Pairing 연산을 이용하는 효율적인 Identity 기반의 전자서명 알고리즘

박동진, 이필중

포항공과대학교, 전자전기공학과

Efficient Identity-Based Signature Scheme from Pairings

Dong Jin PARK, Pil Joong LEE

Department of EEE., POSTECH

요 약

본 논문에서는 pairing 연산을 이용하는 효율적인 identity 기반의 전자서명 알고리즘을 제안한다. Identity 기반의 전자 서명에서는 pairing 연산이 가장 계산량이 많이 필요한 연산이기 때문에, 제안하는 알고리즘은 이 연산을 최소화하도록 설계되었다. 또한 서명 검증과정에 필요한 2번의 pairing 연산 중에서 1번의 연산을 사전 계산해 둘 수 있게 하여서 온라인 계산에 필요한 연산량도 최소화하였다.

I. 서론

Identity 기반의 암호시스템은 Shamir가 최초로 제안하였다 [6]. 초기의 제안된 identity 기반의 암호시스템은 효율적이지 못하거나 안전하지 않은 문제점을 가지고 있었다. 이러한 문제점은 Boneh와 Franklin에 의해 타원곡선상에서 Weil pairing과 Tate pairing을 이용하는 암호화 방법이 제안되면서 해결되었다 [1]. 이후 pairing을 사용하는 많은 알고리즘들이 제안되었다. Identity 기반의 전자 서명의 경우에도 pairing을 사용한 많은 알고리즘들이 제안되었다 [7, 5, 3, 4, 8].

Pairing 연산을 사용하는 암호시스템에서는 이 연산의 횟수가 전체 시스템의 성능을 결정하게 된다. 왜냐하면 현재까지 발표된 결과 중에서 가장 빠른 pairing 연산도 scalar multiplication 연산에 비해서 5배 가량 느리기 때문이다 [2]. 따라서 이런 시스템을 설계할 때는 최소한의 pairing 연산을 사용하도록 하는 것이 중요하다. 본 논문은 이제까지 제안된 어떤 알고리즘보다 연산량이 작은 identity 기반의 전자서명 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2절에서 제안하는 알고리즘을 설명을 하고, 3절은 제안한 알고리즘을 분석한다. 그리고, 4절에서 결론을 맺는다.

II. 제안하는 서명 알고리즘

G_1과 G_2를 큰 소수 q를 위수로 가지는 군이라고 하자. 본 논문에서는 다음 세가지 성질을 만족하는 mapping $\hat{e}$: $G_1 \times G_1 \rightarrow G_2$ 를 사용한다.

- Bilinear: 모든 $P, Q \in G_1$와 $a, b \in Z$가 $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$를 만족한다.

- Non-degenerate: 어떤 $P, Q \in G_1$에 대해서도 $\hat{e}(P, Q)$는 G_2에서 항등원이 아니다.

- Computable: 어떤 $P, Q \in G_1$에 대해서도 $\hat{e}(P, Q)$를 계산하는 효율적인 알고리즘이 존재한다.

현재까지 알려진 이런 mapping의 종류에는 Weil pairing과 Tate pairing이 있다. 이럴 경우 G_1은 타원 곡선상의 군이 되고, G_2는 정수들의 군이 된다. 각각의 군의 연산은 덧셈과 곱셈이다.

1. 알고리즘 설명

1) 초기화 (Setup)

임의의 $P \in G_1$와 $s \in Z_q^*$를 선택한다. sP를 계산해서 P_{pub}로 한다. 다음의 두가지 암호학적 해쉬 함수를 선택한다: H_1: $\{0, 1\}^* \times G_1 \rightarrow G_2$, H_2: $\{0, 1\}^* \rightarrow G_1$. KGA의 마스터키는 s가 되고, (P, P_{pub}, H_1, H_2)는 시스템 변수가 된다.

2) 개인키 추출 (Extract)

어떤 개인의 identity ID가 주어졌을 때 KGA는 $D_{ID} = sH_2(ID)$를 계산하고, 이 값이 ID에 해당하는 개인키가 된다. 이때 ID에 해당하는 공개키의 값은 $Q_{ID} = H_2(ID)$이다. 두 종류의 키들은 $D_{ID} = sQ_{ID}$의 관계가 있음을 확인할 수 있다.

3) 서명 생성 (Sign)

메시지 m과 개인키 D_{ID}를 가지고 서명 값 (U, V)를 계산한다. 계산 순서는 다음과 같다.

- Z_q^*상의 임의의 r을 선택한다.
- $U = rP$.
- $h = H_1(m, U)$.
- $V = (r+h)^{-1}D_{ID}$.

4) 서명 검증 (Verify)

메시지 m과 서명 (U, V)를 가지고 서명의 올바름을 검증한다. 우선 $c = \hat{e}(P_{pub}, Q_{ID})$값을 계산하고, c와 $\hat{e}(U+hP, V)$의 값이 같으면 올바른 서명으로 확인한다. 검증과정의 올바름은 다음의 식으로 확인할 수 있다.

$\hat{e}(U+hP, V) = \hat{e}((r+h)P, (r+h)^{-1}D_{ID})$

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

이때 c값은 공개된 값으로 만들어지기 때문에 메시지 m을 몰라도 알 수 있다. 따라서 이 값은 서명을 받기 전에 미리 계산해 둘 수 있다. 또한, 같은 ID로 된 여러 개의 m에 대한 서명을 한꺼번에 검증할 때 1번만 계산해 둘 수도 있다.

III. 분석

1. 연산량 분석

제안하는 알고리즘은 서명 생성 과정에서 2번의 scalar multiplication이 필요하다. 2번의 scalar multiplication중에서 U를 계산하는 부분은 사전 계산이 가능하다. 서명 검증 과정에서는 1번의 scalar multiplication과 2번의 pairing이 필요하다. 2번의 pairing중에서 c를 계산하는 부분은 사전 계산을 해두거나, 같은 ID에 대해서 여러 개의 서명을 받았을 경우 재사용이 가능하다.

아래의 표는 현재까지 제안된 pairing을 이용하는 identity 기반의 전자서명 알고리즘들과 제안한 알고리즘과의 연산량 비교이다. 표 1에서 M은 scalar multiplication을 SM은 simultaneous scalar multiplication을, E는 modular exponentiation을, P는 pairing을 의미한다. 1M과 1E는 비슷한 정도의 계산 시간으로 생각할 수 있고, 1SM는 대략 1.5M으로 생각할 수 있다. 1P는 5M과 비슷하다 [2]. 서명 생성 과정과 서명 검증 과정에서 ()안에 표시된 연산은 사전 계산이 가능한 부분을 의미한다.

	서명 생성	서명 검증
본 논문	1M + (1M)	1M + 1P + (1P)
[7]	1SM + (1M)	2P + (1P)
[5]	1M + (1M)	2E + 1P + (1P)
[3]	1M + (1M)	1M + 2P
[4]	1M + (1E + 1M + 1P)	1E + 1P + (1P)
[8]	1SM + (1M)	1M + 2P

표 1: 연산량 비교

Hess는 [4]의 서명 생성 과정에 1E + 1M이 필요하다고 주장했다. 하지만 [4]의 Scheme 1은 1P + 1E + 1SM이 필요하다. 사전 연산으로 pairing과 r값을 계산했을 경우 표 1과 같은 연산량을 필요로 한다. [8]은 서명의 길이를 줄이기 위해서 point compression 방법을 응용하고 있다. 표 1의 연산량 비교에서는 공정한 비교를 위해 point compression과 decompression에 필요한 연산량은 포함시키지 않았다.

표 1에서 보면, 전체 계산 시간을 고려했을 때는 본 논문에서 제안된 알고리즘과 [3]에서 제안된 알고리즘들이 가장 효율적임을 알 수 있다. 그리고, 온라인 계산 시간만을 고려하면 본 논문에서 제안된 알고리즘과 [4]에서 제안된 알고리즘이 가장 효율적임을 알 수 있다. 즉, 제안하는 알고리즘이 전체 계산량과 온라인 계산에서 모두 가장 효율적인 알고리즘임을 확인할 수 있다.

2. 안정성 분석

1) KGA의 마스터키 s 누출

KGA의 마스터키 s를 알기 위해서는 $P_{pub} = sP$ 나 $Q_{ID} = sD_{ID}$를 풀어야 한다. 즉, s를 알기 위해서는 타원곡선 상에서의 이산대수문제를 풀어야 한다.

2) 사용자의 개인키 D_{ID} 누출

사용자의 개인키 D_{ID}를 알아내기 위해서는 s를 구해서 sQ_{ID}를 계산하는 방법과 D_{ID}를 직접 알아내는 방법 2가지가 있다. 첫 번째는 이미 어려움을 보였다. D_{ID}를 직접 구하려면 $V = (r+h)^{-1}D_{ID}$에서 $(r+h)^{-1}$을 알아야 하는데, 이것 역시 타원곡선 상에서의 이산대수문제를 풀어야만 한다.

3) 서명 위조 가능성

임의의 메시지 m에 대해서 유효한 서명 (U, V)가 있다고 가정해 보자. 이때 (U, V)는 검증식($\hat{e}(U+hP, V) = \hat{e}(P_{pub}, Q_{ID})$ $(= c)$)을 만족한다. V는 타원곡선 위에 존재하는 점이기 때문에 $V = xB$로 표현될 수 있다. 이 때 x는 공격자가 알고 있는 정수이고, B는 타원곡선 위의 점이다. B는 생성자 P로 만들어진 군에 속하기 때문에 (공격자가 b의 값을 알 수는 없지만) bP로 표현될 수 있다. 비슷하게 Q_{ID}도 qP로 표현이 가능하다. 이때 q 역시 공격자가 그 값을 알지 못한다. b값이 가질 수 있는 경우를 생각해 보자.

i) b가 s나 q가 아닐 경우

A를 $U+H_1(m, U)P$ 라고 하자. 검증식을 만족하기 위해서는 $\hat{e}(A, B)$의 값이 c가 되어야 한다. c는 $\hat{e}(P, P)^{sq}$이다. 따라서, A를 구하는 문제는 (bP, sP, qP)를 알고 있을 때 $b^{-1}sqP$를 구하는 것만큼 어렵다. 그리고, 이 문제를 풀 수 있으면 CDHP를 풀 수 있다.

ii) b가 s일 경우 $(B = P_{pub})$

이 경우에는 A는 qP $(= Q_{ID})$가 되면 된다. $V = xP_{pub}$이므로 검증식에서 $U+hP$는 $x^{-1}Q_{ID}$이 되어야 한다. 따라서 다음의 식을 얻을 수 있다.

$$U + H_1(m, U)P = x^{-1}Q_{ID} \qquad (1)$$

따라서 서명을 위조하기 위해서는 위의 식을 만족하는 U, m을 구할 수 있어야 한다. 암호학적 해쉬함수를 사용한다면 이런 U, m을 구하기가 현실적으로 불가능하다.

iii) b가 q일 경우 $(B = Q_{ID})$

ii)와 같은 방법으로 다음의 식을 얻을 수 있다.

$$U + H_1(m, U)P = x^{-1}P_{pub} \qquad (2)$$

같은 이유에서 위조가 현실적으로 불가능하다.

i) ii) iii)에서 볼 때 CDHP 풀기, 해쉬함수에서 충돌 찾기, 타원곡선에서의 이산 대수 문제 풀기가 어렵다면 임의의 메시지에 대해서 유효한 서명을 만드는 것은 현실적으로 불가능하다. 즉 제안하는 알고리즘에서는 어떤 메시지 m에 대해서도 위조 서명을 만들 수 없다.

IV. 결론

본 논문은 identity 기반의 효율적인 전자서명 알고리즘을 제안하였다. 특히 제안하는 알고리즘은, 서명 검증 과정에서 사용되는 2번의 pairing연산 중 1번에서 메시지에 독립적인 값을 사용하기 때문에, 보다 효율적인 구현이 가능하다. 제안하는 전자서명 알고리즘은 최근까지 제안된 identity 기반의 전자서명 알고리즘 중에서 계산량 측면에서 볼 때 가장 효율적인 것임을 알 수 있다.

참고문헌

[1] D. Boneh and M. Franklin, "Identity based encryption from the Weil pairing," *CRYPTO 2001*, LNCS 2139, pp. 213-229, 2001.

[2] P. S.L.M. Barreto, H. Y. Kim, B. Lynn and M. Scott, "Efficient algorithms for pairing-based cryptosystems," *CRYPTO 2002*, LNCS 2442, pp. 354-369, 2002.

[3] J. C. Cha and J. H. Cheon, "An identity-based signature from gap Diffie-Hellman groups," *PKC 2003*, LNCS 2567, pp. 18-30, 2003.

[4] F. Hess, "Efficient identity based signature schemes based on pairings," *SAC 2002*, LNCS 2595, pp. 310-325, 2002.

[5] K. G. Paterson, "ID-based signature form pairings on elliptic curves," *Electronics Letters*, vol. 38 (18), pp. 1025-1026, 2002.

[6] A. Shamir, "Identity-based cryptosystems and signature schemes," *CRYPTO '84*, LNCS 196, pp. 47-53, 1985.

[7] R. Sakai, K. Ohgishi and M. Kasahara, "Cryptosystems and signature schemes based on pairing," *SCIS 2000*, pp. 26-28, 2000.

[8] X. Yi "An identity-based signature scheme from the Weil pairing," *IEEE communications letters*, vol. 7, no. 2, pp. 76-78, 2003.