.

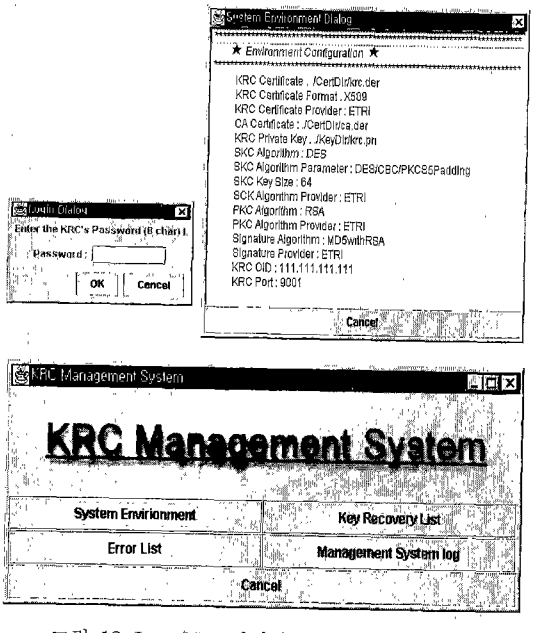


그림 12 Java/NT 버전의 관리용 인터페이스

표 3 개발한 키복구 시스템의 크기(KByte)

서브시스템	부문	C/Unix 버전(*)	Java/NT버전(**)
사용자 키복구 서브시스템	설치 부문	-	16
	운영 부문	382	40
	지원 부문	19	44
키복구 센터 서브시스템	설치 부문	-	16
	운영 부문	382	49
	관리 부문	18	38
	지원 부문	66	44
키복구 에이전트 서브시스템	설치 부문	-	15
	운영 부문	379	38
	관리 부문	66	38
	지원 부문	15	44
계		1327	382

(*) include된 파일 포함, (**) 순수개발 부분

5. 평 가

5.1 성능평가

시스템의 성능이란 키복구 서비스에 소요되는 전체시간을 의미한다. 제안한 시스템은 n개의 키복구 에이전트로 구성되었으며 모든 키복구 에이전트는 동일한 플랫폼으로 구성되었다고 가정한다. 또한 순수 키복구 시간만을 고려한다(즉, 암호화된 자료의 복호 시간을 고려하지 않음). $LOT(i)^{(n)}$를 i 파트의 내부 계산 시간이라 하고, $CCT(X,Y)^{(n)}$을 파트 X와 Y간의 통신 시간이라 한다.

- 전체 계산 시간($LOT^{(n)}$) = $LOT(\mathrm{UA})^{(n)}$ + $LOT(UB)^{(n)}$ + $LOT(KRC)^{(n)}$ + $LOT(KRA)^{(n)}$ × n
- 전체 통신 시간($CCT^{(n)}$) = $CCT(UA,UB)^{(n)}$ + $CCT(UB,KRC)^{(n)}$ + $CCT(KRC,UB)^{(n)}$ + [($CCT(KRC,KRA)^{(n)}$ + $CCT(KRA,KRC)^{(n)}$] × n
- 전체 키복구 시간($TRT^{(n)}$) = $LOT^{(n)}$ + $CCT^{(n)}$

키복구 시간을 정확히 계산하기 위해서는 제안한 시스템에서 사용한 암호모듈들의 실행시간과 통신시간을 측정해야한다. 표 4는 제안한 시스템의 C/Unix버전에서 측정한 각 알고리즘의 수행시간을 보인다.

키복구 에이전트의 개수를 늘이면, 공격자는 많은 에이전트들의 비밀키들을 풀어야하며 공모공격도 어려워지므로 시스템의 보안성은 증가되지만, 키복구에 필요한 통신기간과 계산시간이 많이 소요된다. 최적의 키복구 에이전트 개수는 제안한 시스템을 운영하는 조직의 보안정책, 계산 및 통신속도 등을 고려하여 결정할 수 있다.

표 4 제안한 시스템내의 암호 모듈들의 지연시간 (C/Unix 버전)

암호모듈	지연시간 (ms)(*)	자료크기(bytes)		
		입력	키	출력
DES(암호)	.440	300	16	300
DES(복호)	.128	300	16	300
RSA(암호)	1.635	32	97	64
RSA(복호)	10.331	64	344	32
RSA(서명)	10.263	20	344	64
RSA(검증)	1.332	64	97	20
SHA	.382	97	-	20
KCDSA(서명)	50.643	289	230	120
KCDSA(검증)	85.126	289	180	1(logic)
통신 경로	통신시간	메시지크기		
UA → UB	592.02			4140
UB → KRC	659.66			4613
KRC → KRA1, KRA2	76.93			538
KRA1 → KRC, KRA2 → KRC, KRC → UB	9.15			64

(*) SUN Ultra Sparc, 256 MB, 333 MHz 상에서 측정
(**) 7KByte/Sec 통신대역폭을 가정함(국내의 현실적인 인터넷 속도)

5.2 보안성 기능평가

보안성 평가란 개발한 시스템이 키복구 시스템의 보안 요구사항 명세서(PP)대로 올바른 기능(functionality)을 수행하는 지와, 이 기능들의 보장성(assurance)을 평가하는 것이다[16]. 제안한 시스템에서는 2장에서 주어진 시스템의 보안 요구사항(즉, 보안위협, 보안목적, 보안기능)에 대해, (1) 키복구와 관련된 보안위협들과 보안목적들간의 대응성, (2) 보안목적들과 보안기능들간의 대응성 및 (3) 보안기능들과 제안한 시스템의 보안기능들간의 대응성을 CC 2.0의 평가스킴에 따라 평가하였다. 표 5~7은 평가결과를 보인다.

표 5 보안목적과 보안위협간의 대응성

보안목적 \ 보안위협	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
O1	○		○							
O2		○								
O3				○						
O4					○				○	○
O5						○	○			
O6								○		

6. 관련연구 및 시스템의 응용

6.1 관련연구

키복구 서비스는 Commercial Key Recovery(CKR)[9]에서는 자료복구 센터에 의해 제공되며, CyKey[10]에서는 키복구 당국이 인가한 키복구 요청자에 의해 제공된다. 따라서, 키복구 서비스를 요청하는 모든 사용자들은 키복구 센터와 직접 통신해야 한다. 이 경우, 키복구 센터는 위장 및 사칭공격에 대해 취약할 뿐 아니라 키복구 권한을 독점하게 된다. CKR[9]에서는 access rule index를 이용하여 사용자(주체)들의 접근제어를 수행한다. SecretAgent [11]에서 키복구 서비스는 그림 13-(a)와 같이 1개의 키복구 요청자와 1개의 키복구 에이전트에 의해 제공된다.

IBM의 2단계 CSKR(cryptographic secure key recovery)[12]에서 키복구 서비스는 그림 13-(b)와 같이 1개의 복구 제공자와 2개 이상의 키복구 에이전트에 의해 제공된다.

CSKR에서는 서비스 시간을 줄이기 위해 공개키 암호 연산을 최소화하고 있다.

[단계 1](세션 그룹당 1회 수행): 송신자(user-A)는 수신자(user-B)와 비밀값(S)(난수)를 공유한다. 송신자는 지정된 각 키복구 에이전트에 대해, Key-Generating (KG) 값을 생성하며(S의 해쉬연산 이용) 이를 각 키복구 에이전트의 공개키로 암호화한다. KG값은 세션그룹 내의 단위 세션에서 공통으로 이용하므로 긴 수명을 갖는다.

[단계 2](단위 세션마다 수행): 송신자는 지정된 각 키복구 에이전트에 대해, Key-encrypting Key(KK) 값(난수)을 생성하고 KK들을 이용하여 단위 세션키(ssk)를 여러번 비대칭 암호화하여 비밀값(Xb)을 생성한다. KG값과 Xb값은 키복구 정보(KRI)가 되며 ssk로 암호화된 메시지에 캡슐화하여 송신자로 보낸다.

[복구 단계]: 수신자(키복구 요청자)는 단위 세션키(ssk)를 복구하기 위해, KRI를 키복구 센터를 경유하여 각 키복구 에이전트로 보낸다. 각 키복구 에이전트는 자신의 비밀키를 통해 KG값을 복호화하고(세션 그룹당 1회 수행) KK값을 생성하며(단위 세션마다 수행) 키복구 센터로 보낸다. 키복구 센터는 KK들을 이용하여 Xb에 대해 비대칭 복호화하여 세션키(ssk)를 생성하여 송신자에게 보낸다.

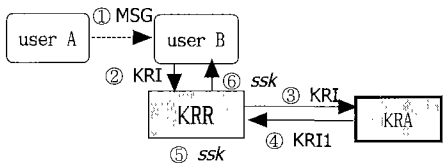
시간이 많이 걸리는 공개키 암호 연산은 KG에 대해서만 실행하며, KG는 KK로부터 유도할 수 없으므로, 세션그룹 내의 단위 세션마다 여러번 사용될 수 있다. CSKR에서는 파라미터 검증스킴을 통해 키의 복구성(recoverability)을 분석하였다.

표 6 보안목적과 CC 2.0의 보안기능간의 대응성

CC의기능 / 보안목적	FAU_ARP.1	FAU_GEN.1	FAU_GEN.2	FDP_ACC.1	FDP_ACF.1	FCO_NRO.1	FDP_UCT.1	FPT_ICT.1	FDP_UIT.1	FDP_DAU.1	FIA_AFL.1	FIA_UAU.1	FIA_UAU.2	FIA_UAU.3	FIA_UAU.4	FIA_UAU.5	FIA_UID.1	FIA_UID.2	FIA_USB.1
O1	○	○	○																
O2				○	○														
O3						○													
O4									○										
O5							○	○											
O6										○	○	○	○	○	○	○	○	○	○

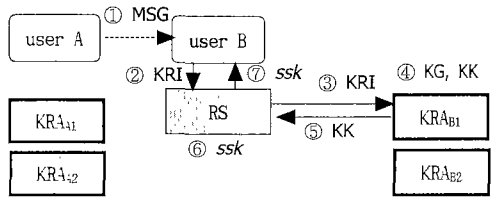
표 7 CC 2.0의 기능 요구사항과 제안한 시스템 기능간의 대응성

시스템 모듈 \ CC의 기능		FAU_ARP.1	FAU_GEN.1	FAU_GEN.2	FDP_ACC.1	FDP_ACF.1	FCO_NRO.1	FDP_UCT.1	FPT_ICT.1	FDP_UIT.1	FDP_DAU.1	FIA_AFL.1	FIA_UAU.1	FIA_UAU.2	FIA_UAU.3	FIA_UAU.4	FIA_UAU.5	FIA_UID.1	FIA_UID.2	FIA_USB.1
사용자 서브 시스템	인증모듈				○	○						○	○	○				○	○	○
	키복구 요청모듈								○	○					○					
	키복구 모듈								○	○	○				○	○	○			
	통신모듈							○												
키복구 센터 서브 시스템	인증모듈				○	○						○	○	○				○	○	○
	관리모듈	○	○																	
	키복구 요청모듈								○	○					○					
	키복구 모듈								○	○	○				○	○	○			
	통신모듈							○												
	감사모듈	○	○	○			○													
키복구 에이전트 서브 시스템	인증모듈				○	○						○	○	○				○	○	○
	관리모듈	○	○																	
	키복구 요청처리모듈								○	○					○					
	키복구 모듈								○	○	○				○	○	○			
	통신모듈							○												
	감사모듈	○	○	○			○													



① MSG = SC_e(*data, ssk*) + KRI
(여기서, KRF = $PC_e(PC_e(ssk, puk_{KRR}), puk_{KRA})$
② ③ KRI
④ KRI1 = PC_d(KRI, prk_{KRA})
⑤ *ssk* = PC_d(KRI1, prk_{KRR})
⑥ *ssk*

(a) SecretAgent



①② MSG = SC_e(*data,* ssk) + KRI
KRI = [B1 + B2]; KG = Hash(S)
B1 = [T1, KG_{B1}' = PC_e(KG, puk_{B1}),
KG_{B2}' = PC_e(KG, puk_{B2})]: 그룹세션당 1회 수행
B2 = [T2, Hash(B1), $Xb^{(i)}$ = $SC_e(SC_e(ssk^{(i)}, KK^{(i)}{}_{B2}), KK^{(i)}{}_{B1})$]: 각 단위세션 *i*에 대해 수행
③ KRI (user B의 요청 또는 법집행)
④ KG_{B1} = $PC_d(KG_{B1}', prk_{B1})$, KG_{B2} = $PC_d(KG_{B2}', prk_{B2})$: 1회 수행
$KK^{(i)}{}_{B1}$ = $Gen(KG_{B1})$, $KK^{(i)}{}_{B2}$ = $Gen(KG_{B2})$: 각 단위세션 *i*에 대해 수행
⑤ $KK^{(i)}{}_{B1}$, $KK^{(i)}{}_{B2}$
⑥⑦ $ssk^{(i)}$ = $SC_d(SC_d(Xb^{(i)}, KK^{(i)}{}_{B1}), KK^{(i)}{}_{B2})$: 각 단위세션 *i*에 대해 수행

(b) CSKR

<주> SCe(x,y): 자료 x를 키 y로 대칭 암호화, SCd(x,y): 자료 x를 y키로 대칭 복호화, PCe(x,y): 자료 x를 키 y로 비대칭 암호화, PCd(x,y): 자료 x를 키 y로 비대칭 복호화, Gen(): KK의 생성함수

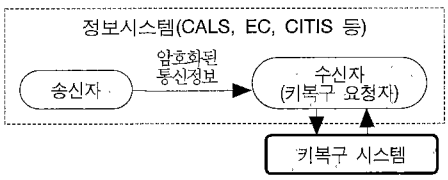
그림 13 기존의 키 캡슐화 방식의 키복구 시스템

그러나, 단위 세션(예; 단위 트랜잭션)들을 그룹세션으로 쉽게 그룹화 할 수 있을 때만 장점을 갖는다. 또한, 수신자(키복구 요청자)는 송신자가 지정한 키복구 에이전트의 식별자 및 공개키를 알아야만 한다.

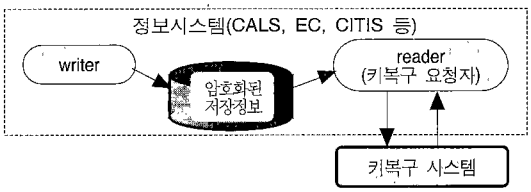
6.2 시스템의 응용

제안한 시스템은 당초에 CALS, 전자상거래(SET, SSL), CITIS(Contractor Integrated Technical Information Service) 인프라 내의 전자문서 복구 솔루션으로 개발되었지만, 암호 통신을 이용하는 모든 정보시스템에서 응용될 수 있다.

또한, 그림 14와 같이 제안한 시스템의 송신자 서브시스템을 기존의 출력 프로세서에 추가하고 수신자 서브시스템을 입력 프로세서에 추가하면 제안한 시스템은 암호화된 저장정보의 복구 시스템이 된다. 즉, 출력 프로세서는 저장할 정보를 암호화하고 키복구 정보를 생성하여 암호화된 정보와 함께 디스크에 저장한다. 키복구시 입력 프로세스(키복구 요청자)는 키복구 정보를 키복구 센터에 보내면 키를 복구할 수 있다.



(a) 암호화된 통신정보의 복구



(b) 암호화된 저장정보의 복구

그림 14 제안한 시스템의 응용

7. 요약 및 결론

본 연구에서 개발한 키복구 시스템은 기존의 키 캡슐화 방식의 시스템들과 비교할 때 다음과 같은 특성을 갖는다. 먼저, 시스템 프로토콜상의 특성은 다음과 같다.

첫째, n개의 키복구 에이전트들을 이용한 키복구 방식이며 기존의 시스템들보다 일반적인 방식이다.

둘째, 송신자(키복구 정보 생성자)는 비밀리에 키복구 에이전트들을 지정할 수 있으며, 수신자(키복구 요청자)는 참여하는 키복구 에이전트들에 관한 정보를 알 필요가 없다. 따라서, 키복구 요청자에게 키복구 서비스에 대한 투명성이 제공된다.

셋째, n개의 키복구 에이전트로 구성된 시스템에서, 세션키(128비트)는 128/n비트로 분할하지 않고, n-1개의 난수키와 세션키를 이용하여 n개의 중간키(128비트)들을 생성하므로, 세션키에 대한 보안성이 증대된다.

넷째, 키복구 요청자는 새로운 키쌍(공개키, 개인키)을 생성하고 이를 이용하여 키복구를 수행한다. 따라서, 키복구 작업 이후에는 손실된 키는 자동적으로 무효화되므로, 이전의 키를 이용한 불법적인 키복구는 불가능하다.

다섯째, 제안한 시스템은 통신정보 뿐만 아니라 저장정보에 대한 키복구가 가능하다. 저장정보에 대한 키복구시, 송신자와 수신자 서브시스템 측은 각각 기존의 디스크 출력 및 입력 시스템 내에 삽입하면 된다.

또한, 시스템 개발상의 특성은 다음과 같다.

첫째, 검증되고 잘 알려진 암호 알고리즘들을 포함하고 있는 범용 암호 라이브러리를 이용함으로써, 시스템의 유지보수성이 높다. 즉, 암호 알고리즘들을 쉽게 교체할 수 있다.

둘째, 2개의 버전을 개발하였으며, 특히 Java/NT버전은 자바환경이 갖는 장점들을 살릴 수 있다.

셋째, 정보보호시스템을 위한 새로운 개발방법론이라 할 수 있는 "파트-보안관계" 방법론을 새롭게 개발하여 활용하였다.

넷째, 국제표준 정보보호시스템 평가기준인 CC 2.0과 키복구 시스템의 사실상의 국제표준인 RKRP를 기반으로 하여 분석, 설계 및 평가하였다.

개발한 시스템을 상용화하기 위한 성능개선 및 패키지화, 파트-보안관계 방법론의 체계화 및 방법론 지원도구의 개발(예, 파트-보안관계 분석기, 템플리트 및 소스코드 생성기 등) 등은 향후 연구할 부분이다.

참 고 문 헌

[1] Technology Committee of Key Recovery Alliance, Cryptographic Information Recovery using Key Recovery, A Working Paper, Version 1.2, http://www.kra.org, 1997.

[2] Denning, D. and Branstad, D., "A Taxonomy for Key Escrow Encryption Systems," *Communications of the ACM*, Vol.39, No.3, pp.34-40, 1996.

[3] Ganesan, R., "How To Use Key Escrow," *Communications of the ACM*, Vol.39, No.3, p.33, 1996.

[4] He, J. and Dawson E., "A New Key Escrow

Cryptosystem," *Lecture Notes in Computer Science*, Vol.1029, pp.105-113, 1995.
[5] Lee Y. and Laih C., "On the key recovery of the Key Escrow System," *Proceedings of 13th Annual Computer Security Applications Conference*, pp.216-220, 1997.
[6] Ganesan, R., "The Yaksha Security System," *Communications of the ACM*, Vol.39, No.3, pp.55-60, 1996.
[7] Jefferies, N., et al., "A Proposed Architecture for Trusted Third Party Services," *Lecture Notes in Computer Science*, Vol.1029, pp.98-104, 1995.
[8] Maher, D., "Crypto Backup and Key Escrow," *Communications of the ACM*, Vol.39, No.3, pp.48-53, 1996.
[9] Walker, S., et al., "Commercial Key Recovery," *Communications of the ACM*, Vol.39, No.3, pp.41-47, 1996.
[10] Cylink, CyKey: A Key Recovery System for Commercial Environments, Cylink Corp., www.cylink.com, 1998.
[11] Markwitz, M. and Sclafly, R., "Key Recovery in SecretAgent," *Digital Signature*, 1997.
[12] Gennaro, R., et. al., Secure Key Recovery, IBM Thomas J. Watson Research Center, 1999.
[13] CC, Common Criteria for Information Technology Security Evaluation, Ver 2.0, CCEB, ISO/IEC 15408, 1998.
[14] RKRP, Requirements for Key Recovery Products, Final Report, Federal Information Processing Standard for Federal Key Management Infrastructure, http://csrc.nist.gov/keyrecovery, 1998.
[15] KRA, Business Requirements for Key Recovery, Key Recovery Alliance, Rel. 3.0, 1997.
[16] 이강수, "선진국 정보보호시스템의 평가제도에 관한 연구", 정보통신학술연구지원국, 정보통신부, 1998.
[17] 강상승 외, "인트라넷 기반의 키 복구 시스템 구현", *CISC'99*, 1999.
[18] H. Gomma, *Software Design Methods for Concurrent and Realtime Systems*, Addison-Wesley, 1993.

강 상 승

1997년 경북대학교 전자공학과 학사. 1999년 경북대학교 전자공학과 석사. 1999년 ~ 현재 한국전자통신연구원 전자상거래연구부 전자지불연구팀 연구원. 관심분야는 전자상거래, 정보보호, 무선 인터넷 기술

임 신 영

1983년 건국대학교 공업화학과 학사. 1985년 건국대학교 화학공학과 석사. 1992년 건국대학교 전자계산학과 석사. 2001년 고려대학교 컴퓨터학과 박사. 1987년 ~ 1988년 (구)시스템공학연구소 올림픽정보화 사업(GIONS 개발). 1989년 ~ 1993년 (구)시스템공학연구소 슈퍼컴퓨터 구축 사업(인터넷 전산망 관리). 1994년 ~ 1997년 (구)시스템공학연구소 연구전산망 구축 및 개발 사업(전산망 보안). 1998년 ~ 현재 한국전자통신연구원 전자상거래연구부 전자지불연구팀 팀장. 관심분야는 전자 지불, 암호키 복구, IC 카드 보안, 홍채 인식, 디지털 콘텐츠 정보보호

고 정 호

1997년 한남대학교 컴퓨터공학과 학사. 1999년 한남대학교 컴퓨터공학과 석사. 1999년 ~ 현재 한남대학교 컴퓨터공학과 박사과정. 2000년 ~ 현재 목원대학교 겸임교수. 관심분야는 전자상거래, 정보보호, 소프트웨어 컴포넌트

전 은 아

1999년 한남대학교 컴퓨터공학과 학사. 2001년 한남대학교 컴퓨터공학과 석사. 2001년 ~ 현재 (주)퓨처시스템 정보통신연구소 보안팀 연구원. 관심분야는 암호응용, 정보보호, CALS/EC, 소프트웨어공학

이 강 수

1981년 홍익대학교 컴퓨터공학과 학사. 1983년 서울대학교 대학원 전산학과 석사. 1989년 서울대학교 대학원 전산학과 박사. 1985년 ~ 1987년 국립대전산업대학교 전자계산학과 전임강사. 1992년 ~ 1993년 미국일리노이대학교 객원교수. 1995년 한국전자통신연구원 초빙연구원. 1998년 ~ 1999년 한남대학교 멀티미디어학부장. 1987년 ~ 현재 한남대학교 컴퓨터공학과 정교수. 관심분야는 소프트웨어공학, 병행시스템 모델링 및 분석, 정보보호시스템 평가, 멀티미디어교육 커리큘럼, 웹 엔지니어링