通信情報保護論文誌
第9券 第2號 1999. 06

안전한 키 위탁 시스템에 관한 연구

채 승철*, 이 임영*

A Study on the Secure Key Escrow System

Seung-chul Chae*, Im-yeong Lee*

요약

암호를 사용함으로써 정보를 위·변조나 노출 위협으로부터 보호할 수 있지만, 암호의 잘못된 사용은 몇 가지 문제점을 발생시킬 수 있다. 만약 키가 손상되면 암호의 적법한 소유자조차도 암호문을 해독할 수 없으며, 범죄 등의 목적에 암호가 사용되었을 경우, 수사 기관이 수사에 어려움을 겪을 수 있다. 이러한 문제점을 해결하기 위해 키 또는 메시지를 복구할 수 있도록 키를 신뢰기관에 위탁하는 키 위탁(key escrow) 방식이 제안되었다. 키 위탁은 유사시 키 복구를 보장하면서 동시에 사용자의 사생활을 침해해서는 안된다. 본 논문에서는 키 위탁 시스템이 가져야할 요구사항을 도출하고, ElGamal 암호 시스템에 기반한 새로운 키 위탁 방식을 제안한다. 제안 방식은 도청기한의 제한, 키의 복구에 관련된 기관의 부정 방지와 이에 대한 사용자의 shadow 공개키 생성 방지 등의 속성을 갖는 새로운 키 위탁 방식이다.

Abstract

Cryptography techniques can prevent eavesdroppers from maliciously intercepting or modifying sensitive information. However, misuses of encryption may cause other problems. First, if the encryption key is lost or damaged, even an authorized access to the original data will be denied. Second, criminals can prevent authorized law enforcement officers from examining the necessary information by using the strong encryption technique. A technique known as Key Recovery, which recovers the encryption key from encrypted data, can provide solutions for the situations. In this paper, we propose a new key escrow system based on the ElGamal cryptosystem. Our system provide time-bound eavesdropping under court authorized permission, protect from trustee's cheating and prevent user's shadow public key generation.

I. 서론

정보화 사회로 진입한 오늘날 모든 정보들은 점차적으로 종이 문서의 형태에서 전자적 형태로 변환되어 전달되고 있다. 이미 개인간의 전화, 팩스, E-mail 등의 이용은 보편화되었으며, 개개인의 건강/인사 기록과 같은 사적인 정보에서부터 은행간의 거래, 중요한 사업상의 정보 등의 기밀성을 요하는 자료 및 비행 관제 시스템, 교통 정보 시스템 등과 같은 국가 기간망에 이르는 많은 정보들이 전자적 형태로 이동되고 있다. 이처럼 전자적인 정보의 유통량이 증가하고 정보의 가치가 높아질수록 정보 보호의 문제가 부각된다. 정보 보호란 보다 안전하고 신뢰성 있게 정보를 전달할 수 있도록 하는 것으로 암호학에 그 기반을 두고 있다. 암호란 키와 수학적인 알고

리즘을 이용하여 평문을 알아보기 힘든 암호문의 형태로 변환시키는 것으로, 안전하게 구현된 암호 시스템에서는, 키를 알지 못하는 사람은 암호화된 데이터를 복호할 수 없다는 것을 전제로 한다.

암호의 사용은 정보의 누출 및 오용을 방지하고 상대의 신원 확인을 가능하게 함으로써, 온라인 상에서의 전자상거래나 전자 계약을 가능하게 하는 등 많은 장점을 가지고 있다.

반면에 암호는 키가 분실되면 그 키를 사용해서 암호화된 정보를 복원할 수 없다는 점과 범죄 집단이 법 집행을 방해할 목적으로 암호를 사용하는 경우 수사가 불가능하다는 등과 같은 위험 요소를 안고 있다. 이러한 문제는 암호 사용이 일반화하는데 장애 요소로 작용하고 있다. 이러한 문제점의 해결책으로 현재 유력하게 대두되는 방법이 키 복구 방식이다. 키 복구란 유사시에 암호문의 키 또는 메시지를 복구하여 위에서 언급한 문제점을 해결해보고자 하는 것이다. 이러한 키 복구는 키가 쉽게 복구되어서 사용자의 사생활을 침해할 수 있는 소지를 가져서는 안되며, 또한 적법한 상황에서는 언제든지 키 또는 메시지의 복구가 가능해야 한다는 두 가지 상반된 목표를 동시에 만족해야 한다.

키 복구 방식은 크게 키를 신뢰성있는 제 3자에게 위탁(Escrow)하고 이것을 통해 키 또는 메시지를 복구하는 키 위탁(Key escrow) 방식과 암호화된 메시지가 생성될 때 메시지에 복구에 필요한 정보를 부가하는 캡슐화(Encapsulation) 방식으로 분류할 수 있다.

일반적으로 캡슐화 방식은 메시지마다 세션키에 관계된 복구 정보가 부가되기 때문에 사생활 침해 가능성이 비교적 적지만 복구 정보의 조작이 쉽기 때문에 복구의 확실성이 떨어진다는 단점이 있다. 키 위탁 방식은 사용자의 키를 직접 신뢰 기관에 위탁하므로 유사시 키나 메시지의 복구가 확실하지만 키 정보 유출 위험성 때문에 사용자의 사생활 침해 가능성이 항상 존재한다. 따라서 키 위탁 시스템은 키 복구의 확실성을 고려하면서 사용자의 사생활을 보호해 주어야 한다.

본 논문에서는 이러한 두 가지 조건에 부합하는 새로운 키 위탁 시스템을 제안하였다. 논문의 구성은 2장에서는 키 위탁 시스템이 가져야할 기본적인 요구사항을 정의하고, 3장에서는 기존에 제안된 시스템들을 살펴본다. 4장에서는 새로운 키 위탁 방식을 제안하고, 5장에서는 제안된 시스템에 대하여 고찰할 것이다.

II. 키 위탁 시스템의 요구사항

키 위탁 시스템(Key escrow system)이란 사전에 약속된 개체에게 키를 위탁하고, 어떤 특정한 조건하에서 허가된 개체에게 암호문의 키나 평문의 복구가 가능한 능력을 제공하는 암호 시스템이라고 할 수 있다. 일반적으로 키 위탁 시스템은 다음과 같은 조건을 만족해야 한다.

[조건 1] 위탁 시스템은 기존 암호 시스템과 동일한 무결성과 기밀성을 유지해야 한다.

키 위탁 기능이 암호 시스템 전체의 무결성과 기밀성에 영향을 미쳐서는 안된다는 것을 의미한다. 즉, 암호문의 송신자와 수신자, 그리고 적법한 상황에서의 복구기관을 제외한 제 3자가 암호문을 해독할 수 없어야 한다.

[조건 2] 유사시에 키(또는 메시지)를 복구할 수 있어야 한다.

키 위탁 시스템은 기본적인 기능인 키 또는 메시지 복구의 확실성을 보장함으로써 유사시에 합법적인 수사기관이나 데이터의 소유자가 암호문에 접근할 수 있도록 해주어야 한다.

[조건 3] 키를 복구할 수 있는 기간이 적절하게 제한되어야 한다. 또한 사용자의 비밀키가 복구되어서는 안된다.

암호의 특성상 일단 키가 복구되면 그 이후의 통신은 모두 노출된다. 키 위탁 시스템은 이러한 상황을 방지하기 위해 반드시 복구 기간을 제한할 수 있어야 한다.

[조건 4] 키의 복구는 반드시 적법한 절차에 따른 상황에서만 이루어져야 한다.

사전에 정의된 합법적인 절차 이외에 복구기관의 공모 등을 통해 키 복구가 이루어질 수 없어야 함을 의미한다. 사용자의 사생활 보호를 위해서 반드시 이러한 조건이 필요하다. 또한 시스템은 적절하지 않은 절차에 따른 키 복구를 방지할 수 있는 기능을 가지고 있어야 한다.

[조건 5] 키의 생성은 사용자가 참여하는 것이 바

람직하다. 단 사용자가 키 생성을 할 경우에는 숨겨진 채널(Subliminal channel) 등을 구성할 수 없도록 하여야 한다. 또한 사용자와 복구기관은 모두 키가 랜덤하게 선택되었다는 것을 확신할 수 있어야 한다.

키의 생성을 복구 기관에서 담당할 경우 복구 기관이 사용자의 비밀키까지 생성하므로 적법하지 않은 상황에서 키의 유출 위험이 높을 뿐만 아니라, 사용자는 자신의 키가 랜덤하게 선택된 것인지 알 수 없다. 따라서 키의 생성에는 사용자가 참여하는 것이 바람직하다. 하지만 사용자가 전적으로 키를 생성하는 경우에는 다음과 같은 Shadow 공개키의 문제가 발생할 수 있다.

일반적으로 사용자는 공개키 비밀키 쌍 (P, S)을 선택한 후에 공개키 P를 공개하고, 비밀키 S에 대한 복구능력을 복구기관에게 준다. 하지만 사용자가 전적으로 키를 생성한다면 일방향 해쉬함수와 같이 널리 알려진 공개된 함수 f를 사용하여 P′=f(P)인 (P, S), (P′, S′)쌍을 생성할 수 있다. 이때 사용자는 P를 공개하고, S에 대한 복구 능력을 복구기관에게 주더라도 P′과 S′을 이용하여 통신을 한다면 복구 기관은 적법한 상황에서도 이 사용자의 암호문을 복구할 수 없다. 사용자가 키를 생성하는 경우, 이러한 문제점에 대한 해결책이 마련되어야 한다.

[조건 6] 키의 위탁 및 복구 과정의 알고리즘은 공개적으로 알려지는 것이 바람직하다.

알고리즘의 신뢰도를 위해 키 위탁 및 복구에 사용되는 모든 알고리즘은 공개적으로 알려지는 것이 좋다. 또한 알고리즘이 공개되어도 전체 시스템의 안전도에 영향을 미쳐서는 안된다.

[조건 7] 시스템의 오용이 어려워야 하고 오용의 감지는 쉽게 할 수 있어야 한다.

키 위탁 시스템이 설계상의 오류로 키를 복구할 수 있는 제 3의 경로를 제공해서는 안된다. 또한 키 위탁 시스템 내에서 키 복구를 회피할 수 있어서는 안된다.

[조건 8] 누구나 쉽게 사용할 수 있어야 하고, 비용이 저렴해야 한다.

키 위탁 시스템은 소프트웨어로 구현되는 것이 바람직하다. 소프트웨어 구현은 비용이 저렴하며, 설치 및 사용이 간편하다.

이상에서 제시한 조건 1~5는 키 위탁 시스템이 필수적으로 갖추어야 하는 조건이며 조건 6~8은 바람직한 속성이다. 본 제안 방식에서는 이러한 요건을 모두 만족한다.

III. 기존의 키 위탁 시스템

키 위탁이란 사용자의 비밀키의 전부 또는 일부를 신뢰받는 제 3자(Trusted Third Party)에게 위탁함으로써 유사시에 키를 복구할 수 있는 능력을 주는 것이라고 말할 수 있다. 또한 키 위탁 방식은 항상 사용자의 프라이버시 보호와 정부의 법 집행 능력 보장이라는 두 가지 상반된 목적을 만족 시켜야 한다. 즉, 복구 능력을 갖는 주체라도 합법적인 절차를 따르지 않는다면 복구가 불가능하도록 구성되어야 한다.

이와 같은 조건을 만족시키기 위해서 많은 연구가 이루어졌다. 가장 먼저 미국에서는 클리퍼(Clipper) 프로젝트라는 이름으로 키 위탁 장치의 개발이 이루어졌다. 클리퍼 칩은 위탁 암호화를 위한 전용 칩으로써 하드웨어로 만들어진 클리퍼 칩 안에 위탁된 키를 삽입해 놓은 형태로 보급되었다.[6]

그러나 클리퍼 칩은 특별한 하드웨어 장치를 필요할 뿐만 아니라 키 위탁 기관의 사생활 침해 가능성이 제기됨으로써 널리 보급되지 못했다. 또한 짧은 오류 검출 비트(Checksum) 등의 내부적인 프로토콜의 설계의 미비함 등으로 인해 여러 가지 공격이 가능하다는 내용이 지적되기도 하였다.[4]

Fair Cryptosystem은 1992년 Silvio Micali가 Crypto 92에서 처음으로 제안하였다.[2] 이 방식은 기존의 공개키 방식에 적용할 수 있는 안전한 키 위탁 방식을 제시하고 있다. Fair Cryptosystem이란 불법적인 도청의 방지와 합법적인 도청의 용이성을 보장해줄 수 있는 암호 시스템을 의미한다.

Fair Cryptosystem에서는 소프트웨어 기반의 키 위탁 시스템과 키의 불법적인 복구를 막기 위해 기존의 공개키 암호를 Fair하게 만드는 방법을 제시하고 있다. 이 시스템에서는 사용자가 자신의 키를 생성해서 n개의 위탁 기관에게 VSS(Verifiable Secret Sharing) 방식을 사용해서 분산 위탁하는 방식이다. 모든 신뢰 기관은 사용자의 위탁 정보가 올바른 키의 부분인지 확인하고 키 승인 정보를 공개키 인증기관에게 보내며, 공개키 인증기관은 모든 위탁 기관

의 승인 정보가 수신되면 키를 인가한다.

이 방식은 전적으로 사용자가 키를 생성함으로써 Shadow 공개키 문제가 생길 수 있다.[3] 또한 이 방식에서는 키의 복구기한을 제한 할 수 있는 방법을 제시하지 못하였다.

이후에 발표된 Failsafe Cryptosystem은 Kilian과 Leighton이 "Fair Cryptosystems, Revisited"라는 논문에서 제안한 방식으로 Fair Cryptosystem에서 발생할 수 있는 Shadow 키와 관련된 문제점을 해결할 수 있도록 사용자와 신뢰 기관이 협력해서 키를 선택하는 방식을 제안하고 있다.[3]

이 방식은 다음과 같은 조건을 만족시킨다.

◦ 각 사용자들은 위탁기관과 인증기관을 신뢰하지 못하더라도 비밀키가 안전하게 선택되었다는 것을 확신할 수 있어야 한다.

◦ 인증기관은 사용자가 랜덤하지 않은 생성기를 사용하더라도 키가 안전하게 선택되었다는 것을 확신해야 한다.

Failsafe Key Escrow 방식은 사용자와 위탁기관이 서로 신뢰하지 못하더라도 비밀키가 안전하게 선택되었다는 것을 확신할 수 있도록 사용자와 키 위탁 기관이 함께 키를 생성한다.

이 방식은 Shadow 공개키의 문제를 해결하였지만, Fair Cryptosystem에 기반한 방식이므로 여전히 복구 기한의 제한 문제가 그대로 남아있다. 또한 위의 두방식 모두 VSS 방식에 전체 시스템의 안전도가 달려있지만, 모든 신뢰기관이 공모할 경우 키를 얻을 수 있기 때문에 안전성에 문제가 생긴다.

최근에 일본에서 발표된 Blind decoding을 이용한 키 위탁 방식에서는 이러한 몇 가지 문제점을 해결하였으나, Blind decoding시에 복호자의 속임수 방지를 위한 과부하 및 키 생성시 사용자가 참여하지 않음으로써 생성된 키가 랜덤한지 사용자가 확신할 수 없다는 문제가 있다.[9]

Ⅳ. 제안 방식

이상에서 살펴본 바와 같이 키 위탁 시스템을 구성할 때에는 복구의 확실성과 사용자의 사생활 보호라는 두가지 측면이 모두 고려되어야 하지만 대부분의 시스템은 2장에서 제시한 조건을 모두 만족시키지 못하고 있다. 키 위탁 시스템은 그 특성상 가장 취약한 점이 전체 시스템의 성능을 결정하기 때문에 제시된 조건들을 하나라도 만족하지 못한다면 높은 신뢰성을 획득할 수 없다.

본장에서 제안하고자 하는 시스템은 다음과 같은 특징을 갖는다.

• 사용자의 공개키/비밀키는 사용자와 신뢰 기관이 협력해서 생성한다.

이것은 Fair cryptosystem에서 발생할 수 있는 Shadow 공개키 문제를 해결할 수 있으며, Blind decoding을 이용한 방식처럼 전적으로 신뢰기관이 키를 생성할 때 생기는 문제점을 방지한다.

• 분산된 키 정보를 보관하고 있는 신뢰기관들이 모두 공모하여도 키를 복원할 수 없다.

Fair Cryptosystem이나 Failsafe 방식, Clipper 방식에서는 키를 분산 보관하는 신뢰기관들이 공모한다면 사용자의 비밀키를 조합할 수 있다. 하지만 제안하는 방식에서는 이러한 점을 미리 방지함으로써 사용자의 사생활을 안전하게 보호한다.

• 사용자의 비밀키 정보는 어떠한 경우에도 유출되지 않는다.

사용자의 비밀키가 위탁되는 과정에서 분산되며, 키 생성과정이나 복구 과정에서 사용자의 완전한 비밀키를 알 수 있는 주체는 사용자 뿐이다. 이후의 복구 과정에서도 비밀키 자체는 복원되지 않는다.

• 사용자의 통신을 감청할 수 있는 기한을 법원에 의해 제한이 가능하다.

프로토콜에 참여하는 주체들은 다음과 같다.

사용자(U) : 키를 위탁하고, 위탁된 키를 사용해서 통신을 하는 주체

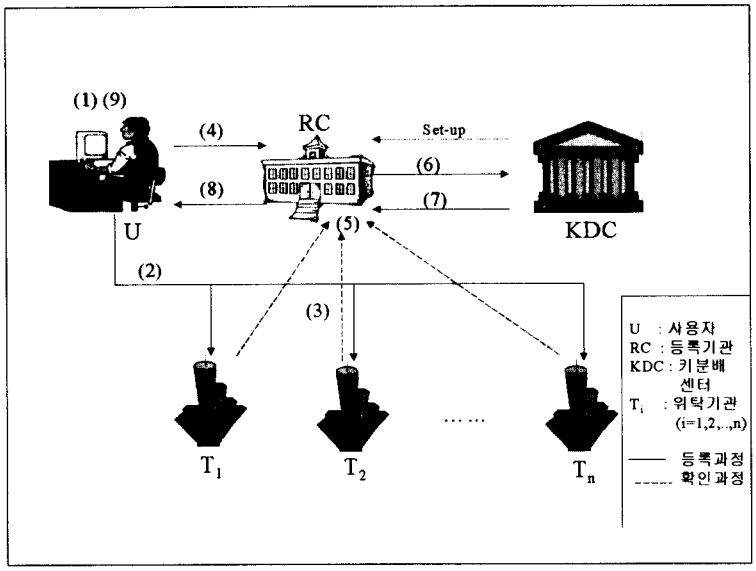
등록/복구 기관(RC) : 키 등록, 인증서 발급, 복구 요청의 처리를 담당하는 주체

키 분배 센터(KDC) : 사용자 등록시 필요한 키쌍을 생성하고 보관하는 주체.

위탁기관(T_i) : 위탁된 키 정보를 보관하는 주체

제안된 키 위탁 시스템에서 사용자는 다음과 같은 과정을 거쳐서 시스템에 등록한다.

4.1 사전 준비 단계



[그림 1] 사용자 등록 과정
Fig 1. User registration process

키 분배 센터 KDC는 랜덤한 생성자 g ($\in Z_p^*$), 소수 p의 시스템 파라미터를 준비한다. 이 값은 시스템의 모든 참여자들에게 공개적으로 알려진다.

KDC와 등록 기관 RC, 위탁 기관 T는 각각 다음과 같은 방식으로 공개키/비밀키 쌍을 생성한다. 공개키는 시스템의 모든 참여자들에게 알려진다.

$$P = g^S \bmod p \quad (S \in Z_{p-1}) \qquad (1)$$

KDC는 다음과 같은 다수의 키 정보 (S_{Bi}, P_{Bi})를 생성한다. 이 정보는 사용자 등록시 사용자가 하나를 선택해서 자신이 생성한 키 정보와 조합되어 사용자의 키를 결정하는데 사용된다.

$$P_{Bi} = g^{S_{Bi}} \bmod p \quad (S_{Bi} : \text{랜덤수}) \qquad (2)$$

또한 랜덤 생성기를 이용하여 랜덤하게 ID_i 값을 생성하고, ID_i 와 위에서 생성된 키 쌍 (S_{Bi}, P_{Bi})를 연결해서 자신의 데이터 베이스에 안전하게 저장한다. KDC의 데이터베이스에는 다수의 (ID_i, P_{Bi}, S_{Bi})의 쌍이 저장된다. 이렇게 생성된 값들 중에서 (P_{Bi}, ID_i) 만을 RC에게 전송한다.

RC는 KDC로부터 전달받은 다수의 (P_{Bi}, ID_i)를 공개 보드를 통해 공개한다. 사용자는 등록 단계에서 이러한 키 쌍중의 하나를 선택하게 된다.

4.2 사용자 등록 단계

(1) 사용자 U는 스스로 다음과 같은 비밀키/공개키 쌍을 생성한다.

$$S_{UA},\ P_{UA} = g^{S_{UA}} \bmod p \quad (S_{UA} : \text{랜덤수}) \qquad (3)$$

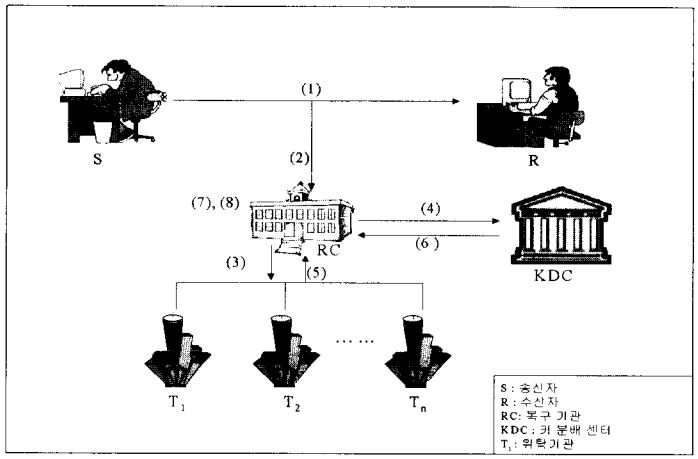
(S_{UA} : 사용자 U의 비밀키, P_{UA} : 사용자 U의 공개키)

사용자는 생성된 자신의 비밀키 정보 S_{UA}를 다음과 같은 방식을 사용하여 n개의 분할 정보 S_{UAi}를 생성하고 각각의 키 분할 정보에 대한 확인 정보 V_i를 만든다. 이것은 다음과 같이 구성될 수 있다.

$$S_{UA} = S_{UA1} + S_{UA2} + \ldots + S_{UAn} \bmod (p-1) \quad (1 \leqq S_{UAi} \leqq p-2) \qquad (4)$$

$$V_i = g^{S_{UAi}} \bmod p \qquad (5)$$

(2) 이후에 사용자는 각 위탁기관 T_i들의 공개키



[그림 2] 데이터 복구 과정
fig 2. Data recovery process

로 키 분할 정보 S_{UAi}와 확인 정보 V_i를 암호화하여 각각의 T_i들에게 전송한다.

(3) T_i는 수신된 V_i와 S_{UAi}를 가지고 다음을 확인한다.

$$g^{S_{UAi}} \bmod p = V_i$$

위의 과정이 올바르면 T_i는 V_i와 승인 정보를 RC로 전송한다.

(4) 사용자 U는 등록기관 RC의 공개된 보드로부터 (ID_i, P_{Bi})값 중의 하나를 선택하고 RC에게 다음과 같은 정보와 함께 공개키 승인 요청을 보낸다. 이때 사용자 U가 선택한 (ID_i, P_{Bi})값을 (ID_U, P_{UB})라고 할 때 공개키 승인 요청은 다음과 같다.

CertRequest = { X_i || ID_U || P_{UA} || Identity }

(ID_U : 사용자가 선택한 ID, P_{UA} : 사용자가 생성한 공개키, Identity : 사용자의 신원정보)

(5) RC는 공개키 승인 요청이 수신되면 위탁기관들이 보내온 각각의 승인 정보를 확인하고 각각의 V_i를 가지고 다음이 올바른지 확인한다.

$$P_{UA} = \prod_{i=1}^{n} V_i \bmod p \qquad (6)$$

$$(\because \prod_{i=1}^{n} V_i = g^{S_{UA1}} g^{S_{UA2}} \ldots g^{S_{UAn}} = g^{S_{UA1}+S_{UA2}+\ldots+S_{UAn}} = g^{S_{UA}} = P_{UA})$$

(6) RC는 수신된 승인 요청이 올바르면 자신의 데이터베이스에 사용자의 신원과 ID 값인 (Identity, ID_i)를 연결해서 저장하고, 사용자가 생성한 공개키 P_{UA}와 사용자가 선택한 ID_U를 KDC에게 전송한다.

(7) KDC는 사용자가 전송한 P_{UB} 에 대응되는 S_{UB}를 자신의 데이터베이스에서 검색하여 P_{UA}로 암호화한 $E_{P_{UA}}(S_{UB})$를 RC에게 전송한다.

(8) RC는 (5)의 과정이 올바르면 사용자의 P_{UA}와 ID_U에 해당하는 P_{UB}를 곱해서 사용자의 최종 공개키를 만든다.

$$P_U = P_{UA} * g^{S_{UB}} = g^{S_{UA}} g^{S_{UB}} = g^{(S_{UA}+S_{UB})} \qquad (7)$$

생성된 공개키를 인증한 공개키 인증서와 KDC로부터 수신된 $E_{P_{UA}}(S_{UB})$를 사용자에게 전송하고, 자신의 데이터베이스에 사용자의 공개키 P_U를 저장한다.

$Sign_{S_{RC}}(P_U) \| E_{P_{UA}}(S_{UB})$(Sign:전자 서명) (8)

(9) 사용자는 (8)에서 전송된 S_{UB}를 복호해서 최종적으로 인증된 공개키 P_U와 다음과 같은 비밀키 S_U를 갖게 된다.

$$S_U = S_{UA} + S_{UB} \bmod p \quad (9)$$

4.3 데이터 복구 단계

데이터 복구 단계에서 RC는 복구 기관의 역할을 하게 된다. 송신자 S가 수신자 R에게 암호문을 전송할 때 RC가 메시지를 복구하는 과정은 다음과 같다.

(1) 송신자 S는 수신자 R에게 다음과 같은 암호문을 전송한다.

$\{E_{SK}(M) \| g^k \bmod p \| SK \cdot P_R{}^k \bmod p\}$ (10)

(M : 메시지, k : 랜덤수, P_R : 수신자의 공개키, SK : 데이터 암호화키)

(2) 복구 기관은 수신자의 메시지로부터 메시지 헤더 분석이나 발신지 추적과 같은 기존의 기술을 이용하여 신원을 추출하고, 자신의 데이터베이스에서 신원과 연결된 ID_R을 검색한다.

(3) RC는 검색된 ID_R과 g^k를 위탁기관 T_i에게 전송한다.

(4) RC는 ID_R과 g^k를 KDC에게 전송한다.

(5) 각 위탁기관들은 수신된 g^k에 자신이 보관하고 있는 수신자의 비밀키의 일부인 S_{RAi} 값을 다음과 같이 RC에게 전송한다.

$$(g^k)^{S_{RAi}} \bmod p \quad (11)$$

(6) KDC는 수신된 ID_R에 해당하는 S_{RB}값을 수신된 g^k에 승산해서 RC에게 전송한다.

$$(g^k)^{S_{RB}} \bmod p \quad (12)$$

(7) 복구기관은 3, 4 단계에서 수신된 정보를 조합해서 다음 값을 계산한다.

$$\begin{aligned} S' &= g^{kS_{RA1}} \cdot g^{kS_{RA2}} \cdots g^{kS_{RAn}} \cdot g^{kS_{RB}} \\ &= g^{k(S_{RA1} + S_{RA2} + \ldots + S_{RAn})} \cdot g^{S_{RB}} \\ &= g^{k(S_{RA} + S_{RB})} = g^{S_R} = P_R^k \end{aligned} \quad (13)$$

(8) 복구 기관은 (7)에서 얻은 값으로 (1)의 암호문에서 데이터 암호화키 SK를 얻을 수 있다.

$$\{SK \cdot P_R^k\}/S'$$
$$= \{SK \cdot P^k{}_R\} / P^k{}_R \bmod p = SK \quad (14)$$

V. 요구사항 고찰 및 분석

본 장에서는 제안된 시스템이 2장에서 기술한 요구조건에 일치하는지 살펴보고, 기존 시스템과의 문제점을 비교한다.

■ 조건 1)에 대한 고찰

본 시스템에서 송신자가 수신자에게 전송하는 암호문은 다음과 같다.

$$E_{SK}(M) \| g^k \bmod p \| SK \cdot P_R{}^k \bmod p \quad (15)$$

(SK : 데이터 암호화키, P_R : 수신자의 공개키)

이러한 암호문은 일반적인 ElGamal 암호 시스템과 완전히 동일하다. 따라서 제안된 시스템의 키 위탁 및 복구 과정만 안전하다면 일반적인 암호 시스템과 동일한 안전성을 갖는다는 것을 알 수 있다.

■ 조건 2)에 대한 고찰

제안 방식에서는 사용자의 비밀키를 여러개로 분할해서 위탁한다. 위탁 과정에서 위탁기관 T_i들이 분할된 정보의 유효성을 확인하고, 다시 RC가 전체 정보의 유효성을 확인하므로 위탁된 정보가 사용자가 받은 인증서에 대응되는 비밀키임을 확신할 수 있다. 따라서 유사시 복구가 가능하다.

■ 조건 3)에 대한 고찰

복구 과정에서 키의 보관자인 T_i들과 KDC는 g^k에 각자 자신이 소유한 비밀값 S_{Ai}와 S_B를 승산해서 반환한다. 이때 복구기관은 g^k을 알지만 이

표 2. 각 방식의 요구사항 만족도
Table 1. Comparison of each scheme

	Fair Cryptosystem	Clipper Chip	Fail Safe Key Escrow	Blind Decoding	제안 방식
소프트웨어 구현	○	×	○	△	○
Shadow public key resistance	×	○	○	○	○
사용자의 키 생성	○	×	○	×	○
도청기한 제한	×	×	×	○	○
Trustee들의 공모	가능	가능	가능	불가능	불가능

산 대수의 어려움에 근거해서 복구기관 RC는 사용자의 비밀키를 알 수 없다. 또한 k는 각 세션에 따라 랜덤하게 선택되는 수이므로 복구 기관은 항상 데이터 암호화 키인 SK 만을 얻을 수 있다. 따라서 복구 기한은 각 세션별로 제한될 수 있다.

■ 조건 4)에 대한 고찰

먼저 사용자는 적절한 수의 위탁 기관들이 정보를 공개하기 전까지는 자신의 메시지가 노출되지 않는다는 것을 확신해야 한다. 각 위탁 정보는 여러 개의 위탁기관에 분산되어 있기 때문에 메시지가 노출되지 않는다는 것을 확신할 수 있다. 먼저 적법하지 않은 상황에서 키를 복구 하기 위해서 T_i들이 공모하는 경우, 키의 일부인 P_A 부분만 복구를 할 수 있다. 따라서 T_i간의 공모 만으로는 키의 복구가 불가능하다. 만약 T_i들과 KDC가 공모하더라도, KDC는 RC의 협조 없이 키의 ID를 알 수 없기 때문에 키의 복구가 불가능하다. T_i들과 RC가 공모하더라도 KDC가 저장하고 있는 키의 S_B 부분을 얻을 수 없기 때문에 불법적인 키의 복구가 불가능해 진다. 따라서 본 시스템은 각 기관들의 공모 위협으로부터 안전하다고 할 수 있다.

■ 조건 5)에 대한 고찰

사용자는 자신의 비밀키 S_A를 랜덤하게 선택한다. 또한 KDC가 선택한 S_B는 사용자의 키 S_A가 제출되기 전에 선택되었다. 따라서 사용자는 S= S_B+ S_A가 랜덤하다는 것을 확신할 수 있다. KDC는 자신이 S_B를 랜덤하게 선택하였으며, 사용자의 키 S_A는 사용자가 S_B를 선택하기 전에 위탁되므로 S= S_A+ S_B가 랜덤하다는 것을 확신할 수 있다. 따라서 사용자와 복구 기관 모두 키가 랜덤하게 선택되었다는 것을 확신할 수 있으며, 사용자의 Shadow 공개키 생성을 방지할 수 있다.

■ 조건 6) ~ 8)에 대한 고찰

제안 방식은 전체 기능이 소프트웨어로 구성될 수 있으며, 비밀 정보를 사용하지 않고 기존의 공개키 암호에 기반하고 있기 때문에 알고리즘이 공개되어도 시스템의 보안에 영향을 미치지 않는다. 또한 제안 방식에서는 사용자의 잘못된 키 위탁, Shadow 공개키 생성 등을 방지할 수 있으므로 사용자가 시스템을 오용할 수 없다.

[표 1]은 제안된 방식을 타 방식과 비교한 것이다. 제안 방식에서는 사용자가 키 생성에 참여함으로써 자신의 키가 랜덤하게 선택되었다는 것을 확신할 수 있다. 클리퍼 칩이나 Blind Decoding을 사용한 방식에서는 키를 전적으로 복구 기관들이 생성하기 때문에 사용자는 자신의 키가 랜덤하게 선택되었다는 것을 확신하지 못하며 전체 시스템에 대한 신뢰를 가질 수 없다.

또한 복구 기관의 메시지 복구는 각 메시지 단위로 이루어지므로 복구기한이 각 세션별로 제한될 수 있다. 또한 사용자의 키 부분을 보관하고 있는 KDC가 사용자의 신원과 자신이 가지고 있는 키의 부분

을 연결시킬 수 없으므로 KDC는 RC의 도움 없이 키 정보를 노출시킬 수 없다.

VI. 결론

암호가 널리 보급될수록 암호를 이용한 범죄나 오남용 사례가 증가할 것으로 판단된다. 따라서 이러한 위협에 대해서 적절하게 대처할 수 있는 능력이 필요하며, 그 대안으로 키 위탁 암호 시스템이 현재 가장 유력하게 떠오르고 있다.

키 위탁 시스템은 기존의 암호 시스템에 메시지를 복구할 수 있는 능력을 부가하는 것으로 설계시에는 항상 사용자의 사생활 보호와 키 복구의 확실성에 대한 보장이 이루어져야 한다.

본 논문에서는 키 위탁을 운영하는 기관들의 공모나 불법적인 키 획득을 통해 사용자의 통신을 복구하는 것을 방지함으로써 사용자의 사생활을 보호하고, 또한 사용자가 키 생성을 통해 숨겨진 채널을 구성하는 등 키 위탁 시스템을 오용하려는 시도를 차단할 수 있는 새로운 키 위탁 시스템을 제안하였다.

향후 연구과제로는 제안된 시스템에는 포함되지 않은 암호화된 메시지의 키의 식별, 메시지의 소유자 식별을 위한 키 복구 영역에 관한 내용이 남아 있다. 키 복구 영역은 일반적으로 메시지 암호화 키의 위탁 여부, 메시지 송수신자 식별 등을 위해 사용된다. 하지만 복구 영역이 추가되는 경우 호환성의 문제가 발생할 수 있고 복구 영역을 위변조 할 수 없도록 하는 기술적 대비책이 필요하다. 향후 호환성 문제와 위/변조 문제를 해결할 수 있는 방안이 연구된다면 키 위탁 시스템이 보다 효율적으로 동작할 수 있을 것이다. 또한 암호문의 소유자 및 키 식별 문제도 연구되어야할 과제이다.

참 고 문 헌

[1] T. ElGamal, "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms.", IEEE Transacions on Information Theory, pp. 469-472, 1985.

[2] Silvio Micali, "Fair Cryptosystems", Advances in Cryptology-CRYPTO '92, pp. 113-138, 1992

[3] Joe Kilian and Tom Leighton, "Fair Crypto systems, Revisited" CRYPTO 95, pp 208-221, 1995

[4] Yair Frankel and Moti Yung, "Escrow Encryption Systems Visited: Attacks, Analysis and Designs", CRYPTO 95, pp. 222-235, 1995

[5] Arjen K. Lenstra, Peter Winkler and Yacov Yacobi, "A Key Escrow System with Warrant Bound", CRYPTO 95, pp. 197- 207, 1995

[6] Dorothy E. Denning and Miles Smid, "Key Escrowing Today", IEEE Communications, Vol. 32, pp. 58-68, 1994

[7] David Paul Maher, "Crypto Backup and Key Escrow", Communications of the ACM, volume. 39, pp. 48-53, 1996

[8] Dorothy E.Denning, "A Taxonomy for Key Recovery Encryption System", Communica tions of the ACM, Vol. 39, pp. 34-40, 1996

[9] Kouichi Sakurai, Yoshinori Yamane, "A key escrow system with Protecting User's Privacy by Blind Decoding", 1998

[10] 이임영, 채승철, "Key recovery 시스템에 관한 고찰", 한국통신정보보호 학회지, 제 7권 4호, pp. 45-58, 1997.

[11] 최용락, 소우영, 이재광, 이임영, "통신망 정보보호", 도서출판 그린, 1996

□ 著者紹介

채승철(Seung-chul Chae) **학생회원**

1997년 8월 순천향대학교 전산학과 졸업

1997년 8월~현재 순천향대학교 전산학과 석사과정

<관심분야>

암호 이론, 컴퓨터 보안

이임영(Im-yeong Lee) **정회원**

81년 8월 홍익대학교 전자공학과 졸업
86년 3월 오사카대학 통신공학과 석사
89년 3월오사카대학 통신공학과 박사
89년 1월~94년 2월 한국전자통신연구원 선임연구원
94년 3월~현재 : 순천향대학교 컴퓨터학부 교수

<관심분야>
암호이론, 정보이론, 컴퓨터 보안