情報保護學會論文誌
第15券 第2號, 2005. 4

동적 ID 정보가 포함된 신원기반 암호시스템에서 효율적인 키 재발급 모델*

김 동 현,[1†] 김 상 진,[2] 오 희 국,[1‡] 구 본 석,[3] 유 권 호[3]
[1]한양대학교, [2]한국기술교육대학교, [3]국가보안기술연구소

A New Efficient Private Key Reissuing Model for Identity-based Encryption Schemes Including Dynamic Information

Donghyun Kim,[1†] Sangjin Kim,[2] Heekuck Oh,[1‡] Bonseok Koo,[3] Gwonho Ryu[3]
[1]Hanyang University, [2]Korea University of Technology and Education,
[3]National Security Research Institute

요 약

신원기반 암호기법에서 PKG(Private Key Generator)의 권한 남용을 막기 위해 일반적으로 threshold 기법을 사용한다. 하지만 이 방법은 사용자 개인키 발급과정에서 보다 많은 인증, pairing 연산, 통신비용을 요구한다. 이 논문에서는 PKG의 권한을 분산시키는 신원기반 암호기법에서 같은 사용자에게 다수의 개인키를 수시로 발급하거나 만료된 또는 철회된 키를 재발급하는 경우 이를 효율적으로 처리해줄 수 있는 새로운 모델을 제안한다. 새 기법에서 사용자의 개인키는 서로 다른 신뢰기관에 의해서 발급되는 두 개의 요소인 KGK(Key Generation Key)와 KUD(Key Usage Descriptor)로 구성된다. 이 중 KGK는 다수의 신뢰기관인 KIC(Key Issuing Center)가 threshold 방법으로 발급하며, KUD는 단일 신뢰기관인 KUM(Key Usage Manager)이 발급한다. 이 시스템의 장점은 키 재발급 비용이 상수시간이며 공개채널을 통한 키 발급이 가능하다는 것이다. 또한 Gentry가 제안하였던 time-slot 기반의 개인키 철회기법을 다른 신원기반 암호기법보다 효율적으로 적용할 수 있다. 이 논문은 새 시스템의 안전성에 대해서 증명하고 타 시스템과의 비교 분석을 통해 그 효율성을 보여준다.

ABSTRACT

The main obstacle hindering the wide deployment of identity-based cryptosystem is that the entity responsible for creating the private key has too much power. As a result, private keys are no longer private. One obvious solution to this problem is to apply the threshold technique. However, this increases the authentication, computation, and communication cost during the key issuing phase. In this paper, we propose a new efficient model for issuing multiple private keys in identity-based encryption schemes based on the Weil pairing that also alleviates the key escrow problem. In our system, the private key of a user is divided into two components, KGK (Key Description Key) and KUD (Key Usage Desscriptor), which are issued separately by different parties. The KGK is issued in a threshold manner by KIC (Key Issuing Center), whereas the KUD is

접수일 : 2004년 12월 1일 ; 채택일 : 2005년 3월 24일
* 본 연구는 국가보안기술연구소에서 지원하는 위탁과제로 수행하였습니다.
† 주저자 : dhkim@cse.hanyang.ac.kr
‡ 교신저자 : hkoh@cse.hanyang.ac.kr

issued by a single authority called KUM (Key Usage Manager). Changing KUD results in a different private key. As a result, a user can efficiently obtain a new private key by interacting with KUM. We can also adapt Gentry's time-slot based private key revocation approach to our scheme more efficiently than others. We also show the security of the system and its efficiency by analyzing the existing systems.

Keywords : *Web, Steganography, Database*

Ⅰ. 서 론

1984년 Shamir[1]에 의해 최초로 그 개념이 소개된 신원기반 암호시스템(identity-based cryptosystem)은 사용자의 공개키를 잘 알려진 ID(IDentity)로부터 유도하는 암호시스템이다. Shamir는 RSA를 이용하여 신원기반 서명기법을 제안하였지만, 사실 신원기반 암호기법까지 염두에 두지는 않았다. 이후로 많은 다양한 신원기반 암호기법이 제안되었지만[2-4] 2001년 Boneh와 Franklin[5]이 Weil pairing을 이용하여 신원기반 암호기법을 제안하기 전까지 사람들은 '완전한' 신원기반 암호기법을 제안하지 못했다. 여기서 완전하다는 것은 시스템의 안전성이 어떤 특정한 가정에 의존하지 않으며, 어느 정도 수준의 효율성을 갖춘 시스템을 말한다. 예를 들어 Desmedt와 Quisquater의 방법[2]은 조작할 수 없는 하드웨어가 필요하며, Tanaka의 방법[3]은 사용자들이 공모하지 않아야 안전하다. Maurer와 Yacobi의 방법[4]의 경우에는 사용자의 개인키를 생성하는 비용이 비효율적이다. 한편 Cock[6]은 이차잉여를 이용한 또 다른 완전한 신원기반 암호기법을 제안하였다. 그러나 현재 대부분의 신원기반 암호기법에 관한 연구는 pairing을 이용하여 진행되고 있으며[7-10], 이 논문도 pairing을 이용한 신원기반 암호기법을 제안한다.

신원기반 암호시스템은 기존의 PKI(Public Key Infrastructure)에 비해 몇 가지 이점을 갖는다. 첫째, 모든 사용자의 공개키는 e-mail과 같은 그 사용자의 잘 알려진 ID로부터 유도된다. 따라서 신원기반 암호시스템에는 인증서가 필요 없으며, 송신자는 수신자나 제3자와 접촉하지 않고 수신자에게 메시지를 암호화하여 전달할 수 있다. 둘째, 인증서가 필요 없으므로 신원기반 암호시스템에는 인증서 디렉터리와 같은 인증서 관리를 위한 PKI 요소들이 필요 없다. 물론 신원기반 암호시스템에서도 시스템에서 사용하는 공개 파라미터 정보가 관리되어야 하지만 그 비용은 모든 사용자의 인증서를 관리하는 것에 비해 매우 저렴하다.

이와 같은 장점에도 불구하고 신원기반 암호시스템이 기존 PKI의 대안이 되기 위해서는 다음과 같은 문제들에 대한 해결이 선행되어야 한다. 첫째, 신원기반 암호시스템에서 사용자의 개인키를 PKG(Private Key Generator)라는 신뢰기관이 직접 계산하여 발급해 준다. 따라서 PKG는 강력한 키 복구 능력을 가지고 있으며, PKG는 모든 사용자의 암호문을 해독하거나 서명을 위조할 수 있다. 둘째, PKG는 사용자에게 개인키를 발급하기 전에 사용자를 인증해야 하며, 생성한 개인키를 안전하게 사용자에게 전달할 수 있어야 한다. 후자는 현재 키 발급 프로토콜에서 간단하게 해결할 수 있지만[9] 전자는 신원기반 암호시스템만으로 제공하기가 어렵다.

PKG의 강력한 능력을 제한할 수 있는 가장 간단한 방법은 다수의 PKG에게 키 발급 권한을 분산시킨 후, 사용자의 개인키를 threshold 방식으로 발급하는 것이다. 이렇게 하면 다른 암호시스템에 비해 키 복구 기능을 쉽게 제공할 수 있는 이점이 있으며, 키 복구가 필요한 응용에서는 신원기반 시스템이 매우 유용하게 사용될 수 있다. Boneh와 Franklin은 이러한 방식을 사용하는 키 발급 방법을 제안하였다.[5] 이 기법에서 사용자의 개인키를 발급할 때 사용되는 마스터 키는 다수의 PKG들에게 비밀로 분배되며, 일정한 수 이상의 PKG가 협력해야 사용자의 개인키를 생성할 수 있다. 이것을 통해 PKG의 강력한 능력을 제한할 수 있지만 사용자는 개인키를 얻기 위해 다수의 PKG와 통신해야 하는 부담을 갖는다. 결국 키 발급과정은 보다 많은 인증, pairing 연산, 통신비용이 요구된다. 이 문제를 해결하기 위해 이병천 등[9]은 인증 부담을 줄이면서 PKG의 능력을 제한할 수 있는 방법을 제안하였다. 이 시스템에서 사용자는 개인키를 발급받기 위해서 단 한번의 인증을 수행한다. 또한 이 시스템은 간단한 은닉 기법을 사용하여 개인키를 안전하게 사용자에게 전달할 수 있다. 하지만 사용자는 모든 PKG와 순차적으로 상호작용해야 하는 부담을 갖는다.

Gentry[7]는 사용자가 선택한 난수 값을 개인키를 생성할 때 포함하도록 하는 새로운 접근방법을 제안하였다. 이 기법에서 PKG는 사용자가 선택한 난수를 모르기 때문에 사용자의 개인키를 복구할 수 없다. 하지만 이렇게 함으로써 사용자들 역시 다른 사용자의 공개키를 신원정보로부터 바로 계산할 수 없게 되었다. 즉, 이 기법은 신원기반 암호시스템처럼 PKG가 개인키를 발급하지만 기존 PKI와 유사하게 사용자가 선택한 난수 값에 대한 인증서가 필요하므로 엄밀한 의미에서 신원기반 암호시스템이라고 할 수 없다. Al-Riyami와 Paterson[8]은 Gentry의 시스템을 확장하여 인증서를 필요로 하지 않는 공개키 암호시스템을 제안했다. 하지만 이 기법은 사용자의 공개키에 대한 암시적 인증만을 제공한다. 따라서 각 사용자는 자신이 가지고 있는 상대방의 공개키가 올바른 것인지 확신할 수 없다는 문제점이 있다.

Boneh와 Franklin은 신원기반 암호기법에서 사용되는 사용자의 ID에 키에 대한 부가적인 서술자를 추가하여 보다 여러 가지 이점을 얻는 방법을 제시하였다.[5] 예를 들어, 키의 수명, 키의 용도, 키 소유자의 권한 등이 서술자로 사용될 수 있다. 이 경우 한 사용자는 자신의 유일한 신원정보로부터 다양한 ID를 유도할 수 있으며 그에 대한 다수의 개인키를 갖게 된다. 하지만 이 방법을 threshold 기법을 사용하는 시스템[5,9]에 그대로 적용하면 발급 과정에서 필요한 많은 비용 때문에 실제 활용하기가 부담스럽다.

전통적인 PKI의 경우에는 철회된 인증서에 관한 정보를 사용자들에게 알려주기 위해 보통 인증서 디렉토리에 최신의 CRL(Certificate Revocation List)이 유지된다. 철회된 인증서는 그 유효기간 동안에 CRL에 유지되며, 각 사용자는 인증서를 사용하기 전에 CRL를 통해 인증서의 철회 여부를 확인해야 한다. Gentry는 인증서 기반의 암호시스템에서 인증서 철회하기 위한 time-slot 기법이라는 또 다른 기법을 제안하였다[7]. 이 기법의 기본적인 생각은 인증서의 유효기간을 매우 짧게 만들고 유효기간이 경과되면 사용자가 다시 개인키를 발급받도록 하는 것이다. 이 때 문제가 발생하면 다음 기간에 해당하는 개인키를 발급하지 않으므로 철회 문제를 해결하고자 하였다. 이 방식은 time-slot마다 모든 사용자에게 개인키를 발급해 주어야 하는 문제와 이미 발급된 개인키를 철회하지 못하는 문제가 있다. 전자 문제는 계층 구조를 통해 해결하고자 하였고, 후자는 유효기간이 짧게 하여 해결하고자 하였다. 한편, PKI의 경우와 마찬가지로 신원기반 암호기법 또한 개인키 철회방법이 필요하다. 하지만 현재 신원기반 암호기법을 위해 제안된 효율적인 개인키 철회방법은 없다. 물론 기존 PKI와 마찬가지로 CRL을 이용할 수 있지만 CRL을 사용하면 제 3자나 수신자와 접촉 없이 메시지를 암호화하여 전달할 수 있다는 신원기반 암호기법의 장점이 퇴색된다. 따라서 신원기반 암호시스템의 장점을 유지하기 위해서는 Gentry와 같은 접근 방법이 보다 현실적이다.

앞에서 언급된 다양한 문제점을 고려하여 이 논문에서는 다음과 같은 특성을 갖는 새로운 신원기반 암호기법을 제안한다. 먼저, threshold 개념을 사용하여 PKG의 능력을 제한하였다. 둘째, 사용자의 개인키는 KGK(Key Generation Key)와 KUD(Key Usage Descriptor)로 구성하였다. KGK는 기존의 신원기반 암호시스템에서와 같은 방법으로 사용자의 ID로부터 유도된다. 이 키는 KIC(Key Issuing Center)라 명명된 다수의 신뢰기관에 의하여 threshold 방식으로 발급된다. 키 발급의 대략적인 절차는 Boneh와 Franklin이 제안한 방식과 유사하지만 제안하는 시스템에서는 안전한 채널 없이도 가능하다는 것이 장점이다. KUD는 KGK를 유도할 때 사용되었던 사용자 ID와 time-slot과 같은 가변 정보를 이용하여 발급된다. 단일 신뢰기관인 KUM(Key Usage Manager)는 사용자의 인증과정 없이 사용자로부터 은닉된 KGK를 받은 후 그 유효성을 확인한다. 만약 사용자가 철회되지 않은 경우, KUM은 KGK에 맞는 적절한 KUD를 발급해 준다. 셋째, 비록 사용자의 개인키가 서로 다른 인증기관에 의하여 발급되는 키들의 합으로 구성되지만, 누구나 사용자의 공개키를 유도할 수 있다. 넷째, 사용자는 KUD를 새로 발급 받음으로서 새로운 개인키를 얻을 수 있다. 따라서 사용자는 매우 효율적으로 같은 신원정보에 대한 다수의 ID에 해당하는 개인키를 얻을 수 있다. 또한 제안되는 기법은 Gentry가 제안한 time-slot 기반의 키 철회기법을 적용할 수 있다. 비록 기존의 다른 신원기반 암호기법에도 Gentry의 기법을 적용할 수 있지만 이 논문에서 제안한 키 발급 방법은 보다 작은 인증, pairing 연산, 통신비용을 요구한다.

이 논문의 나머지 부분은 다음과 같이 구성된다. 2장에서는 논문과 관련된 수학적 배경에 대하여 설명하고, 3장에서는 관련 연구에 대하여 살펴본다. 4장에서는 제안되는 기법을 소개하고, 5장에서는 제

안되는 기법의 안전성을 분석하고 관련 연구와 비교한다. 마지막으로 6장에서 결론 및 향후 과제에 대하여 논한다.

Ⅱ. 수학적 배경

이 논문에서는 다음과 같은 표기법을 사용한다: 1) $\mathbb{G}_1$은 위수가 q인 타원곡선상의 덧셈군을 나타낸다. 2) $\mathbb{G}_2$는 위수가 q인 유한체상의 곱셈군을 나타낸다. 3) $\mathbb{G}_1^* = \mathbb{G}_1/\{O\}$이다. 여기서 O는 $\mathbb{G}_1$의 항등원이다. 4) $\mathbb{Z}_q^*$는 q를 법으로 하는 곱셈군이다. 4) a, b, c는 $\mathbb{Z}_q^*$의 원소이다. 5) P, Q, R은 $\mathbb{G}_1$의 임의의 원소이다.

정의 1 (Admissible bilinear map). 다음과 같은 특성을 만족하는 $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$를 admissible bilinear map이라 한다.

- **Bilinear**: 임의의 P, Q, R에 대하여 다음이 성립한다.
  - $\hat{e}(P, Q+R) = \hat{e}(P, Q) \cdot \hat{e}(P, R)$
  - $\hat{e}(P+Q, R) = \hat{e}(P, R) \cdot \hat{e}(Q, R)$
- **Non-degenerate**: 어떤 P, Q에 대하여 $\hat{e}(P, Q) \neq O$ 이다.
- **Computable**: 어떤 P, Q에 대하여 $\hat{e}(P, Q)$를 계산할 수 있는 효율적인 알고리즘이 존재한다.

Bilinear map의 bilinear 특성은 $\hat{e}(aP, bQ) = \hat{e}(aP, Q)^b = \hat{e}(P, bQ)^a = \hat{e}(P, Q)^{ab}$가 성립한다는 것을 의미한다. 여기서 $a, b \in \mathbb{Z}_q^*$이다. 타원곡선상의 Weil 또는 Tate pairing을 응용하는 경우 이 같은 admissible bilinear map을 구현할 수 있다.[5]

정의 2 (Discrete Logarithm Problem (DLP) in $\mathbb{G}_1$). DLP는 주어진 $\langle P \neq O, aP \rangle$로부터 a를 계산하는 문제이다.

정의 3 (Computational Diffie-Hellman Problem (CDHP) in $\mathbb{G}_1$). CDHP는 주어진 $\langle P \neq O, aP, bP, cP \rangle$로부터 abP를 계산하는 문제이다.

정의 4 (BDH Parameter Generator). 다음과 같은 특성을 갖는 확률적 알고리즘 G를 BDH 파라미터 생성기라 한다. 1) 안전성 파라미터 $k \geq 1$를 입력으로 받는다. 2) 다항시간 알고리즘이다. 3) 위수가 q인 두 개의 군 $\mathbb{G}_1$과 $\mathbb{G}_2$의 명세와 admissible bilinear map $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$를 돌려준다.

정의 5 (Bilinear Diffie-Hellman Problem (BDHP)). BDHP는 주어진 $\langle P \neq O, aP, bP, cP \rangle$로부터 $\hat{e}(P, P)^{abc} \in \mathbb{G}_2$를 구하는 문제이다.

현재 타원곡선상의 점들로 이루어진 덧셈군 $\mathbb{G}_1$과 유한체의 원소로 이루어진 곱셈군 $\mathbb{G}_2$간의 BDHP문제는 어려운 것으로 알려져 있다.[5]

Ⅲ. 관련연구

이 장에서는 제안하는 시스템과 관련된 두 개의 기법을 소개한다. 이 기법들을 소개함에 있어 다음과 같은 표기법을 사용한다. 1) ID는 사용자의 신원정보이다. 2) $H_1: \{0,1\}^* \rightarrow \mathbb{G}_1^*$, $H_2: \mathbb{G}_2 \rightarrow \{0,1\}^l$, $H_3: \mathbb{G}_2 \rightarrow \mathbb{Z}_q^*$는 충돌회피 해쉬함수이다. 여기서 l은 메시지 블록의 길이를 나타낸다. 3) "||"는 두 문자열을 결합(bitwise concatenation)하는 연산자이다. 소개되는 각 기법은 시스템 설정 단계를 갖으며, 이 단계에서 신뢰기관은 BDH 파라미터 생성기 G를 수행하여 $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$를 얻은 후, $\mathbb{G}_1$의 임의의 생성자인 P를 얻는다.

3.1 Boneh와 Franklin의 기법

Boneh와 Franklin은 다수의 신뢰기관이 (t, n) threshold에 기반하여 개인키를 발급하게 하여 단일 PKG의 능력을 제한하였다.[5] 이 기법에서 사용자는 개인키 발급을 위하여 t개의 PKG에게 자신의 개인키 몫(공유 비밀)을 요청해야 한다. 이 때 사용하는 프로토콜은 다음과 같다.

- **시스템 설정**: 이 단계에서, n개의 PKG 중 임의의 PKG가 G를 수행한다. 또한, $P \in \mathbb{G}_1^*$, H_1, H_2를 선택한다. 각 PKG는 자신의 마스터

키 $s_i \in \mathbb{Z}_q^*, (1 \le i \le n)$을 Gennaro 등의 방법[11]을 이용하여 생성한다. 이후 각 PKG는 자신의 공개키 $P_i = s_i P \in \mathbb{G}_1^*$를 계산한다. 전체 PKG의 공개키는 $P_{pub} = \Sigma_{i \in I} L_i s_i P$이다. 여기서 I는 1보다 크고 n보다 작은 임의의 t개의 수들의 집합이며, L_i는 적절한 라그랑즈(Lagrange) 계수이다. 시스템 공개 파라미터는 다음과 같다.

$\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, l, P, P_1, \cdots, P_n, P_{pub}, H_1, H_2 \rangle$

- **개인키 추출**: 사용자는 t개의 서로 다른 PKG에게 자신의 ID를 보낸후 자신을 인증한다. 각각의 PKG는 자신의 마스터 키 s_i를 이용하여 사용자의 개인키 몫 $d_{ID}^{(i)} = s_i Q_{ID} = s_i H_1(ID) \in \mathbb{G}_1^*$를 계산한다. 이후 안전한 통신 채널을 통하여 $d_{ID}^{(i)}$를 사용자에게 돌려준다. 사용자는 자신이 받은 $d_{ID}^{(i)}$의 올바름을 다음의 식으로 확인해 볼 수 있다.

$\hat{e}(P, d_{ID}^{(i)}) \stackrel{?}{=} \hat{e}(P_i, Q_{ID})$

- **개인키 계산**: t개의 서로 다른 올바른 개인키 몫을 얻은 사용자는 다음과 같이 자신의 개인키를 계산한다.

$d_{ID} = \Sigma_{i \in I} L_i s_i Q_{ID}$

사용자는 d_{ID}의 올바름을 다음의 식으로 확인할 수 있다.

$\hat{e}(P, d_{ID}) \stackrel{?}{=} \hat{e}(P_{pub}, Q_{ID})$

- **암호문 작성**: 메시지 $m \in \{0,1\}^l$를 보내고자 하는 송신자는 수신자의 ID를 이용하여 다음과 같은 과정을 수행한다.
  - $Q_{ID} = H_1(ID) \in \mathbb{G}_1^*$을 계산한다.
  - 임의의 $r \in \mathbb{Z}_q^*$를 선택한다.
  - $g = \hat{e}(Q_{ID}, P_{pub})$을 계산한다.
  - 암호문은 다음과 같다.

$C = \langle rP, m \oplus H_2(g^r) \rangle \in \mathbb{G}_1^* \times \{0,1\}^l$

- **암호문 해독**: 수신자는 자신의 개인키 d_{ID}를 이용하여 송신자가 전송한 암호문 $C = \langle U, V \rangle$를 다음과 같이 해독한다.

$m = V \oplus H_2(\hat{e}(U, d_{ID}))$

이 기법에서 사용자의 개인키를 복구하기 위해서는 t개 이상의 PKG들이 공모해야 한다. 이 기법은 단일 PKG에 의한 키 복구 문제를 훌륭하게 해결하고 있지만 그에 따르는 인증 및 통신 양이 단일 PKG 환경에 비하여 t배 증가한다. 또한 개인키 발급 시 $2t+2$번의 pairing 연산을 수행하여야 한다.

3.2 이병천 등의 기법

이병천 등은 다수의 신뢰기관으로 키 복구 문제를 해결함과 동시에 Boneh와 Franklin의 방법에 비하여 인증 부담이 적은 새로운 키 발급 기법을 제안하였다.[9] 이 방법에서 KGC(Key Generation Center)와 n개의 KPA(Key Privacy Agency)들은 사용자의 개인키를 순차적인 방법으로 만들어 준다. 이 프로토콜은 다음과 같이 진행된다.

- **시스템 설정**: 이 단계에서 KGC는 G를 수행한다. 또한, $P \in \mathbb{G}_1^*$, H_1, H_2, H_3를 선택한다. KGC는 또한 자신의 마스터 키 $s_0 \in \mathbb{Z}_q^*$를 선택한 뒤, 공개키 $P_0 = s_0 P \in \mathbb{G}_1^*$를 계산한다.

- **시스템 공개키 설정**: n개의 KPA들은 각각 자신의 개인키 $s_i \in \mathbb{Z}_q^*, (1 \le i \le n)$를 선택한 뒤 대응되는 공개키 $P_i = s_i P \in \mathbb{G}_1^*$를 계산한다. 또한 다음과 같이 전체 신뢰기관의 공개키를 계산한다.

$P_{pub} = s_0 s_1 \cdots s_n P \in \mathbb{G}_1^*$

시스템 공개 파라미터는 다음과 같다.

$\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, l, P, P_0, \cdots, P_n, P_{pub}, H_1, H_2, H_3 \rangle$

- **개인키 발급**: 사용자는 자신의 비밀 값 $x \in \mathbb{Z}_q^*$를 선택한 뒤 은닉 값 $X = xP \in \mathbb{G}_1^*$를 계산한다. 이후, 사용자는 자신의 ID와 X를 KGC에게 공개된 채널로 전송하며 개인키 발급을 요청한다. KGC는 아래와 같은 절차를 통해 은닉된 부분 개인키를 발급해 준다.
  - 단계 1. 사용자를 인증한다.
  - 단계 2. 사용자의 공개키 $Q_{ID} = H_1(ID \| KGC \| KPA_1 \| \cdots \| KPA_n) \in \mathbb{G}_1^*$를 계산한다. 여기서 KGC와 $KPA_i, (1 \le i \le n)$는 신뢰기관의 ID이다.

- 단계 3. 사용자의 은닉된 부분 개인키인 $\overline{Q}_0 = H_3(\hat{e}(s_0X, P_0))s_0Q_{ID} \in \mathbb{G}_1^*$와 $\overline{Q}_0$에 대한 KGC의 서명 $Sig_0(\overline{Q}_0) = s_0\overline{Q} \in \mathbb{G}_1^*$를 계산한다.
- 단계 4. $\overline{Q}_0$와 $Sig_0(\overline{Q}_0)$를 사용자에게 전송한다.

• **개인키 보호**: $i, (1 \le i \le n)$번째 KPA에게 사용자는 ID, X, $\overline{Q}_{i-1}$과 $Sig_{i-1}(\overline{Q}_{i-1})$을 전송한다. 사용자로부터 요청을 받은 i번째 KPA는 다음 절차를 수행한다.

- 단계 1. $\hat{e}(Sig_{i-1}(\overline{Q}_{i-1}), P) \stackrel{?}{=} \hat{e}(\overline{Q}_{i-1}, P_{i-1})$ 여부를 확인한다.
- 단계 2. $\overline{Q}_i = H_3(\hat{e}(s_iX, P_i))s_i\overline{Q}_{i-1} \in \mathbb{G}_1^*$과 $Sig_i(\overline{Q}_i) = s_i\overline{Q}_i \in \mathbb{G}_1^*$를 계산한다.
- 단계 3. $\overline{Q}_i$과 $Sig_i(\overline{Q}_i)$를 사용자에게 돌려준다.

위의 과정은 n개의 KPA들에 대하여 순차적으로 적용된다. 모든 KPA들과 통신을 마친 후 사용자는 다음과 같은 자신의 은닉된 개인키를 얻게 된다.

$\overline{Q}_n = H_3(\hat{e}(s_nX, P_n))s_n\overline{Q}_{n-1} \in \mathbb{G}_1^*$

• **개인키 추출**: 사용자는 자신의 비밀 값 x를 이용하여 은닉된 개인키의 은닉요소를 제거한 후 자신의 개인키 D_{ID}를 얻는다.

$$D_{ID} = \overline{Q}_n / H_3(\hat{e}(P_0, P_0)^x) \cdots H_3(\hat{e}(P_n, P_n)^x)$$
$$= s_0s_1 \cdots s_nQ_{ID} \in \mathbb{G}_1^*$$

사용자는 다음 수식을 통하여 자신의 개인키 D_{ID}의 올바름을 확인 한다.

$\hat{e}(D_{ID}, P) \stackrel{?}{=} \hat{e}(Q_{ID}, P_{pub})$

• **암호문 작성/해독**: 이 과정은 Boneh와 Franklin의 기법[5]과 동일하다.

이 기법에서 사용자는 자신의 개인키를 발급받기 위해서 KGC에게 단 한번만 자신을 인증한다. 하지만 사용자는 KGC 및 n개의 KPA들과 순차적으로 키 발급절차를 수행해야 한다. 이 기법은 단일 PKG가 존재하는 원래의 기법에 비하여 $n+1$배의 통신 양을 요구한다. 또한 $4n+4$번의 pairing 연산을 수행한다.

Ⅳ. 제안하는 기법

4.1 용어 정의

제안하는 시스템을 자세히 기술하기에 앞서 구별이 모호할 수 있는 키 발급(key issuance), 키 재발급(key reissuance), 키 복구(key escrow), 키 철회(key revocation)의 의미를 제안하는 시스템 관점에서 먼저 설명한다. 이 시스템에서 키 발급이란 사용자가 기존과 다른 신원정보를 이용하여 최초로 개인키를 획득하는 것을 말한다. 키 재발급은 특정 신원정보를 이용하여 개인키를 이미 발급받은 사용자가 키가 만료되거나 철회되어 동일 신원정보를 이용하여 다시 발급받아야 하는 경우와 현재 키에는 문제가 없지만 동일 신원정보를 이용하여 다른 용도의 키를 발급받아야 하는 경우를 말한다. 키 복구는 키가 분실되거나 법 집행기관의 요구에 의해 사용자의 개인키를 사용자로부터 얻지 않고 키 발급 기관을 통해 계산하는 것을 말한다. 키 철회는 보통 현재 사용하고 있는 개인키를 더 이상 합법적으로 사용할 수 없도록 만드는 것을 말하지만 이 논문에서는 신원기반 암호시스템의 장점을 그대로 유지하기 위해 현재 사용되고 있는 개인키를 사용하지 못하도록 하는 대신에 이 사용자는 더 이상 새로운 개인키를 발급받지 못하도록 하는 방법을 사용하고 있다.

4.2 디자인 특징

제안하는 시스템은 다음과 같은 목표를 갖는다: 1) 개인키를 발행해 주는 PKG의 키 복구 능력을 제한해야 한다. 즉, 능력 자체를 제거하지는 않지만 남용할 수 없도록 해야 한다. 2) 같은 신원정보에 대한 다수의 개인키 발급이 효율적이어야 한다. 3) 사용자의 공개키를 잘 알려진 ID로부터 유도할 수 있는 신원기반 암호시스템의 특성을 유지해야 한다.

현재까지 키 복구 능력을 제한할 수 있는 가장 좋은 방법은 threshold 기법을 적용하는 것이다. 하지만 Boneh와 Franklin의 기법에서 알 수 있듯이, 단순히 다수의 PKG들에게 마스터키를 분배하는 방법은 개인키 발급을 비효율적으로 만든다. 이와 같은 특성은 다수의 키를 발급할 때 더욱 심각해진다. 보다 효율적인 개인키 재발행을 위해 이 논문에서는

Gentry의 아이디어[7]를 응용하였다. 그의 기법에서 사용자의 개인키는 두개의 요소로 구성된다. 이와 유사하게 이 논문에서 제안하는 기법에서 사용자의 개인키는 두개의 요소로 구성된다. 다만 세 번째 목적을 달성하기 위해 두 개의 요소는 서로 다른 신뢰기관이 발급한다. 또한 첫 번째 목적을 달성하기 위해 한 개의 요소는 threshold 방식으로 발급한다. 결과적으로 제안하는 시스템에서 n개의 신뢰기관은 threshold 방식으로 사용자의 개인키 몫을 발급해 주며, 단일 신뢰기관이 나머지 몫을 발급해 준다. 이러한 구조는 사용자가 단일 신뢰기관이 발급하는 개인키 몫을 변경함으로써 보다 효율적으로 다수의 개인키를 얻을 수 있도록 해준다.

4.3 사용자 ID의 구조

일반적으로 신원기반 암호시스템에서는 e-mail과 같은 잘 알려진 공개된 정보를 사용자의 ID로 사용한다. 하지만 어떠한 사유에 의해 사용자의 개인키를 변경해야 하면 사용자가 자신의 ID를 변경하거나 PKG의 마스터 키를 변경해야 한다. 하지만 두 가지 방법 모두 현실적이지 못하다. 전자의 경우 사용자의 ID는 사용자를 유일하게 식별해 줄 수 있는 주민등록번호와 같은 정보이거나 e-mail과 같이 다른 사용자들에게 널리 알려진 정보이다. 따라서 이러한 정보를 변경하는 것은 어렵다. 후자의 경우 모든 사용자의 개인키가 재발급 되어야 하기 때문에 이 역시 어렵다. 이 같은 문제를 해결하기 위해 사용자의 ID에 키의 수명과 같은 동적인 정보를 포함할 수 있으며, 개인키를 변경할 때 이런 동적인 정보만 변경하면 된다. 이와 같은 방법을 응용하면 한 사용자가 자신의 고유한 정보에 해당하는 다양한 공개키/개인키 쌍을 얻을 수 있다. 하지만 신원기반 암호기법에서 PKG의 키 복구 능력을 제한하기 위해 threshold 기법을 적용하는 경우 다수의 ID에 해당하는 다수의 개인키를 독립적으로 발급하는 것은 효율적이지 못하다.

이 논문에서는 다수의 키를 보다 효율적으로 발행해 주는 방법을 제안한다. 제안하는 방법에서 사용자의 개인키는 KGK와 KUD의 두 가지 요소로 구성된다. KGK는 사용자의 고유한 신원정보를 입력으로 하여 다수의 신뢰기관들인 KIC들에 의하여 threshold 방식으로 발급된다. KUD는 사용자의 고유한 신원정보와 키 서술자를 입력으로 하여 단일 신뢰기관인 KUM에 의하여 발급된다. 이 경우, 서술자는 Gentry가 언급했던 time-slot[7]이나 키의 용도 같은 것을 사용할 수 있다. KGK와 KUD는 동일한 사용자의 고유한 신원정보를 이용하여 생성되므로 서로 연관된다. 이 같은 키 발급의 예는 표 1에 있다. 이때 KGK가 특정기간 이후 재 발급되어야 할 필요가 있으면 KGK를 생성할 때 사용된 사용자 고유의 정보에 유효기간을 붙여서 KGK를 발급해 주면 된다. 비슷한 방법으로 KUD에도 유효기간을 지정할 수 있다. 사용자는 KGK나 KUD를 재발급 받음으로써 새로운 개인키를 얻을 수 있다. 특히 KUD를 재발급 받는 경우에는 보다 효율적으로 개인키를 얻을 수 있다. 만약 KGK을 재발급 받는다면, 그와 관련된 모든 KUD가 재발급 되어야 한다. 하지만 KUD는 단일 신뢰기관으로부터 사용자 인증과정 없이 공개된 통신채널을 통하여 발급되므로 비용이 저렴하다.

표 1. 사용자의 공개키 유도에 사용되는 문자열

예	KGK	KUD
1	foo@x.com‖2004	foo@x.com‖2004‖Role1
2	foo@x.com‖2004	foo@x.com‖2004‖16,Oct.

4.4 시스템 참여자

제안하는 시스템의 참여자에는 사용자, KUM, n개의 KIC가 있다. 각각의 역할은 다음과 같다.

- **n개의 KIC**: KIC는 사용자의 KGK를 발급해 준다. 사용자는 n개중 적어도 t개 이상의 KIC에게 키 발급을 요청해야 한다. 각 KIC는 공개된 통신 채널을 통하여 사용자에게 은닉된 KGK 몫을 전송해 준다.
- **KUM**: KUM은 사용자를 위하여 KUD를 발급해 준다. 사용자는 자신의 신원정보, 은닉된 KGK, 원하는 동적 정보를 KUM에게 전송한다. KUM은 공개된 통신 채널을 통하여 KUD를 사용자에게 전송한다.
- **사용자**: 사용자는 t개의 KIC로부터 각각의 KGK 몫을 얻은 후, 이것을 이용하여 KGK를 계산한다. 또한 KUM으로부터 필요한 KUD를 얻는다. 사용자의 개인키는 KGK와 KUD를 합한 것이 된다.

4.5 프로토콜

제안하는 프로토콜은 KUM 설정, KIC 설정, KGK 발급, KUD 발급, 암호문 작성, 암호문 해독 등 6개의 세부 과정으로 구성된다.

- **KUM 설정**: KUM은 BDH 파라미터 G를 수행하여 $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}\rangle$를 얻는다. 또한 $\mathbb{G}_1$의 임의의 생성자인 P와 다음과 같은 두 개의 해쉬함수를 선택한다.
  $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$, $H_2 : \mathbb{G}_1^* \rightarrow \{0,1\}^l$
  여기서 l은 메시지 블록의 길이를 의미한다. 마지막으로 KUM은 마스터 키 $s \in \mathbb{Z}_q^*$를 선택한 후 공개키 $P_{KUM} = sP \in \mathbb{G}_1^*$를 계산한다.

- **KIC 설정**: n개의 KIC들은 Gennaro 등[11]의 기법을 응용하여 분산된 방식으로 각각의 마스터 $x_i \in \mathbb{Z}_q^*$를 선택한다. 이때 i번째 KIC의 공개키는 $P_{KIC_i} = x_i P \in \mathbb{G}_1^*$가 된다. 또한 전체 KIC들의 마스터 키는 $x = \Sigma_{i \in I} L_i x_i$이고 공개키는 $P_{KIC} = xP \in \mathbb{G}_1^*$이다. 전체 시스템의 공개 파라미터는 다음과 같다.
  $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, l, P, P_{KUM}, P_{KIC}, P_{KIC_1}, P_{KIC_2}, \cdots, P_{KIC_n}, H_1, H_2\rangle$

- **KGK 발급**: ID를 사용자의 신원 정보라 하자. 사용자는 t개의 서로 다른 KIC에게 다음의 절차를 먼저 수행한다. 다음의 절차를 수행하기 전에 KIC는 사용자를 인증해야 한다. 이때 사용자의 인증은 기존 PKI를 사용한다고 가정한다.
  - 단계 1. 사용자는 은닉 값 $b_i \in \mathbb{Z}_q^*$를 선택한 후 $b_i Q_{ID} = b_i H_1(ID)$를 계산한다.
  - 단계 2. 사용자는 i번째 KIC에게 다음을 전송한다.
    $\langle ID, b_i P, b_i Q_{ID}\rangle$
  - 단계 3. i번째 KIC는 사용자를 인증한 뒤 아래의 확인식을 이용하여 $b_i Q_{ID}$의 올바름을 확인한다.
    $$\hat{e}(Q_{ID}, b_i P) = \hat{e}(Q_{ID}, P)^{b_i} \stackrel{?}{=} \hat{e}(b_i Q_{ID}, P) = \hat{e}(Q_{ID}, P)^{b_i}. \quad (1)$$
  - 단계 4. i번째 KIC는 다음과 같이 은닉된 KGK를 계산한다.
    $b_i d_{ID}^{(i)} = x_i b_i Q_{ID} \in \mathbb{G}_1^*$
    마지막으로 공개된 통신채널을 통하여 $b_i d_{ID}^{(i)}$를 사용자에게 전송한다.
  - 단계 5. 사용자는 자신의 은닉 값 b_i를 이용하여 KGK 몫 $d_{ID}^{(i)} = b_i^{-1} b_i d_{ID}^{(i)}$를 계산한다. 이후 아래의 식을 이용하여 $d_{ID}^{(i)} = x_i Q_{ID}$의 올바름을 확인한다.
    $$\hat{e}(P, d_{ID}^{(i)}) = \hat{e}(P, x_i Q_{ID}) = \hat{e}(P, Q_{ID})^{x_i} \stackrel{?}{=} \hat{e}(P_{KIC_i}, Q_{ID}) = \hat{e}(x_i P, Q_{ID}) = \hat{e}(P, Q_{ID})^{x_i}. \quad (2)$$
  사용자는 t개의 서로 다른 KIC와 위의 절차를 수행한 뒤 다음의 식을 이용하여 KGK를 계산해 낸다.
  $$d_{ID} = \sum_{i \in I} L_i d_{ID}^{(i)} = \sum_{i \in I} L_i x_i Q_{ID} \in \mathbb{G}_1^*$$
  계산된 KGK d_{ID}의 올바름은 다음의 식을 이용하여 확인할 수 있다.
  $$\hat{e}(P, d_{ID}) = \hat{e}(P, xQ_{ID}) = \hat{e}(P, Q_{ID})^x \stackrel{?}{=} \hat{e}(P_{KIC}, Q_{ID}) = \hat{e}(xP, Q_{ID}) = \hat{e}(P, Q_{ID})^x$$

- **KUD 발급**: 사용자는 새로운 은닉 값 $b \in \mathbb{Z}_q^*$를 선택한 뒤 $\langle ID, T, X = bP_{KIC}, Y = bd_{ID}\rangle$를 KUM에게 전송하여 KUD를 요청한다. 여기서 T는 발행되는 KUD에 대한 서술자이다. 만약 사용자가 KUD를 받을 권한이 있다면 KGC는 다음의 절차를 수행하여 KUD를 발급해 준다.
  - 단계 1. KUM은 사용자가 올바른 KGK를 전송했는지 다음의 수식을 이용하여 확인한다.
    $$\hat{e}(Q_{ID}, X) = \hat{e}(Q_{ID}, bP_{KIC}) = \hat{e}(Q_{ID}, bxP) \stackrel{?}{=} \hat{e}(P, bd_{ID}) = \hat{e}(P, bxQ_{ID}) = \hat{e}(Q_{ID}, bxP) \quad (3)$$
  - 단계 2. KUM은 $Q_{\langle ID, T\rangle} = H_1(ID \| T) \in \mathbb{G}_1^*$를 계산한 뒤, KUD $d_{\langle ID, T\rangle} = sQ_{\langle ID, T\rangle} \in \mathbb{G}_1^*$를 계산한다.
  - 단계 3. KUM은 공개된 통신채널을 이용하여 $d_{\langle ID, T\rangle}$를 사용자에게 전송한다.

  사용자는 아래의 식을 이용하여 KUM으로부터 받은 $d_{\langle ID, T\rangle}$의 올바름을 확인한다.

$$\hat{e}(P, d_{\langle ID,T\rangle}) = \hat{e}(P, sQ_{\langle ID,T\rangle}) = \hat{e}(P, Q_{\langle ID,T\rangle})^s \stackrel{?}{=} \hat{e}(P_{KUM}, Q_{\langle ID,T\rangle}) = \hat{e}(sP, Q_{\langle ID,T\rangle}) = \hat{e}(P, Q_{\langle ID,T\rangle})^s \quad (4)$$

마지막으로 사용자는 KGK와 KUD를 더하여 자신의 개인키 $D_{\langle ID,T\rangle} = d_{ID} + d_{\langle ID,T\rangle} \in \mathbb{G}_1$을 얻는다.

만약 계산된 $D_{\langle ID,T\rangle}$가 $\mathbb{G}_1$의 항등원인 경우 키의 서술자를 변경해야 한다. 하지만 이러한 가능성은 무시할 수 있을 정도로 매우 작다. 따라서 $D_{\langle ID,T\rangle} \in \mathbb{G}_1^*$로 가정한다. KUD $d_{\langle ID,T\rangle}$를 공개된 채널을 통하여 보내는 이유는 시스템 전체의 threshold 특성을 유지하기 위해서이다. 즉, $d_{\langle ID,T\rangle}$는 공개되어 있으므로 사용자의 개인키를 복구하기 위해서는 서로 다른 t개의 KIC가 협력하면 가능하다. 만약 $d_{\langle ID,T\rangle}$가 안전한 비밀 통신채널을 통하여 전송되는 경우, 사용자 개인키 $D_{\langle ID,T\rangle}$의 복구는 반드시 KUM의 참여를 필요로 하게 된다.

- **사용자 개인키 재발급**: 사용자는 필요에 따라 자신의 새로운 개인키를 얻을 수 있다. 이 과정은 구체적으로 다음과 같은 두 가지 경우로 나눌 수 있다.
  - KUD가 만료되거나 기존과 다른 KUD가 필요한 경우: 현재 사용하고 있는 KUD가 만료되거나 기존과 다른 KUD를 발급받아 또 다른 개인키를 얻고자 하면 사용자는 새로운 T를 선택한 후 KUM으로부터 T와 기존의 KGK에 따르는 새로운 KUD을 얻는다. 이것은 이미 소개된 KUD 발급 절차를 다시 수행함으로서 완수된다.
  - KGK가 만료된 경우: 현재 사용하고 있는 KGK가 만료되면 사용자는 먼저 다수의 KIC들로부터 새로운 KGK를 획득한 후에 필요한 KUD를 다시 발급받아야 한다. 이것은 앞서 언급된 KGK 발급 과정과 KUD 발급 과정을 다시 수행함으로서 완수된다.
- **암호문 작성**: 암호문 작성에 앞서, 메시지의 송신자와 수신자는 사전에 T에 대하여 동의하였다고 가정한다. 먼저 송신자는 $Q_{ID} = H_1(ID) \in \mathbb{G}_1^*$과 $Q_{\langle ID,T\rangle} = H_1(ID\|T) \in \mathbb{G}_1^*$을 계산한 후, 다음을 계산한다.

$$g = \hat{e}(P_{KIC}, Q_{ID})\hat{e}(P_{KUM}, Q_{\langle ID,T\rangle}) \in \mathbb{G}_2$$

마지막으로, 임의의 비밀 값 $r \in \mathbb{Z}_q^*$를 선택한 후 메시지 $m \in \{0,1\}^l$에 대한 암호문을 다음과 같이 구한다.

$$C = \langle rP, m \oplus H_2(g^r)\rangle \in \mathbb{G}_1^* \times \{0,1\}^l$$

- **암호문 해독**: 암호문 $C = \langle U, V\rangle$를 받은 수신자는 개인키 $D_{\langle ID,T\rangle}$를 이용하여 다음과 같이 평문을 구한다.

$$m = V \oplus (H_2(\hat{e}(U, D_{\langle ID,T\rangle})))$$

제안되는 암호기법의 올바름은 아래의 식을 이용하여 쉽게 확인할 수 있다.

$$\begin{aligned}\hat{e}(U, D_{\langle ID,T\rangle}) &= \hat{e}(U, xQ_{ID} + sQ_{\langle ID,T\rangle}) \\ &= \hat{e}(U, xQ_{ID})\hat{e}(U, sQ_{\langle ID,T\rangle}) \\ &= \hat{e}(rP, xQ_{ID})\hat{e}(rP, sQ_{\langle ID,T\rangle}) \\ &= \hat{e}(P, Q_{ID})^{xr}\hat{e}(P, Q_{\langle ID,T\rangle})^{sr} \\ &= \hat{e}(xP, Q_{ID})^r\hat{e}(sP, Q_{\langle ID,T\rangle})^r \\ &= \hat{e}(P_{KIC}, Q_{ID})^r\hat{e}(P_{KUM}, Q_{\langle ID,T\rangle})^r = g^r\end{aligned}$$

V. 시스템 분석

5.1 키 발급 프로토콜의 안전성 증명

이 장에서는 제안된 키 발급 기법의 안전성을 증명한다. 자세한 증명에 앞서 제안하는 시스템의 안전성에 대해 정의를 내린 후, 그 정의에 입각하여 안전하다는 것을 증명할 것이다.

정의 6. 제안하는 키 발급기법은 다음의 세 가지 조건을 만족할 경우 안전하다.

- 각 참여자는 서로 다른 참여자의 능력을 얻을 수 없다.
- 의도된 수신자는 다른 참여자의 잘못된 응답을 감지할 수 있다.
- 공개된 통신 채널을 통하여 전송된 메시지가 변조된 경우 의도된 수신자에 이것을 확인할 수 있다.

시스템의 안전성 증명을 위하여 다음과 같은 가정을 사용한다.

가정 1. Threshold 메커니즘에 의하여 분배되어

있는 비밀을 복구하기 위해서는 $t \leq n$개 이상의 KIC가 공모해야 한다.

가정 2. $\mathbb{G}_1$에서의 DLP와 CDHP를 해결하는 것은 계산적으로 어렵다.

가정 3. 추가적으로 다음과 같은 사항들을 가정한다.

- 모든 참여자는 다음과 같은 공개 파라미터를 알고 있다.
  $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, l, P, P_{KUM}, P_{KIC}, P_{KIC_1}, P_{KIC_2}, \cdots, P_{KIC_n}, H_1, H_2 \rangle$
- 모든 참여자는 임의의 $\langle ID, T \rangle$쌍에 대해 다음을 알고 있다.
  $\langle ID, T, X = bP_{KIC}, Y = bd_{ID} \rangle$, $d_{\langle ID,T \rangle}$
- 각 KIC는 자신의 마스터 키인 x_i를 알고 있다.
- KUM은 자신의 마스터 키인 s를 알고 있다.

이 가정들을 바탕으로 앞으로 네 개의 보조정리를 먼저 제시한 후, 그것들에 대한 증명을 보일 것이다. 보조 정리에 대한 증명이 끝나면 제안하는 키 발행기법이 정의 6에 준하여 안전하다는 것을 증명한다.

보조정리 1. t개 이상의 KIC가 공모하여 x를 누설하지 않는 한, 임의의 참여자는 x를 얻을 수 없다.

증명. 만약 공격자가 x를 복구해 낼 수 있다면, 공격자는 사용자의 개인키를 복구할 수 있다. 우선, 가정 1에 의하여 $t < n$개 미만의 KIC들이 공모하는 경우에는 x를 복구할 수 없다. 또한, 가정 2에 의하여 $\langle ID, T, X = bP_{KIC}, Y = bd_{ID} = bxQ_{ID} \rangle$를 이용하여 x를 계산해 내는 것은 어렵다.

한편, 자신의 ID에 해당하는 개인키를 가지고 있는 공격자가 타인의 ID'에 해당하는 개인키를 얻으려 할 수 있다. 이때, KIC들은 자신에게 개인키를 요청하는 사용자의 신원을 확인하므로, 공격자는 다음과 같은 두 가지 공격을 시도할 수 있다. 첫째, 공격자는 $Q_{ID'} = aQ_{ID}$를 만족하는 a를 찾아낸 후, ID'에 해당하는 KGK를 얻으려 할 수 있다. 둘째, 만약 공격자가 $Q_{ID'} = H_1(ID')$과 $Q_{ID'} = bQ_{ID}$를 만족하는 ID'를 계산할 수 있다면, 공격자는 ID'에 해당하는 KGK를 구할 수 있다. 여기서 $b \in_R \mathbb{Z}_q^*$는 공격자가 선택한 임의의 값이다. 하지만 이러한 공격들은 각각 가정 2와 해쉬함수 H_1의 안전성에 의하여 계산적으로 어렵다. 게다가 후자의 경우 공격자가 ID'을 계산해 낸다 하여도 올바른 신원정보가 되기는 어렵다. 따라서 보조정리 1은 성립한다.

보조정리 2. KUM이 s를 누설하지 않으면 KUM을 제외한 다른 참여자가 s를 얻는 것은 계산적으로 어렵다.

증명. 가정 2에 의해 보조정리 2는 자연히 성립한다.

보조정리 3. t개 이상의 KIC가 공모하지 않으면 임의의 참여자는 사용자의 개인키를 복구해 낼 수 없다.

증명. 사용자의 개인키는 KGK인 d_{ID}와 KUD인 $d_{\langle ID,T \rangle}$의 합으로 이루어진다. 여기서 KUD는 공개된 정보이므로 공격자는 KGK만 얻으면 된다. 하지만 보조정리 1에 의하여 이것은 계산적으로 어려운 일이다. 결론적으로 보조정리 3은 성립한다.

보조정리 4. 공개된 네트워크를 통하여 전송되는 데이터가 변조된 경우, 의도된 수신자는 이를 감지할 수 있다.

증명. KGK의 요청이 있으면 사용자는 i번째 KIC에게 b_iP와 b_iQ_{ID}를 전송한다. 이것의 올바름은 식 (1)로 확인할 수 있다. $b_id_{ID}^{(i)}$는 KGK발급 과정에서 사용자가 i번째 KIC로부터 얻는 값이다. 이것의 올바름은 식 (2)로 확인할 수 있다. 다음은 KUD의 요청이 있으면 사용자가 KUM에게 전송하는 값이다.

$\langle ID, T, X = bP_{KIC}, Y = bd_{ID} = bxQ_{ID} \rangle$

여기서 X와 Y의 올바름은 식 (3)으로 확인할 수 있다. 마지막으로 $d_{\langle ID,T \rangle}$는 KUD를 사용자가 KUM으로부터 얻는 값이다. 이 값의 올바름은 식 (4)를 이용하여 확인할 수 있다. 따라서 보조정리 4는 성립한다.

정리 1. 제안하는 프로토콜은 정의 6의 모든 조건을 만족하므로 안전하다.

증명. 보조정리 1부터 보조정리 3에 의하여 각 참여자는 다른 참여자의 권한을 얻을 수 없다. 또한

보조정리 4에 의하여 네트워크상의 변조는 감지된다. 따라서 제안하는 기법은 정의 6의 모든 조건을 만족하므로 안전하다.

5.2 관련 연구와의 비교

이 장에서는 제안하는 키 발급 모델의 성능을 Boneh와 Franklin의 방법[5], 이병천 등의 방법[9]과 비교한다. 특히 키 발급 절차의 효율성, 키 복구 절차, 키 노출에 따른 개인키의 안전성의 세 가지 관점에서 비교한다. 비교에 앞서 Boneh와 Franklin의 방법과 제안하는 기법이 threshold 방식으로 키를 발급하지만 이병천 등의 방법[9]은 그렇지 않다는 것을 강조한다. 또한 Boneh와 Franklin의 방법과 제안하는 기법은 동시에 다수의 신뢰기관이 키 발급 절차를 진행 하지만 이병천 등의 방법의 경우에는 순차적으로 진행해야 한다는 것이 다르다. 따라서 신뢰기관의 수가 늘어나는 경우, 이병천 등의 방법은 그 수에 비례해서 더 많은 시간을 필요로 한다. 이 비교에서 Boneh와 Franklin의 방법과 제안하는 새로운 기법의 threshold값 t는 이병천 등의 방법의 KPA의 수와 같다고 가정한다. 즉, Boneh와 Franklin의 방법과 새로운 기법의 신뢰기관의 수는 동일하며 이병천 등의 방법은 한 개의 추가적인 신뢰기관이 더 있다는 것을 의미한다. 이러한 설정은 비교를 보다 공정하게 하기 위함이다.

먼저 각 기법들을 인증 횟수, pairing 연산 수, 통신 횟수를 기준으로 비교한다. 표 2에서 볼 수 있듯이 초기 키 발급의 경우 Boneh와 Franklin의 방법은 가장 적은 pairing 연산을 수행한다. 하지만 제안하는 기법에서 안전한 채널을 확보하기 사용했던 은닉기법을 Boneh와 Franklin의 방법에 적용하는 경우 Boneh와 Franklin의 방법은 pairing 연산이 $4t+2$가 필요하다. 이러한 설정에 따라서 초기에 키를 발행할 경우 제안하는 새로운 키 발행 기법은 Boneh와 Franklin의 방법보다 4번의 pairing 연산을, 이병천 등의 방법보다 2번의 pairing연산을 더 필요로 한다. 비록 키의 재발급 시 제안되는 기법에서 KGC가 다른 기법들에 비하여 약간 많은 2번의 pairing연산을 요구하지만 사용자는 단지 2번의 pairing 연산을 수행하면 된다. 결론적으로 키가 재발급되는 경우에 새 기법은 계산양은 단 4번에 불과하며 이것은 다른 기법에 비하여 매우 작다. 통신 횟수의 측면에서 볼 때 초기 키 발행 시에는 Boneh와 Franklin의 방법이 가장 효율적이다. 이것은 키 발급에 참여하는 신뢰기관의 수가 가장 적기 때문이다. 반면 추가적으로 순차적인 키 발급절차로 인하여 이병천 등의 방법은 보다 많은 시간을 필요로 한다. 키를 다시 발급하는 경우 제안하는 새 기법의 통신 횟수는 다른 기법과는 다르게 상수이다. 마지막으로 인

표 2. 키 발급에 따르는 비용

		초기 키 발행 시			키 재발행 시		
		[5]	[9]	새로운 기법	[5]	[9]	새로운 기법
Pairing 연산	사용자	$2t$ + 2*	t + 3	($2t$ + 2) + 2	$2t$ + 2*	t + 3	2
	PKG/KPA/KIC	0	$3t$	$2t$	0	$3t$	0
	KGC	-	1	2	-	1	2
	계	$2t$ + 2*	$4t$ + 4	($4t$ + 2) + 4	$2t$ + 2*	$4t$ + 4	4
통신 횟수	사용자	t	t + 1	t + 1	t	t + 1	1
	PKG/KPA/KIC	t	t	t	t	t	0
	KGC	-	1	1	-	1	1
	계	$2t$	$2t$ + 2	$2t$ + 2	$2t$	$2t$ + 2	2
인증 횟수	PKG/KPA/KIC	t	0	t	t	0	0
	KGC	-	1	0	-	1	0
	계	t	1	t	t	1	0

1. t는 Boneh와 Franklin의 방법[5]에서의 개인키 발급에 참여하는 PKG의 수이다. 또한 t는 이병천 등의 방법[9]의 KPA와 제안하는 기법의 KIC의 수와도 같다.
2. Boneh와 Franklin의 방법*은 안전한 통신채널을 가정하지만 이병천 등의 방법과 이 논문에서 제안하는 기법은 안전한 통신채널을 가정하지 않는다. 제안하는 새 시스템에서 안전한 통신채널을 확보하기 위하여 사용하였던 은닉기법을 Boneh와 Franklin의 방법에 적용하는 경우 pairing 연산의 개수는 $4t$ + 2번으로 증가한다.

증 횟수의 관점에서 볼 때, 초기 키 발급 시에는 Boneh와 Franklin의 방법이 가장 효율적이지만 재발급 시에는 제안하는 기법은 인증 절차가 필요 없다. 결론적으로 이 논문에서 제안하는 새 기법은 다수의 키를 발급하는 경우 다른 기법들에 비하여 매우 효율적이다.

다음으로 키 복구에 관하여 논한다. 앞에서 언급한 바와 같이 비교가 되고 있는 모든 기법들은 PKG의 개인키 복구 능력을 제한하고 있다. Boneh와 Franklin의 방법의 경우 threshold 환경에서의 일반적인 비밀 복구 절차에 의하여 개인키를 복구할 수 있다. 이러한 과정은 pairing 연산을 필요로 하지 않는다. KUD가 공개되어 있다고 가정하는 경우 제안하는 기법은 Boneh와 Franklin의 방법과 동일한 방법으로 개인키를 복구할 수 있다. 반면에 이병천 등의 방법의 경우 $t+1$개의 모든 신뢰기관이 순차적으로 키 복구 프로토콜에 참여해야 한다. 이것은 이병천 등의 방법이 다른 기법들에 비하여 보다 많은 시간을 요한다는 것을 의미한다.

제안하는 새로운 키 발급 기법은 다수의 ID에 대한 개인키를 기존의 키 발급 방법[5,9]에 비하여 매우 효율적으로 발행해 준다. 하지만 KUD $d_{\langle ID,T\rangle}$가 공개된 정보이므로, 사용자의 개인키 $D_{\langle ID,T\rangle}=d_{ID}+d_{\langle ID,T\rangle}$가 노출된 경우 동일한 사용자의 KGK d_{ID}에 해당하는 모든 KUD를 도청한 임의의 공격자는 그것들에 대한 개인키를 계산해 낼 수 있다. 따라서 제안하는 시스템에서 사용자는 자신의 개인키가 노출된 경우 KGK와 그에 따르는 KUD를 모두 재발급 받아야 하는 불편함을 갖는다. 앞에서 언급한 것처럼 사용자가 은닉 기법을 이용하여 $d_{\langle ID,T\rangle}$를 안전하게 받는 방법을 사용하면 임의의 공격자가 사용자의 모든 개인키를 복구하는 것을 막을 수 있다. 하지만 사용자의 개인키 $D_{\langle ID,T\rangle}$를 얻은 KUM은 $d_{\langle ID,T\rangle}$를 계산하여 사용자의 KGK를 복구할 수 있다. 또한 이 경우 제안하는 기법은 threshold 특성을 잃게 되며, 사용자의 개인 키 복구를 위해서는 t개 이상의 KIC들과 KUM의 참여를 필요로 하는 단점이 있다.

Ⅵ. 결 론

이 논문에서는 다수의 ID에 대한 개인키를 효율적으로 발행해 주는 새로운 신원기반 암호기법을 제안하였다. 새 시스템에서 사용자의 개인키는 서로 다른 신뢰기관에 의하여 발행되는 두 개의 구성요소인 KGK와 KUD로 이루어진다. 첫 번째 구성요소인 KGK는 사용자의 고유 정보를 입력으로 하여 KIC라는 n개의 신뢰기관에 의하여 threshold 방식으로 발급되며, 두 번째 구성요소인 KUD는 사용자의 고유 정보와 동적인 정보를 입력으로 하여 단일 신뢰기관인 KUM에 의하여 발급된다.

키를 처음 발급하는 경우에 제안하는 새로운 기법은 기존의 다른 신원기반 암호기법의 키 발급 절차와 유사한 효율성을 갖는다. 하지만 KUD를 수정하는 방법으로 사용자에게 다수의 키를 발급하거나 유효기간 만료 등의 이유로 키를 재발급하는 경우, 새로운 기법은 훨씬 효율적이다. 추가적으로 새 기법은 간단한 은닉기법을 사용하여 공개 채널을 통한 키 발급을 가능하게 하였다. 또한 다른 신원기반 암호기법에 비하여 보다 효율적으로 Gentry의 time-slot 기반의 키 철회 기법을 적용할 수 있다.

이 논문에서 제안하는 키 발급 방법은 신원기반 서명이나 계층적 암호기법 등에도 적용될 수 있어서 방법이 신원기반 암호시스템을 보다 실용적으로 만들어 줄 것이라고 믿는다. 특히 새로운 키 발급기법에 의하여 생성된 개인키는 Boneh와 Franklin이 제안하였던 개인키와는 다르다. 비록 Boneh와 Franklin의 방법을 이용하여 만든 많은 암호학적 도구들이 새로운 시스템 설정에 맞게 적용할 수 있지만 추가적인 pairing 연산이 현재로서는 요구된다. 따라서 Boneh와 Franklin의 방법과 동일한 개인키를 발급해 주면서, 보다 안전하고 효율적으로 개인키를 발급해 줄 수 있는 방법에 관한 연구가 필요하다. 또한, 이와 같은 이유로 키 발급 부담의 감소에 비하여 pairing 연산의 부담이 큰 응용에는 제안되는 시스템이 적용되기 어려울 수 있다. 마지막으로, threshold 특성을 유지시켜 주면서 발행된 각 키의 독립성을 유지시켜 주는 방법에 관한 연구가 필요하다.

참 고 문 헌

[1] A. Shamir, "Identity-based Cryptosystems and Signature Scheme," *Advances in Cryptology, Crypto 1984*, LNCS 196, pp. 47-53, 1985.

[2] Y. Desmedt, J. Quisquater, "Public-key Systems based on the Difficulty

of Tam-pering," *Advances in Cryptology. Crypto 1986.* LNCS 263, pp. 111-117, 1987.

[3] H. Tanaka, "A Realization Scheme for the Identity-based Cryptosystem," *Advances in Cryptology. Crypto 1987.* LNCS 293, pp. 341-349, 1988.

[4] U. Maurer, Y. Yacobi, "Non-interactive Public-key Cryptography," *Advances in Cryptology. Crypto 1991.* LNCS 547, pp. 498-507, 1991.

[5] D. Boneh, M. Franklin, "Identity-based Encryption from Weil pairing," *Advances in Cryptology. Crypto 2001.* LNCS 2139, pp. 213-229, 2001.

[6] C. Cocks, "Identity-based Encryption Scheme Based on Quadratic Residues," *Proc. of the 8th IMA Conf. on Crypto-graphy and Coding.* LNCS 2260, pp. 360-363, 2001.

[7] C. Gentry, "Certificate-based Encryption and the Certificate Revocation Problem," *Advances in Cryptology. Eurocrypt 2003.* LNCS 2656, pp. 490-497, 2003.

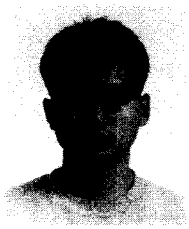
[8] S. Al-Riyami, K. Paterson, "Certificateless Public Key Cryptography," *Advances in Cryptology. Asiacrypt 2003.* LNCS 2894, pp. 452-473, 2003.

[9] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang, S. Yoo, "Secure Key Issuing in ID-based Cryptography," *Proc. of AISW 2004.* CRPIT 32, pp. 69-74, 2004.

[10] 김태구, 염대현, 이필중, "보다 효율적인 Hierarchical ID-based Cryptosystem," 한국정보보호학회논문지 13권 3호, pp. 129-134, 2003.

[11] R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, "Secure Distributed Key Generation for Discrete-Log Based Cryptosystems," *Advances in Crypto-logy. Eurocrypt 1999.* LNCS 1592, pp. 295-310, 1999.

〈著者紹介〉



김 동 현 (Donghyun Kim) 학생회원
2003년 2월: 한양대학교 전자컴퓨터공학부(학사)
2005년 2월: 한양대학교 컴퓨터공학과(석사)
〈관심분야〉 암호기술 응용, 신원기반 암호기법

김 상 진 (Sangjin Kim) 종신회원
1995년 2월: 한양대학교 전자계산학과(학사)
1997년 2월 한양대학교 전자계산학과(석사)
2002년 8월 한양대학교 전자계산학과(박사)
2003년 3월~현재: 한국기술교육대학교 인터넷미디어공학부 조교수
〈관심분야〉 암호기술 응용
URL:http://infosec.kut.ac.kr/sangjin/

오 희 국 (Heekuck Oh) 종신회원
1983년: 한양대학교 전자공학과(학사)
1989년: 아이오와주립대학 전자계산학과(석사)
1992년: 아이오와주립대학 전자계산학과(박사)
1993년~1994년: 한국전자통신연구원 선임연구원
1995년 3월~현재: 한양대학교 컴퓨터공학과 부교수
〈관심분야〉 암호프로토콜, 네트워크 보안
URL:http://infosec.hanyang.ac.kr/~hkoh/

구 본 석 (Bonseok Koo) 정회원
1998년: 경북대학교 전자공학과(학사)
2000년: 포항공과대학교 전자전기공학과(석사)
2000년 9월~현재: ETRI 부설 국가보안기술연구소 근무
〈관심분야〉 정보보호, 공개키 암호 구현

유 권 호 (Gwonho Ryu) 정회원
1999년: 포항공과대학교 전자전기공학과(학사)
2001년: 포항공과대학교 전자전기공학과(석사)
2002년 9월~현재: ETRI 부설 국가보안기술연구소 근무
〈관심분야〉 정보보호, 네트워크 보안, PKI