2000년 한국정보처리학회 추계 학술발표논문집 제7권 제2호

효율적인 Okamoto-Schnorr 기반 은닉 서명

이진호° 윤마루 이명선 김태윤
고 려 대 학 교 컴 퓨 터 학 과
{jhlee}@netlab.korea.ac.kr

Efficient Blind Signature based on Okamoto-Schnorr scheme

Jean-Ho Lee° Ma-ru Yun Myoung-sun Lee Tai-Yun Kim
Dept. of Computer Science & Engineering, Korea University

요약

은닉 서명은, 정보 은닉성이 가지는 정보와 서명 사이의 무연관성(unlinlability) 때문에 전자 지불 시스템에서 응용되고 있다. 은닉 서명 생성 과정에 사용되는 계산식에 나타나는 역원 인자 연산은, 다른 솟수나 이산 대수 기저값과는 달리, 전처리 과정에서 계산될 수 없는 실시간 계산을 추가적으로 요구한다. 이것은 은닉 서명이 전자 지불 시스템에 사용되는 경우, 전체 시스템에 대한 오버헤드로 작용할 수 있는 문제점이 될 수 있다. 본 연구에서는 Okamoto가 제시한 Schnorr 기반 은닉 서명 기법을 근간으로, 은닉 서명 계산 과정에서 역원 계산을 제거하여 은닉 서명 생성 속도를 향상시킬 수 있는 방안을 제시한다.

1. 서론

전자 지불 환경은 지불 처리 속도와 익명성에 대한 문제가 매우 중요하다. 이런 환경에 적합하게 사용되는 서명 기법인 은닉 서명 기법은, Chaum[1]에 의해 처음으로 제시하였다. 은닉 서명 기법은 전자 서명이 제공하는 기본적인 특징인 서명자 인증, 위조 방지, 부인 방지, 서명 내용의 무결성 보장 등의 성질 이외에, 비연관성(unlinkability)을 추가로 제공한다. 은닉 서명 기법은 서명자로부터 메시지의 내용을 은폐시킬 수 있기 때문에, 주로 전자 화폐 시스템이나 전자 투표 시스템과 같이 메시지 내용과 관련 정보에 대한 비밀성이 보호되는 응용분야에서 사용되고 있다.

은닉 서명의 은닉성을 이용하는 전자 화폐 시스템이 효율적으로 구현되기 위해서는, 수행 속도의 문제가 제기된다. 전자 화폐를 발행, 입금, 출금하는 과정에서 발생하는 은닉 서명의 생성이나 검증 단계의 계산 처리 속도는 전자 화폐 시스템 전체의 실행 시간에 영향을 미치게 된다. 따라서, 은닉 서명의 계산 처리 속도를 개선한다면, 은닉 서명을 사용하는 전자 지불 시스템의 처리 속도를 향상시킴으로써, 전체 시스템의 수행 시간을 줄이고 수행 비용을 줄임으로써, 시간과 비용 측면에서 성능 향상을 가져올 수 있다.

본 연구에서는, 은닉 서명 계산 과정을 고찰한 다음, 은닉 서명의 생성과 검증 처리 속도에 영향을 미치는 요소들을 살펴보고, 구체적으로 Okamoto가 제안한 Okamoto- Schnorr 계산 과정을 변경하여, 계산 처리 시간의 향상을 높이는 은닉 서명 방식을 제안한다.

2장에서는 은닉 서명 기법에 대해 고찰해 보고, 3장에서는 Okamoto-Schnorr 은닉 서명 기법의 계산 과정을 개선하여 실제로 속도 향상 방안을 적용시켜 본다. 4장에서는 속도를 개선한 Okamoto-Schnorr 변형 은닉 서명 기법의 성능을 분석하고, 5장에 결론 및 향후 과제를 제시한다.

2. 관련연구

2.1 은닉 서명 기법

은닉 서명 기법은 크게 RSA 기반과 ElGamal 기반 방식의 2가지로 나누어 볼 수 있다. Chaum 은 RSA 방식[7]에 기반한 서명자와 서명 내용과의 비연관성을 제공하는 은닉 서명 기법을 제시하였다[3]. Okamoto는 Schnorr 기법[2]을 확장시켜 은닉서명 방식으로 제안하였다[4]. Okamoto-Schnorr 기법은 해쉬 함수를 적용하여 은닉 메시지를 생성함으로써, 서명 대상의 크기가 줄어들게 되어,해쉬 함수를 사용하지 않는 은닉 서명 방식보다, 서명 생성 과정에서 계산 시간을 단축시킬 수 있다. 또한 기존의 삭제-와-선택(cut-and-choose) 기법 대신에 비트 커미트먼트(bit commitment)를 사용하는 영지식 기법을 도입하여 통신 비용을 줄일 수 있는 장점이 있다. Horster는 ElGamal 서명기법[8]의 변형들로 이루어진 Meta-ElGamal 기법에 은닉 서명 방식을 적용시켜,

서명자와 검증자 모두에게 지수(exponent) 계산량을 줄임으로써 효율성을 제공했다[6]. Camenisch와 Stadler는 DSA 서명 기법과 Nyberg- Rueppel서명 기법을 은닉 서명 방식에 적용시켰다[5].

은닉 서명의 생성 과정을 고찰하기 위해, 기존 방식의 프로토콜들을 비교해보면 그림 1과 같다:

(*은닉 서명 프로토콜의 매개 변수

p,q : 솟수(|p|≥512bits, |q|≥140bits, q|(p-1)),

n = p*q,

e,d : Z의 원소(gcd(e,ψ(n))=1, e*d≡1(mod ψ(n))

g : 이산 대수 기저($g^q \equiv 1 \pmod p$, |g|≥212bits)

$x \in Z_q$, $y=g^x \pmod p$,

m : 은닉 서명할 대상 메시지,

서명자의 RSA기반 (공개키, 개인키) = ((n,e), (p,q,d))

서명자의 ElGamal 기반 (공개키,개인키) = (x,y)

역원을 인자로 가지는 연산의 수행 시간은, 순인자로 이루어진 연산을 수행시간보다 더 걸리며, 컴퓨팅 자원의 소모가 더 많다. 왜냐하면 연산을 수행하기에 앞서, 역원 값을 계산해내야 하기 때문에, 여기에서 추가적인 계산적 오버헤드가 발생하게 되는 것이다. 따라서, 서명 프로토콜을 수행할 때, 역원 인자가 포함된 계산을 사용하지 않는다면, 추가로 발생하는 오버헤드를 제거할 수 있기 때문에, 계산 속도를 향상시킬 수 있을 것이다.

또 한가지 방법은, 역원 인자가 포함된 연산을 위해, 전처리 과정을 통해 미리 역원 인자값을 계산하는 것이다. 그러나, 이것은 역원 인자가 미리 정해져 있는 경우에만 가능하기 때문에, 역원 인자로 임의의 숫자(random number)가 사용되는 프로토콜에 적용하기에는 비현실적이다.

본 연구에서는, 은닉 서명 생성 과정의 계산 속도를 향상시키기 위해, 기존의 은닉 서명 기법을 수정하여, 역원 인자가 포함된 연산식을 제거시키고자 한다. 이를 위해, Okamoto-Schnorr 은닉 서명 기법에서 역원 인자 연산식을 제거시킨, Okamoto-Schnorr 은닉 서명의 변형 형태를 제시하고자 한다.

은닉서명 기법 / 단계	Chaum의 RSA기반 방식		Okamoto의 Schnorr기반 방식		Horster의 ElGamal기반 방식	
	User	Signer	User	Signer	User	Signer
단계 1	$r \in Z_n^*$ $m' = m * r^e \pmod n$		$a,b \in Z_q$ $t = r' y^{-a} g^b \pmod p$	$k \in Z_q$ $r' = g^k \pmod p$	$a,b \in Z_q$ $r = r'^a g^b \pmod p$	$k \in Z_q$ $r' = g^k \pmod p$
단계 2		$s' \equiv (m')^d \pmod n$	$r = h(t,m)$ $t' = r + a \pmod q$		$m' = a^{-1}(m+r) - r' \pmod q$	
단계 3				$s' = xt' + k \pmod q$		$s' = x(m'+r') - k \pmod q$
서명 방정식	$s = s' * r^{-1} \pmod n$		$s = s' + b \pmod q$		$s = as' - b \pmod q$	
검증 방정식	$s^e \pmod n = m \pmod n$		$t = g^s y^{-r} \pmod p$		$g^s = y^{m+r} r^{-1} \pmod p$	

그림 1 기존 은닉 서명 프로토콜의 비교

3. 은닉 서명의 효율성

3.1 은닉 서명의 효율성

스마트 카드를 위한 효율적인 서명 기법을 위해 Schnorr 는 해쉬함수를 사용해서 메시지 교환량의 크기가 줄이고, 따라서 계산량도 축소시킴으로써, 계산 용량이 작은 마이크로 프로세서에 적합한 은닉 서명 방식을 제안하였다[Sch91]. 이와 비슷하게, 계산량보다는 계산 속도의 관점에서 보면, 은닉 서명 기법이 전자 지불 시스템에 적용될 때, 은닉 서명의 생성과 검증 과정의 처리 속도가 지불 시스템 전체의 처리 성능에 영향을 끼친다고 볼 수 있다. 은닉 서명의 생성과 검증 과정에서 속도에 영향을 줄 수 있는 요소들은 여러 가지가 존재하지만, 계산 과정을 구성하고 있는 연산자와 인자로 이루어진 연산의 효율성에 초점을 맞추어 고찰하고자 한다.

서명 방정식을 계산하는 과정을 살펴보면, 다음과 같이, 방정식의 매개 변수의 역원(inverse of parameter) 값이 인자로서 사용되는 연산이 요구된다:

Chaum[3]의 경우, 획득 은닉 서명 $s=s'*r^{-1} \pmod n$

Okamoto[4]의 경우, 은닉 인자 $t = r' y^{-a} g^{-b} \pmod p$ 와, 서명 검증 방정식 $t == g^s y^{-r} \pmod p$

Horster[6]의 경우, 은닉 메시지 $m'=a^{-1}(m+r)-r' \pmod q$ 와, 서명 검증 방정식 $g == y^{m+r} r^{-1} \pmod p$

3.2 변형된 Okamoto 은닉 서명 기법

기존의 Okamoto-Schnorr 은닉 서명 기법에서 계산 속도를 개선시키기 위해, 기존 프로토콜의 가정과 단계를 그대로 이용하며, 역원 인자가 포함된 연산식을 제거하고 연관된 연산식을 개발한다.

① 사용자가 서명자로부터 받은 커미트먼트 r'을 사용하여 중간 은닉 인자 r을 생성하기 위해 중간 인자 t를 계산할 때의 계산식에서 역원 계산을 제거시킬 수 있는데 2가지 방법이 있다:

$t = r' y \; g \pmod p$

⇒ i) $t = r' y^b g^a \pmod p$

ii) $t = r' y^a g^b \pmod p$

새로 얻은 중간 인자 t를 가지고, 기존 방식대로 해쉬함수를 사용하여 중간 은닉 인자 r=h(t,m)을 계산한다.

② 사용자가 은닉 메시지 m'을 계산할 때, 역원을 제거한 연산식이 호환되게 하기 위해, 계산식을 변형한다:

m′= r+a (mod q)
⇒ i) m′ = r-b (mod q)
ii) m′ = r-a (mod q)

③ 서명자는 은닉 서명 s′과 서명 방정식 s를 변형시켜 계산한다:

s′ = k+m′x (mod q)
s = s′+b (mod q)
⇒ I) s′=k-m′x (mod q)
s = s′ + a (mod q)
ii) s′ = k-m′x (mod q)
s = s′ + b (mod q)

④ 사용자는 서명 방정식 (r,s)를 가지고, 은닉 서명 검증을 한다:

t == $g^s y^{-r}$ (mod p)
⇒ i) t == $g^s y^r$ (mod p)
(→) : t = $r'y^b g^a$(mod p) = $(g^k)(g^x)^b g^a$(mod p)
= g^{k+bx+a}(mod p)
(←) : $g^s y^r$ (mod p) = $g^{s'+a}$ (g^{rx}) (mod p)
= $g^{(k-m'x)+a} g^{rx}$(mod p)
= $g^{k-x(r-b)+a+rx}$(mod p) = g^{k+bx+a}(mod p)
ii) t == $g^s y^r$ (mod p)
(→) : t = $r'y^a g^b$ (mod p) = $g^k(g^{ax})g^b$ (mod p)
= g^{k+ax+b} (mod p)
(←) : $g^s y^r$ (mod p) = $g^{s'+b} g^{rx}$ (mod p)
= $g^{k-m'x+b} g^{rx}$(mod p)
= $g^{k-x(r-a)+b} g^{rx}$(mod p) = g^{k+ax+b}(mod p)

본 연구에서 제안하는 은닉 서명 기법은, 기존의 Okamoto- Schnorr 은닉 서명 기법과 동일하게, 키인증 센타(KAC,Key Authenticaion Center) 혹은 신뢰 센타(TTC,Trusted Third Party)의 존재를 가정하고, 이것의 공개키와 개인키를 생성하는 초기화 단계를 수행한다.

아래와 같이 4단계의 은닉 서명 프로토콜을 수행한다.

(p,q : 솟수 (q|(p-1))
g: 이산 대수 기저 (g ≡ 1 (mod p))
h: Z *Z → {0,…, 2 -1} : 해쉬 함수
(x,y) : 서명자의 개인키와 공개키)

[단계 1: 서명자의 초기 설정 단계]

단계1-1: 개인키 x 에 대한 공개키 $y \equiv g^x$ (mod p)를 생성한다.

단계1-2: 임의의 수 k∈Z를 선택하고, 비트 커미트먼트 $r'=r'y^b g^a$ (mod p)를 계산하여 사용자에게 전달한다.

[단계 2: 사용자의 메시지 전송 단계]

단계2-1:임의의 수 a,b∈Z를 선택하고, 서명자로부터 받은 r′을 사용하여, $t=r'y^b g^a$(mod p)를 계산한다.

단계2-2: 중간 은닉 인자 r=h(t,m)을 계산한다.

단계2-3: 은닉 메시지 m′=r-b(mod q)를 계산하여 서명자에게 전달한다.

[단계 3: 서명자의 은닉 서명 생성 단계]

단계3-1: 사용자로부터 받은 m′에 대해 은닉 서명 s′=k-m′x(mod q)를 생성한다.

단계3-2: 은닉 서명 s′을 사용자에게 전달하고, (r′,m′,k,s′)을 보관한다.

[단계 4: 사용자의 은닉 서명 검증 단계]

단계4-1: 서명자로부터 받은 은닉 서명 s′에 대해, s=s′+a (mod q)를 계산한다.

단계4-2: $g^s y^r$ == t (mod p) 인지 검사하여, 만족하는 경우 올바른 은닉 서명으로 받아들인다.

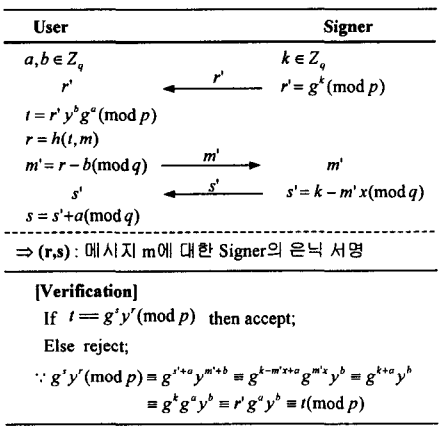

그림 2 변형된 Okamoto 은닉 서명 기법

4. 성능 평가

본 연구에서 제안한 기법은, 기존의 Okamoto-Schnorr 은닉 서명 기법에서 역원 인자 계산을 제거했기 때문에, 실제로 은닉 서명의 크기나 프로토콜 과정에서 교환되는 데이터 량이, 2가지 방식 모두가 거의 차이가 없이 동일하다. 그러나, 서명 계산 과정에서, 처리 시간의 오버헤드가 되는 역원 인자를 포함하는 연산을 피할 수 있기 때문에 계산 시간을 줄일 수 있고, 따라서 서명 생성 과정의 전체적인 수행 시간을 줄일 수 있게 된다.

제안한 Okamoto 변형 은닉 서명 기법의 성능 분석을 위해, 기존의 은닉 서명 기법들과 비교하였고, 표1과 같다. 성능 비교를 위해, p를 512비트, q를 140비트, 해쉬함수의 크기를 128비트, 안전성 인자의 크기를 20비트라고 가정하였다.

기법 / 비교항목	Chaum	Horster	Okamoto	Okamoto-변형
안전성 근거	소인수 분해	이산 대수 문제	이산 대수 문제	이산 대수 문제
전처리 기능	가능	가능	가능	가능
서명 크기	1024	652	268	268
데이터 교환량	2048	792	792	792
역원 계산 회수	1회	2회	2회	사용안함

<표1. Okamoto 변형 은닉 서명 기법의 성능 비교>

5. 결론

본 연구에서는, 은닉 서명이 가지는 효율성에 대해 고찰해보고, 계산량보다는 계산 시간의 측면에서 은닉 서명 생성 속도를 개선할 수 있는 방안을 제시하였다.

구체적으로, Okamoto-Schnorr 은닉 서명을 기반으로, 은닉 서명 생성 과정의 역원 인자 연산을 제거시킴으로써, 계산 속도를 개선시킨 은닉 서명 방식을 제시하고, 이를 기존의 은닉 서명 방식과 성능을 비교하였다.

본 연구에서 제안하는 Okamoto-Schnorr 변형 은닉 서명 방식은, 해쉬 함수를 사용하는 Schnorr 서명을 기반으로 하기 때문에, 전자 지불 시스템에 응용될 수 있고 기존의 전자 지불 시스템의 성능을 보다 향상시킬 수 있을 것이다.

은닉 서명이 전자 지불 시스템에 적용되기 위해서는 안전성에 대한 문제가 논의되어야 할 것이다. 이를 위해, 향후 과제로는 전자 지불 시스템에서의 여러 가지 공격에 대해 은닉 서명이 가지는 안전성에 대한 연구가 요구된다.

6. 참고 문헌

[1] David Chaum, "Blind Signature for Untraceable Payments", Proc. Crypto'82, pp.199~203, LNCS, Springer-Verlag, 1983.

[2] C.P.Schnorr, "Efficient Identification and Signatures for Smart Cards", Proc. Crypto '89, pp.239~252, LNCS, Springer-Verlag, 1990.

[3] David Chaum, "Blinding for Unanticipated Signatures", Proc. Eurocrypt '87, pp. , LNCS, Springer-Verlag, 1988.

[4] Tatsuaki Okamoto, "Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes", Proc. Crypto'92, pp.31~63, LNCS, Springer-Verlag, 1993.

[5] Jan L. Camenisch, Jean-Marc Piveteau, Markus A. Stadler, "Blind Signatures based on the Discrete Logarithm Problem", Proc.Eurocrypt'94, pp.428~432, LNCS, Springer-Verlag, 1995.

[6] Patrick Horster, Markus Michaels, Holger Petersen, "Efficient Blind Signature Schemes based on the Discrete Logarithm Problem", Technical Report TR-94-6-D, Univ. of Tech. Chemnitz-Zwichau, Dept. Computer Science, June, 1994.

[7] Ronald L. Rivest, Adi Shamir, Leonard M. Addleman, "A Method for Obtaining Digital Signatures and Public Key Cryptosystems", CACM,21(2):120~126, 1978.

[8] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE Trans. on Information Theory 31, pp.469~472, 1985.