ECDLP를 기반으로 하는 Blinding ECDSA

전 병 욱° 권 용 진
한국항공대학교 통신정보공학과

A Blinding ECDSA based on the Elliptic Curve Discrete Logarithm Problems

Byeong Wook Jun Yong Jin Kwon
Dept. of Telecommunication and Information Engineering
Hankuk Aviation University

요 약

전자 상거래에 대한 다양한 프로토타입(prototype)이 구현되고 있고, 확대 적용의 현실성이 증대되고 있는 작금의 상황을 반영하여 관련 연구가 활발해 지고 있으며, 그 중에서 보다 안전하고 효율적인 전자지불방식에 대한 현실적 요구가 증대하고 있다. 전자지불방식의 하나인 전자화폐는 실물 화폐와 유사한 성질들을 만족해야 하며, 이러한 성질들 중에서 필수적인 익명성을 얻기 위한 방법으로는 D. Chaum이 제안한 Blind Signature가 대표적이다. 본 논문에서는 기존의 암호시스템의 문제점을 극복할 수 있는 시스템으로써 주목받고 있는 타원곡선 암호시스템 상에서 익명성을 제공하는 Blinding ECDSA를 제안한다.

Ⅰ. 서 론

전세계적인 인터넷의 보급은 초기의 디지털 정보 전달 및 이용이라는 측면에서 벗어나 현재에는 인터넷 방송, 원격 진료, 전자상거래 등 사회 여러 분야의 정보화를 촉진시키는 역할을 담당하고 있다. 이 중 전자상거래는 기업과 기업간의 거래뿐만 아니라, 기업

과 소비자간의 구매 분야에서도 이용되고 있으며 그 사용량이 꾸준히 증가되고 있는 추세이다. 전자상거래를 이용함으로써 구매자는 인터넷을 통한 상품 검색, 주문, 결재까지 가능하므로 시간적, 공간적 불편함을 줄이고, 기업의 입장에서는 판매 촉진이라는 이익을 얻을 수 있다.

이러한 전자상거래 환경의 활성화를 위해 고려해야 할 사항들 중 큰 비중을 차지하고 있는 것이 안전한 대금 결재이다. 기존 통신 판매에서의 대금 결재는 은행지로 등을 통해 이루어지고 있으며, 이는 구매자가 직접 은행에 가야한다는 불편함으로 인해 전자상거래 활성화의 저해 요인으로 지적되고 있다. 따라서, 네트워크를 통한 안전한 대금 결재 시스템의 개발이 요구되고 있으며, 현재 다양한 전자지불 시스템이 개발되고 있는 상황이다.

전자지불 시스템 중의 하나인 전자화폐는 그 사용 방식의 특성상 실물 화폐와 유사한 성질을 만족해야 하며, 그 성질들로는 비의존성(independence), 보안성(security), 익명성(privacy), 오프라인성(off-line), 양도성(transferability), 분할성(divisibility) 등이 제안되고 있다[1](그림 1). 이 외에도 개인의 Privacy를 보장하면서 돈세탁 등의 화폐 부정 사용을 방지하기 위한 조건부 추적가능성에 대한 요구도 또한 제기되고 있다.

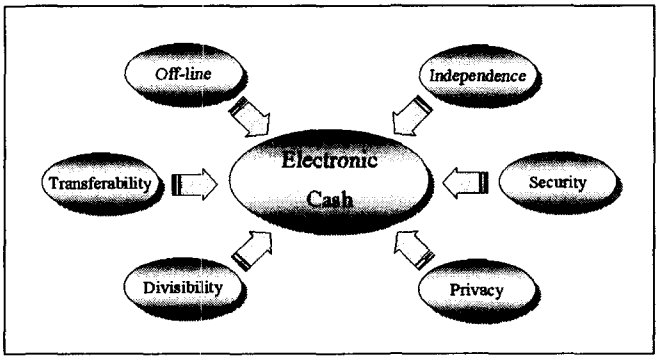

그림 1. 전자 화폐의 요구 사항

Figure 1. Requirements of Electronic Cash

전자 화폐의 요구사항 중 사용자의 익명성은 은행이나 상점이 사용자의 화폐 사용 내역을 추적할 수 없도록 하는 것으로서, 화폐 사용의 자유를 보장하는 성질이다. 익명성을 제공하는 방법 중 가장 대표적인 것으로는 D. Chaum이 제안한 Blind Signature[2]와 J. Camenisch 등이 제안한 서명 방식[3] 등이 있으며, 이들은 각각 소인수분해와 이산대수문제를 기반으로 하고 있다.

소인수분해 문제 및 이산대수문제 등을 기반으로 하고 있는 대부분의 공개키 암호시스템들[4,5,6]은 키분배 문제 해결, 디지털 서명 개념 등의 많은 장점과 함께, 긴 암호화/복호화 시간, 넓은 키 공간(Key space) 등과 같은 구현상의 제한을 가지고 있다. 공개키 암호시스템이 가지는 이러한 단점들은 스마트 카드처럼 작은 계산력과 제한된 양의 메모리를 갖는 디바이스에 적합하지 않으므로, 그 사용 분야에 제약을 받게 하는 원인이 되고 있다.

전술한 공개키 암호시스템의 문제점들은 타원곡선(Elliptic Curve)을 이용한 공개키 암호시스템으로 해결 가능하다. 즉, 타원 곡선 위에서의 이산대수문제는 일반적인 그룹에서 정의되는 이산대수문제보다 더욱 어렵고, 키 공간과 계산량의 문제를 어느 정도 해결할 수 있으므로 스마트 카드 등의 제한된 디바이스에도 적용이 가능하다.

본 논문에서는 타원곡선 디지털 서명 방식인 ECDSA를 변형하여 사용자의 익명성을 보장하는 타원곡선 Blind signatue를 제안하고자 한다. 2장에서는 기존의 타원곡선 암호시스템에 대하여 소개한다. 3장에서는 타원곡선 디지털 서명에서 익명성을 제공하는 타원곡선 Blind Signature를 제안하고 마지막 장에서 결론을 맺는다.

Ⅱ. 타원곡선 암호시스템

1. 타원곡선

1985년, Neil Koblitz[7]와 Victor Miller[8]는 타원곡선 상의 점들에 대한 이산 대수문제에 기반을 둔 타원곡선 암호시스템(Elliptic Curve Cryptosystem, ECC)을 각자 독립적으로 제안하였다. 이러한 타원곡선 암호시스템은 암호화 방식뿐만 아니라 디지털 서명 방식으로도 사용될 수 있다.

(1) 타원곡선 이산대수문제(The Elliptic Curve Discrete Logarithm Problem, ECDLP)

q가 소수의 멱승 형태일 때, F_q는 q개의 원소를 포함하는 유한체(finite field)를 의미한다. 실제 응용에 있어서 q는 일반적으로 2의 멱승(2^m) 또는 홀수인 소수(p)가 된다. 이 때, 타원곡선 이산대수문제는 다음과 같다 : F_q에 대해 정의된 타원 곡선 E,

order n의 점 $P \in E(F_q)$와 $Q \in E(F_q)$가 주어졌을 때, $Q = dP$를 만족시키는 정수 $d(0 \le d \le n-1)$이 존재한다면 그 값을 구한다.

타원곡선 이산대수문제는 소인수분해 문제나 이산대수문제보다 상당히 어려운 것으로 알려져 있다. 이들을 이용한 암호시스템에서 동일한 암호학적 강도를 가정했을 때 타원곡선 암호시스템의 경우, 다른 시스템들에 비하여 키의 길이가 매우 짧아지는 장점을 갖는다. 예를 들어, 2048-bit의 RSA나 DSA에 비해 300-bit ECC(Elliptic Curve Cryptosystem)의 경우가 더욱 안전한 것으로 알려져 있다[9].

(2) 타원 곡선의 정의

타원곡선은 일반적으로 임의의 유한체 상에서 정의될 수 있으며, 특히 F_{2^m}의 경우 연산 수행에 있어 더욱 효율적이다. 여기에서는 설명을 단순화하기 위해 Z_p(p는 3보다 큰 소수)상에서의 타원곡선에 대해 설명한다.

Z_p에 대한 타원곡선 E는 다음과 같은 형태로 정의된다.

$$y^2 \equiv x^3 + ax + b \pmod{p}$$

여기서, $a, b \in Z_p$는 $4a^3 + 27b^2 \neq 0$인 상수이며 타원곡선은 무한원점(point at infinity)이라고 하는 원소 O를 포함한다.

타원곡선 E는 적절한 연산을 적용함으로서 abelian 그룹으로 구성할 수 있는데, 일반적인 그룹을 정의하는 것처럼 타원곡선 위의 점에 대해 다음과 같이 덧셈을 정의한다. 단, 모든 연산은 Z_p위에서 정의된다.

1. 모든 점 $P \in E(Z_p)$에 대하여 $P + O = O + P = P$ 이 성립한다.
2. 만약 $P = (x, y) \in E(Z_p)$이면, $(x, y) + (x, -y) = O$가 된다. (점 $(x, -y)$는 $-P$로 표시하고, P의 negative라고 한다.)
3. $P = (x_1, y_1) \in E(z_p)$, $Q = (x_2, y_2) \in E(z_p)$라고 할 때, $P + Q = (x_3, y_3)$가 된다.

여기서,

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & (P \neq Q \text{ 일때}) \\ \dfrac{3x_1^2 + a}{2y_1} & (P = Q \text{ 일때}) \end{cases}$$

위와 같은 덧셈 규칙은 기하학적으로 잘 설명된다[10]. 타원곡선 E상의 서로 다른 두 점 P와 Q의 합 $R=(x_3, y_3)$는 다음과 같이 정의된다. 첫 번째로 P와 Q를 통과하는 선을 그린다; 이 선은 세 번째 점에서 타원곡선과 교차한다. 이 때, R은 이 점의 x축에 대한 투영(reflection)이다(그림 2). 그림에서 타원곡선은 타원(ellipse)과 무한 곡선(infinite curve)의 두 부분으로 구성된다.

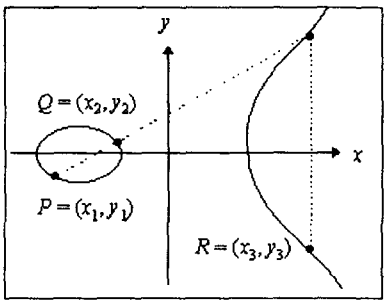

그림 2. 서로 다른 두 점의 덧셈에 대한 기하학적 표시 : P+Q=R

Figure 2. Geometric description of the addition of two distinct elliptic curve points : P+Q=R

$P=(x_1, y_1)$일 때 P의 doubling, $R=(x_3, y_3)$은 다음과 같이 정의된다. 첫 번째로 타원곡선 상의 점 P에 대한 접선(tangent line)을 그린다. 이 선은 두 번째 점에서 타원곡선과 교차한다. 이 때, R은 이 점의 x축에 대한 투영이다(그림 3).

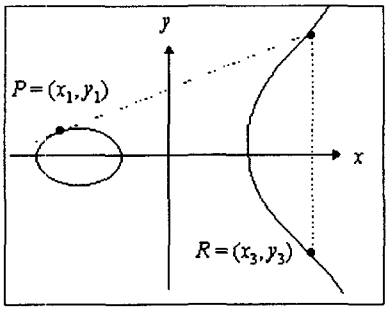

그림 3. 한 점의 doubling에 대한 기하학적 표시 : P+P=R

Figure 3. Geometric description of the doubling of an elliptic curve points : P+P=R

2. 타원곡선 암호시스템

타원곡선 이산대수문제를 기반으로 하는 타원곡선 암호시스템으로는 암호화 방식인 ECES(Elliptic Curve Encryption Scheme)와 디지털 서명 방식인 ECSS(Elliptic Curve Signature Scheme), ECDSA(Elliptic Curve Digital Signature Algorithm), 그리고 키 설정 프로토콜인 ECKEP(Elliptic Curve Key Establishment Protocol) 등이 있다[11]. 이 장에서는 본 논문의 주요 대상인 ECDSA에 대하여 간략히 살펴보기로 하자.

(1) 시스템 셋업 및 키 생성

시스템 셋업(System Setup)

기초적인 유한체 F_q가 선택된다. F_q상에서 정의된 타원곡선 E와 E에서 order가 소수 n인 점 P가 선택된다. 체 F_q, 타원곡선 E, 점 P 및 order n은 시스템 파라미터를 의미하며, 공개 정보이다.

키 생성(Key Generation)

각 Entity들은 다음과 같은 동작을 수행한다.

1. 구간 [1, n-1]에서 랜덤한 정수 d를 선택한다.
2. 점 $Q = dP$를 계산한다.
3. Entity의 공개키는 점 Q로 구성된다.
4. Entity의 비밀키는 정수 d이다.

(2) 타원곡선 서명 방식(ECDSA)

ECSS와 ECDSA 방식에서 서명 대상인 메시지는 먼저 고정된 길이의 메시지 다이제스트로 해쉬된 후 서명이 이루어지고, 서명의 검증(Verification)을 위해서는 서명문과 원문이 모두 필요하다. ECDSA는 NIST Digital Signature Algorithm(DSA)를 타원곡선 상에서 구현한 것이다.

ECDSA에 대한 서명 생성

(Entity B가 Entity A를 위해 메시지 M에 서명하는 경우)

B는 다음의 단계들을 수행한다.

1. 메시지 M을 이진 스트링으로 표현한다.
2. 해쉬 알고리즘을 사용하여 해쉬값 $e = H(M)$ 를 계산한다.
3. 구간 [1, n-1]에서 랜덤한 정수 k를 선택한다.
4. 점 $(x_1, y_1) = kP$를 계산하고, $r = x_1 \bmod n$ 으로 설정한다.
5. 비밀키 d를 사용하여 $s = k^{-1}(e+dr) \bmod n$ 을 계산한다.
6. B는 메시지 M과 서명 (r, s)를 A에게 전송한다.

만약 $r = 0$ 또는 $s = 0$ 이면, 서명 검증에 실패할 것이다. 그러나, k가 랜덤하게 선택된다면 $r = 0$ 또는 $s = 0$ 일 확률은 무시할 수 있을 정도로 작다.

ECDSA에 대한 서명 검증

(Entity A가 메시지 M에 대한 B의 서명 (r, s)를 검증하는 경우)

A는 다음의 단계들을 수행한다.

1. B의 공개키 Q를 알아낸다.
2. 만약 $(r \bmod n) = 0$ 이면 서명을 부인한다.
3. 해쉬값 $e = H(M)$ 를 계산한다.
4. $s^{-1} \bmod n$ 을 계산한다.

5. $u = s^{-1}e \bmod n$ 와 $v = s^{-1}r \bmod n$ 을 계산한다.
6. 점 $(x_1, y_1) = uP + vQ$ 를 계산한다.
7. 메시지 M에 대한 B의 서명을 수락하기 위한 필요충분조건은 $(x_1 \bmod n) = r$ 이다.

(Notes)

(a) 해쉬값 e는 modulo n에 의해 감소된다. 따라서 $H \bmod n$ 도 또한 암호학적으로 안전한 해쉬함수가 되도록 해쉬함수 H와 n이 선택되어야 한다.

(b) 서명 생성 단계 4에서 부여된 조건 $r \neq 0$ 는 보안성에 관한 조건이다. 만약 $r = 0$ 이면, 서명식 $s = k^{-1}(e + dr)$ 는 비밀키 d를 포함하지 못하게 된다.

(c) 만약 k가 랜덤하게 선택된다면, $r = 0$ 또는 $s = 0$ 일 확률은 무시할 수 있을 정도로 작다.

Ⅲ. 타원곡선 은닉서명

1. 타원곡선 은닉서명

D. Chaum, J. Camenisch 등에 의해 제안된 Blind signature 기법은 서명자라고 하더라도 메시지와 서명문을 서로 연관시키지 못하도록 하는 서명 기법이다. 결과적으로 메시지에 서명을 한 서명자는 서명문의 정당성은 검증할 수 있으나 서명된 메시지를 가지고 있는 수신자의 신원은 알 수 없게 된다. 이 프로토콜을 사용한 전자화폐 방식은 은행에서 인출된 전자화폐와 사용자를 연결시키지 못하게 함으로써 사용자의 privacy를 보장하는데 이용될 수 있다. 본 절에서는 타원곡선 이산대수문제를 기반으로 하는 ECDSA를 변형하여, 사용자의 익명성을 제공하는 Blinding ECDSA를 제안하고자 한다.

키 생성(Key Generation)

키 생성 방식은 2장의 ECDSA와 동일하며, Entity A, B의 공개키는 각각 Q_A, Q_B, 비밀키는 d_A, d_B 이고, 공개정보 F_q, E, P, n 은 공유한다.

Blinding ECDSA에 대한 서명 생성

(Entity B가 Entity A를 위해 메시지 M에 서명하는 경우)

A는 다음의 단계들을 수행한다.

1. 구간 [1, n-1]에서 랜덤한 정수 k_{A1}를 선택한다.
2. 점 $(x_1, y_1) = k_{A1}P$를 계산하여 B에게 전송한다.

B는 다음의 단계들을 수행한다.

1. 구간 [1, n-1]에서 랜덤한 정수 k_B를 선택한다.
2. 점 $k_B(x_1, y_1) = (x_2, y_2)$를 계산하여 A에게 전송한다.

A는 다음의 단계들을 수행한다(Blinding 단계).

1. 구간 [1, n-1]에서 랜덤한 정수 k_{A2}를 선택한다.
2. 점 $k_{A2}(x_2, y_2) = (x_3, y_3)$를 계산한다.
3. 점 $(x_4, y_4) = (x_2, y_2) + (x_3, y_3)$를 계산한다.
4. $x_4 = x_3\alpha$를 만족하는 α를 계산한다.
5. 메시지 M을 이진 스트링으로 표현한 후, 해쉬값 $e = H(M)$를 계산한다.
6. $e' = e\alpha^{-1}x_2x_3^{-1}$를 계산하여 B에게 전송한다.

B는 다음의 단계들을 수행한다(Signing 단계).

1. $r_B = x_2 \bmod n$을 계산한다.
2. $s_B = k_B^{-1}\{e' + d_Br_B\}$를 계산하여 A에게 전송한다.

A는 다음의 단계들을 수행한다(Unblinding 단계).

1. $r = x_4 \bmod n$으로 설정한다.
2. $s = s_B\alpha x_2^{-1}x_3(k_{A1} + k_{A1}k_{A2})^{-1}$를 계산한다.

Blinding ECDSA에 대한 서명 검증

(Entity A가 메시지 M에 대한 B의 서명 (r, s)를 검증하는 경우)

A는 다음의 단계들을 수행한다.

1. B의 공개키 Q_B를 알아낸다.

2. $s^{-1} \bmod n$ 을 계산한다.
3. $u = s^{-1}e \bmod n$ 와 $v = s^{-1}r \bmod n$ 을 계산한다.
4. 점 $(x_4, y_4) = uP + vQ_B$ 를 계산한다.
5. 메시지 M에 대한 B의 서명을 수락하기 위한 필요충분조건은 $(x_4 \bmod n) = r$ 이다.

검증식 $uP + vQ_B = (x_4, y_4)$ 는 다음과 같이 유도될 수 있다.

$$\begin{aligned} uP + vQ_B &= s^{-1}eP + s^{-1}rQ_B \\ &= s^{-1}\{s(k_{A1} + k_{A1}k_{A2})k_B - d_Br\}P + s^{-1}rQ_B \\ &\quad (\because s = (k_{A1} + k_{A1}k_{A2})^{-1}k_B^{-1}\{e + d_Br\}) \\ &= k_{A1}k_BP + k_{A1}k_{A2}k_BP \\ &= (x_2, y_2) + (x_3, y_3) \\ &= (x_4, y_4) \end{aligned}$$

사용자 A가 선택한 랜덤한 정수 k_{A1}과 k_{A2}를 서명자가 알아내기 위해서는 타원곡선 이산대수문제를 풀어야 하며, 이는 계산량적으로 매우 어려운 문제이다. 또한, 서명자는 k_{A2} 및 α의 랜덤성으로 인하여, x_4로부터 x_3와 α의 관계를 유도해내기 어렵다. 따라서, 서명자가 원 서명문 (r_B, s_B)와 서명 결과인 (r, s)를 연결하는 것은 극히 어렵게 된다. 사용자 A의 입장에서는 서명자의 서명문 s_B가 계산되기 전에 e'을 계산하여야 하므로, 메시지의 해쉬값 e를 위조할 수 없다.

본 제안 프로토콜은 타원곡선 디지털 서명 방식으로 표준화 작업 중에 있는 ECDSA의 서명문 생성식과 검증식을 변경하지 않으면서 익명성을 부여하는 방법을 제시하고 있다. 이러한 점은 실제 응용에 있어서 Blind Signature를 생성하기 위한 별도의 서명 모듈을 설계하지 않고도 기존의 ECDSA 기본 알고리즘에 몇 개의 부가적인 처리를 위한 모듈을 추가하면 된다는 장점을 갖는다. 즉, 일반 서명과 Blind 서명에 대한 모듈을 별도로 설계할 필요없이 단일 서명 모듈을 설계하고, 여기에 Blind 서명에 필요한 몇 가지 처리를 추가하면 된다.

Ⅳ. 결 론

본 논문에서는 타원곡선 이산대수문제를 기반으로 하는 디지털 서명 기법 중 ECDSA를 변형한 타원곡선 은닉서명을 제안하였다. 제안한 타원곡선 은닉서명은 피서명자의 익명성을 보장하는 서명기법이며, 그 안전성은 ECDLP의 어려움에 기반을 두고 있다. 또한 본 논문에서 제안하고 있는 타원곡선 은닉서명은 Chaum의 전자지불 프로토콜과 마찬가지로, 고객에게 화폐의 불추적성(untraceability)을 제공하는 전자지불 시스템에 적용할 수 있다. 향후 연구 방향으로는 타원곡선 암호시스템 상에서 화폐의 부정사용을 막고, 고객에게 제한된 익명성을 제공할 수 있는 Fair Cryptosystem과 화폐 처리의 효율성을 보장하기 위한 Off-line성 등에 대한 연구가 요구된다.

[참고 문헌]

[1] T. Okamoto and K. Ohta, "Universal Electronic Cash", Advance in Cryptology-Crypto'91, Lecture Notes in CS, Springer-Verlag, pp32-37, 1992

[2] D. Chaum, "Blind signature for untraceable payments", Crypto'82, pp199-203, 1982

[3] J. Camenisch, J. M. Piveteau, M. Stadler, Blind Signature Based on the Discrete Logarithm Problem, Proc. Eurocrypt 94, pp428-432.

[4] T. ElGamal, "A public key cryptosystem and a signature scheme based on the discrete logarithm", IEEE Trans. Vol. 31, No. 4, pp469-472, 1985

[5] M. O. Rabin, "Digitalized Signatures and Public Key Functions as Intractable as Factorization", Technical Report, MIT/LCS/TR212, MIT Lab, Computer Science, Cambridge, Mass. 1979

[6] R. L. Rivest, A. Shamir, L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp120-126, 1978

[7] N. Koblitz, "Elliptic curve cryptosystems", Mathematics of Computation, number 48, pp203-209, 1987

[8] V. S. Miller, "Use of elliptic curves in cryptography", Advances in Cryptology -

Proceedings of CRYPTO'85, Springer Verlag Lecture Notes in Computer Science 218, pp417-426, 1986

[9] A Certicom Whitepaper, "Remarks on the security of the elliptic curve cryptosystem", September, 1997

[10] D. B. Johnsom, A. J. Menezes, "Elliptic Curve DSA(ECDSA): An Enhanced DSA", A Certicom Whitepaper, 1997

[11] WORKING DRAFT, IEEE P1363 STANDARD, November, 1995