타원곡선 상에서의 익명성 철회 가능한 전자화폐 시스템 설계

김희선, 백준상, 김광조

한국정보통신대학원대학교

Design of revocable electronic cash system over elliptic curve

Heesun Kim, Joonsang Baek, Kwangjo Kim

Information and Communications University

요 약

본 연구에서는 안전성과 효율성을 제공하는 익명성 철회 가능한 전자화폐 시스템의 구현을 위하여 유한체 상에서 설계된 안전한 전자화폐 시스템에 타원곡선 암호 시스템을 적용하여, 전자화폐 프로토콜을 설계하였다. 설계된 전자화폐 프로토콜은 유한체 상의 이산대수 문제에 비교하여 짧은 키의 길이로도 유한체와 동일한 수준의 안전성을 제공하는 타원곡선 상의 이산대수 문제에 안전성을 기반하고 있고 짧아진 키의 길이로 인해 프로토콜 내의 메시지의 길이에 대한 효율성을 증가시킨 특징이 있다.

1 서론

기존의 화폐 개념을 네트워크 상으로 옮겨 디지털화한 무형의 화폐 또는 지불 수단인 전자화폐는 이미 전자상거래 활성화에 중요한 기반기술이 되고 있으며, 그에 대한 연구와 실용화도 빠른 속도로 발전하고 있다. 기존의 신용력에 기반을 두고 종이 화폐가 가지고 있었던 불편함을 해소하기 위해 원격지 이송에 따른 통신기능, 휴대 및 보관 관리의 편리성, 위조 방지 기능을 기존 화폐기능에 추가하여 그 응용의 범위 또한 빠르게 확산되어가고 있다[16].

전자화폐에 대한 연구는 Chaum 등에 의해서 은닉 서명을 이용한 익명성이 제공[11]되면서 빠른 속도로 연구가 진행되어 왔다. Chaum 등이 제안한 시스템은 분할선택 방식을 이용하여 전자화폐의 이중 사용자 추적을 실현하였지만 이 방식은 저장 데이터와 통신량의 증가

로 비실용적인 측면을 가지고 있다. Okamoto와 Ohta가 제안한 전자화폐[7, 8]는 분할성, 양도성의 요구사항을 만족시키면서 전자화폐가 가지는 제한성을 극복함으로써 실물화폐보다 효율적인 기능을 제공하게 되었다. 또한, Brands에 의해서 분할선택 방식의 비효율성이 개선되고 제한적 내용은닉 서명 기법을 이용한 전자화폐 시스템이 제안되었다[9]. Brand의 기법은 Schnorr 디지털 서명과 군 표현 문제를 기반으로 하고 있으며, 사용자 U가 이중 사용한 화폐의 정보 $(r_1, r_2), (r_1', r_2')$를 $(r_1 - r_1') / (r_2 - r_2')$와 같이 계산하여 사용자 정보 u_i를 노출시켜 이중 사용자를 검출한다.

그러나 이와 같이 전자화폐에 제공되는 완전한 익명성을 악용한 범죄의 가능성이 지적되었고[13], 화폐에 대한 블랙메일링, 돈세탁, 키 강탈 등의 문제점이 제시되면서[5, 13, 14], 불법 사용자 추적이 가능한 공정한 지불 시스템(fair payment system)이 제안되었다. 블랙메일링과 돈세탁 방지에 관한 연구[14]를 시작으로, 이러한 공격들을 방지할 수 있는 기법들이 연구되었다[4, 5, 6, 18, 21]. 특히, Camenisch 등이 제안한 시스템[4]은 카운터를 사용하여 다중 지불을 방지하였으나 고객별 사용 빈도 수에 따른 적정 길이 및 순서 유지가 쉽지 않으며 지불과 예치단계가 고객, 상점, 은행이 온라인으로 연결됨으로 인해 오프라인 시스템에 비해 비효율적인 특성을 지닌다. 또한 은행을 전적으로 신뢰해야 하는 프로토콜 설계로 은행의 모함 공격 (framing attack)이 가능하다. 주목할 만한 연구로서 은행을 이용하거나 은행에 가해지는 공격에 대응한 전자화폐 시스템이 Jakobsson과 Yung에 의해 제안[5]되면서 은행강탈 공격의 개념이 소개되었다. 이들은 이중 확인 서명을 통하여 블랙메일링, 돈세탁 등의 공격을 포함하여 강력한 은행의 강탈 공격에 대응하는 시스템을 설계하였고, Challenge semantics 기법을 통해 분할성, 전자수표, 신용카드 구입 등으로의 기능 확장을 용이하게 하였다. 그러나 이들의 제안은 인출단계에서의 옴버즈맨의 온라인 형태의 개입으로 오프라인에 비해 효율성이 낮다는 단점을 지니고 있다. 이러한 시스템의 단점을 Petersen과 Poupard가 강탈 공격에 대응한 오프라인 전자화폐[6]를 제안함으로써 효율적으로 개선시켰다. Jakobsson과 Yung의 시스템은 신뢰기관이 매 인출단계마다 개입되어 통신 비용을 증가시키는데 반해 Petersen과 Poupard가 제안한 시스템은 등록단계에서 사용자가 신뢰기관에게 의사 공개키를 등록시킴으로써 비효율성을 개선시켰으며, 기타 다른 시스템에서 고려하고 있는 공격 모델들을 포함한, 사용자, 상점, 신뢰기관에 대해 발생할 수 있는 비밀키 강탈 공격, 사용자에 대한 모함 공격, 위장 공격, 돈세탁, 블랙메일링, 신뢰기관의 눈속임 공격, 화폐의 강탈 공격 등에 대해 안전하게 설계되었다. 또한, 이를 위해 데이터베이스와 철회 목록(블랙리스트와 화이트리스트)을 구체적이고 효율적으로 사용하였다. 그러나 이것은 시스템이 데이터의 저장공간과 그 메시지 처리를 위한 막대한 양의 비용을 감수할 것을 요구한다.

본 연구의 전자화폐 시스템은 스마트 카드의 사용이 가능한 10만원 미만의 소액거래를 목표로 하고 있다. 오프라인 네트워크형으로서, 블랙메일링, 돈세탁, 화폐의 이중사용 등의 공격에 안전할 수 있도록 익명성 철회가 가능한 시스템을 목표로 하여 안전하면서도 효율적인 전자화폐 프로토콜의 구현에 초점을 두어 설계하였다. 이를 위해, 강탈 공격을 비롯한 다양한 공격 모델에 안전하며, 각 단계 프로토콜과 데이터베이스 및 철회 목록의 구체적인 설

계로 구현이 용이한 Petersen과 Poupard의 전자화폐 시스템[6](이하 PePo97이라 함)을 기반 모델로 삼아, 타원곡선 암호 기법을 적용하여 효율성을 개선한 전자화폐 프로토콜을 설계하였다. 설계된 본 논문의 제안 시스템은 유한체 상에서의 이산대수 문제보다 더욱 어렵다고 알려져 있는 타원곡선 상의 이산대수 문제에 안전성을 기반으로 하고 있다. PePo97 전자화폐 시스템은 비밀키 강탈 공격, 사용자에 대한 모함 공격, 위장 공격, 돈세탁, 블랙메일링, 신뢰기관의 눈속임 공격, 화폐의 강탈 공격 등을 대응하기 위한 오프라인 프로토콜을 구현한 시스템이고 가장 다양한 공격 모델에 대응하고 있는 시스템으로서, 같은 강탈공격에 강한 온라인 프로토콜인 Jakobsson과 Yung의 시스템[5]에 비해서도 보다 효율적인 시스템을 제안하였다. 그러나 PePo97 시스템은 막대한 양의 데이터 저장공간과 통신 메시지를 다루어야 한다는 단점이 있어, 스마트 카드와 같이 저장공간이 제한된 하드웨어 상의 구현에 있어서는 적합하지 못하다. 따라서 본 연구는 PePo97 시스템을 타원곡선 상에서 구현함으로써 유한체 상의 PePo97 전자화폐 프로토콜에 비해 보다 짧은 키 길이로도 동등한 보안 수준을 제공하는 시스템을 설계하여 메시지의 길이에 대해 약 6.4배가량의 개선도를 보였고, 감소된 메시지 길이로 저장 비용에 대한 효율성을 향상시켰다. 설계한 타원곡선 상의 전자화폐 시스템은 위조 불가능성, 이중 사용 방지, 익명성, 추적 불가능성, 완전정보화, 효율성, 익명성 취소, 사용자 추적, 강탈 추적, 연결 불가능성, 환불가능성, 공정성, 모함 방지, 화폐 추적, 속임 방지의 요구사항을 만족시킨다.

본 고에서는 이러한 연구 결과를 제시하기 위해 2장에서 설계한 전자화폐 프로토콜의 구성과 시스템의 보안 요구사항 및 전자화폐의 분류에 대해 기술하고, 3장에서는 본 연구를 위해 기반으로 하고 있는 PePo97 기법을 간략히 설명하고 있다. 또한, 4장에서는 본 연구에 이용된 암호학적 기법과 프로토콜의 타원곡선 상에서의 설계에 대해 다루고 있다. 제안한 시스템에 대한 안전성 분석 및 기본 모델과의 효율성 분석은 5장에서 설명하고 있고, 6장에서 결론을 기술하였다.

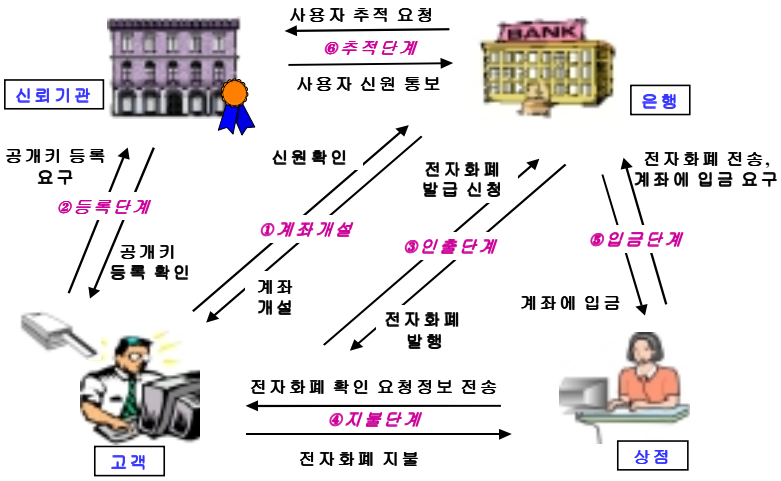
2 전자화폐 시스템의 구성

전자화폐의 불법적인 사용이나 공격 등으로 인하여 발생할 수 있는 위협 요인으로부터 정직한 사용자를 보호할 수 있도록 익명성 철회 가능한 모델을 구현하였다. 시스템의 전체 구성도는 (그림 1)과 같다. 본 전자화폐 시스템은 사용자, 은행, 상점의 기본 구성 요소와 익명성 철회를 위해 요구되는 신뢰기관으로 구성되어 있으며, 각 개체들간에 계좌 개설, 등록단계, 인출단계, 지불단계, 입금단계, 추적단계 프로토콜을 수행한다.

본 연구의 전자화폐 시스템은 스마트 카드의 사용이 가능하며, 통신에 대한 부하가 적은 오프라인 네트워크형의 구현을 목표로 설계되었다. 또한 신용카드를 소지할 수 없는 사람이라도 누구든지 사용 가능하도록 10만원 미만의 소액거래를 위한 전자화폐 시스템을 고려한 것으로 익명성을 제공하는 시스템을 고려하였다. 일반적으로 정상적인 거액 거래(예를 들면, 가옥 매입, 자동차 매입 등)의 경우엔 누구든지 자신의 지불 사실을 익명으로 처리하고 싶

어하지는 않는다. 따라서 이러한 경우의 대부분은 익명성 거래를 요구하지는 않을 것이다. 반면, 소액 거래(예를 들면, 비디오 테이프, 성인 용품 등)의 경우엔 자신의 지불 내역을 익명으로 처리하고 싶은 경우도 많을 것이다. 따라서 소액거래를 위해서는 익명성을 보장한 지불이 요구된다. 그러나, 완전한 익명성의 보장은 돈세탁, 블랙메일링 등의 익명성을 악용한 공격의 가능성이 존재한다[13]. 따라서, 익명성을 보장하되 필요시 익명성을 철회하여 정당한 사용자들을 보호할 수 있어야 하며, 또한 시스템의 안전성을 위해 익명성 철회를 비롯한 돈세탁 방지, 블랙메일링 방지, 속임 프로토콜 참여 방지, 위장 공격 방지, 모함 방지 등의 안전성 요구사항에 대해 고려하고 있는 전자화폐 시스템을 고려하였다. 그리고, 스마트 카드의 이용을 기반으로 하고 있으므로 하드웨어 구현에 적합한 효율성도 고려하였다.

이를 위해 본 연구 시스템은 PePo97의 전자화폐 시스템을 기반 모델로 삼았다. PePo97은 익명성 철회 가능한 전자화폐 시스템으로서 사용자, 은행, 신뢰기관의 비밀키 강탈공격을 비롯한 화폐의 강탈, 모함 공격, 위장 공격, 돈세탁, 블랙메일링, 속임 프로토콜 참여 등의 공격 모델에 대해 안전한 오프라인 시스템이다. 이러한 공격 모델에 대응하기 위한 여러 기법들이 제시되어 왔으나[4, 5, 12, 14, 18, 19, 20, 21], PePo97이 제시하는 다양한 공격 모델들에 강한 시스템은 Jakobsson과 Yung의 시스템[5] 뿐이다. 그러나, Jakobsson과 Yung의 제안은 온라인 시스템을 근간으로 하고 있으므로, PePo97의 오프라인 시스템에 비해서는 효율성이 떨어지며 구체적인 프로토콜을 제시하고 있지 않다. 따라서, 다양한 공격 모델에 강한 오프라인 전자화폐 시스템인 PePo97을 제안 시스템의 기반으로 하게 되었다. 특히, PePo97의 시스템은 구체적이며 단순한 DB의 사용과 각 단계 프로토콜의 설계로 구현을 용이하게 하는 장점이 있다. 반면, DB를 다루기 위한 데이터 저장 비용 및 처리 비용이 높다는 단점이 있어, 시스템에 스마트 카드의 이용을 고려하기 위해서는 메시지 길이를 효과적으로 줄일 필요가 있다. 따라서, PePo97 시스템에 타원곡선 암호 시스템을 적용시켜 메시지의 길이를 효율적으로 줄임으로써 동일한 안전성을 제공하면서도 효율성을 증가시킨 개선된 전자화폐 시스템을 제안하게 되었다.



(그림 1) 시스템 전체 구성도

가. 전자화폐의 요구사항 분석

전자화폐 시스템은 프로토콜의 안전성 측면에 대해, 위조 불가능성, 이중 사용 방지, 익명성, 추적 불가능성, 완전정보화, 효율성의 기본적인 요구사항 이외에도 익명성 취소, 사용자 추적, 강탈 추적, 연결 불가능성, 환불가능성, 공정성, 모함 방지, 화폐 추적, 속임 방지, 초과사용 추적, 분할성, 양도성의 부가적인 요구사항을 필요로 한다. 각 요구사항에 대한 설명은 <표 1>에 기술되어 있다[5, 6, 15].

본 구현 시스템이 만족시키고 있는 요구사항은 <표 1>의 요구사항 I, II에 해당하며, 본 시스템의 안전성에 관한 분석은 본 고의 5장에서 설명하고 있다.

<표 1> 전자화폐 프로토콜의 기능적 측면의 요구 사항

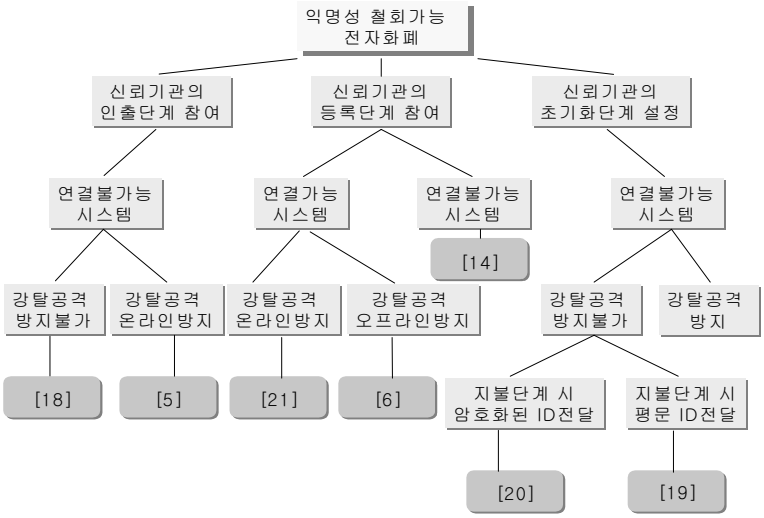
요구사항		설명
요구사항 I	위조 불가능성(Unforgeability)	권한이 부여된 은행만이 전자화폐를 발행권을 가짐
	이중 사용 방지(Double-spending prevention)	같은 전자화폐를 두 번 사용 불가능
	익명성(Anonymity)	은행이 신뢰기관의 협조 없이 화폐와 정직한 사용자의 신원을 연결시키는 것이 불가능
	추적불가능성(Untraceability)	권한이 부여된 익명성 취소 이외에 전자화폐와 사용자 사이의 관계 추적은 불가능
	완전정보화	디지털화, Bit의 기본요소로 구성
	효율성(Efficiency)	저장용량, 통신량, 계산량에 있어서의 효율성
요구사항 II	익명성 취소(Revocability)	은행의 강탈자가 얻은 어떠한 돈도 사용 불가능하도록 공개됨
	사용자 추적(User-tracing):	은행과 신뢰기관은 협동하여 사용된 전자화폐의 사용자를 추적
	강탈추적(Extortion-tracing)	은행과 신뢰기관은 협동하여 사용된 또는 예치된 전자화폐의 연결되는 정보를 계산
	속임 방지(Blindfolded-freeness)	은닉된 특정한 화폐를 은행이 모르는 가운데 은행으로부터 얻는 것이 불가능
	화폐 추적(Coin-tracing)	은행과 신뢰기관은 협동하여 사용될 전자화폐와 예치된 전자화폐와의 정보를 연결
	연결불가능성(Unlinkability)	동일 사용자가 지불한 서로 다른 전자화폐를 연결시키는 것이 불가능
	환불가능성(Refundability)	올바르게 인출된 돈이 은행이나 신뢰기관에 의해 받아들여지지 않은 경우 금액의 환원 가능
	공정성(Fairness)	신뢰기관과 은행의 각각에 대한 사용자의 익명성이 보장되어야 하며, 이 두 기관의 합법적인 협조에 의해서 익명성이 조건적으로 철회 가능
	모함 방지(Framing-freeness)	어떤 사용자나 상점도 은행이나 신뢰기관에 의해서 거짓으로 고소될 수 없음
요구사항 III	초과사용 추적(Overspent-racing)	전자화폐를 초과 사용한 사용자의 신분 추적
	양도성(Transferability)	한번 인출된 전자화폐를 다른 사용자에게 양도
	분할성(Divisibility)	전자화폐의 액면금액을 더 작은 단위의 금액으로 나눔

나. 익명성 철회 가능한 전자화폐 시스템의 분류

익명성 철회 가능한 전자화폐 시스템에 관한 논의는 전자화폐 시스템이 갖는 익명성 제공에 대한 문제점이 지적되면서 다양하게 연구가 되어왔다. 1982년 Chaum이 익명성 전자화폐에 대한 개념[22]을 소개하고, 1987년 익명성을 제공하는 온라인 전자화폐 시스템[10]과 1988년 오프라인 전자화폐 시스템[11]을 제안하면서 익명성을 제공하는 전자화폐 시스템에 대한 연구가 다양하게 진행되었다. 그러나 Solms에 의해서[13] 익명성은 완전한 범죄를 위해 오용될 수 있음이 지적되면서, 화폐에 대한 블랙메일링, 돈 세탁, 비밀키 강탈이나 도난, 혹은 은행이나 신뢰기관의 눈속임 프로토콜의 참여 등 발생 가능한 공격들이 고려되어 왔다[5, 13, 14]. 결국 이러한 공격들을 방지하기 위해 익명성 철회 기법이 제안되었다.

익명성 철회 기법은 발생 가능한 공격들에 대해 화폐의 추적이 인가를 받은 제 삼자, 즉 신뢰기관이나 신뢰기관과 같은 집단에 의해 추적 가능한 기법이다. 블랙메일링과 돈세탁을 방지하는 첫번째 연구는 [12]와 [14]에 의해 제안되었고, 그 이후로 또한 이러한 공격들을 방지할 수 있는 기법들이 연구되었다[4, 5, 6, 18, 21]. 모든 기법들은 계좌개설 단계나 화폐의 인출단계에서 신뢰기관의 참여를 요구하며, 두 개의 시스템만[19, 20]이 시스템의 초기화 단계에서만 신뢰기관의 참여를 요구한다. 그러나 이러한 시스템들은 강탈 공격과 눈속임 프로토콜의 이용을 방지하는 것이 불가능하다. 강탈 공격과 눈속임 프로토콜을 방지할 수 있는 시스템은 [5, 6, 21]이며, [5, 21]의 프로토콜들은 지불단계 프로토콜에서 신뢰기관과의 통신이 요구되는 점이 비효율성으로 지적된다. 특히, 이러한 공격들이 보고되면, 이 기법들은 불법적인 화폐의 사용을 막기 위해 사용자, 상점, 신뢰기관 사이에 지불단계 프로토콜을 요구한다. 그에 비해 [6]의 프로토콜은 오프라인 상에서 인출단계 및 지불단계를 수행하므로, 이들 시스템보다 높은 효율성을 제공한다.

이와 같은 연구에 대한 도표를 (그림 2)에서 나타내고 있다[6]. (그림 2)는 연구 되어온 익명성 철회 가능한 전자화폐 시스템들을 분류해 놓은 그림으로서, 철회 단계 이외에 신뢰기관이 프로토콜에 참여하는 단계별 분류, 지불한 화폐간의 연결성에 따른 분류, 강탈 공격의 방지 여부와 방지 기법에 따른 분류, 지불단계 시 사용자 ID의 암호화여부 등에 따라 연구사례를 분류하고 있다. 본 연구에서 구현하려는 시스템은 (그림 2)에서 표시된 [6]의 기법이다. 그림은 [6]의 기법에 대해 신뢰기관이 철회단계 이외에 등록단계에 참여하고 있으며, 사용자의 같은 의사 공개키에 대해 지불 화폐간의 연결이 가능한 시스템이고, 강탈공격에 대응 가능한 오프라인 시스템임을 설명하고 있다.



(그림 2) 익명성 철회 가능한 전자화폐의 분류

3 PePo97 기법 분석

PePo97 전자화폐 시스템은 오프라인의 익명성 철회 가능한 전자화폐 시스템으로서 다음과 같은 각 단계 프로토콜을 수행한다.

- 초기화 : 시스템 변수와 모든 참여자들의 키 쌍을 생성한다.
- 계좌개설 단계 : 은행이 사용자의 계좌를 개설하고 개인 정보를 등록한다.
- 등록 단계 : 사용자가 의사 공개키를 발행하여 신뢰기관에게 등록한다.
- 인출 단계 : 사용자가 은행 계좌로부터 자신의 사용 장치(PC 혹은 스마트 카드)에 화폐를 인출한다.
- 지불 단계 : 사용자가 자신의 장치에 저장된 화폐를 이용하여 상점에게 지불을 수행한다.
- 입금 단계 : 상점이 은행으로 전자화폐를 전송하고 은행은 상점의 계좌에 입금한다.
- 익명성 철회 : 신뢰기관이 인출 단계의 통신정보로부터 화폐를 계산해내거나 지불정보로부터 사용자의 신원을 계산해낸다.

PePo97은 사용자, 은행, 신뢰기관의 비밀키에 대한 강탈공격, 화폐 강탈, 블랙메일링, 위조 공격, 모함 공격, 위장 공격, 눈속임 프로토콜 참여 등의 다양한 공격 모델에 안전한 시스템이다. PePo97의 시스템은 이러한 요구사항들을, 적합하게 선택된 메시지 공격에 대한 서명의 위조가 알려져 있는 계산량적으로 어려운 문제들(인수분해 혹은 이산대수 문제)과 동등한 문제로 증명된다면, 서명 및 검증 기법이 확률적으로 계산량적으로 안전하다는 정의를 기반으로 하여 증명하고 있다. 강탈 공격에 대한 대응과 오프라인 프로토콜의 구성은 인출 단계 전에 사용자가 신뢰기관에게 의사 공개키를 등록시키는 것과 안전한 철회 목록을 이용하여 가능하게 하였다. 구체적이고 체계적인 DB의 사용으로 구현이 쉽다는 장점은 있으나,

이를 관리하고 다루기 위한 저장 비용 및 시간 비용은 축적되는 메시지로 인해 효율성을 저하시키는 원인이 될 것이다. 따라서, 제한적인 저장 공간을 가진 스마트 카드와 같은 하드웨어 상의 구현을 위해서는 저장 비용을 절약할 필요성이 있다. 이를 위해 PePo97에서는 인터넷 지불 시스템과 전자지갑 시스템의 두 가지 모델을 제시하고 있다. 그래서, 스마트 카드를 사용하는 전자지갑의 모델에서는 메시지 길이를 줄이기 위해 효율적인 암호 알고리즘들을 이용하고 있다. 그러나, 이 모델은 효율성을 위해, 일부 제한적인 프로토콜을 수행하고 있으며, 일부 메시지에만 메시지 길이의 감소를 가져왔다.

따라서, 본 연구에서는 PePo97의 전자화폐 시스템에 대해 동일한 수준의 안전성을 제공하면서도 동시에 효율성을 높일 수 있도록, 이에 타원곡선 암호 기법을 적용시킴으로써 시스템을 효율적으로 개선하였으며, 다음 장에서 그 설계에 대해 설명하고 있다.

4 시스템 설계

이번 장에서는 전자화폐 시스템의 설계에 대해 다루고 있다. 본 전자화폐 시스템은 타원곡선 상에서 설계되어 유한체 상에서 설계된 시스템에 비해 메시지의 길이가 짧아져 보다 효율적이며, 스마트 카드와 같은 하드웨어 기반의 구현에 적합한 시스템이다.

또한 정직한 고객에 대하여 은행과 신뢰기관은 거짓 결탁하지 않을 것을 가정하며, 은행과 사용자는 신뢰기관에 대한 전적인 신뢰를 바탕으로 하고 있다.

가. 설계 시 사용되는 암호학적 기법

전자화폐 프로토콜의 설계에 관한 설명에 앞서, 설계에서 수행하는 암호학적 기법에 대해 설명하고자 한다.

두 실체 간의 신원확인 및 프로토콜의 인증된 메시지의 교환을 위해 키 합의 프로토콜을 수행하여 세션키를 생성한 후, 세션키로 블록 암호화 된 메시지를 전달한다. 본 연구에서는 세션키 생성을 위한 키 합의 프로토콜로서 Diffie-Hellman 기반의 새로운 키 합의 프로토콜[23]을 적용하였다. 또한 은행이 사용자를 추적하는 공격에 대해 안전하도록 전자화폐의 불추적성을 만족시키기 위해 PSLC-2 (Provably Secure Length-saving public-key encryption scheme based on Computational D-H assumption)공개키 암호화 기법[24]을 사용하였다.

1) 키 합의 프로토콜

본 연구에서는 등록단계와 인출단계에서 두 실체 간의 인증된 메시지의 전달을 위해 세션키를 생성하고, 이 세션키를 이용하여 모든 전달 메시지들을 블록 암호화하여 전달한다. 본 연구에 적용시킨 Song과 Kim이 제안한 Diffie-Hellman 기반의 새로운 AK 프로토콜[23]은 묵시적 키 인증성(IKA, Implicit Key Authentication) 알려진 키에 대한 안전성(K-KS, Known-Key security), 전향적 보안성(FS, Forward Secrecy), 키 위장(K-CI, Key-Compromise Impersonation)에

대한 안전성, 미지의 키 공유(UK-S, Unknown Key-Share)에 대한 안전성을 만족하는 프로토콜이다. 다음은 이들이 제안한 새로운 AK 프로토콜의 타원곡선 상에서의 구현 기법을 기술하였다.

● 표기 정의

A, B	정직한 실체들
E(GF(p))	타원곡선
#E(GF(p))	*E(GF(p))*의 위수
P	*E(GF(p))* 상의 위수 *q*인 점(기저점)
q	*#E(GF(p))*를 나누는 큰 소수
x_A, x_B	*A, B*의 장기 개인키로서, $x_A, x_B \in_R [2, q-1]$
Q_A, Q_B	*A, B*의 장기 공개키로서, $Q_A = x_A \cdot P, x_B = Q_B \cdot P$
k_A, k_B	*A, B*의 임시 개인키로서, $k_A, k_B \in_R [2, q-1]$
R_A, R_B	*A, B*의 임시 공개키로서, $R_A = k_A \cdot P, R_B = k_B \cdot P$
F	키 유도 함수 (key derivation function)
w	*cofactor,* *w= #E(GF(p))/q*

● 공개키 검증

공개키 $Q = (x_Q, y_Q)$ 는 다음과 같은 단계를 수행함으로 검증될 수 있다.

① Q는 O와 같지 않다.

② x_Q 와 y_Q 는 *GF(p)*상의 원소이다.

③ Q는 타원곡선 E의 방정식을 만족한다.

④ $qQ = O$

단기 키에 대한 검증의 비용을 줄이기 위해, ④는 키 검증단계에서 생략될 수도 있다. 이때의 키 검증을 embedded 공개키 검증이라 한다[28]

● 프로토콜

① A는 $k_A \in_R [2, q-1]$ 를 선택하고, B에게 $R_A = k_A \cdot P$ 와 $Cert_A$를 전송한다.

② B는 $k_B \in_R [2, q-1]$ 를 선택하고, A에게 $R_B = k_B \cdot P$ 와 $Cert_B$를 전송한다.

③ A는 R_B에 대한 embedded 공개키 검증을 수행하고, 검증이 실패하면, A는 프로토콜을 실패로 종료하며, 그렇지 않으면, A는 공유 비밀정보 $Z = (k_A \cdot Q_B) + ((x_A + k_A) \cdot R_B)$ 과 $k = F(w \cdot Z)$ 를 계산한다. 이때, 만일 $w \cdot Z = O$ 이면, A는 프로토콜을 실패로 종료한다.

④ B는 R_A에 대한 embedded 공개키 검증을 수행하고, 검증이 실패하면, B는 프로토콜을 실패로 종료하며, 그렇지 않으면, B는 공유 비밀정보 $Z = (k_B \cdot Q_A) + ((x_B + k_B) \cdot R_A)$ 과 $k = F(w \cdot Z)$ 를 계산한다. 이때, 만일 $w \cdot Z = O$ 이면, B는 프로토콜을 실패로 종료한다.

⑤ 세션키는 k이다.

2) 공개키 암호화 기법

최근의 공개키 암호 기법의 안전성에 관한 연구는 암호학적으로 매우 강한 공격인 적응적 선택 암호문 공격(adaptively chosen-ciphertext attack)에 대하여 안전성이 수학적으로 엄밀히 증명되는 기법의 제안에 초점이 맞추어지고 있다. 본 논문에서 설계한 전자 화폐 시스템에서는 최근 Baek 등[26]이 제안한 안전성이 증명되는 공개키 암호화 기법을 사용하고 있다. 이 암호화 기법은 결정 Diffie-Hellman 가정(decisional Diffie-Hellman assumption)에 근거한 여러 제안된 암호화 기법보다 약한 가정으로 알려져 있는 계산 Diffie-Hellman 가정 (computational Diffie-Hellman assumption)에 적응적 선택 암호문 공격에 대한 안전성을 기반하고 있으면서 암호문의 길이도 기존의 Diffie-Hellman 기반 암호화 기법에 비하여 현저히 짧아진 특징을 가지고 있다. 또한 이 기법은 암호학적 축소 (cryptographic reduction) 기법을 이용하여 랜덤 오라클 모델[27]하에서 적응적 선택 암호문 공격에 대한 안전성이 수학적으로 증명되는 특징도 가지고 있다. 본 전자화폐에서는 효율성을 더욱 증가시킨 이 기법의 타원곡선 버전인 PSLC-2(Provably Secure Length-saving public key encryption scheme based on Computational D-H assumption) [24]을 사용하고 있다. 다음은 PSLC-2를 기술한 것이다.

- 키 생성자 K
  - Galois 체 $GF(p)$, $E(GF(p))$에서 정의된 비초특이 타원곡선을 선택하고 $E(GF(p))$의 위수 $\#E(GF(p))$를 계산한다. q는 $\#E(GF(p))$를 나누는 큰 소수이고 P는 $E(GF(p))$ 상의 위수 q의 점이라 한다.
  - $pk = (E, P, q, W(=uP))$, $sk = (E, P, q, u)$이며, 여기서 $u \in_R GF(q)$이고, $|p| = k = k_0 + k_1$이다.

- 해쉬 함수 (두개의 랜덤 오라클)
  - 두 개의 해쉬 함수 H와 G를 다음과 같이 선택한다.
    $H : \{0,1\}^k \rightarrow GF(q),\ G : GF(p) \rightarrow \{0,1\}^k$

- 암호화 기법 E
  - $R = tP$와 $S = tW$를 계산한다. 여기서, $t = H(m \| s),\ m \in \{0,1\}^{k_0},\ s \leftarrow_R \{0,1\}^{k_1}$ 이다.
  - 암호문은 $E_{pk}(m,s) = (A,B) = (R, G(x_S) \oplus (m \| s))$ 을 생성하며, x_S 는 S의 x좌표이다.

- 복호화 기법 D
  - $S' = uA$ 와 $t' = H(B \oplus G(x_{S'}))$ 을 계산한다.
  - $A = t'P$ 이면, 복호문은 $D_{sk}(A,B) = [B \oplus G(x_{S'})]^{k_0}$ 이며, 그렇지 않으면, 'null'을 생성한다. 여기서 $x_{S'}$ 는 S'의 x 좌표이며 $[B \oplus G(x_{S'})]^{k_0}$ 은 $[B \oplus G(x_{S'})]$ 의 첫번째 k_0 bit 이다.

나. 타원곡선 상에서의 전자화폐 프로토콜 설계

타원곡선 암호 시스템의 사용은 스마트 카드와의 통합을 고려한 것으로 타원곡선 암호의 구현은 하드웨어 상에서 고속연산 처리와 메시지 저장 공간에 대해 장점을 지니고 있다.

타원곡선 E는 유한체 $GF(p)$상에서 정의되었으며, 기저점은 P로 정의하며, q는 사용하는 부분군의 위수로서 $\#E(GF(p))$를 나눈다. 시스템의 구성 요소인 각 개체들은 사용자 U, 신뢰기관 T, 은행 B, 상점 S로 표기한다. 각 개체들을 제외한 알파벳 대문자는 타원곡선 상의 점을 의미하여, 알파벳 소문자는 정수값을 의미한다. 또한 표기의 통일을 위해, 유한체 상의 두 정수 a, b의 곱셈 연산은 ab와 같이 표기하며, 타원곡선 상의 점에 대한 스칼라 곱(scalar multiplication)은 $a \cdot P$와 같이 표현하였다. 그리고, 유한체 상의 정수의 덧셈과 타원곡선 상의 점의 덧셈은 모두 '+'기호로 표기하였다. 공개키 디렉토리에 개체 Z의 장기(long term) 공개키가 등록되어 있으며, 공개키 Q_{Z_i} 가 공개되어 있다고 가정한다. 각 공개키 Q_{Z_i} 는 $[2,q-1]$에서 랜덤하게 생성한 비밀키 x_{Z_i} 의 P에 대한 스칼라 곱으로 $Q_{Z_i} = x_{Z_i} \cdot P$ 이며, 타원곡선 E 상의 점이다. 여기서, Z은 $Z \in \{U, T, B, S\}$이며, i는 $i \in \{1, 2, ..\}$인 정수값으로 각 개체가 여러 개의 비밀키, 혹은 공개키를 생성할 수 있음을 의미한다. 장기 비밀키인 x_{Z_i} 는 $[2,q-1]$상에서 랜덤하게 선택된 정수값이다. 또한, 각 프로토콜을 수행하면서 세션마다 임시로 생성하게 되는 비밀키는 $[2,q-1]$에서 랜덤하게 생성하여 k_{Z_i} 로 정의하였으며, 각 비밀키에 대응하는 공개키 R_{Z_i} 는 $R_{Z_i} = k_{Z_i} \cdot P$ 이다. 등록 단계에서 생성하게 되는 사용자 U의 의사 비밀키 x_{ps}와 의사 공개키 Q_{ps}는 먼저 $[2,q-1]$에서 랜덤하게 생성해 내며, 의사 공개키 Q_{ps}는 $x_{ps} \cdot P$로 계산된다.

각 프로토콜에 사용된 시스템 표기 및 정의는 <표 2>에서 설명하고 있고, 각 단계에서 수행된 정보들을 저장할 데이터베이스에 대한 설명은 <표 3>에서 기술하고 있다. 타원곡선 상으로의 프로토콜의 변환은 [1, 2, 3]을 참고로 하였다.

다음은 타원곡선 상에서 설계한 전자화폐 프로토콜을 설명하고 있다.

1) 계좌 개설 단계

(가) 사용자 U가 은행 B에게 자신의 신원을 확인시킨다.

(나) 은행 B는 신원을 확인하고 사용자 U의 계좌 acc_U를 개설하여 전달한다.

2) 등록 단계

(가) 등록 단계에서의 실체간의 신원확인과 인증된 메시지의 교환을 위해 (그림 3)과 같이 Diffie-Hellman 기반의 새로운 AK 프로토콜[23]을 수행하여 사용자 U와 신뢰기관 T 간에 세션키 $K_{U,T}$를 생성한다. 등록단계에서 사용자 U와 신뢰기관 T 사이의 모든 주고받는 메시지들은 생성한 세션키 $K_{U,T}$를 이용하여 블록 암호화 된 후 전달된다.

(나) (그림 4)와 같이 등록단계를 수행한다. 사용자 U는 화폐 발급에 이용할 의사 비

밀키 x_{ps}를 *[2, q-1]*에서 랜덤하게 생성하여, 기저점 *P*와의 스칼라 곱 $Q_{ps} = x_{ps} \cdot P$으로 의사 공개키 Q_{ps}를 생성한다. 의사 공개키와 사용자의 식별정보 id_U에 대한 사용자 *U*의 서명 s_U를 생성하기 위해, 비밀키 k_{U_2}와 대응하는 공개키 R_{U_2}를 생성한다. 서명 s_U를 공개키 R_{U_2}와 연접하여 σ_U를 생성하고, 의사 공개키 Q_{ps}와 세션키 $K_{U,T}$로 암호화한 후, 사용자의 식별정보 id_U와 함께 신뢰기관 *T*에게 전달한다.

$$
\begin{aligned}
&Choose\ x_{ps}, k_{U_2} \in [2, q-1] \\
&Q_{ps} = x_{ps} \cdot P \\
&R_{U_2} = k_{U_2} \cdot P \\
&s_U = x_{U_2} h(R_{U_2} \| id_U \| Q_{ps}) + k_{U_2} \\
&\sigma_U = (R_{U_2} \| s_U)
\end{aligned}
\tag{1}
$$

<표 2> 표기 및 정의

표기	설명	표기	설명
P	기저점	x_{Z_i}	Z의 장기 비밀키, ($x_{Z_i} \in_R [2, q-1], i \in \{1,2,..\}$)
#E(GF(p))	타원곡선 상의 점의 개수	Q_{Z_i}	Z의 장기 공개키, ($Q_{Z_i} = x_{Z_i} \cdot P$, $i \in \{1,2,..\}$)
E(GF(p))	타원곡선	k_{Z_i}	Z의 단기 비밀키, ($k_{Z_i} \in_R [2, q-1], i \in \{1,2,..\}$)
q	*#E(GF(p))*를 나누는 부분군의 위수	R_{Z_i}	Z의 단기 공개키, ($R_{Z_i} = k_{Z_i} \cdot P$, $i \in \{1,2,..\}$)
w	cofactor, *#E(GF(p)) / q*	x_{ps}	U의 의사 비밀키, ($x_{ps} \in_R [2, q-1]$)
U	사용자	Q_{ps}	U의 의사 공개키, ($Q_{ps}=x_{ps} \cdot P$)
S	상점	$K_{U,T}$	U와 T 간의 키 합의 프로토콜에 의한 세션키
B	은행	$K_{U,B}$	U와 B 간의 키 합의 프로토콜에 의한 세션키
T	신뢰기관	Z_{PK}, Z_{SK}	Z의 공개키, 비밀키
Z	실체, *Z∈{U, B, S, T}*	s_Z	Z의 서명값
c	64비트로 랜덤하게 생성한 화폐	σ_Z	$Z \| \| R_{Z_i}$, $i \in \{1,2,..\}$
||	메시지의 연접	s_c	c에 대한 U의 서명값
msg	상점의 challenge 정보	σ_c	$c \| \| R_c$
id_Z	Z의 식별정보	E_K, D_K	대칭키 K를 이용한 암호화, 복호화 기법
acc_Z	Z의 계좌번호	$E_{Z_{PK}}, D_{Z_{SK}}$	공개키 암호화 기법, 복호화 기법
Ind(j)	저장된 *j*에 대한 색인	S_Z, V_Z	Z의 서명 기법, 검증 기법
h(A||B||..)	입력 *A,B,...*의 연접에 대한 충돌방지 해쉬함수의 결과값		

<표 3> 데이터 베이스 목록 및 철회 목록

구분	항목	접근자	저장 정보
Coin-DB	$c, Ind(Q_{ps}), \sigma_B$	U	인출 단계 시 B로부터 받은 화폐와 서명값
User-DB	$id, Name, acc,$ $Address, e-mail$	B	고객 관리 DB로서 시스템의 초기화에서 설정. 고객의 식별정보 id와 고객의 이름 $Name$, 계좌번호 $acc(acc_U$과 acc_S를 포함),주소 $Address$, e-$mail$ 주소 등의 고객정보
PsdPub-DB	$id_U, Q_{ps}, \sigma_U, Ind(x_{T_2})$	T	U의 의사 공개키 등록 DB
Pay-DB	$c, Q_{ps}, msg, \sigma_B, \sigma_T, \sigma_c$	S	지불 단계 시 U로부터 받은 화폐 정보
PsdPrv-DB	$x_{ps}, Q_{ps}, R_{T_2}, \sigma_T$	U	등록 단계 시 U가 의사 비밀키와 등록 단계 정보
With-DB	$id_U, Ind(x_{B_1}), e', \varepsilon_T$	B	인출 단계 시 B가 받은 정보, e'는 은닉화폐, ε_T 은 신뢰기관의 공개키를 이용한 화폐정보의 공개키 암호화 값
Dep-DB	$c, Q_{ps}, id_S, \sigma_T, \sigma_B, \sigma_c$	B	입금 단계 시 S로부터 받은 화폐 정보
User-BL	Q_{ps}	T, S	U의 강탈 당한 의사 비밀키에 대응하는 의사 공개키 목록
Bank-BL	Q_{B_1}	T, S	B의 강탈 당한 비밀키에 대응하는 공개키 목록
Trust-BL	Q_{T_2}	T, S	T의 강탈 당한 비밀키에 대응하는 공개키 목록
Coin-WL	Q_{ps}, c	T, S	B의 강탈 당한 비밀키로 서명된 인출 후 사용되지 않은 정당한 화폐들의 목록
PsdPub-WL	Q_{ps}	T, S	T의 강탈 당한 비밀키로 서명된 정직한 사용자의 의사 공개키 목록

(다) 신뢰기관 T은 사용자의 서명 σ_U 을 식(2)과 같이 검증한 후, 사용자 U의 의사 공개키 Q_{ps}에 대한 신뢰기관의 서명 s_T와 이를 위해 생성한 공개키 R_{T_2} 를 연접하여 식 (3)와 같이 σ_T 을 생성하고 사용자에게 전달한 다음, 사용자의 식별정보 id_U와 의사 공개키 Q_{ps}, 사용자의 서명 σ_U , 신뢰기관이 서명에 사용한 비밀키 x_{T_2} 의 포인터 $Ind(x)$를 PsdPub-DB에 저장한다.

$$\begin{aligned} &if\ ((s_U \cdot P != h(R_{U_2} \| id_U \| Q_{ps}) \cdot Q_{U_2} + R_{U_2}))\quad reject \\ &else\quad accept \end{aligned} \tag{2}$$

$$\begin{aligned} &Choose\ \ k_{T_2} \in [2, q-1] \\ &R_{T_2} = k_{T_2} \cdot P \\ &s_T = x_{T_2} h(R_{T_2} \| Q_{ps}) + k_{T_2} \\ &\sigma_T = (R_{T_2} \| s_T) \end{aligned} \tag{3}$$

(라) 사용자 U는 신뢰기관이 전달한 서명 σ_T 을 식 (4)와 같이 검증한 후, 생성한 의사 키 쌍과 전달받은 정보들을 PsdPrv-DB에 저장한다.

$$\begin{array}{l} {}^{UT}\sigma_T)) = \sigma_T \\ {}^{UT}_{T} \cdot \quad = \quad _{T_2}) \cdot \quad _{T_2} + \quad _{T_2}) \quad eect \\ ee \quad accept \end{array} \tag{4}$$

3) 인출 단계

(가) 등록단계와 마찬가지로, 실체간의 신원확인과 인증된 메시지의 전달을 위해 Diffie-Hellman 기반의 새로운 AK 프로토콜[23]을 (그림 3)과 같이 수행하여 사용자 U와 은행 B 간에 세션키 $K_{U,B}$를 생성하며, 이 세션키를 이용하여 이 단계에서 전달되는 메시지들을 암호화하여 전달하게 된다.

(나) 인출단계는 (그림 5)와 같이 수행된다. 사용자는 화폐 c를 랜덤한 24비트로 생성하여, 의사 공개키 Q_{ps}와 함께 신뢰기관 T의 공개키로 PSLC-2 공개키 암호화[24]를 하여 암호화 값으로 $E_{pk}\ m\ s)$를 은행에게 전달한다. 아래 식 (5)는 PSLC-2 공개키 암호화 기법을 본 전자화폐 프로토콜에 적용시켜 본 것이다. $E_{pk}\ m\ s)$은 은행에 대한 강탈공격 시에 모두 신뢰기관으로 전달된다. $E_{pk}\ m\ s)$를 받은 은행은 비밀키 k_{B_1}로 공개키 R'_{B_1}를 생성하여 사용자 U에게 전달한다.

$$\begin{array}{l} Choose \ \ s \leftarrow_R \{0,1\}^k \\ m = [h(Q_{ps} \| c)]^k \\ t = H(m \| s) \\ R_U \ = t \cdot P \\ S = t \cdot Q_{T_2} \\ E_{pk}(m,s) = (A,B) = (R, G(x_S) \oplus (m \| s)) \end{array} \tag{5}$$

(다) 사용자 U는 화폐 c, 의사 공개키 Q_{ps}, 은행 B가 전달한 공개키 값 R'_{B_1}를 자신이 생성한 비밀키 u, v으로 은닉 정보 e'를 만들어 은행 B에게 전달한다.

$$\begin{array}{l} \quad\quad u, \ \in [2, \ -1] \\ {}^{'}_{B_1} \quad {}^{B_1} \\ _{B_1} = u \cdot \ {}'_{B_1} + \ \cdot \\ = \ (\ _{B_1} \| \ \| \ _{ps}) \\ '= \ -u \end{array} \tag{6}$$

(라) 은행 B는 은닉 정보 e'에 은닉 Schnorr 서명[25]을 하여 서명값 s'_B를 사용자 U에게 전달한다.

$$\begin{array}{l} \quad\quad k_{B_1} \in \quad\quad - \\ '_B = \ _{B_1} '+ k_{B_1} \end{array} \tag{7}$$

(마) 사용자 U는 은행 B가 전달한 은닉서명 s'_B를 자신이 알고 있는 비밀값 u, v를 이용하여 발행화폐의 서명값 s_B를 계산하고, 서명 s_B와 R_{B_1}의 연접 σ_B를 식 (8)과 같이 생성한다. 생성한 서명을 식 (9)과 같이 검증한 후, 서명 σ_B를 화폐 정보 c, $Ind(Q_{ps})$와 함께 Coin-DB에 저장한다.

$$s_B = s'_B u + v$$
$$\sigma_B = (R_{B_1} \| s_B) \quad (8)$$

$$if \quad s_B \cdot P == h\ R_{B_1} \quad c \quad Q_{ps}) \cdot Q_{B_1} + R_{B_1})) \quad then \quad accept \quad (9)$$

(바) 은행 *B*는 With-DB에 사용자에게서 받은 정보들을 저장한다.

4) 지불 단계

(가) 상점 *S*은 (그림 6)에서와 같이 화폐를 지불하려는 사용자 *U*에게 challenge 값으로서 상점 *S*의 식별정보 id_S와 시간 정보를 이용하여 *msg*를 계산하여 전달한다.

$$msg = h(id_S \| time) \quad (10)$$

(나) 사용자 *U*는 인출받은 화폐 *c*와 의사 공개키 Q_{ps}, 서명값 σ_T , σ_B 와 아래 식 (11)과 같이 계산된 화폐에 대한 서명값 σ_c 을 상점에 전달한다. 이때, 상점에 대한 모함을 방지하기 위해서는 서명값 σ_c 은 상점의 공개키로 암호화하여 전달하는 것이 안전하다.

$$Choose\ k_{U_4} \in [2, q-1]$$
$$R_{U_4} = k_{U_4} \cdot P$$
$$s_c = x_{ps} h(R_{\ 4} \| c \| id \ \| msg \| Q\ ) + k_{\ 4}$$
$$\sigma_c = (R_{\ 4} \| s_c) \quad (11)$$

(다) 상점 *S*은 전달받은 인출 정보들의 서명값 σ_T , σ_B , σ_c 을 식 (12)과 같이 검증하고 정보들을 Pay-DB에 저장한다. 그러나, 강탈공격이 신고된 후에는 서명값들이 모두 올바르게 검증이 되었는지를 확인하고, 블랙리스트와 화이트리스트를 검사하여야 한다. 사용자 *U*의 의사 비밀키 x_{ps}가 강탈당한 경우, 사용자의 Q_{ps}가 User-BL에 기록되어 있는지를, 은행 *B*의 비밀키 x_{B_1} 가 강탈당한 경우, 은행의 공개키 Q_{B_1} 가 Bank-BL에 기록되어 있고 강탈당한 비밀키로 서명된 화폐 *c*가 Coin-WL에 기록되어 있는지를, 신뢰기관 *T*의 비밀키 x_{T_2} 가 강탈당한 경우, 신뢰기관의 공개키 Q_{T_2} 가 Trust-BL에 기록되어 있고, 강탈당한 비밀키로 서명한 사용자 *U*의 의사 공개키 Q_{ps}가 PsdPub-WL에 기록되어 있는지를 체크한다. 확인이 되면 받은 정보들을 Pay-DB에 저장한다.

$$Obtain\ c, Q_{ps}, \sigma_B, \sigma_T, \sigma_c\ from\ the\ user$$
$$if\ ((s_T \cdot P == h(R_{T_2} \| Q_{ps}) \cdot Q_{T_2} + R_{T_2}) \&\&$$
$$(s_B \cdot P == h(R_{B_1} \| c \| Q_{ps}) \cdot Q_{B_1} + R_{B_1}) \&\&$$
$$(s_c \cdot P == h(R_{U_4} \| c \| id_S \| msg \| Q_{ps}) \cdot Q_{ps} + R_{U_4})$$
$$then\ accept \quad (12)$$

5) 입금 단계

(가) 입금단계는 (그림 7)과 같이 수행된다. 상점 *S*는 사용자 *U*에게서 받은 화폐 *c*, 의사 공개키 Q_{ps}, 은행 *B*의 서명값 σ_B, 신뢰기관 *T*의 서명값 σ_T을 은행 *B*에게 전달하고 은행 *B*는 받은 서명값들을 지불단계의 식 (12)과 같이 검증한다.

(나) 서명값이 모두 검증되고, 은행 *B*의 DB에 예치된 화폐의 정보 중 같은 화폐 *c*와 같은 의사 공개키 Q_{ps}가 존재하면, 그 예치된 화폐 정보에 대한 화폐의 서명값 σ'_c을 찾아 상점 *S*에게 전달한다(이중 입금 검출 단계).

(다) 상점 *S*은 은행 *B*로부터 화폐의 서명값 σ'_c을 수신할 경우 자신이 지불할 화폐의 서명값 σ_c과 같은지를 체크하여 같은 값일 경우 reject하고, 다른 값일 경우에만 은행 *B*에게 전달한다.

(라) 만일, 같은 화폐 *c*와 의사 공개키 Q_{ps}의 쌍이 존재하지 않거나, 존재하더라도 상점 *S*으로부터 이미 예치된 정보와 다른 서명값 σ_c을 가졌을 경우, 상점 *S*은 은행 *B*에게 상점의 식별정보 id_S, 상점의 계좌번호 acc_S, msg, σ_c을 전달한다. 은행 *B*은 서명값 σ_c을 검증한 다음, User-DB의 상점 *S*의 계좌 acc_S에 금액을 입금하고, Dep-DB에 받은 정보들을 저장한다.

(마) 이때, 은행 *B*의 DB에 이미 예치된 화폐의 정보 σ'_c가 존재하며, 화폐의 서명값 σ'_c이 상점 *S*로부터 받은 서명값 σ_c과 동일하지 않으면 같은 화폐가 인출자에 의해서 이중 사용된 것이므로(이중 사용), 은행 *B*는 이중 사용된 화폐의 의사 공개키 Q_{ps}와 화폐 *c*를 이용하여 다음의 추적 단계를 수행하고 이중 사용자를 검출한다.

6) 사용자 추적 단계

(가) 화폐를 이중 사용한 이중사용자의 신원을 파악하기 위해, 은행 *B*과 신뢰기관 *T*가 결탁한다. 따라서, 은행 *B*는 이중 사용된 화폐 $c, Q_{ps}, \sigma_B, \sigma_T, \sigma_c$ 와 Dep-DB로부터 $c, Q_{ps}, \sigma_B, \sigma_T, \sigma'_c$ 를 검출하여 이 정보들을 모두 신뢰기관 *T*에게 전달한다.

(나) 신뢰기관 *T*는 화폐에 대한 서명값들을 지불단계의 식 (12)에서와 같이 검증한 후, 의사 공개키 관리 DB인 PsdPub-DB로부터 화폐의 의사 공개키 Q_{ps}에 대한 사용자의 식별정보 id_U와 사용자의 서명값 σ_U을 검출하여 은행 *B*에게 전달한다.

7) 강탈 추적 단계

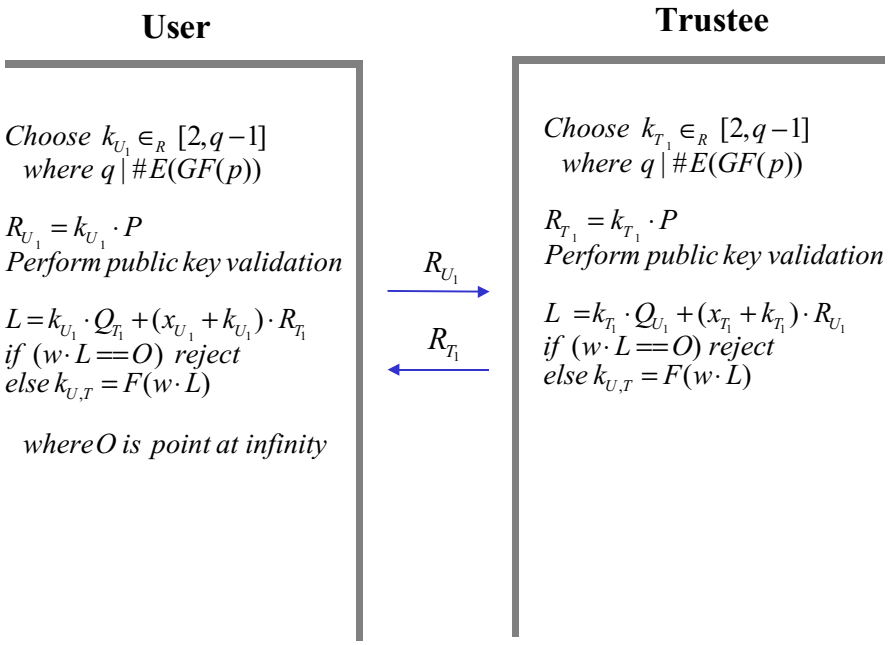
(가) 사용자 *U*의 비밀 키 x_{ps}의 강탈 공격에 대응 : 사용자 *U*가 신뢰기관 *T*에게 이 공격을 보고하며, 신뢰기관 *T*는 자신의 PsdPub-DB에서 (Q_{ps}, id_U)를 확인한다. 신뢰기관 *T*는 User-BL에 Q_{ps}를 기록하여 즉시 상점에게 배포한다. 또한 신뢰기관 *T*는 사용자 *U*가 Q_{ps}로 인출한 사용하지 않은 화폐를 입금하거나 교환할 수 있도록 (Q_{ps}, id_U)에 대한 신뢰기관 *T*의 서명을 발급하여 준다.

(나) 은행 B의 비밀 키 x_{B_1} 의 강탈 공격에 대응 : 은행 B의 비밀키 x_{B_1} 의 노출로 전자화폐의 위조가 가능해진다. 따라서 With-DB에 있는 이 비밀키 x_{B_1} 와 연관된 모든 정당한 화폐에 대한 $E_{T_{PK}}(h(Q_{ps} \| c))$ 를 신뢰기관 T에게 전달한다. 신뢰기관 T는 $h(Q_{ps} \| c)$를 복호화하고 Coin-WL에 이 값들을 기록한다. 또한 x_{B_1} 에 대응되는 공개키 값 Q_{B_1} 를 Bank-BL에 기록하여 Coin-WL와 Bank-BL를 즉시 상점 S에게 전달한다. 은행 B는 새로운 키 쌍을 생성해야 한다.

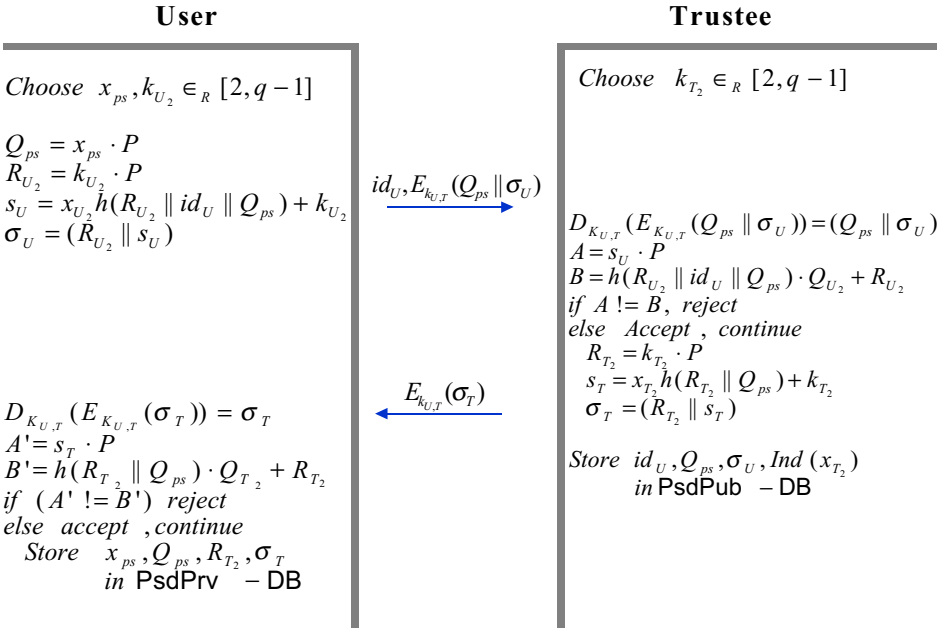
(다) 신뢰기관 T의 비밀 키 x_{T_2} 의 강탈 공격에 대응 : 신뢰기관 T는 x_{T_2} 로 서명된 PsdPub-DB의 모든 공개키 값 Q_{ps}들을 PsdPub-WL에 기록한다. 동시에 x_{T_2} 에 대응되는 공개키 값 Q_{T_2} 를 Trust-BL에 기록하여, PsdPub-WL와 Trust-BL을 모두 상점들에게 배포한다. 신뢰기관 T는 새로운 키 쌍을 생성해야 한다.

(라) 은행 B의 비밀 키 x_{B_1} 에 대한 화폐의 눈속임 공격에 대응 : 키 합의 프로토콜이 사용자 U와 은행 B간의 신원으로 수행되기 때문에 화폐에 대한 투명한 눈속임 프로토콜의 수행은 불가능하다.

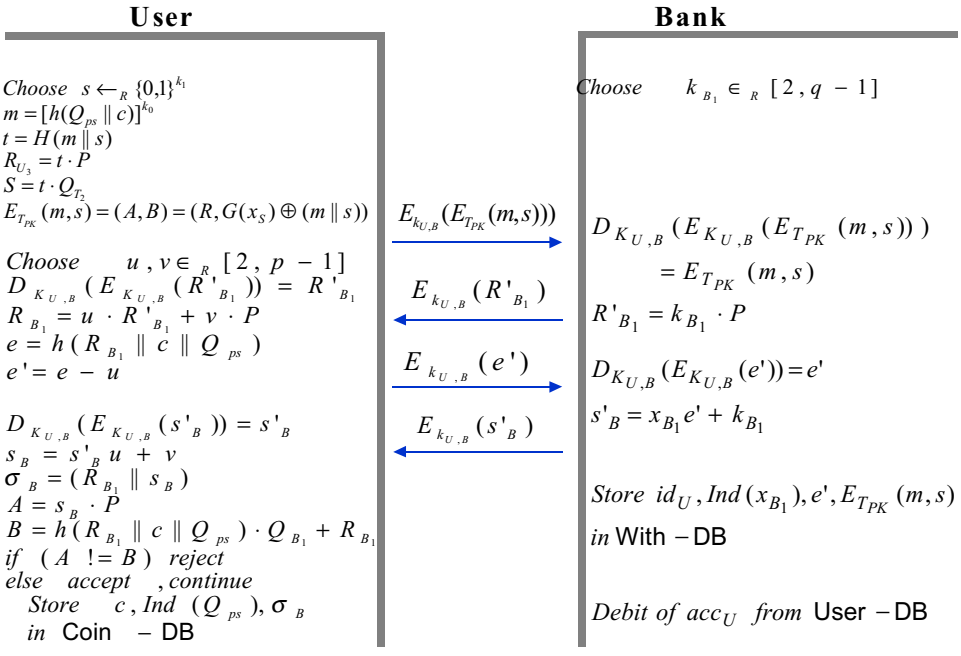
(마) 신뢰기관 T의 비밀 키 x_{T_2} 에 대한 x_{ps}, Q_{ps}의 눈속임 공격에 대응 : 위조 불가능한 서명 기법의 사용으로 이러한 공격은 불가능하다.



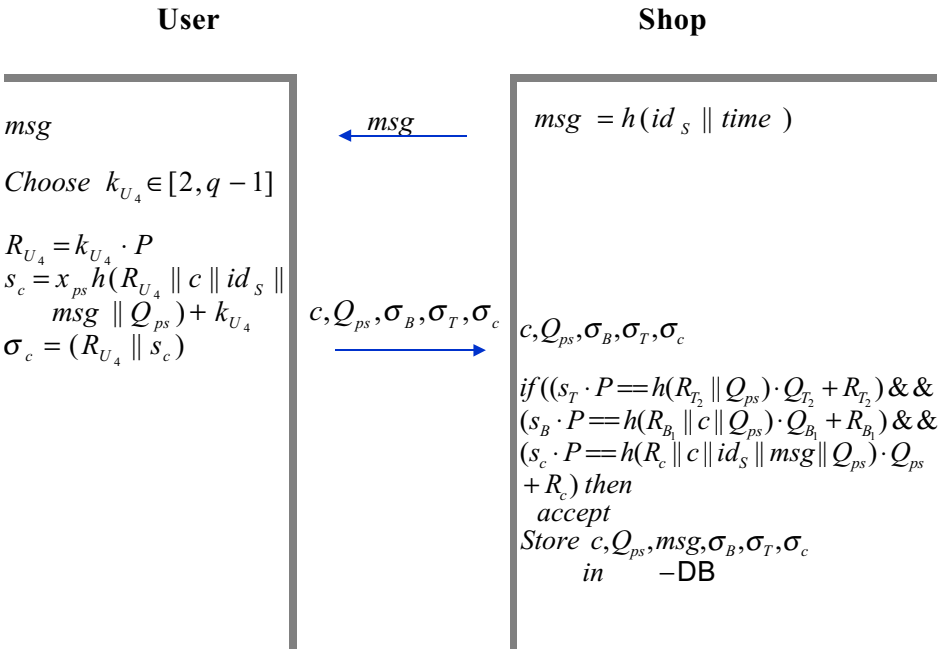
(그림 3) 키 교환 프로토콜



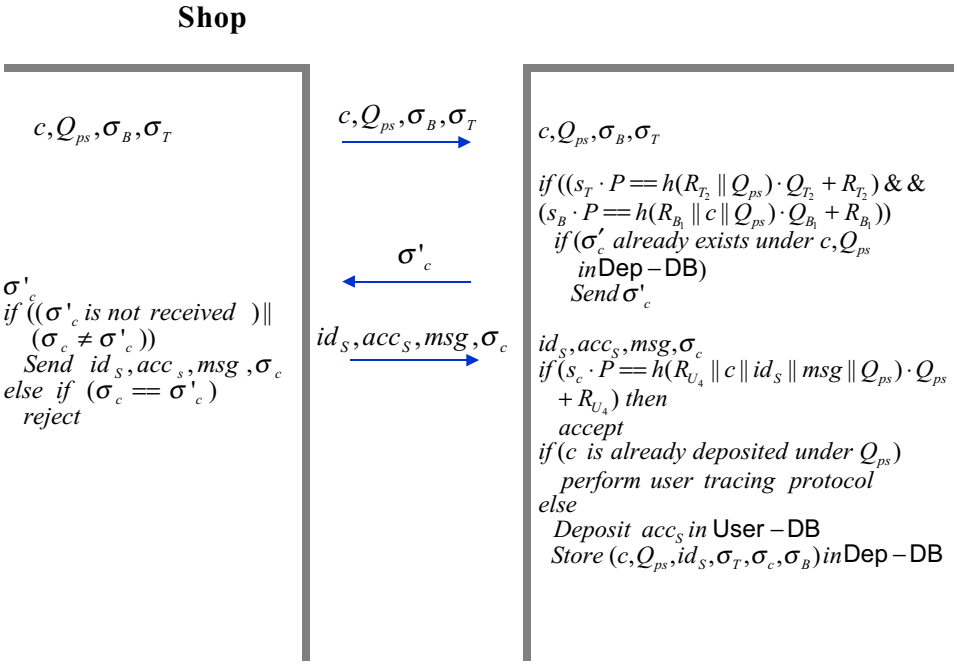
(그림 4) 등록 프로토콜



(그림 5) 인출 프로토콜



(그림 6) 지불 프로토콜



(그림 7) 입금 프로토콜

5 시스템의 안전성 및 성능 비교

이번 장에서는 설계한 시스템의 안전성 및 효율성에 대해 분석하고 있다.

가. 안전성 분석

본 논문에서 제안하고 있는 시스템은 유한체 상에서의 이산대수 문제보다 더욱 어렵다고 알려져 있는 타원곡선 상의 이산대수 문제에 안전성을 기반으로 하여, 위조 불가능성, 이중 사용 방지, 익명성, 추적 불가능성, 완전정보화, 효율성, 익명성 취소, 사용자 추적, 강탈 추적, 연결 불가능성, 환불가능성, 공정성, 모함 방지, 화폐 추적, 속임 방지의 요구사항을 만족시키고 있다. 시스템의 이러한 요구사항의 만족은, 적합하게 선택된 메시지 공격에 대한 서명의 위조가 알려져 있는 계산량적으로 어려운 문제들(인수분해 혹은 이산대수 문제)과 동등한 문제로 증명된다면, 서명 및 검증 기법(S, V)이 확률적으로 계산량적으로 안전하다는 정의 하에 증명되며, PePo97에서 제시하고 있는 기법[6]과 동일한 수준의 안전성을 제공한다. 다음은 만족시키는 요구사항에 대해 기술하고 있다.

제시하고 있는 시스템 파라미터의 표기는 <표 2>와 동일하다.

가정 ① : (S_B, V_B)가 계산량적으로 안전한 확률적 타원곡선 은닉 서명이다.

가정 ② : h 함수는 충돌 불가능 함수이다.

가정 ③ : $_{T_K} D_{T_{SK}}$)는 강한 확률 암호 기법이다.

가정 ④ : (S_C, V_C)가 계산량적으로 안전한 확률적 타원곡선 서명 기법이다.

가정 ⑤ : (S_U, V_U)가 계산량적으로 안전한 확률적 타원곡선 서명 기법이다.

가정 ⑥ : (S_T, V_T)가 계산량적으로 안전한 확률적 타원곡선 서명 기법이다.

가정 ⑦ : 키 합의 프로토콜이 적극적인 공격과 수동적인 공격에 대해 안전하며, 대칭키 암호시스템 (E_K, D_K)에 대해 세션키 K를 모르고 공격을 하는 것이 불가능하다.

정리 1. *가정 ① 가정 ② 가정하면,시스템은 화폐의 위조에 대해 안전하다.*

증명 : S_B는 existential forgery에 대해 안전하므로, 모든 메시지 $h(c||Q_{ps})$에 대한 화폐의 서명 σ_B은 은행에 의해 생성되어야만 한다. 해쉬함수 h는 사용자에 의해 충돌 불가능 함수이므로 사용자가 $h(c'||Q_{ps}) = h(c||Q_{ps})$인 $c' \neq c$를 찾거나 혹은 $h(c||Q'_{ps}) = h(c||Q_{ps})$인 $Q'_{ps} \neq Q_{ps}$를 찾는 것이 어렵다. 따라서 시스템은 화폐의 위조 불가능성의 요구사항을 만족한다.

정리 2. *가정 ① 가정 ③ 가정하면,시스템은 은행이 사용자를 추적하는 것에 대해 안전하다.*

증명 : (S_B, V_B)이 완전한 은닉 서명이므로, 서명 받은 화폐(c, σ_B)은 (c', σ'_B)와 같은 값을 아는 것으로는 추적 불가능하다. 또한 $_{T_K} D_{T_{SK}}$)이 확률적인 암호 기법이므로,

은행은 은닉된 화폐와 함께 전달되는 암호화 값 ε_T 을 주어진 화폐와 연관시키지 못하도록 한다.

정리 3. *가정 ④를 가정하면,시스템은 위장 공격, 은행에 의한 사용자 모함, 신뢰기관의 도움으로 은행이 사용자를 추적하는 것에 대해 안전하다.*

증명 : 사용자의 의사 비밀키 x_{ps}를 모르는 사람은 누구든지 $V_C(Q_{ps}, \sigma_c, (c, id_S, msg))=true$를 만족시키는 (c, id_S, msg)에 대한 서명 σ_c를 생성할 수 없으므로, 은행에 의한 위장 공격은 불가능하다. 만일, 은행이 사용자가 이중 사용을 하였다고 주장할 경우에는 사용자가 지불 시 생성한 서명 σ_c, σ'_c을 제시해야 하는데, 마찬가지로 비밀키 x_{ps}를 모르고 서명값을 생성할 수 없으므로, 은행의 사용자에 대한 모함도 계산량적으로 어렵다. 마지막으로, 은행이 신뢰기관의 도움으로 사용자를 추적하는 것은 이전과 마찬가지의 이유로, 화폐의 서명값에 대한 정보만으로는 그 화폐의 새로운 서명값을 생성할 수 없으므로, 이러한 공격 또한 계산량적으로 어렵다.

정리 4. *가정 ⑤를 가정하면 시스템은 신뢰기관에 의한 사용자의 모함에 대해 안전하다.*

증명 : 신뢰기관은 사용자 U를 모함하기 위해서는 주어진 Q_{ps}와 알려진 어떠한 id_U에 대한 유효한 서명 $S_U(x_{U_2}(id_U, Q_{ps}))$ 를 생성해야 하지만, 이것은 (S_U, V_U)가 확률적으로 계산량적으로 안전한 서명 기법이므로 이러한 공격이 계산량적으로 어렵다.

정리 5. *가정 ⑥을 가정하면 시스템은 사용자추적이 가능하므로, 돈세탁, 신뢰기관의 눈속공격에 대해 안전하다.*

증명 : S_T가 위조 불가능하므로, 서명값 σ_T는 신뢰기관이 사용자의 신원과 서명된 의사 공개키 Q_{ps} 사이의 관계를 알고 있다는 것을 입증하는 증거가 된다. 사용자 추적 단계의 기법에서 신뢰기관은 은행이 보낸 정보 $(c, Q_{ps}, msg, \sigma_B, \sigma_T, \sigma_c)$의 모든 서명값을 검증한다. 이때, 서명값은 그 이전의 단계에서 올바른 개체에 의해서 필수적으로 생성된 것이 이미 검증되었기 때문에, σ_B는 화폐 $h(c|| Q_{ps})$를 인증하고, σ_c는 c가 의사 공개키 Q_{ps}하에서 지불되었다는 것을 증명하며, σ_T는 신뢰기관이 id_U에 c를 연결시킬 수 있는 (id_U, Q_{ps})을 알고 있다는 것을 증명한다. 따라서 돈세탁과 같은 공격을 방지할 수 있다. 또한 반대로 사용자가 서명자와 단 한번의 상호 프로토콜을 수행한 후에 두개의 서명값을 알게 된다는 것은 S_T의 위조 불가능성과는 모순이 되므로, 신뢰기관이 σ_T를 위한 눈속임 프로토콜에 투명하게 참여한다는 것은 계산량적으로 어렵다.

정리 6. *가정 ⑦을 가정하면 시스템은 화폐 의사키의 도청, 은행의 상점에 대한 모함에 대해 안전하다.*

증명 : 등록 단계와 인출 단계에서의 사용자의 통신은 신뢰할 만한 키 합의 프로토콜을

수행하여 생성된 신뢰할 만한 세션키 하에서 보호된다. 이 세션키는 도청과 man-in-the-middle 공격에 강하기 때문에 통신 프로토콜은 이러한 속성을 상속받는다. 사용자와 상점 간의 σ_c의 전송은 상점의 공개키로 암호화 되기 때문에, 은행은 σ_c을 알 수 없으며 상점을 모함할 수 없다.

정리 7. *User-BL이 올바르게 사용된다면, 시스템은 모든 강탈공격에 대한 추적이 가능하므로, 화폐의 강탈에 대해서 안전하다고 말할 수 있다.*

증명 : 화폐의 구조에 화폐 c와 Q_{ps}의 관계가 포함되어 은행의 서명에 의해 보호되므로, 화폐를 사용하거나 입금시키기 위해 사용자로부터의 화폐 강탈은 불가능하다. 그러므로, 화폐 c와 함께 x_{ps}가 강탈당했다면, c는 User-BL에 있는 Q_{ps}에 대응되는 x_{ps}과 함께 지불단계에서 서명되어 있어야 한다.

정리 8. *User-BL, Bank-BL, Trust-BL이 올바르게 사용된다면, 시스템은 모든 비밀키 강탈에 대해 안전하다.*

증명 : 강탈당한 사용자의 비밀키에 대응되는 공개키가 이미 블랙리스트에 기록되어 있으면, 해당 공개키로 인출된 어떤 화폐도 상점에 의해 거부될 수 있다. ②은행의 비밀키가 강탈당한 경우, Bank-BL에 이 비밀키에 대응하는 공개키가 기록되어 이후의 이 비밀키의 사용은 방지된다. ③신뢰기관의 비밀키가 강탈당한 경우에도 마찬가지로 이 비밀키로 수행된 어떠한 서명도 다시 재발행된다.

정리 9. *Coin-WL과 PsdPub-WL이 올바르게 사용된다면, 정직한 사용자는 사용하지 않은 어떠한 화폐도 잃어버릴 수 없다.*

증명 : 정직한 사용자가 자신의 비밀키가 강탈당하고 그 비밀키에 대응하는 공개키가 블랙리스트에 기록이 되면, 신뢰기관의 서명 $S_T(x_{T_2}(Q_{ps}, id_U))$ 으로 사용되지 않은 화폐는 교환이나 환불이 가능하다. ②은행의 비밀키가 강탈당하고 블랙리스트에 기록되더라도, 사용자는 이 비밀키로 정당하게 서명된 화폐는 여전히 사용할 수 있다. 이 화폐는 Coin-WL에 $h(Q_{ps}||c)$로 기록되어 있으므로, 누구도 위조할 수 없으며, 그 화폐와 공개키 값을 알 수도 없게 된다. ③신뢰기관의 비밀키가 강탈당하고 블랙리스트에 기입이 되더라도, 그 비밀키로 인증을 받는 공개키는 PsdPub-WL에 기록되어 있으므로 여전히 사용이 가능하다. 그러나 대응하는 x_{ps}를 누구든지 얻지는 못하므로, 그 공개키는 재사용될 수는 없다.

나. 성능 분석

타원곡선 상에서 구현한 본 전자화폐 시스템은 유한체 상에서 구현한 PePo97과 비교하여 메시지 길이와 그에 따른 저장공간의 감소로 효율성이 크게 증가하였다. PePo97에서는 유한체 $p|q-1$를 만족하는 1,024 bit의 소수인 모듈러 p와 160 bit의 q를 가정하였고, 타원곡선

상에서 설계한 본 시스템은 이와 같은 수준의 안전성을 보장하도록 160 bit의 점 P와, 160 bit의 q를 가정하여 시스템의 메시지를 비교해 보았다. 그에 따른 비교 메시지 및 길이 계산은 <표 4>와 같다. 등록단계와 인출단계 시에 전달되는 메시지는 세션키로 블록 암호화되므로, 체에 관계없이 128-bit의 길이를 갖는다. 그러나, 세션키 생성을 위해 키 합의 프로토콜 수행 시 전달되는 공개키는 PePo97 기법에서는 1,024-bit이나 본 구현 기법에서는 160-bit이므로 약 85% 가량의 메시지 길이의 감소 (6.4배 개선도)를 가져왔다.

가장 많은 메시지 길이를 보내게 되는 지불단계의 전송 메시지는 화폐 c, 의사 공개키 Q_{ps}, 은행 서명 σ_B, 신뢰기관 서명 σ_T, 화폐에 대한 서명 σ_c가 연접되어 전달되는데, 입금 단계 및 추적 단계의 네트워크 메시지는 지불단계의 전송 메시지가 그대로 적용되므로, 여기서는 지불단계에 대해서만 살펴보았다. PePo97 시스템은 스마트 카드 구현을 위해 24-bit로 화폐를 생성하며, 본 구현에서도 24-bit의 화폐 정보를 이용한다. 화폐의 각 서명값들은 서명과 공개키의 연접으로 구성되어 있으므로, PePo97 시스템은 (1024+160) bit의 길이를, 본 구현 기법에서는 (160+160) bit의 메시지 길이를 갖는다. 따라서 지불단계에서 연접되어 전달되는 메시지를 비교해보면, 본 연구의 구현 기법에서는 약 76% 가량의 메시지 길이가 감소(4.0배 가량의 개선도) 했다.

메시지 길이의 감소와 그에 따른 저장공간의 감소는 스마트 카드와 같은 적은 양의 메모리를 사용하는 하드웨어를 이용하거나 전자화폐 시스템과 같이 막대한 양의 데이터베이스를 구축해야 하는 구현에 있어서는 큰 장점이 된다. 따라서, 본 연구의 전자화폐 프로토콜은 타원곡선 상에서 구현함으로써 저장공간에 대한 효율을 향상시켰다.

<표 4> 메시지 길이 비교

메시지	구분	PePo97 기법(A)	본 구현 기법(B)	개선도(A/B)
R_{U_1}	세션키 생성을 위해 전달하는 공개키	1024 bit	160 bit	1024/160 ≒ 6.4 (85%)
$c\|\|Q_{ps}\|\|\sigma_B\|\|\sigma_T\|\|\sigma_c$	지불단계의 전달 메시지	(1024 + 1024 + 1024+160) + (1024+160) + (1024+160) = 4600 bit	(160 + 160 + 160+160) + (160+160) + (160+160) = 1144 bit	4600/1144 ≒ 4.0 (76%)

6 결론

본 연구는 스마트 카드의 사용이 가능한 10만원 미만의 소액거래의 구현을 위한 설계를 수행하였다. 실제적인 구현에 적용하기 위해 네트워크 통신에 효율적인 오프라인 네트워크 형의 전자화폐 시스템을 규격으로 삼았으며, 블랙메일링, 돈세탁, 화폐의 이중사용 등의 악

명성을 악용한 공격에 안전할 수 있도록 익명성 철회가 가능한 시스템을 목표로 하여 안전하면서도 효율적인 전자화폐 프로토콜의 구현에 초점을 두어 설계하였다. 이를 위해, 강탈 공격을 비롯한 다양한 공격 모델에 안전하며, 각 단계 프로토콜과 데이터베이스 및 철회 목록의 구체적인 설계로 구현이 용이한 PePo97 전자화폐 시스템을 타원곡선 암호 기법을 적용하여 효율성을 개선한 전자화폐 프로토콜을 설계하였다. 본 시스템은 타원곡선 상의 이산대수 문제에 안전성을 기반으로 하고 있으며, 위조 불가능성, 이중 사용 방지, 익명성, 추적 불가능성, 완전정보화, 효율성, 익명성 취소, 사용자 추적, 강탈 추적, 연결 불가능성, 환불가능성, 공정성, 모함 방지, 화폐 추적, 속임 방지의 요구사항을 만족시킨다. 또한, PePo97에 비해 동일한 수준의 안전성을 제공하면서도 메시지 길이에 대해 약 6.4배까지의 개선도를 보임으로써 데이터 처리 및 저장공간을 다루는 데에 효율성을 증가시켰다. 이러한 효율성은 스마트 카드와 같은 하드웨어 기반의 시스템과 많은 양의 DB를 다루는 시스템에 장점을 제공한다. 또한 본 전자화폐 프로토콜의 등록 및 인출 단계에 사용되는 키 합의 프로토콜로서 알려진 키, 전향적 보안, 키 위장, 미지의 키 공유에 대한 안전성을 만족하고, 실체 당 온라인 상에서 계산해야 할 모듈러 지수승은 2회로 줄어든 Song과 Kim의 Diffie-Hellman 기반의 새로운 AK 프로토콜을 이용하였으며, 인출 단계의 공개키 암호화 기법으로는 최근 Baek 등이 제안한 선택 암호문 공격(adaptively chosen-ciphertext attack)에 대하여 안전성이 타원곡선 상의 계산 Diffie-Hellman 가정 하에 수학적으로 엄밀히 증명되는 PSLC-2를 사용되고 있다.

현재 본 연구 결과로 제시된 타원곡선 상의 전자화폐 프로토콜은 암호 라이브러리 ICUCLIB-v2(ICU Cryptographic LIBrary-v2)[17]를 이용하여 구현 중에 있다. ICUCLIB-v2는 전자화폐 시스템과 같은 응용 연구의 구현에 암호 프리미티브를 쉽고 간단하게 제공할 수 있도록 자체 개발한 암호 라이브러리로서 전자화폐 시스템의 차후 확장성 및 통합성을 고려하여 이용하였다.

7 참고문헌

[1] IEEE P1363 Draft Version 9. “Standard Specifications For Public Key Cryptography,” 1999, http://grouper.ieee.org/groups/1363/

[2] Don B. Johnson and Alfred J. Menezes, “Elliptic Curve DSA (ECDSA): An Enhanced DSA,” Certicom Corp, ECC Whitepapers, http:// www.certicom.com

[3] Aleksan Jurisic and Alfred J. Menezes, “Elliptic Curves and Cryptography,” Certicom Corp, ECC Whitepapers, http:// www.certicom.com

[4] J. Camenisch, J. M. Piveteau and M.Stadler, “An efficient Fair Payment System,” Proc. of 3rd ACM Conference on Computer and Communications Security, ACM Press, pp. 88-94, 1996

[5] M. Jakobsson and M. Yung, “Revokable and Versatile E-money,” Proc. 3rd annual ACM Conference On Computer and Communication Security, pp. 76-87, March 1996.

[6] Holger Petersen and Guillaume Poupard, “Efficient Scalable Fair Cash with Off-line Extortion Prevention,” Proc. of Int. Conference on Information and Communications Security (ICICS’97), LNCS Vol.1334, Springer-Verlag, pp. 463-477, Nov. 1997.

[7] T. Okamoto and K. Ohta, "Universal Electronic Cash," In Advances in Cryptology- Proc. CRYPTO'91, LNCS Vol. 576, Springer-Verlag, pp. 324-337, Aug. 1991

[8] T. Okamoto, "An Efficient Divisible Electronic Cash Scheme," In Advances in Cryptology-Proc. of CRYPTO'95, LNCS, Vol. 963, Springer-Verlag, pp.438-451, Aug. 1995.

[9] S. Brands, "Untraceable Off-line Cash in Wallets with Observers," In Advances in Cryptology-Proc. of CRYPTO'93, LNCS, Vol. 773, Springer-Verlag, pp.302-318, Aug. 1994.

[10] D. Chaum, “Privacy Protected Payments: Unconditional Payer and/or Payee Anonymity,” Smart Card 2000: The future of IC Cards, North-Holland, pp. 69-92, 1989.

[11] D. Chaum, A. Fiat and M. Noar, “Untraceable Electronic Cash,” In Advances in Cryptology-Proc. of CRYPTO'88, LNCS, Vol. 403, Springer-Verlag, pp.319-327, Aug. 1989.

[12] M. Stadler, J. M. Piveteau and J. Camenisch, “Fair-Blind Signatures,” In Advances in Cryptology: Proc. of EUROCRYPT’95, LNCS Vol. 921, Springer-Verlag, pp.209-219, 1995.

[13] S. von Solms and D. Naccache, “On Blind Signatures and Perfect Crimes,” Computers and Security, pp. 581-583, Nov. 1992.

[14] E. Brickell, P. Gemmell and D. Kravitz, “Trustee-based Tracing Extensions to Anonymous Cash and the Making of Anonymous Exchange,” Proc. of 6th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 457-466, 1995.

[15] 김희선, 서문석, 백준상, 김광조, "전자현금 프로토콜의 요구 사항과 비교 분석", WISC99, 1999. 9

[16] 전자화폐시스템 모델(안) 연구, 기술기준연구 98-4, 한국정보보호센터, 1998.12

[17] 분산 환경에서의 정보보안 및 인증 기술 연구, 암호 및 정보보안 연구실, 1999. 9

[18] D.M’Raihi, “Cost Effective Payment Schemes with Privacy Regulations,” In Advances in Cryptology-Proc. of ASIACRYPT’96, LNCS Vol.1163, Springer-Verlag, pp.266-275, Nov. 1996.

[19] J. Camenisch, U. Maurer and M. Stadler, “Digital payment Systems with Passive Anonymity-Revoking Trustees,” Proc. of ESORICS’96, LNCS Vol. 1146, Springer-Verlag, pp.31-43, 1996.

[20] Y. Frankel, Y. Tsiounis and M. Yung, ””Indrect discourse Proofs” : Achieving Efficient Fair Off-Line E-Cash,” In Advances in Cryptology-Proc. of ASIACRYPT’96, LNCS Vol.1163, Springer-Verlag, pp.286-300, Nov. 1996.

[21] E. Fujisaki and T. Okamoto, “Practical Escrow Cash System,” Proc. of 1996 Cambridge Workshop on Security Protocols, LNCS Vol. 1189, Springer-Verlag, pp.33-48, 1997.

[22] D. Chaum, “Blind Signatures for Untraceable Payments,” In Advances in Cryptology: Proc. of CRYPTO’82, Plenum Press, pp. 199-203, 1983.

[23] 송보연, 김광조, “키 공유 확인이 가능한 새로운 키 합의 프로토콜,” WISC2000(to appear),

Sep. 2000.

[24] Joonsang Baek, Byoungcheon Lee, and Kwangjo Kim, “Provably Secure Length-saving Public-Key Encryption Scheme under the Computational Diffie-Hellman Assumption,” ETRI Journal (제출), 2000.

[25] C. P. Schnorr, “Efficient Identification and Signatures for Smart Cards,” In Advances in Cryptology: Proc. of CRYPTO’89, LNCS Vol. 435, Springer-Verlag, pp.239-251, 1990.

[26] J. Baek, B. Lee, and K. Kim, "Secure Length-saving ElGamal Encryption under the Computational Diffie-Hellman Assumption"(updated), Proc. of Fifth Australian Conference on Information Security and Privacy (ACISP '2000), LNCS 1841, Brisbane, Australia, Springer-Verlag, pp. 49-58, 2000.

[27] M. Bellare and P. Rogaway, “Random oracles are practical : A paradigm for designing efficient , protocools,” ACM Conference on Computer and Communications Security, pp. 62-73, 1993.

[28] L. Law, A. Menezes, M. Qu, J. Solinas and S. Vanstone, “An Efficient protocol for Authenticated Key Agreement Protocol,” Technical Report CORR 98-5, University of Waterloo, Canada, March 1998.