사용자만이 비밀키를 복구할 수 있는 키 복구 시스템

최 희 봉[†]·유 회 종[††]·오 수 현[†††]·원 동 호[††††]

요 약

1998년 A. Young 등은 공개키 기반구조(PKI)를 이용한 키 복구 시스템인 ARC를 제안하였다. 그리고 1999년 P. Paillier 등은 ARC를 개선하여 사용자의 증명서 저장공간이 필요없는 SE-PKI 키 복구 시스템을 제안하였다. 또한 2001년 유회종 등은 Paillier가 제안한 SE-PKI 키 복구 시스템에 비밀분산 개념을 추가하여 다수의 키 위탁 기관이 참여하는 키 복구 시스템을 제안했다. 본 논문에서는 새로운 scheme을 추가하여 사용자의 비밀키를 사용자만이 인증기관의 도움을 받아 키 위탁 기관으로부터 온 라인 상에서 안전하게 복구할 수 있는 키 복구 시스템을 제안한다. 이 키 복구 시스템에서는 사용자가 비밀키를 자주 변경하는 경우 이전 암호문을 복호화하기 위해 필요한 비밀키 관리가 용이하다.

The Key Recovery System for Users to Recover Their Own Secret Key

Hee-bong Choi[†]·Hui-jong Yu[††]·Soo-hyun Oh[†††]·Dong-ho Won[††††]

ABSTRACT

In 1998, A. Young and M. Yung introduced the concept of ARC that conjugates functionalities of a typical PKI with the ability to escrow private keys of the system users. And in 1999, P. Paillier and M. Yung proposed a new notion - called SE-PKI - which presents other additional advantages beyond ARC. Also in 2000, Hi-Jong, Yu et al proposed the key recovery system with multiple escrow agents which has shared secret theory. In this paper, we propose the key recovery system that can recover the secret key of users on on-line communication. Only a user can recovery private key at the system. In case of changing frequently user's secret key, it is easy to manage the secret key in order to decrypt ciphertexts in the key recovery system

키워드 : 공개키 기반 구조(Public Key Infrastructure), 키복구 시스템(Key Recovery System), 사용자 비밀키(Secret Key of Users)

1. 서 론

현대 사회가 점차 고도의 정보화 사회로 발전해 가면서 다양한 정보의 개방과 공유, 네트워크를 통한업무 처리의 일반화는 무한한 가능성과 편리함을 주었지만 정보의 침해라는 문제를 발생시켰다. 이로 인하여 정보 보호의 문제가 부각되었으며, 이전에 군사상의 목적 등 국가적 차원에서 주로 이용되었던 암호의 사용이 민간 부문으로 확대되었다.

암호를 사용함으로써 다음과 같은 문제점이 발생하게 되었다.

첫째, 키의 분실이나 손상으로 인하여 사용자가 자신의 정보에 접근할 수 없는 경우이다. 이 경우 적법한 키의 소유자라고 할지라도 자신의 정보에 대한 접근을 할 수 없으므로 많은 손실을 가져올 수 있다.

†정 회 원 : 국가보안기술연구소
††정 회 원 : 한국전자통신연구원
†††준 회 원 : 성균관대학교 대학원 전기전자컴퓨터공학부
††††종신회원 : 성균관대학교 전기전자컴퓨터공학부 교수

논문접수 : 2000년 10월 18일, 심사완료 : 2001년 3월 23일

둘째, 국가가 범죄수사 등의 합법적인 이유로 키에 접근해야 할 필요성이 있을 경우 발생되는 문제점이다. 암호는 키를 아는 사람만이 암호문을 복호할 수 있는 기밀성 기능을 포함하고 있기 때문에, 범죄자들은 암호를 사용함으로써 합법적인 수사를 방해할 수 있다

셋째, 암호가 오용됨으로써 발생할 수 있는 잠재적 위협이 존재하는 경우로, 사업장에서 피고용인이 중요한 정보를 암호화하고 키를 담보로 금품을 요구할 수도 있으며, 키의 도난이나 손상 등의 위협이 항상 존재한다.

위에서 살펴본 바와 같이 키 복구는 암호의 대중화와 함께 발생한 여러 역기능을 해결할 수 있는 방법이다. 그러나 키 복구 방식을 현재의 암호 사용자 환경에 무조건 도입하는 것은 시간과 비용, 방법적인 측면에서 무리가 있다.

현재 각 나라에서는 전자 상거래와 같은 암호 응용 분야에서 유용하게 사용될 수 있는 공개키 기반구조 구축이 진행되고 있으며 여러 선진국에서는 이미 구축이 되어 서비스가 진행중이다. 따라서 이러한 공개키 기반구조에 키 복구 방식을 대입하는 것은 효율적인 방법이라 할 수 있으며

또한 현재까지 이러한 연구가 상당히 이루어진 상태이다.

1998년 A. Young과 M. Yung은 공개키 기반구조를 이용하여 다수의 위탁 기관이 참가 가능한 키 복구 시스템(ARC, Auto-Recoverable Auto-Certifiable Cryptosystem)을 제안하였다[1]. 사용자는 위탁 기관의 공개키를 이용하여 자신의 비밀키를 생성하고 이를 인증 기관에 증명함으로써 공개키에 대한 인증서를 발급 받는다. 1999년 P. Paillier와 M. Yung은 ARC를 개선시킨 키 복구 시스템 SE-PKI(Self- Escrowed Public Key Infrastructure)를 제안하였다[2]. 2001년 유희종, 최희봉, 정찬주, 원동호는 키 복구 시스템 SE-PKI에 비밀분산개념과 안전한 키 복구 기능을 도입한 공개키 기반 구조와 연동 가능한 키 복구 시스템을 제안하였다[8]. 본 논문에서 유희종 등의 키 복구 시스템에 새로운 Scheme을 추가하여 사용자의 비밀키를 사용자만이 신뢰기관의 도움을 받아 키 위탁 기관으로부터 온 라인 상에서 안전하게 복구할 수 있는 키 복구 시스템을 제안한다. 다수의 위탁기관이 참여된 키 복구 시스템에 새로운 scheme을 추가함으로써 모든 키 위탁기관 중 한 개 이상이 공모하지 않는다는 조건에서 사용자의 비밀키가 사용자 외에 다른 기관에 누출됨이 없이 자신의 키를 잃어버린 사용자의 비밀키를 복구할 수 있는 시스템이다. 이 키 복구 시스템은 비밀키를 자주 변경하는 사용자에게 비밀키 관리가 용이하다. 예를 들면 장기로 보관하는 문서를 암호화한 경우 비밀키를 변경할 때 암호화된 문서를 다시 복호화하여 새로운 공개키로 암호화할 필요 없이 처음에 암호화한 공개키를 함께 보관하면 되기 때문에 편리하다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문의 배경이 되는 PKI와 연동하는 키 복구 시스템에 대하여 알아보고 3장에서는 다수 키 위탁기관이 참여 할 수 있도록 비밀분산 개념을 키 복구에 적용한 시스템을 설명하고 4장에서는 사용자의 비밀키를 사용자만이 복구할 수 있는 키 복구 Scheme을 제안한다. 5장에서는 제안 키 복구 시스템의 안전성과 효율성에 대하여 논의하며 6장은 결론으로 구성되어 있다.

2. 배 경

PKI와 연동하는 키 복구 시스템에 대해 일반적인 사항을 설명한다. 먼저 신뢰 위탁 기관들이 마스터 비밀/공개키를 생성하여 마스터 공개키를 공개하면 사용자는 이것을 이용하여 자신의 비밀/공개 키 쌍을 생성한다. 그리고 사용자는 자신의 키 쌍이 위탁 기관의 마스터 비밀키를 사용하여 복구 가능하다는 것을 증명하는 증명서P와 자신의 공개키를 인증기관에 전달한다. 인증기관은 전송 받은 P를 검증하고 이 검증에 성공하면 사용자의 공개키에 대한 인증서를 발급한다.

차후에 법 집행 기관이나 사용자의 키 복구는 위탁 기관들의 마스터 비밀키를 사용함으로써 이루어진다. 시스템의 자세한 과정은 2.1절에 나타내었고 SE-PKI 시스템은 2.2절에 설명하였다.

2.1 PKI와 연동 가능한 키 복구 시스템

$S=\langle G^S, E^S, D^S\rangle$는 사용자의 키 생성 알고리즘, 암/복호 알고리즘이며 $T=\langle G^T, E^T, D^T\rangle$는 위탁 기관의 마스터 키 생성 알고리즘, 암/복호 알고리즘이다.

2.1.1 시스템 설정

위탁 기관들은 G를 사용하여 마스터 비밀/공개 키 (X, Y)를 생성한다.

2.1.2 키 생성과 공개 위탁 검증

각 사용자는 $G(Y,1^k)$ 알고리즘을 이용하여 공개/비밀키 쌍 (x, y)를 생성하고 $y=E_X^T(x)$를 만족하는 증명서 P를 공개키 y와 함께 인증기관에 전송한다.

인증기관은 P를 검증하여 검증에 성공하면 사용자의 공개키 y에 대한 인증서를 발급한다.

2.1.3 암/복호화

사용자는 인증기관으로부터 수신자의 공개키와 인증서를 얻은 후 이를 검증하고 검증이 성공하면 전송하고자 하는 메시지를 수신자의 공개키로 암호화하여 전송한다.

수신자는 비밀키를 이용하여 암호문을 복호화한다.

2.1.4 키 복구

어떤 사용자에 대하여 키 복구가 인증되면 위탁 기관은 자신의 비밀정보를 이용하여 $x=D_X^T(y)$를 계산하여 사용자의 비밀키를 복구해낸다.

2.2 SE-PKI(Self-Escrowed Public Key Infrastructure)

SE-PKI에서 마스터의 암호방식은 1999년 제안된 P. Paillier의 확률론적 암호방식을 결정적 암호방식으로 수정하여 사용하고[5], 사용자의 암호 방식으로는 Diffie-Hellman방식과 ElGamal의 방식을 이용한 시스템을 사용했다.

본 장에서는 SE-PKI를 소개하고자 한다. (1)절에서는 P. Paillier의 확률론적 암호방식을 결정적 암호방식으로 수정한 마스터 암호방식을 설명하고, (2)절에서는 시스템 설정, 암/복호 프로토콜을 설명하고 (2)절에서는 키 복구가 가능한지를 증명하는 증명서를 생성하고 이를 검증하는 프로토콜을 설명한다.

2.2.1 P. Paillier의 암호방식

P. Paillier가 n이 RSA modulus $n=pq$인 $Z_{n^2}^*$상에서 연산되는 공개키 확률 암호 scheme을 제안하였다[2]. n의 소인수를 알면 discrete logarithms modulo n^2을 빨리 계산할

수 있다. 여기서 밑수 $g \in Z^*_{n^2}$는 $\gcd(n, \alpha) = 1$를 만족하는 어떤 α에 대하여 $n\alpha$라는 위수를 갖는다.

$\lambda = \lambda(n) = \mathrm{lcm}(p-1, q-1)$가 n과 서로 소인 조건에서 최대 위수 $n\lambda$를 g로 선택한다.

상에서 다음 정수 값인 $L(u) = (u-1)/n$을 갖는 함수를 정의한다. 여기서 나눗셈은 Z에서 닫혀있다. 이 때 공개키는 (n, g)쌍이고 개인키는 λ혹은 인수 p, q이다.

$m < n$인 $U_n = \{u < n^2 | u = 1 \bmod n\}$평문의 암호화는 다음과 같다. $[0, 2^l]$에서 랜덤하게 정수 r를 선택한다. 여기서 l은 n의 비트 길이를 나타낸다. 암호화는

$$c = g^{m+nr} \bmod n^2$$

로 표시된다.

복호화은

$$m = \frac{L(c^{\lambda} \bmod n^2)}{L(g^{\lambda} \bmod n^2)} \bmod n$$

로 표시된다.

이 암호시스템을 결정적 암호시스템으로 변환한다.

메시지 $m < n$의 암호화는 단순한 연산을 적용하여 $c = g^m \bmod n^2$으로 수행되고, 복호화는 방정식(2)에 의하여 수행된다. 이에 대한 보안수준은 감소될 지 몰라도 비밀요소를 알지 못하고서 이산대수를 풀 지 못한다[7]. <표 1>에서 이 결정적 암호시스템을 표시한다.

〈표 1〉 SE-PKI의 암/복호 과정

공 개 키	n, g(최대위수)
비 밀 키	$\lambda = \mathrm{lcm}(p-1, q-1)$
암 호 화	평 문 $m < n^2$ 암호문 $c = g^m \bmod n^2$
복 호 화	암호문 $c < n^2$ 평 문 $m = \frac{L(c^{\lambda} \bmod n^2)}{L(g^{\lambda} \bmod n^2)} \bmod n$

2.2.2 시스템 설정과 암/복호 프로토콜

본 절에서는 SE-PKI에서 제안된 두 방식에 대하여 설명한다. 위탁 기관의 마스터 암호 방식으로는 1999년 제안된 P. Paillier의 확률론적 암호 방식을 결정적 암호방식으로 수정하여 사용하고[5], 사용자가 수행하는 방식으로는 Diffie-Hellman 방식과 ElGamal의 방식을 이용한 시스템을 소개한다.

첫번째 방식인 Diffie-Hellman방식은 <표 2>에 나타낸 것과 같이 사전에 n을 RSA modulus로 두고 $n = pq$인 그룹 $Z^*_{n^2}$을 설정하여 키 위탁기관이 p, q를 가지면 $x < n$인 이산대수 $y = g^x \bmod n^2$를 수행하여 복구할 수 있다.

〈표 2〉 SE-PKI의 Diffie-Hellman 키 설정 과정

설 정	키 위탁기관은 $n = pq$를 생성하고 n을 공개함.
프로토콜	1. 사용자A는 난수 $a < n$를 생성하 여 사용자B에게 $g^a \bmod n^2$를 송 신한다. 2. 사용자B는 난수 $b < n$를 생성하여 사용자A에게 $g^b \bmod n^2$를 송 신한다. 3. 양쪽에서를 $K = g^{ab} \bmod n^2$계산 한다.
키 복 구	키 위탁기관은 g^a에서 a를 계산하고 g^b에서 b를 계산하여 K를 복구

둘째 방식인 ElGamal의 방식에 대한 기본적인 암/복호 과정은 다음과 같으며 전체 과정을 <표 3>에 나타내었다.

〈표 3〉 SE-PKI의 ElGamal 암/복호 과정

마스터 공개키	n, g, $l = 2\|n\|$
마스터 비밀키	lcm(p-1, q-1)
사용자의 공개키	$y = g^x \bmod n^2$ where $x <_R n$
사용자의 비밀키	$x < n$
암 호 화	plaintext $m < n^2$ ciphertext $c = (my^k, g^k)$ where $k <_R 2^l$
복 호 화	ciphertext $c = (a, b)$ plaintext $m = a/b^x \bmod n^2$

① 위탁 기관은 마스터 비밀/공개키를 생성한다. P. Paillier의 프로토콜에 따라 마스터 공개키는 $n = pq$, $g \in Z^*_{n^2}$($\gcd(n, \alpha) = 1$인 어떤 α에 대한 $n\alpha$를 위수로 갖음), $l = 2|n|$이며 마스터 비밀키는 lcm(p-1, q-1)이다. 위탁 기관은 마스터 공개키를 공개한다.

② 사용자는 공개된 마스터 공개키를 이용하여 자신의 비밀/공개키 쌍을 생성한다. 사용자의 비밀키는 $x < n$, 공개키는 $y = g^x \bmod n^2$이며 인증기관에게 키 복구 가능한 공개키에 대한 인증서를 발급 받으면 암호 통신을 할 수 있다. 여기서 인증기관의 키 복구 가능한 공개키 검증은 ZKIP에 의한다[5].

③ 사용자들이 암호 통신을 할 때에 이용되는 암호 방식은 ElGamal 방식이며 평문이 $m < n^2$이면 암호문은 $c = (my^k, g^k)$, $k <_R 2^l$이며 이를 전송 받은 측에서는 암호문을 $c = (a, b)$로 설정하고 $m = a/b^x \bmod n^2$을 계산하여 복호한다.

3. 다수 키 위탁기관의 참여 가능한 키 복구 시스템

위탁 기관들의 키 생성 시 마스터 비밀키의 분산을 위해서 1997년 D. Boneh등이 제안한 방법[3]을 사용하여 SE-PKI에 적용한 것은 유희종 등의 논문[8] 참고하면 된다. 이 논문에서 간단하게 설명한다.

3.1 시스템 설정

위탁기관의 마스터 공개키는 n(=pq), g, 마스터 비밀키는 $\varphi(n)$이다. 위탁 기관들이 나누어 갖게 되는 비밀키는 $\varphi(n)$ 값이다. 두 위탁 기관 A와 B에 대하여 각각 φ_a, φ_b를 나누어 갖게 되는 것이다[3].

이에 대한 설명은 다음과 같다.

$$n = pq = (p_a + p_b)(q_a + q_b)$$
$$\varphi(n) = (n - p_a - q_a) - (p_b + q_b)$$

즉, A는 φ_a를 가지고, B는 φ_b를 가지게 되는 다음의 형태로 비밀키의 분산이 가능하게 된다.

$$\varphi(n) = (n - p_a - q_a) - (p_b + q_b) = \varphi_a + \varphi_b$$

3.2 암/복호화

위탁 기관들의 암/복호 알고리즘은 1999년 P. Paillier가 제시한 알고리즘이며 사용자들의 암/복호 알고리즘은 ElGamal 암호 알고리즘이다[6].

암/복호화 과정을 <표 4>에 정리하였다.

〈표 4〉 다수의 키 위탁기관이 참여한 키 복구 시스템의 암/복호화 과정

마스터 공개키	n, g, $l = 2\|n\|$
마스터 비밀키	$\varphi(n) = \varphi_a + \varphi_b$
사용자의 공개키	$y = g^x \bmod n^2$ where $x <_R n$
사용자의 비밀키	$x < n$
암 호 화	plaintext $m < n^2$ ciphertext $c = (my^k, g^k)$ where $k <_R 2^l$
복 호 화	ciphertext $c = (a, b)$ plaintext $m = a/b^x \bmod n^2$

3.3 키 복구

키 복구 과정에서는 먼저 위탁 기관 A와 B가 각각 φ_a와 φ_b를 이용하여 $g^{\varphi_a}, y^{\varphi_a}$와 $g^{\varphi_b}, y^{\varphi_b}$를 계산하여 키 복구 수행자에게 전송한다.

$$\frac{L((g^x)^{\phi_a} \cdot (g^x)^{\phi_b}) \bmod n^2}{L(g^{\phi_a} \cdot g^{\phi_b})} \bmod n$$
$$= x$$

일반적으로 임의의 참여기관 수만큼 비밀분산이 성립하기 때문에[3] i개 참여한 위탁기관에 대해서도 성립한다[8].

$$n = pq$$
$$= (p_1 + p_2 + \ldots + p_i)(q_1 + q_2 + \ldots + q_i)$$
$$\varphi(n) = (n - p_1 - q_1) - (p_2 + q_2) - \ldots - (p_i + q_i)$$
$$\varphi(n) = \varphi(1) + \varphi(2) + \ldots + \varphi(i)$$

이러한 다수의 위탁기관 참여 가능한 SE-PKI 키 복구 시스템[8]에 새로운 Scheme을 추가함으로써, 사용자의 비밀키를 키 위탁 기관은 물론 다른 기관에 노출시키지 않고 온 라인 상에서 사용자만이 복구할 수 있음을 다음 장에서 설명한다.

4. 사용자만이 비밀키를 복구할 수 있는 키 복구 Scheme 제안

만약 어떤 사용자가 비밀키를 분실한다면 암호화된 데이터를 사용하지 못하므로 분실 한 사용자의 비밀키 복구해야 한다. 이 때 사용자의 비밀키를 키 위탁기관은 물론 다른 기관에 노출하지 않고서도 키 복구를 수행한다면 사용자는 안심하고 비밀키를 복구하여 재 사용할 수 있다. 여기서 제안하는 Scheme은 정당한 법 집행관에 의해 사용자의 비밀키를 복구하는 목적이 아니고 사용자가 자신의 비밀키를 복구하는 시스템이다.

본 논문에서 제안하는 키 복구 Scheme은 유희종 등의 키 복구 시스템[8]에 암호 프로토콜을 추가한 것이다. 즉 시스템 설정과 PKI 운용시에는 유희종 등의 키 복구 시스템과 같지만 키 복구 기능을 수행할 때는 다르다. 인증기관은 이 논문에서 제안한 암호 프로토콜을 이용하여 사용자를 인증하고 키 복구용 공개키를 검증하여 인증서을 발행하고, 이 인증서로 사용자와 키 위탁 기관간에 통신하여 사용자의 비밀키를 복구할 수 있게 된다. 여기서 인증기관은 시스템 설정 시에 이용한 인증기관과 같다.

다음은 운용중인 사용자가 자신의 비밀키를 분실했을 경우 키 복구용 공개키/비밀키 생성과 인증기관의 인증, 키 위탁기관에 키 복구 의뢰, 키 복구까지의 과정을 다음에서 설명한다.

지금 현재 키 위탁에 참여중인 키 위탁기관이 i개라 하면 각 키 위탁 기관이 마스터의 비밀키를 아래와 같이 분산되어 갖고 있다.

$$\varphi(n) = (n - p_1 - q_1) - (p_2 + q_2) - \ldots - (p_i + q_i)$$
$$\varphi(n) = \varphi(1) + \varphi(2) + \ldots + \varphi(i)$$

키복구과정은 다음과 같은 순서로 수행된다.

① 자신의 비밀키 복구가 필요한 사용자는 $G(Y,1^k)$ 알고리즘을 이용하여 새로운 공개/비밀키 쌍 (x',y')를 생성하고 $y'=E_Y^T(x')$를 만족 하는 증명서P를 공개키 y'와 함께 인증기관에 전 송한다.

② 인증기관은 증명서P를 받아 복호화하여 검증하 고 새로 생성한 사용자의 공개키 y'에 대한 인증 서를 발급한다.

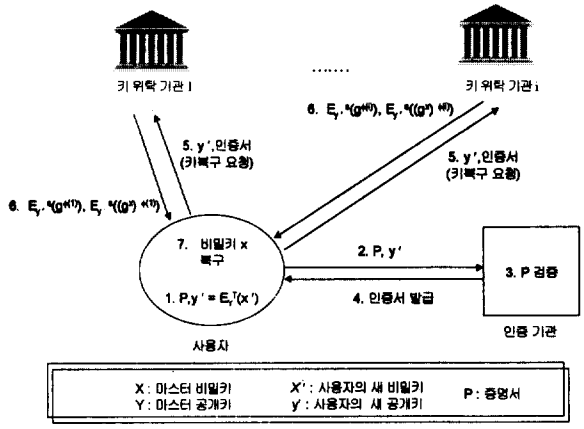
③ 사용자는 공개키 y', 공개키 인증서 및 키 복구 요청서를 모든 키 위탁기관에 보낸다.

④ 각 키 위탁 기관은 사용자의 공개키 인증서를 검 증하여 성공하면 사용자의 공개키로 다음 데이터 $g^{\phi(i)},(g^x)^{\phi(i)}$를 $E_{y'}^s(g^{\phi(i)}),E_{y'}^s((g^x)^{\phi(i)})$와 같이 암호화하여 키 복구를 요청한 사용자에게 보낸다. 여기서 x는 사용자가 복구하고자 하는 비밀키이다.

⑤ 사용자는 모든 키 위탁 기관으로부터 수신한 데이 터를 복호화하여 P. Paillier 암호방식을 수행하여 분실한 사용자의 비밀키를 복구한다.

$$\frac{L((g^x)^{\phi(1)}\cdot(g^x)^{\phi(2)}\cdot\cdot(g^x)^{\phi(i)})\bmod n^2}{L(g^{\phi(1)}\cdot g^{\phi(2)}\cdot\cdot g^{\phi(i)})\bmod n^2}\bmod n$$
$$=\frac{L((g^x)^{\phi(n)}\bmod n^2)}{L(g^{\phi(n)}\bmod n^2)}\bmod n$$
$$=x$$

위의 과정을 (그림 1)에 나타내었다. (그림 1)은 사용자의 비밀키를 복구하기 위한 과정이 온라인 통신상에서 이루어지도록 한 것이다. 여기서 키 복구용 비밀키/공개키 쌍을 생성할 때에도 키 복구가 가능하도록 하여 비밀키/공개키 쌍을 사용자가 여러 번 독자적으로 생성하여 사용할 수 있도록 하였다.



(그림 1) 사용자의 비밀키 분실후 사용자에 의한 키 복구 과정

사용자는 자신만이 사용하는 공개키를 생성하여 사용할 수 있다. 예를 들면 장기적으로로 저장하는 대량의 문서를 암호화하여 보관하는 경우 비밀키가 변경되는 경우 다시 새로운 비밀키/공개키로 복호화 및 암호화를 수행해야 하나 이전 공개키와 함께 문서를 보관하면 필요할 때 키 복구 프로토콜을 이용하여 비밀키를 복구 받아 복호화할 수 있다. 즉 사용자가 자신의 비밀키를 신뢰성 높은 시스템인 키 위탁 기관에 위임하고 필요할 때 새로운 비밀키/공개키 쌍을 생성하여 다시 복구 받을 수 있다. 여기서 키 위탁 기관은 사용자의 비밀키를 모두 관리할 필요가 없으며 자신의 마스터 비밀키만 관리하고 있으면 된다.

키 복구할 때 반드시 사용자의 인증이 필요하기 때문에 인증기관으로부터 인증서를 받는 프로토콜을 추가함으로써 키 복구를 원하는 사용자를 인증할 수 있다.

5. 안전성과 효율성

제안하는 키 복구 시스템은 시스템 설정 과정과 시스템 운영과정, 비밀키 복구과정으로 나누어 안전성과 효율성을 설명한다.

5.1 안전성

시스템 설정 과정에서는 유회종 등의 키 복구 시스템과 같다. 키 복구의 안전성을 위한 복구 가능한 공개키 인증과 안전한 비밀분산 프로토콜이 필요하다. 여기서 사용자가 생성한 공개키를 키 복구 가능한가 검증하는 인증기관은 키 위탁 기관의 마스터 비밀키를 알지 않고 검증해야 하기 때문에 ZKIP를 사용하였다. 안전한 비밀분산은 참고문헌[3]에 따른다.

시스템 운영할 때 일반 PKI과 같은 프로토콜을 사용함으로써 연동할 수 있는 PKI의 안전성에 의존하게 된다.

비밀키 복구할 때 공개키 소유자만이 비밀키를 복구해야 하기 때문에 사용자의 인증이 필수적이다. 또한 새로 생성한 비밀키 복구용 공개키가 키 복구 가능한가 검증도 받아야 한다. 이것은 시스템 설정할 때 이용한 인증기관의 도움을 받는다. 사용자의 비밀키를 키 위탁 기관은 물론 다른 기관에서 알지 못하도록 하는 안전성은 공모하지 않은 다수의 위탁 기관과 복구과정에서의 이산대수의 어려움, 안전한 복구 프로토콜으로 확보할 수 있다.

일반적인 키 복구 시스템의 안전성인 Squeezing 공격에는 사용자 인증에 의해 보호할 수 있고, 대량 감청 문제 대해서는 비밀키 소유자만이 비밀키를 복구함으써 해당 사항이 없다.

5.2 효율성

시스템의 설정할 때 PKI를 이용한 키 복구 가능한 공개키 인증과 비밀 분산을 위한 프로토콜이 있다. 이것은 유회종 등의 키 복구 시스템과 같다.

시스템 운용할 때 일반 PKI와 똑 같이 운용되므로 효율성이 일반 PKI와 같다

비밀키 복구할 때 사용자와 인증기관간에 PKI를 이용한 키 복구 가능한 공개키 인증 프로토콜이 추가되고, 키 위탁 기관에서는 마스터 키로 복구하고자 하는 공개키를 멱승하는 과정과 이 정보를 전달하기 위한 암호화 과정이 수행되고 사용자는 키 복구용 정보를 복호화하는 과정과 P. Pailler 암호방식을 이용하여 자신의 비밀키를 복호화하는 과정이 추가된다.

제안한 시스템은 비밀키 복구할 때 암호화/복호화 연산과 검증 및 인증서 발행, 추가된 통신 프로토콜로 인해 효율성이 떨어지나 이것은 사용자의 비밀키 복구 보호를 위한 대책이다. 그리고 사용자의 비밀키를 신뢰성 높은 시스템에 보관하는 할 수 있는 장점이 있다. 또한 키 복구시에 PKI 시스템에 있는 인증기관을 그대로 이용함으로써 새로 인증기관을 구축할 필요는 없다.

6. 결 론

본 논문에서는 공개키 기반 구조와 연동 가능한 효율적인 비밀키 복구 시스템을 제안하였다. 공개키 기반 구조와 연동 가능한 키 복구 시스템은 이미 구축된 PKI 시스템을 기반으로 하여 키 복구 방식을 구현할 수 있으므로 사용자들에게 요구되는 비용이 적기 때문에 매우 효과적인 방법이다. 따라서 이를 이용한 여러 키 복구 시스템들이 연구되었으며 실제 프로토콜들이 소개되었다.

본 논문에서 제안하고 있는 비밀키 복구 시스템은 사용자이외의 어떤 기관도 사용자의 비밀키를 복구할 수 없다. 그러나 문제는 참여하고 있는 키 위탁기관들 모두 공모한다면 모든 사용자의 비밀키를 복구할 수 있다는 점이다. 따라서 사용자 비밀키 복구 보호를 완벽하게 하도록 시스템을 설계하려면 참여하고 있는 키 위탁 기관 중 적어도 하나라도 공모하지 못하도록 하는 대책이 필요하다.

참 고 문 헌

[1] A. Young, M. Yung, "Auto-Recoverab le and Auto-Certifiable Cryptosystems," A dvanced in Cryptology-Eurocrypt'98, Spri nger-Velag, Lecture Notes in Computer Sci ence, Springer-Verlag, pp.17-31, 1998.

[2] P. Paillier, Moti Yung, "Self-Escrowed Public Key Infrastructures," Proceedings of ICISC'99, The 2nd International Conference on Information Security and Cryptology. Springer-Verlag, Lecture Notes in Computer Science, LNCS, Dec. 9 10, 1999.

[3] D. Boneh, M. Franklin, "Efficient generation of shared RSA keys," In Proceedings Crypto'97, Lecture Notes on Computer Science, Vol.1223, Springer-Verlag. pp.425-439, 1997.

[4] G. Poupard, J. Stern, "Generation of Shared RSA Keys by Two Parties," Advanced in Cryptology-Asiacrypt'98, Springer-Velag, Lecture Notes in Computer Science, Springer-Verlag. LNCS 1514, pp.11-24, 1998.

[5] P. Paillier, "Public-Key Cryptosystem Based on Composite Degree Residuosity Classes," Advanced in Cryptology-Eurocrypt'99, Springer-Velag, Lect ure Notes in Computer Science, Springer-Verlag, pp.223-238, 1999.

[6] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarith ms," In Crypto'84, pp. 10-18, 1984.

[7] A. Camenisch, Y. Frankel and Y. Tsiounis. Easy Come-Easy Go Divisible Cash. In Advances in Cryptology, Eurocrypt'98, LNCS 1403, pp.561- 575, Springer Verlah, 1998.

[8] 유회종, 최희봉, 정찬주, 원동호, "다수의 위탁기관 참여 가능한 SE-PKI 키 복구 시스템", 한국통신정보보호학회 종합학술발표회 논문지, 2000. 12.

최 희 봉

e-mail : hbchoi@etri.re.kr
1984년 부산대학교 전기공학과 학사
1987년 부산대학교 전기공학과 석사
1997년~현재 성균관대학교 전전컴공학부 박사과정
1987년~2000년 국방과학연구소 선임연구원
2000년~현재 국가보안기술연구소 선임연구원
관심분야 : 암호이론, 네트워크보안, 보안시스템 설계

유 회 종

e-mail : anny5@etri.re.kr
1999년 성균관대학교 정보공학과 학사
2001년 성균관대학교 전전컴공학부 석사
2001년~현재 한국전자통신 연구원 연구원
관심분야 : 암호이론, 정보이론

오 수 현

e-mail : shoh@dosan.skku.ac.kr
1998년 성균관대학교 정보공학과 학사
2000년 성균관대학교 전전컴공학부 석사
2000년~현재 성균관대학교 전전컴공학부 박사과정
관심분야 : 암호이론, 정보이론

원 동 호

e-mail : dhwon@simsan.skku.ac.kr
1976년 성균관대학교 전자공학과 학사
1978년 성균관대학교 전자공학과 석사
1988년 성균관대학교 전자공학과 박사
1978년~1980년 한국전자통신연구원 연구원
1985년~1986년 일본 동경공업대 개원연구원
1982년~현재 성균관대학교 전전컴공학부 정교수
관심분야 : 암호이론, 정보이론