通信情報保護論文誌
第9券 第4號 1999. 12

공정한 은닉 KCDSA 서명에 기반한 추적 가능한 전자화폐 시스템

오형근*, 이임영*, 김지연**, 박성준**

A Traceable Electronic Cash System based on Fair Blinding KCDSA Signature Scheme

Hyung-geun Oh*, Im-yeong Lee*, Jee-yeon Kim**, Sung-jun Park**

요 약

은닉 서명 방식은 검증자가 보낸 메시지를 서명자가 그 메시지 내용을 보지 못한 채 서명자로부터 메시지 m에 대한 유효한 서명을 받도록 하는 프로토콜을 말한다. 이것은 만약 은행이 상점과 결탁하더라도 동전에 대한 사용자를 추적할 수 없다는 것을 의미한다. 그러나 사용자의 익명성이 보장되는 전자화폐는 또한 돈 세탁, 돈 약탈 그리고 불법적인 구매 행위와 같은 각종 범죄 행동을 용이하게 한다. 이에 본 논문에서는 국내 전자서명 표준인 KCDSA에 기반하여 공정한 은닉 서명을 제안하고 이를 전자화폐 시스템에 적용시킨다. 특히, 제안된 전자화폐 시스템은 은행이 신뢰 기관의 도움으로 전자화폐를 부정한 수단으로 이용한 사용자를 추적할 수 있는 익명성 제어 기능을 가지고 있다.

Abstract

A blind signature scheme is a protocol allowing verifier to obtain a valid signature for a message m from a signer without him seeing the message. This means that the bank, in collaboration with the shop, cannot trace the electronic cash to user. However, anonymous electronic cash also facilitates fraud and criminal acts, such as money laundering, anonymous blackmailing and illegal purchaes. Therefore, in this paper we propose fair blind signature scheme based on KCDSA which is a domestic digital signature scheme and it apply a electronic cash system. In particularly, a proposed electronic cash system have an anonymity control ability which trace a user who make use a electronic cash illegally in association with a trusted center.

Keyword : Blind Signature, KCDSA, Fair Blind Signature, Electronic Cash, Anonymity Control

* 순천향대학교 컴퓨터학부 정보보호연구실(hgoh@ai-cse.sch.ac.kr, imylee@asan.sch.ac.kr)
** 한국정보보호센터(jykim@kisa.or.kr, chaos@kisa.or.kr)

※ 본 연구는 정보통신부의 대학 S/W 연구센터 지원 사업에 의해 수행된 것임.

I. 서 론

전자화폐 시스템은 현찰 사용시와 마찬가지로 네트워크 상에서 대금 지불시 사용자 익명성을 제공하여야한다. 즉, 현찰을 받는 상점이 사용자의 개인 신상 정보와 화폐 일련 번호 등을 기록해 놓거나 또는 카메라로 녹화해 놓기 전에는 사용자의 사생활은 완벽히 보호가 될 수 있다. 이러한 점으로 인해 사용자들은 현찰 사용시 익명성을 유지 할 수가 있으며 이는 사용자 측면에서 굉장히 중요한 문제이다. 그러나 전자화폐는 발행시 화폐 발행에 필요한 개인 식별 정보를 발행 은행에 제공하기 때문에, 은행이 발급한 전자화폐와 사용자 식별 정보를 연계시킴으로서 사용자를 추적할 수가 있다. 정당한 사용자의 화폐 사용 내역은 알려져서는 안되며 이러한 요구 조건이 만족되지 못할 때 사용자는 추적당하게 되고 사용자의 사생활은 침해받게 된다.

정보화 사회에서 개인의 사생활 보호는 중요한 이슈가 되고 있으며 이에 대한 요건이 만족되지 못할 때 전자화폐는 사용자로부터 외면당하게 될 것이다. 이는 전자상거래 확대에 장애물이 될 수 있다. 따라서 전자화폐 사용자의 사생활은 보호되어야 하며 그것은 사용자와 사용자 구입내용과의 관계가 어느 누구에 의해서도 추적 불가능해야 한다는 것을 의미한다. 즉, 은행과 상점이 결탁하더라도 전자화폐의 사용자를 추적할 수가 없어야 한다. 이를 위해 1982년 D.Chaum은 은닉 서명[1]을 처음으로 제안하였다. 이 은닉 서명은 기존하는 서명의 새로운 형태로서 개인의 프라이버시를 보호하기 위한 것이다. 그러나 은닉 서명은 완전한 익명성을 제공하기 때문에 돈 세탁, 약탈 그리고 불법 무기 구매와 같은 불법적인 범죄 행위들에 이용이 될 수 있으며 이때 사용자를 추적할 수 없도록 한다. 이를 방지하기 위해 일정한 조건 아래에서 익명성을 제어하기 위한 연구들이 진행되어 오고 있으며 여러가지 익명성 제어 방식들이 제안되고 있다. 본 논문에서는 KCDSA[2]에 기반한 새로운 공정한 은닉 서명을 제안하고 이를 전자화폐 시스템에 적용시킨다. 제안한 전자화폐 시스템은 사용자의 익명성을 보장하면서 부정 사용시 신뢰 기관과 은행의 협력으로 사용자를 추적할 수 있는 익명성 제어 기능을 가진다.

II. 은닉서명 연구 동향

1. 은닉 서명 방식

D.Chaum이 전자화폐를 사용하는데에 있어서 사용자의 프라이버시를 보호하기 위해 은닉 서명이라는 새로운 서명 방법을 제시한 이후로 전자화폐에 있어서 기본 요구 사항인 '정당한 사용자에 대한 불추적성', 즉 '프라이버시 보호'를 만족시키기 위해 여러 가지 형태의 은닉 서명 방식이 제안되어 오고 있다. 전자화폐에서 사용되는 은닉 서명 방식은 크게 RSA 방식에 근거한 기법[3]과 이산대수 문제를 이용하는 ElGamal 방식에 근거한 기법[4]으로 나누어 볼 수가 있다. 이들은 모두 메시지 복원형 방식을 취하고 있다. 이것은 은행으로부터 서명을 받으려는 데이터 자체가 계좌 번호라든지 혹은 사용자 식별값 등으로 사용자의 신원을 확인할 수 있는 값들이거나, 추후에 은행이나 기타 추적 기관에 의해 사용 후 추적 당할 수 있는 데이터들이기 때문에 은행의 서명시 부가형을 사용하지 못하고 복원형을 사용하게 된다.

1.1 RSA 암호 시스템을 이용한 방식

RSA 방식에 근거한 기법은 D.Chaum이 은닉 서명 방식을 처음 제안한 이후 초기에 이러한 방식을 이용한 여러 제안 방식들이 나왔다. 이러한 종류의 은닉 서명 방식을 이용하는 전자화폐 프로토콜은 D.Chaum[5], T.Okamoto-K.Ohta[6], T.Okamoto[7], M.Jakobsson-M.Yung[8] 등에 의해 제안이 되었다. RSA 방식을 이용하는 은닉 서명 방식들은 송신자가 서명을 받기 위한 메시지가 아닌 다른 변형된 메시지를 서명자에게 보내더라도 이를 서명자가 알 수가 없다는 문제점을 해결하기 위해 일반적으로 cut-and-choose 방식을 사용하고 있다. 그러나 cut-and-choose 방식은 통신량이 많이 발생하고 전자화폐 구성 항(term)이 많아지기 때

문에 효율성 측면에서 문제점을 안고 있다. 이러한 문제점을 해결하기 위해 T.Okamoto[7]는 이산대수 문제에 기반한 새로운 Bit Commitment를 사용하는 zero-knowledge 방법, 그리고 Range Bounded Commitment(RBC)의 개념을 형식화하고 이산대수 가정에 근거한 방법을 사용하고 있다.

1.2 ElGamal 서명 형태를 이용한 은닉 서명 방식

1976년에 Diffie와 Hellman은 공개키 암호와 디지털 서명 방식에 대한 개념을 제안[9]하였다. 이와는 별도로 Merkle는 키 분배 방식에 대한 개념을 제안하였으며 이는 개방 통신로 상에서 두 통신 상대자들이 암호키를 비밀로 공유할 수 있는 방식이였다. 이를 Diffie와 Hellman이 이산대수 문제를 이용하여 실현하였으며 이러한 이산대수 문제를 이용하는 최초의 서명 방식을 ElGamal이 제안[4]하였다. 이 방식에서 서명문은 RSA 방식에서의 소수 p와 같은 크기를 가짐으로서 서명문의 크기가 RSA 방식의 약 2배 정도로 크고 계산량은 약 4배 정도 많다. 이러한 단점을 개선하여 Schnorr는 p-1의 소인수 q를 위수로 갖는 생성을 사용하고 있다[10].

이산대수를 이용한 방식으로는 1992년 Chaum-Pedersen 방식[11] 이외에 representation problem을 이용하는 제한적인 은닉 서명이라는 새로운 primitive를 제안하고 있는 S.Brands가 제안한 방식[12]과 기존의 DSA서명 방식을 변형한 방식, 그리고 Nyberg-Ruppel 서명 방식에 기반한 방식[13] 등이 있다. 이밖에 대부분의 많은 전자화폐 프로토콜들은 이러한 ElGamal 형태의 서명 방식을 이용한 은닉 서명 방식을 사용하고 있다. 이산대수를 이용하는 방식들은 대부분 디지털 서명의 효율성을 높이기 위해 Schnorr의 서명 방식을 이용하고 있다.

2. 공정 은닉 서명 방식

전자화폐에 있어서 익명성은 개인의 사생활 보호라는 긍정적인 측면 이외에 돈 세탁, 돈 약탈 그리고 마약 구매나 무기 구매 자금 등의 불법적인 구매 행위 등과 같은 각종 범죄 행동을 용이하게 한다[14]. 이것은 전자화폐가 약탈자에 의해 약탈되더라도 그 익명성은 유지하게 되며 약탈자는 이러한 전자화폐를 아무런 제재 없이 사용할 수가 있게 되기 때문이다. 이와 같이 익명성이 보장되는 전자화폐는 범죄자들에 의해 이용이 가능하며 비록 작은 금액으로 사용이 되더라도 그 거래량이 굉장히 많아질 수 있기 때문에 전체 통화 시스템에 큰 위험 요소를 제공할 수가 있다. 따라서 다양한 단계로 제공될 수 있는 전자화폐의 익명성은 그 강도가 증가할수록 비례하여 잠재적인 위험성도 증가하게 된다. 이와 같이, 범죄를 예방해야 하는 상황에서 사용자의 익명성을 무조건적으로 보장하는 것은 바람직하지 않으며, 익명성을 가지는 지불 시스템이 정부나 금융기관들에 의해 받아들여지기 위해서는 어떠한 특정한 조건 아래에서 사용자의 익명성을 제어하는 메카니즘을 제공해야 한다. 이에 각국 정부에서는 익명성 조절에 관심을 가지고 화폐 소유자의 프라이버시에 대한 유연성을 갖기 위해 제어 파라메타를 가지는 익명성을 도입하려하고 있다. 따라서 등장하게 된 요구 조건이 익명성 제어(Anonymity Control)[8,15,16,17]이다. 그러나 이 요구 조건 역시 남용하였을 경우 개인의 사생활 침해와 연관될 수가 있기 때문에 사용자 입자에서의 익명성과 추적 기관의 익명성 제어 부분을 절충한 방법이 필요한데, 이로부터 공정성[15,16,17,18,19]이라는 개념이 돌출 된다. 다시 말해, 공정성이란 사용자의 프라이버시를 만족시키면서 동시에 적법한 과정을 통해 익명성을 제어할 수 있는 기능을 말한다.

공정한 지불 시스템(Fair payment systems)이라고도 불리는 익명성 취소 가능한 지불 시스템에 대한 개념은 Solms-Naccache[14]가 처음으로 소개하였으며 Brickell-Gemmell-Kravitz[20]에 의해서 그 방안이 제안이 되었다. 이 방안에서는 전자화폐의 소유자를 식별하는 소유자 추적(Owner Tracing) 개념을 소개하고 있다. 또한 Stadler-Piveteau-Camenisch[21]는 전자화폐의 소유자 추적과 화폐 추적(Cash Tracing)의 두 가지 익명성 취소 모델에 대해 소개하고 있다.

III. 새로운 공정 은닉 서명

본 장에서는 국내 전자서명 표준인 KCDSA를 기반으로 하여 익명성을 유지하며 필요시 신뢰기관의 도움으로 익명성을 제어할 수 있는 새로운 공정 은닉 서명 기법을 제안하고 있다.

1. 시스템 파라메터

시스템 파라메터는 KCDSA에서 사용한 변수들과 같으며 메시지를 은닉시키기 위한 파라메터가 추가된다. 서명문은 서명자가 제공한 파라메터로부터 사용자가 추출한다.

- p : $2^{|p|-1} < p < 2^{|p|}$, $|p| = 512 + 256i$ ($0 \le i \le 6$)의 크기를 가지며 $(p-1)/2q$ 역시 소수이거나 최소한 q보다 큰 소수들의 곱으로 구성되는 소수.
- q : $p-1$을 나누는 소수로 $2^{|q|-1} < q < 2^{|q|}$, $|q| = 128 + 32j$ ($0 \le j \le 4$)의 크기를 가짐.
- g : $a^{(p-1)/q} \bmod p$, $1 < a < p-1$이고, $a^{(p-1)/q} \bmod q > 1$을 만족.
- α, β : 검증자가 선택하는 은닉 인자. $\alpha, \beta \in {}_R Z_q^*$.
- $h(\cdot)$: $|q|$ 비트 길이의 출력값을 갖는 충돌저항성의 해쉬함수.
- x : $0 < x < q$인 비공개 서명키.
- y : $y \equiv g^x \bmod p$로 계산되는 공개 검증키.
- $Cert\_Data$: 서명자의 ID, 시스템 변수 p, q, g와 공개 검증키 y 등을 포함하는 공개키 확인서의 생성에 이용되는 사용자의 인증 데이터.
- Z : $Cert_Data$의 해쉬코드이다. 즉, $Z = h(Cert\_Data)$.
- I : 서명될 메시지로서 사용자 식별값이며 $0 < I < p$.
- H : 은닉된 식별값 I'과 Z의 해쉬코드값. 즉, $H = h(Z \,||\, I')$. 여기서, $I' \equiv \gamma\, \beta^{-1} \bmod q$이다.
- k' : $0 < k' < q$인 일회용 난수값.
- r : 서명의 첫 부분으로 $r \equiv I g^{\alpha} r'^{\beta} \bmod p$. 여기서, r'은 서명 단계에서 서명자가 생성한 일회용 난수 k'을 이용하여 다음과 같이 생성되는 값이다. $r' \equiv g_1^{k'} \bmod p$.
- E : $E \equiv (I' + H) \bmod q$로 계산되는 서명과정 중의 중간값.
- s : 서명의 두 번째 부분에 사용될 부분으로 $s' \equiv (xE + k') \bmod q$로부터 검증자가 추출. 즉, $s \equiv (s'\beta + \alpha) \bmod q$.
- 서명 : (r, s)

2. 초기화 단계

이 단계에서는 프로토콜을 시작하기 전에 각 참여 객체가 생성해야 할 파라메터들을 생성한다.

- 사용자
  - ID : 사용자 식별값으로 사용자가 랜덤하게 선택하며 비밀로 보관. $ID \in {}_R Z_p$
  - I : 사용자가 생성하여 서명자에게 등록. $I \equiv (g_1)^{ID} \bmod p$ 여기서, g_1은 GF(p)상의 원시원으로 서명자가 생성하여 공개한 값이다.
- 서명자
  - p, q : KCDSA 서명 방식에서 사용한 소수로서 서명자가 생성하여 공개.
  - g_1, g_2, g_3 : GF(p)상의 원시원으로 서명자가 생성하여 공개.
  - 개인키 : $x \in {}_R Z_p$
  - 공개키 : $y \equiv (g_1)^x \bmod p$
- 신뢰기관
  - 개인키 : $X_T \in {}_R Z_p^*$
  - 공개키 : $y_T \equiv (g_2)^{X_T} \bmod p$

3. 서명 단계

서명 단계를 통해 사용자는 서명자로부터 은닉된 서명문을 얻는다. 이때 사용자의 프라이버시 보호를 위해 KCDSA 프로토콜을 변형하여 은닉 서명문을 생성하며, 부정행위 발생시에 신뢰기관에 의해 사용자의 신원을 검출 할 수 있도록 하기 위해 사용자가 추적 인자를 생성하

여 서명자에게 제공한다. 이때 추적 인자값은 서명자가 공개한 값과 사용자가 생성한 I값, 그리고 신뢰기관의 공개키 값을 이용하여 생성하며 서명자는 사용자가 생성한 추적 인자가 올바르게 생성되었는지 확인한다.

- Step 1

사용자는 랜덤하게 $v \in_R Z_q$를 생성한다. 그리고 추적인자 생성에 사용될 A_1'과 A_2'을 계산하여 서명자에게 전송한다.

$$A_1' \equiv (y_T)^v \bmod p,$$

$$A_2' \equiv I \cdot g_2 \cdot (g_3)^{v^{-1}} \bmod p$$

- Step 2

서명자는 사용자가 A_1', A_2'이 올바르게 생성하였는지 증명과정을 수행한 뒤 $k' \in_R Z_q$를 생성하고 이를 이용하여 다음과 같이 r'을 생성한다. 그리고 r'을 사용자에게 전송한다.

prove :

$$\log_{g_3}(A_2'/Ig_2) = \log_{A_1' y_T}$$

$$r' \equiv g_1^{k'} \bmod p$$

여기서, $\log_{g_3}(A_2'/Ig_2) = \log_{A_1' y_T}$에 대한 증명은 [그림 1]과 같이 v값을 노출시키지 않고 추적인자 생성에 대한 증명을 수행하도록 한다. 그와 같은 증명은 G.Davida, Y.Frankel, Y.Tsiounis, 그리고 M.Yung[16] 등에서 언급되고 있으며 그것들은 knowledge에 대한 Schnorr 증명에 기반하고 있다[10].

- Step 3

사용자는 자신의 식별값 I를 은닉시키기 위한 은닉 인자 $\alpha \in_R Z_q^*$와 $\beta \in_R Z_q^*$를 생성하고 이 파라메터와 서명자가 전송한 r'을 이용하여 r값을 계산한다.

$$r \equiv Ig_1^{\alpha} r'^{\beta} \bmod p$$

다시 이 r값과 은닉 인자 β를 이용하여 s값을 구하기 위해 은닉된 값 I'을 계산하여 서명자에게 전송한다.

$$I' \equiv r\beta^{-1} \bmod q$$

prove $\log_{g_3}(A_2'/Ig_2) = \log_{A_1' y_T}$

$\delta \in_R Z_q$

$a \equiv (y_T)^{\delta} \bmod p$

$b \equiv (g_3)^{\delta} \bmod p$

$A_1' \equiv (y_T)^v \bmod p$

$A_2' \equiv I \cdot g_2 \cdot (g_3)^{v^{-1}} \bmod p$

$r = wv + \delta$

$a,\ b,\ A_1',\ A_2'$ →

← w

$w \in_R Z_q$

r →

verify :

$(y_T) \stackrel{?}{=} a \cdot A_1'^{w}$

$(g_3)^{-r} \stackrel{?}{=} (A_2'/Ig_2) \cdot b$

[그림 1] 추적인자 확인 과정

[Fig. 1] Verification of Trace factor

• Step 4

서명자는 사용자로부터 받은 I'과 Cert_Data의 해쉬코드값 Z를 연접하여 새로운 해쉬코드 값 H를 계산하고 이 값과 다시 I'을 이용하여 E값을 계산한다. 그리고 은행의 비밀키 값 x를 이용하여 s'을 다음과 같이 계산하고 이를 사용자에게 전송한다.

$$H = h(Z \| I'),\ E \equiv I' + H \bmod q,$$

$$s' \equiv xE + k' \bmod q$$

• Step 5

사용자는 서명자가 계산하여 보내 준 값 s'으로부터 서명의 두 번째 부분에 사용될 s를 추출

해 낸다. 사용자는 서명자로부터 전송되어 온 s'값을 이용하여 서명문이 올바른 서명문인지 검사하게 된다.

$$s \equiv s'\beta + \alpha \bmod q$$

$$I \stackrel{?}{=} g_1^{-s} y^{r+\beta H} r \bmod p$$

$$= g_1^{-(s'\beta+\alpha)} g_1^{x(r+\beta H)} I g_1^{\alpha} r'^{\beta} \bmod p$$

$$= g_1^{-xE\beta - k'\beta - \alpha + xr + x\beta H + \alpha + k'\beta} I \bmod p$$

$$= g_1^{-x\beta I - x\beta H - k'\beta - \alpha + xr + x\beta H + \alpha + k'\beta} I \bmod p$$

$$= g_1^{-x\beta I + xr} I \bmod p$$

$$= g_1^{-x\beta r\beta^{-1} + xr} I \bmod p$$

$$= I \bmod p$$

여기서, H는 인출단계를 수행하기 전에 서명자로부터 받은 Z값을 이용하여 계산한다.

$$H = h(Z \| I')$$

그리고 나서, 서명문을 구성하기 전에 사용자는 공개 파라메터 p, y_T, g_2와 Step 1에서 생성한 v, A_2'를 바탕으로 신뢰기관에 의해 추적인자로 사용될 추적 파라메터 A, A_1, A_2, A_3를 생성하여 서명문을 구성한다.

$$A \equiv (A_2' \cdot y_T)^v \cdot I \bmod p,$$

$$A_1 \equiv g_2^v \bmod p,\ A_2 \equiv I^v \bmod p,$$

$$A_3 \equiv I \cdot (y_T)^v \bmod p$$

서명문 : $[(r, s), A, A_1, A_2, A_3]$

이때, 추적 파라메터에 대한 유효성 검증은 수신자에 의해 이루어지며 유효하지 않을 경우 서명문의 수신은 거부가 된다 .

$$A \stackrel{?}{=} A_1 \cdot A_2 \cdot A_3 \cdot g_3 \bmod p$$

3.4 사용자 신원 검출 단계

사용자가 은닉된 서명문을 부정하게 사용한 경우에 있어서 사용자가 제시한 은닉 서명문을 받은 서명자는 사용자 신원을 파악하기 위해 추적인자 A_1, A_3를 신뢰기관에 제공함으로써 사용자 추적이 이루어진다.

• Step 1 : 서명문을 수신한 서명자는 서명문으로부터 추적인자 A_1, A_3를 신뢰기관에 전송한다.

• Step 2 : 신뢰기관은 다음과 같이 A_1과 A_3로부터 $A_3' \equiv I^{X_T^{-1}} \cdot g_2^v \bmod p$을 구하고, 신뢰기관의 비밀키 X_T를 이용하여 다시 I를 계산한다.

$$A_3' \equiv A_3^{X_T^{-1}} \bmod p \equiv I^{X_T^{-1}} \cdot g_2^v \bmod p$$

$$A_3'/A_1 \bmod p \equiv I^{X_T^{-1}} \cdot g_2^v / g_2^v \bmod p$$

$$\equiv I^{X_T^{-1}} \bmod p$$

$$\therefore\ I = (I^{X_T^{-1}})^{X_T} \bmod p$$

IV. 새로운 전자화폐 시스템

제안된 방식은 크게 시스템 초기화 단계, 인출 단계, 지불단계 그리고 예치단계의 기본적인 4단계와 그리고 부정행위 발생시에 부정 행위자를 추적할 수 있는 추적단계로 구성이 된다. 그리고 인출단계에서 사용자의 프라이버시를 보호하기 위해 3장에서 제안한 KCDSA에 기반한 공정한 은닉 서명을 적용하여 전자화폐를 인출한다. 먼저 각 개체는 시스템을 시작하기 전에 다음과 같은 초기화 단계를 수행하여 각 단계에서 사용할 파라메터를 생성한다.

사용자		서명자
$v \in_R Z_q$		
$A_1' \equiv (y_T)^v \bmod p$,		
$A_2' \equiv I \cdot g_2 \cdot (g_3)^{v^{-1}} \bmod p$		
	prove	
	$\log_{g_3}(A_2'/Ig_2) = \log_{A_1} y_T$	
		$k' \in_R Z_q$,
$\alpha \in_R Z_q^*$, $\beta \in_R Z_q^*$	r'	$r' \equiv g_1^{k'} \bmod p$
$r \equiv Ig_1^{\alpha} r'^{\beta} \pmod p$		
$I' \equiv r\beta^{-1} \pmod q$	I'	
		$H = h(Z \parallel I')$,
		$E \equiv I' + H \bmod q$,
		$s' \equiv xE + k' \bmod q$
$s \equiv s'\beta + \alpha \bmod q$	s'	
$I \stackrel{?}{=} g_1^{-s} y^{r+\beta H} r \bmod p$		
$A \equiv (A_2' \cdot y_T)^v \cdot I \bmod p$		
$A_1 \equiv g_2^v \bmod p$, $A_2 \equiv I^v \bmod p$,		
$A_3 \equiv I \cdot (y_T)^v \bmod p$		
서명문		
: $[(r, s), A, A_1, A_2, A_3]$		

[그림 2] KCDSA에 기반한 공정한 은닉 서명
[Fig. 2] Fair blind signature based on KCDSA

1. 초기화 단계

- 사용자(u_i)
  - ID_{u_i} : 사용자 식별값으로 사용자가 랜덤하게 선택하며 비밀로 보관한다. $ID_{u_i} \in_R Z_p$
  - I_{u_i} : 사용자가 생성하여 은행에 계좌 개설시에 등록하는 값으로 사용자 식별값 ID_{u_i}을 이용하여 생성한다. $I_{u_i} \equiv g_1^{ID_{u_i}} \bmod p$
  - w_i : i번째 사용자가 $i+1$번째 사용자에게 양도한 금액.
  - C : 사용자가 은행으로부터 발급 받은 전자화폐 데이터.
  - $sign_B(w)$: 사용자가 은행으로부터 발급 받은 전자화폐 금액(w)에 은행이 서명한다.
- 은행
  - p, q, $Cert\_Data$, $h(\cdot)$, Z, k' : 3장에서 제안한 공정한 은닉 KCDSA 서명 방식에서 사용한 파라메터들
  - g_1, g_2, g_3 : GF(p)상의 원시원으로 은행

이 생성하여 공개한다.

· 개인키 : $x \in_R Z_p$

· 공개키 : $y \equiv (g_1)^x \bmod p$

· H : 해쉬코드 값으로 $H = h(Z \| I_{u_i}')$이다.

· TS : Time Stamp

● 추적기관($Trustee$)

· 개인키 : $X_T \in Z_p^*$

· 공개키 : $y_T \equiv (g_2)^{X_T} \bmod p$

2. 인출단계

인출단계를 통해 사용자는 은행으로부터 전자화폐(C)를 발급받는다. 이때 사용자의 프라이버시 보호를 위해 3장에서 언급하였던 공정한 은닉 KCDSA 프로토콜을 이용하며, 부정행위 발생시에 법원의 명령에 의해 은행과 신뢰할 수 있는 추적기관이 추적할 수 있도록 하기 위해 사용자가 추적 인자를 생성하여 은행에게 제공한다. 이때 추적 인자값은 은행이 공개한 값과 사용자가 생성한 I_{u_1}값, 그리고 추적기관의 공개키 값을 이용하여 생성하며 은행은 사용자가 생성한 추적 인자가 올바르게 생성되었는지 확인한 뒤 전자화폐를 발행한다.

● Step 1

사용자는 랜덤하게 $v \in_R Z_q$를 생성한다. 그리고 다음과 같이 나중에 추적인자 생성에 사용될 $A_{u_1 1}'$과 $A_{u_1 2}'$을 생성하여 은행에 전송한다.

$$A_{u_1 1}' \equiv (y_T)^v \bmod p,$$

$$A_{u_1 2}' \equiv I_{u_1} \cdot g_2 \cdot (g_3)^{v^{-1}} \bmod p$$

● Step 2

은행은 [그림 1]과 같이 $A_{u_1 1}'$, $A_{u_2 2}'$를 올바르게 생성하였는지 확인한 뒤 $k' \in_R Z_q$를 생성하고 이를 이용하여 r'을 생성한다. 그리고 r'을 사용자에게 전송한다.

prove :

$$\log_{g_3}(A_{u_1 2}' / I_{u_1} g_2) = \log_{A_{u_1 1}'} y_T$$

$$r' \equiv (g_1)^{k'} \bmod p$$

● Step 3

사용자 u_1은 사용자 식별값 I_{u_1}을 은닉시키기 위한 은닉 인자 $\alpha \in_R Z_q^*$와 $\beta \in_R Z_q^*$를 생성하고 이것과 은행이 전송한 r'을 이용하여 서명의 첫 부분으로 사용될 r값을 계산한다.

$$r \equiv I_{u_1} \cdot (g_1)^{\alpha} \cdot r'^{\beta} \bmod p$$

다시 이 r값과 은닉 인자 β를 이용하여 은닉된 값 I_{u_1}'을 은행에 전송한다.

$$I_{u_1}' \equiv r\beta^{-1} \bmod q$$

● Step 4

은행은 사용자로부터 받은 I_{u_1}'과 Z를 연접하여 해쉬코드 값 H를 계산하고 이 값과 다시 I_{u_1}'을 이용하여 E값을 계산한다.

$$H = h(Z \| I_{u_1}'), \quad E \equiv I_{u_1}' + H \bmod q$$

그리고 은행의 비밀키 값 x를 이용하여 s'을 계산하고 사용자의 계좌로부터 인출한 인출 금액 w에 은행의 서명을 한 뒤 s'과 $sign_B(w)$를 사용자에게 전송한다.

$$s' \equiv xE + k' \bmod q$$

● Step 5

사용자는 은행이 계산하여 보내 준 값 s'으로부터 s를 추출해 낸다.

$$s \equiv s'\beta + \alpha \bmod q$$

Step 3에서 계산한 r과 s'으로 추출한 s값이 서명문으로 구성되며 이 서명문을 이용하여 전자화폐 C를 구성한다. 사용자는 은행으로부터 전송되어 온 s'값을 이용하여 서명문이 올바른 서명문인지 검사하게 된다.

서명문 : (r, s)

서명문 검사 : $I_{u_1} \stackrel{?}{=} g_1^{-s} y^{r+\beta H} r \bmod p$

여기서, H는 인출단계를 수행하기 전에 은행으로부터 받은 Z값을 이용하여 계산한다.

$$H = h(Z \| I_{u_1}')$$

그리고 나서, 화폐를 구성하기 전에 사용자는 3장에서와 같이 공개 파라메터 p, y_T, g_2와 Step 1에서 생성한 v, $A_{u_1 2}'$를 바탕으로 신뢰기관에 의해 추적 인자로 사용될 추적 파라

메터 A, A_1, A_2, A_3를 생성하여 화폐를 구성한다.

$$A \equiv (A_{u_i 2}' \cdot y_T)^v \cdot I_{u_i} \bmod p$$

$$A_{u_i 1} \equiv g_2^v \bmod p.$$

$$A_{u_i 2} \equiv (I_{u_i})^v \bmod p.$$

$$A_{u_i 3} \equiv I_{u_i} \cdot (y_T)^v \bmod p$$

서명문이 올바른 서명문으로 확인이 되면 전자화폐 C를 다음과 같이 구성한다.

$$C = [(r, s), A_{u_i}, A_{u_i 1}, A_{u_i 2}, A_{u_i 3}]$$

3. 지불단계

사용자 u_1이 전자화폐 C를 이용하여 인출금액보다 적은 금액 w_1 ($\leq w$)를 상점 V에게 지불하기 원한다고 가정하면 다음과 같은 단계를 수행한다.

- Step 1

사용자는 먼저 $\theta \in {}_R Z_p^*$를 선택하고 대금지불의 유효성을 확인하기 위한 파라메터 $V_{u_i 1}$, $V_{u_i 2}$을 계산한다.

$V_{u_i 1} \equiv (g_2)^\theta \bmod p$, $V_{u_i 2} \equiv g_1^{ID_{u_i} \cdot \theta} \bmod p$

그리고 은행으로부터 받은 전자화폐 C와 화폐금액 w, 지불하기 원하는 금액 w_1을 연접시키고 이 값에 사용자 u_1의 서명을 수행한다.

$$T_{u_i} = sign_{u_i}(C \| w_1 \| sign_B(w))$$

또한 $B_{u_i} = [B_{u_i 1}, B_{u_i 2}]$을 계산하여 대금 지불을 위해 이 값들을 모두 상점에 전송한다.

$$B_{u_i 1} \equiv (g_1)^\alpha \bmod p,$$

$$B_{u_i 2} \equiv (g_2)^\beta \bmod p,$$

$$(V_{u_i 1}, V_{u_i 2}, B_{u_i}, w_1, C, T_{u_i})$$

- Step 2

상점은 사용자가 지불한 금액의 유효성을 검사하기 위해 우선 T_{u_i}에 대한 사용자 u_1의 서명을 확인하고 C에 포함된 사용자의 추적인자 값들에 대한 유효성 검증을 수행한다. 만약, 이때 추적인자 값들이 유효하지 않을 경우 전자화폐의 수신은 거부가 된다.

$$A \stackrel{?}{=} A_{u_i 1} \cdot A_{u_i 2} \cdot A_{u_i 3} \cdot g_3 \bmod p$$

그리고 유효하다면 사용자가 상점에 보낸 파라메터들의 유효성을 확인하기 위해 challenge 값으로서 d값을 계산하여 사용자에게 전송한다.

$$d = h(V_{u_i 1}, V_{u_i 2}, B_{u_i}, C, T_{u_i}, TS)$$

- Step 3

사용자는 상점이 전송해 온 d값을 이용하여 response값으로 r_1'과 r_1''을 계산하여 상점에 다시 전송한다.

$$r_1' \equiv d \cdot ID_{u_i} \cdot \theta + \alpha \bmod p$$

$$r_1'' \equiv d\theta + \beta \bmod p$$

- Step 4

상점은 사용자가 보내 온 r_1'과 r_2'' 값을 이용하여 Step 1에서 사용자가 전송한 파라메터들에 대한 유효성을 확인한 뒤 유효하다면 사용자가 지불한 금액을 받아들인다.

$$g_1^{r_1'} \stackrel{?}{=} (V_{u_i 2})^d B_{u_i 1} \bmod p$$

$$g_2^{r_1''} \stackrel{?}{=} (V_{u_i 1})^d B_{u_i 2} \bmod p$$

4. 예치단계

상점 또는 최종 전자화폐 수신자는 자신이 받은 전자화폐 w_i을 은행에 전송하기 위해서 거래내역서 D를 은행에 전송한다.

$$D = (C, sign_B(w_i), I_{u_i}, T_{u_i}, V_{u_i 1}, V_{u_i 2}, B_{u_i}, r_i', r_i'')$$

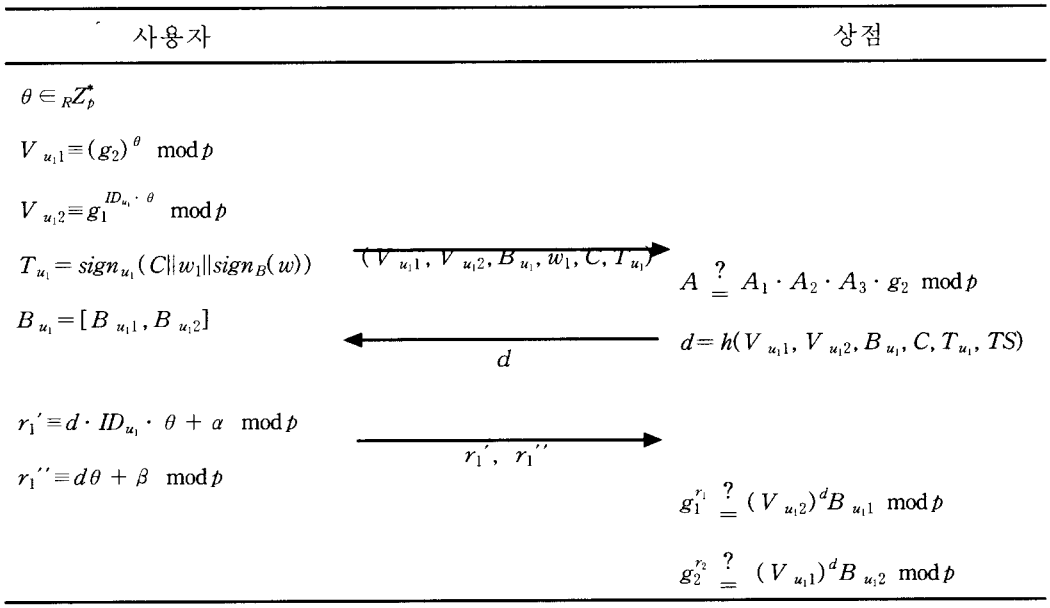
은행은 자신이 받은 전자화폐들의 합계액이 $\sum w_i \leq w$인지를 판단한다.

V. 제안 방식의 고찰

1. 전자화폐의 양도

사용자 u_2(대금 지불인)가 다른 사용자 u_3

(전자화폐 수취인)에게 자신이 받은 전자화폐 2. 익명성 제어



[그림 3] 지불 단계
[Fig. 3] Payment Step

w_1보다 작은 전자화폐 w_2를 전송하기 위해서 사용자 u_2와 사용자 u_3는 4장에서 수행한 지불 단계를 수행한다. 여기서 사용자 u_2는 $(V_{u_1 1}, V_{u_1 2}, B_{u_1}, w_2, C, T_{u_2})$를 사용자 u_3에게 전송한다. 나머지 지불 단계는 동일하며 유효성을 검사해서 일치하면 지불을 받아들인다. 여기서 초기 사용자 u_1이 받은 w만큼의 금액은 w가 넘지 않는 한도 내에서 지불인의 서명 아래에서 사용된다. 이때 i번째 사용자는 자신이 받은 w_{i-1} 금액보다 적은 금액을 사용해야 하며 그 한도액과 사용하고자 하는 금액이 T_{u_i}에 들어가게 되어 i+1번째 사용자가 전자화폐를 사용한 i번째 사용자의 서명을 확인함으로써 금액의 초과 사용을 검증할 수가 있다. 또한 같은 금액을 여러 번 다른 사용자에게 사용하더라도 해당 거래 내역서를 수신한 전자화폐 발행 은행에서 중복 사용된 전자화폐 데이터가 예치되면 T_{u_i}값들을 확인함으로써 부정한 사용자의 서명을 검사를 통해 전자화폐를 중복사용한 사용자를 밝혀낼 수가 있다.

익명성 제어는 사용자가 부정사용 하였을 경우에 거래 내역서에 포함된 추적인자를 통해 사용자 식별자를 드러내거나 또는 화폐 사용시 부가되는 화폐 고유 식별값을 드러냄으로서 이루어진다. 이러한 익명성 제어 모델은 크게 두 가지로 구분해 볼 수가 있다. 하나는 전자화폐의 소유자를 식별하는 소유자 추적(Owner Tracing)과 은행으로부터의 화폐 인출을 식별하기 위한 동전 추적(Coin Tracing)이 있다. 소유자 추적에 있어서 익명성 제어 파라메타는 추적기관이 지불이 이루어지고 난 후, 화폐의 소유자를 판별해 낼 수 있도록 해준다. 이것의 목적은 지불이 이루어지고 난 후에 많은 화폐 유통들에 대해 합법적인 단속 요구로 이중 사용이나 위·변조와 같은 불법 사용이 일어나지 않았더라도 추적하는 것을 가능하게 해준다. 반면에 화폐의 일련번호를 추적하는 것과 유사한 동전 추적은 물건을 구입하기 전에 추적하는 기능을 제공한다. 화폐 추적에 있어서 추적기관은 은행으로부터 인출된 화폐를 확인하고 물품 구입에 사용한 것과 인출된 화폐를 연결시킬 수가 있다.

2.1 화폐 추적

화폐 추적 기능을 통해 추적기관은 법원의 명령을 통해 사용자가 전자화폐를 사용하기 전에 은행에 추적 기능을 부여 할 수가 있다. 즉, 추적기관은 인출 단계에서 사용자가 은행에 전송한 인출 사본 중 A_{u_i1}'으로부터 A_{u_i1}을 생성하고 이를 은행에 재 전송해 줌으로써 은행 측에서는 이를 통해 화폐를 추적한다. 즉, 해당 화폐를 블랙리스트에 올림으로서 상점측에서 인수를 거부하도록 할 수도 있으며, 사용화폐와 인출화폐를 연결함으로써 화폐를 추적할 수가 있다. 화폐 발행 단계에서는 다음 과정을 수행시킴으로서 화폐 추적 기능을 제공한다.

- Step 1 : 은행은 사용자가 제시한 인출 사본 중 A_{u_i1}'을 추적기관에게 제공한다.
- Step 2 : 추적기관은 A_{u_i1}'로부터 A_{u_i1}을 계산해낸다.

$$(A_{u_i1}')^{X_T^{-1}} \equiv [(y_T)^v]^{X_T^{-1}}$$
$$\equiv (g_2)^{X_T \cdot v \cdot X_T^{-1}} \equiv (g_2)^v \equiv A_{u_i1}$$

- Step 3 : 추적기관은 A_{u_i1}을 은행에게 전송한다.

이때 은행은 추적기관이 전송해 준 A_{u_i1}을 사용자가 생성하여 지불 단계에서 상점에 제공하는 A_{u_i1}과 연결시킴으로서 물품을 구입하기 전에 지불의 적법성과 상관없이 추적 기능을 제공한다.

2.2 사용자 추적

사용자 추적 단계는 지불이 성립되고 난 후에 사용자를 판별하는 방법으로서 합법적인 화폐 교환이 이루어지고 난 후에 사용자 추적을 가능케 한다. 이러한 기능을 통해 추적기관은 사용자가 불법적인 물품을 구입한 장소로부터 전자화폐를 찾아 그 화폐의 사용자를 추적하게 된다. 이 단계는 예치 단계에 추가하여 구성되며 사용자가 상점에 대금 지불시 ($A_{u_i3} = I_{u_i} \cdot (y_T)^v \bmod p$)가 추가된다.

- Step 1 : 은행은 상점이 예치한 거래 내역서로부터 A_{u_i1}, A_{u_i3}를 추적기관에 전송한다.
- Step 2 : 추적기관은 A_{u_i1}과 A_{u_i3}로부터 $A_{u_i3}' \equiv (I_{u_i})^{X_T^{-1}} \cdot g_2^v \bmod p$을 구하고, 다시 ID_{u_i}를 계산한다.

$$A_{u_i3}' \equiv (A_{u_i3})^{X_T^{-1}} \bmod p$$
$$\equiv (I_{u_i})^{X_T^{-1}} \cdot (g_2)^v \bmod p$$
$$\equiv (I_{u_i})^{X_T^{-1}} \cdot (g_2)^v / (g_2)^v \bmod p$$
$$A_{u_i3}' / A_{u_i1} \bmod p \equiv ID_{u_i}^{X_T^{-1}} \bmod p$$
$$\therefore\ I_{u_i} \equiv ([I_{u_i}]^{X_T^{-1}})^{X_T} \bmod p$$

- Step 3 : 추적기관은 A_{u_i1}와 A_{u_i3} 그리고 I_{u_i}에 자신의 서명을 한 후 은행의 공개키로 암호화하여 은행에 전송한다.

$$E_{K_B^{r'}}(A_{u_i1} \| A_{u_i3} \| I_{u_i} \| sign_T(A_{u_i1} \| A_{u_i3} \| I_{u_i}))$$

VI. 결 론

은닉 서명은 전자화폐 시스템에서 사용자의 프라이버시를 보호하기 위해 사용되는 중요한 특수 서명이다. 초기의 전자화폐 제안 방식들은 완전한 익명성을 제공하기 위해 연구가 진행되었으나 사용자 익명성만을 강조하여 완전한 익명성을 제공하게 되면 전자화폐의 부정 사용시 해당 전자화폐의 사용을 금지시키거나 부정 사용자를 추적할 수 없게 된다. 따라서 최근에 와서는 익명성을 제어하기 위한 방안들이 연구되고 있다. 특히, 공정성이라는 개념이 도입되어 익명성 제어 기능의 남용으로 사생활 침해의 우려를 방지할 수 있는 전자화폐 시스템이 많이 연구되고 있다. 이에 본 제안 방식에서는 먼저 국내 전자서명 표준인 KCDSA를 기반으로 하여 새로운 공정한 은닉 서명 방식을 제안하고 있다. 제안한 방식은 익명성 제어 파라메터를 서명 생성시에 생성함으로써 나중에 신뢰기관이 서명 사용자에 대한 검출이나 또는 은닉된 메시지를 검출할 수 있도록 하고 있다. 그리

고 이를 전자화폐 시스템에 적용함으로써 은행으로부터 전자화폐를 발행받은 사용자가 돈 세탁이나 불법 구매 자금으로 사용하는 것을 방지하고 있다. 또한 본 제안 방식은 전자화폐 추적과 사용자 추적의 두 가지 방식으로 익명성을 제어하고 있으며 전자화폐 시스템에 선택적으로 적용이 가능하다.

〔참고 문헌〕

〔1〕 D.Chaum, "Blind Signautures for untraceable payments", In Advances in Cryptology, Crypto'82, pp 199-203, 1983

〔2〕 정보통신단체표준, 부가형 전자서명 방식 표준 - 제2부 : 확인서 이용 전자서명 알고리즘, 1998, http://www.tta.or.kr

〔3〕 R.Rivest, A.Shamir and L.Adleman, "A Method for Obtining Digital Signatures and Public-Key Cryptosystem", Communications of the ACM, Vol.21, No. 2, pp.120-126

〔4〕 T.ElGamal, "A Public Key Cryptosystem and A Signature Scheme Based On Discrete Logarithms", Advances in Cryptology : Proceedings of Crypto'84, Springer LNCS 196, pp10-18, 1985

〔5〕 T.Okamoto and K.Ohta, "Universal Electronic Cash", In Advances in Cryptology-Crypto'91, pp324-337

〔6〕 T.Okamoto and K.Ohta, "Disposable Zero-knowledge Authentications and Their Applications to Untraceable Electronic Cash", Proceeding of Crypto'89, pp481-496, 1989.

〔7〕 T.Okamoto, "An Effieient Divisible Electronic Cash Scheme", Advances in Cryptology, Proceeding of Crypto '95, pp438-451, 1995

〔8〕 M.Jakobsson and M.Yung, "Revokable and Versatile Electronic Money", In Proceedings of the third annual ACM security '96, 1996

〔9〕 W.Diffie and M.E.Hellman, "New Directions in Cryptography", IEEE Trans. Info. Theory IT-22, Nov. 1976, pp.644-654

〔10〕 C.P.Schnorr, "Efficient Signature Generation for Smart Cards", Proceeding of Cypto'89, pp. 239-252, 1989

〔11〕 D.Chaum and T.P.Pederson, "Wallet databases with observes", Proproceeding of CRYPTO '92, Springer - Verlay, pp89-105, 1992

〔12〕 S.Brands, "Untraceable off-line Cash in wallets with observers", In Advances in Cryptology-Crypto'93, Proceedings, pp302-318, 1993

〔13〕 J.Camenisch, J.M. Piveteau, and M.Stadler, "Blind signatures based on the discrete logarithm problem", Advances in Cryptology-EUROCRYPT '94, volume 950 of Lecture Notes in Computer Science, page2428-432, 1994

〔14〕 S. von Solms and D. Naccache, "On Blind Signatures and Perfects Crimes", Computers and Security, 11 (1992) pp. 581-583

〔15〕 L.Law, S.Sabett and J.Solinas, "How to make a mint : the cryptography of anonymous electronic cash", No. 96-10-17, National Security Agency, Office of Information Security Research and Technology, Cryptology Division, jun 18 1996.

〔16〕 G.Davida, Y.Frankel and Y.Tsiounis, "Anonymity Control in E-Cash Systems", In Proceedings of the 1st Financial Cryptography conference, Anguilla, BWI,

February24-28, 1997. http://www.ccs.neu.edu/home/hiannis/pubs.html
[17] 오형근, 이임영, "익명성 제어와 화폐 분할 기능을 가지는 효율적인 전자화폐 프로토콜", 정보과학회 논문지(A) Vol 6, No 7, pp.839-846, 1999.
[18] M.Stadler, J-M.Piveteau and J.Camenisch, "Fair Blind Signatures", Advances in Cryptology-Proceedings of Eurocrypto '95, pp.209-219, 1995.
[19] 오형근, 이임영, "전자화폐에 있어서 공정성에 관한 연구", 한국멀티미디어학회 춘계학술발표논문집 Vol 2, No 1, pp.156-161, 1999
[20] E.F.Brickell, P.Gemmell and D.Kravitz, "Trustee-based tracing extension to anonymous cash and the making of anonymous change", In Symposium On Distributed Algorithms(SODA), Albuquerque, NM. Available at http://www.cs.sandia.gov/~psgemme/, 1995
[21] M.Stadler, J.M.Piveteau and J.Camenisch, "Fair blind signatures", In Advances in Cryptology, Eurocrypto '95, pp209-219, 1995

著者紹介

오 형 근

1998년 2월 순천향대학교 전산학과 졸업(학사)
1998년 3월 ~현재 순천향대학교 대학원 전산학과 석사과정 재학중

주관심분야 : 전자화폐, 전자상거래, 암호이론

이 임 영

1981년 홍익대학교 전자공학과 졸업(학사)
1986년 일본 오오사카대학 통신공학부(석사)
1989년 일본 오오사카대학 통신공학과(박사)
1992년 ~1994년 한국전자통신연구원 선임 연구원
1994~ 현재 순천향대학교 컴퓨터학부 교수

김 지 연

1995년 2월 성균관대학교 정보공학과 졸업(공학사)
1997년 2월 성균관대학교 대학원 정보공학과 졸업(공학석사)
1996년 ~ 현재 한국정보보호센터 연구원

주관심분야: 전자화폐, 키관리, 암호 프로토콜

박 성 준(Sung Joon Park) 정회원

1983년 2월 :한양대학교 수학과 졸업
1996년 2월 : 성균관대학교 박사
1985년 1월~1994년 3월 : 한국전자통신연구원
1996년 4월~현재 : 한국정보보호센터 기반기술팀장
〈관심분야〉 암호학, 정보이론