키 노출 공격에 안전한 ID-기반의 강한 지정된 검증자 서명 기법

Secure ID-based Strong Designated Verifier Signature Scheme Against Key-compromise Attack

이지선*, 장직현**, 이동훈*
고려대학교*, 서강대학교**

Ji-Seon Lee(jslee702@korea.ac.kr)*, Jik-Hyun Chang(jchang@sogang.ac.kr)**,
Dong-Hoon Lee(donghlee@korea.ac.kr)*

요약

강한 지정된 검증자 서명(Strong Designated Verifier Signature Scheme)은 지정된 검증자에게 서명자와 같이 서명을 생성할 수 있는 능력을 줌으로써 주어진 서명이 실제 서명자와 검증자 중에 누가 만든 것인지를 알 수 없도록 하는 서명자의 익명성을 보장하는 특별한 서명 방식으로 소프트웨어 분배 또는 전자투표 등에서 이용될 수 있다. 본 논문에서는, 강한 지정된 검증자 서명 방식의 중요한 성질인 소스 하이딩(source hiding)과 키 노출 공격 (key-compromise attack)의 관계를 살피고 두 성질을 동시에 만족하는 강한 지정된 검증자 서명을 생성할 수 없음을 보인다. 마지막으로 키 노출 공격에 안전한 ID 기반의 강한 지정된 검증자 서명 기법을 제안한다.

■ 중심어 : | 강한 지정된 검증자 서명 | ID-기반 서명 | 소스 하이딩 | 키 노출 공격 |

Abstract

A strong designated verifier signature scheme is a special type of signature scheme which provides signer anonymity by enabling the specified recipient, called a designated verifier, to simulate a signature which is indistinguishable from the signer's signature. It has many applications such as software distribution or electronic voting. In this paper, we consider two important security properties of strong designated verifier signature scheme - source hiding and security against key-compromise attack. We show that the two properties cannot be achieved at the same time. Finally, we present a new ID-based strong designated verifier signature scheme which is secure against key-compromise attack.

■ keyword : | Strong Designated Verifier Signature | ID-based Signature | Source Hiding | Key-compromise Attack |

I. 서 론

전자서명(digital signatures)은 메시지를 보낸 사람의 신원을 증명하기 위해 사용되는 암호 기법으로 서명을 수신한 사람은 누구나 실제 서명자에 의해 생성된 서명임을 알 수 있다. 이러한 전자 서명 기법의 중요한 성질 중에 하나는 부인 방지(non-repudiation)인데, 이는 서명 송신 후에 서명 생성자가 자신이 서명을 보냈

* 본 연구에 참여한 연구자(의 일부)는 '3 단계 BK21 사업'의 지원비를 받았음.
접수번호 : #090901-004
접수일자 : 2009년 09월 01일
심사완료일 : 2009년 09월 17일
교신저자 : 이동훈, e-mail : donghlee@korea.ac.kr

다는 사실을 부정할 수 없도록 하는 기능으로 어떤 계약이 이루어진 후 생길 수 있는 분쟁을 막을 수 있다. 하지만, 때때로 이러한 부인 방지 기능이 적절치 않을 수 있는데, 특히 익명성을 제공해야 하는 상황에서는 부인 방지 기능을 제공하지 않는 특별한 서명 기법이 필요하다.

1996년에 Jakobsson 외 2인[1]에 의해 제안된 지정된 검증자 서명 (designated verifier signatures)은 서명자가 지정한 수신자만이 실제 서명자에 의해 서명이 생성되었음을 확신할 수 있도록 하는 특별한 기법으로 서명자가 자신이 만든 서명을 수신자 외에는 확신할 수 없도록 하고 싶은 경우에 유용한 방식이다. 즉, A가 만든 서명을 지정된 수신자 B만이 서명 생성자가 A임을 확신할 수 있고, A와 B 외의 제 3자는 서명이 A와 B 중 누구에 의해 생성되었는지 알 수 없도록 한다. 이는 지정된 검증자(designated verifier) B가 실제 서명자 A가 만든 서명과 구별 불가능한 서명을 생성할 수 있도록 함으로써 가능하다. 따라서 지정된 검증자 서명은 서명자와 지정된 검증자 중에 누가 서명을 생성했는지 알 수 없도록 하는 서명자 익명성을 보장하고 소프트웨어 분배, 전자 투표 등과 같은 분야에서 이용될 수 있다[1].

Jakobsson 외 2인이 제안한 기법에서는, 서명자가 자신의 비밀키로 서명을 생성하거나 지정된 검증자가 자신의 비밀키로 서명자의 서명과 구별 불가능한 서명을 생성할 수 있고, 서명을 검증하기 위해서는 서명자의 공개키와 지정된 검증자의 공개키가 필요하다. 하지만 서명 검증에 서명자와 지정된 검증자의 공개키만이 필요하다는 것은 서명자가 누구인지를 노출시킬 수 있는 위험성을 가지고 있다. 다음의 시나리오를 생각해 보자. 만약에 서명자 A로부터 지정된 검증자 B에게 전달되는 서명을 제 3자가 훔쳤을 경우에 아직 B가 서명을 수신하지 않았다는 것을 확신할 수 있는 상황이라면, 제 3자는 A와 B의 공개키를 이용하여 훔친 서명을 검증할 수 있다. 따라서 공격자는 그 서명이 옳음을 검증할 수 있고 실제 A가 생성한 서명이라는 것을 확신할 수 있게 된다. 이러한 문제를 해결하기 위해 서명 검증시에 지정된 검증자의 비밀키가 포함되도록 구성한 강한 지정된 검증자 서명(strong designated verifier signatures)이 제안되었다[2]. '강함(strongness)' 성질은 지정된 검증자가 서명을 검증할 때 자신의 비밀키를 사용하도록 요구하는 것을 의미한다. 이는 서명 검증조차 오직 지정된 검증자만이 할 수 있도록 함으로써 보다 안전하게 구성한 것이다. 이후로 제안된 대부분의 지정된 검증자 서명 기법은 이러한 강함 성질을 제공한다. ID-기법의 서명 기법이 제안된 이후로는[3] ID-기반의 강한 지정된 검증자 서명이 제안되었다[4][5].

최근에 Zhang과 Mao는 ID-기반의 강한 지정된 검증자 서명을 제안하고 제안한 기법이 다음의 보안 요구사항을 만족함을 보였다[5].

- 위조불가능성 (Unforgeability) : 서명자의 비밀키 또는 지정된 검증자의 비밀키 없이 강한 지정된 검증자 서명을 생성하는 것은 계산적으로 불가능하다.
- 소스 하이딩 (Source hiding) : 메시지와 그 메시지에 대한 서명이 주어졌을 때, 서명자와 지정된 검증자의 비밀키가 모두 주어진다고 하더라도, 그 서명을 둘 중에 누가 생성했는지 알 수가 없다.

논문 [6]에서 Lee와 Chang은 강한 지정된 검증자 서명이 키 노출 공격에 안전해야 서명자 익명성을 보장할 수 있음을 보였다. 만약에 제 3자인 C가 서명자 A의 비밀키를 어떤 경로를 통해서든 획득하였고 A로부터 지정된 검증자 B에게 전달되는 서명을 C가 훔쳤다고 가정해 보자. 이 때 C가 아직 B가 서명을 수신하지 않았다는 것을 확신할 수 있는 상황이고 서명이 B의 비밀키뿐 아니라 A의 비밀키로도 검증 가능하다면, C는 B의 비밀키가 없더라도 A의 비밀키를 이용하여 훔친 서명을 검증할 수 있다. 따라서 공격자는 그 서명이 옳음을 검증할 수 있고 실제 A가 생성한 서명이라는 것을 확신할 수 있게 된다. Lee와 Chang은 이러한 상황에 대해서도 서명의 익명성을 보장하기 위하여 서명자의 비밀키가 노출되더라도 서명 검증이 불가능하여 서명자의 익명성을 보장할 수 있어야 한다고 주장하였다. 이는 다음과 같이 정리할 수 있다.

- 키 노출 공격에 대한 안전성 (Security against key-compromise attack) : 메시지와 그 메시지에 대한 서명이 주어졌을 때, 서명자의 비밀키가 노출되더라도 서명자와 지정된 검증자 중에 누가 그 서명을 생성했는지 알 수 없다.

본 논문에서는 최근에 제안된 Zhang과 Mao가 제안한 ID-기반의 강한 지정된 검증자 서명 기법이[5] 소스 하이딩 성질은 만족하지만 키 노출 공격에는 안전하지 않음을 보인다. 또한 본 논문에서는 소스 하이딩과 키 노출 공격에 대한 안전성을 동시에 제공하는 강한 지정된 검증자 서명을 설계하는 것은 불가능함을 보인다. 2장에서는 이 두 성질간의 관계를 살피고 3장에서는 기존에 제안된 Kumar et al.의 기법과 Zhang과 Mao의 기법을 살펴본다. 4장에서는 키 노출 공격에 강한 새로운 ID-기반의 강한 지정된 검증자 서명 기법을 제안하고, 제안한 기법을 분석한다. 마지막으로 5장에서는 결론을 맺는다.

II. 소스 하이딩 vs 키 노출 공격에 대한 안정성

논문 [5]에서 Zhang과 Mao는 위조 불가능성과 소스 하이딩 성질을 만족하는 ID 기반의 강한 지정된 검증자 서명을 제안하였다. 소스 하이딩을 만족하기 위해서는 서명자의 비밀키로도 서명 검증이 가능하고 지정된 검증자의 비밀키로도 서명 검증이 가능해야 한다. 따라서 제 3자에게 서명자와 검증자의 비밀키가 주어지더라도 누가 생성한 서명인지 알 수 없도록 할 수 있다.

키 노출 공격에 안전한 강한 지정된 검증자 서명을 만들기 위해서는 서명자의 비밀키가 노출되더라도 검증할 수 없도록 해야 한다. 만약에 공격자가 서명자의 비밀키를 가지고 있는 상황에서 서명자 A로부터 검증자 B로 전달되는 서명을 가로챘을 때, 서명자의 비밀키를 이용하여 검증함으로써 올바른 서명임을 확신할 수 있고 아직 서명이 검증자에게 도달하지 않았음을 알기 때문에 서명자가 생성한 서명이라는 것을 알아낼 수 있다. 따라서 키 노출 공격에 안전하려면 서명이 오로지 지정된 검증자의 비밀키로만 검증 가능하여야 한다. 따라서, 소스 하이딩을 만족하고 동시에 키 노출 공격에 안전한 강한 지정된 검증자 서명은 설계하는 것은 불가능하다.

III. 기존에 제안된 ID-기반의 강한 지정된 검증자 서명 기법들

이번 장에서는 최근에 제안된 두 기법을 살펴 본다. 우선 Kumar et al.의 기법을 살펴 볼텐데, 이 기법은 키 노출 공격에는 안전하지만, 소스 하이딩 성질은 만족하지 않는다. 다음으로 Zhang과 Mao의 기법을 살펴 본다. Zhang과 Mao의 기법은 반대로 소스 하이딩 성질은 만족하지만, 키 노출 공격에는 안전하지 않다.

본 논문에서 G_1을 소수인 위수 q를 갖는 덧셈 연산 군이라 하고 G_2를 같은 위수를 갖는 곱셈 연산 군이라고 하자. 그리고 P는 G_1의 생성자이다. 이 때 G_1과 G_2에서 이산대수문제(DLP)는 어렵다고 가정한다. 임의의 $P, Q \in G_1$와 $a, b \in Z_q^*$에 대하여 아래와 같은 조건을 만족하는 함수 $e : G_1 \times G_1 \rightarrow G_2$를 admissible bilinear map이라 한다.

- Bilinearity : $e(aP, bQ) = e(P, Q)^{ab}$.
- Non-degeneracy : $e(P, Q) \neq 1$을 만족하는 $P, Q \in G_1$가 존재한다.
- Computability : 모든 $P, Q \in G_1$에 대하여 $e(P, Q)$를 효율적으로 계산할 수 있는 알고리즘이 존재한다.

1. Kumar et al.의 기법

ID-기반의 기법에서는 신뢰기관인 키 생성 기관(KGC, Key Generation Center)이 주어진 ID에 대응되는 비밀키를 생성하여 각 사용자에게 보내준다. Zhang-Mao 서명 기법은 다음과 같이 Setup, KeyExtract, SigGen, SigVer, SigSimul 알고리즘으로 구성된다.

Signer A		Designated Verifier B
$r_1, r_2, r_3 \in Z_q^*$ $U_1 = r_1 Q_B$ $U_2 = r_2 Q_A$ $U_3 = r_1 r_3 Q_B$ $H = H_2(m, e(r_2 Q_B, S_A))$ $V = r_3 H + r_1^{-1} S_A$	$\xrightarrow{\{U_1, U_2, U_3, V\}}$	Verification $H = H_2(m, e(U_2, S_B))$ $e(U_1, V) \stackrel{?}{=} e(U_3, H) e(S_B, Q_A)$ Simulation $r_1', r_2', r_3' \in Z_q^*$ $U_1' = r_1' Q_A$ $U_2' = r_2' Q_B$ $U_3' = r_1' r_3' Q_A$ $H' = H_2(m, e(r_2' Q_A, S_B))$ $V' = r_3' H' + (r_1')^{-1} S_B$

그림 1. Kumar et al.'s Scheme

(1) Setup : KGC는 그룹 G_1과 G_2를 선택하고, 마스터 비밀값 $s \in Z_q^*$를 선택한 후 공개키 $P_{pub} = sP$를 계산한다. 또한 암호학적 일방향 해쉬 함수 $H_1 : \{0,1\}^* \rightarrow G_1$과 $H_2 : \{0,1\}^* \times G_2 \rightarrow G_1$ 를 선택한다.

KGC는 마스터 비밀키 s를 비밀로 하고 다음과 같은 공개 시스템 파라미터를 공개한다.

$$param = (G_1, G_2, e, q, P, P_{pub}, H_1, H_2)$$

(2) KeyExtract : 공개 확인자가 ID_U인 유저는 비밀키를 얻기 원할 때, KGC로부터 비밀키 $S_{ID_U} = s Q_{ID_U}$를 받는다. 여기에서 공개 확인자 ID_A를 갖는 서명자 A는 자신의 키 쌍 (S_A, Q_A)를 갖고 지정된 검증자 B는 (S_B, Q_B)를 갖는다.

(3) SigGen : 서명자 A는 메시지 m에 대해서 임의의 세 수 $r_1, r_2, r_3 \in Z_q^*$를 선택하여 다음과 같이 서명 $\sigma = (U_1, U_2, U_3, V)$를 생성한다.

$$U_1 = r_1 Q_B$$
$$U_2 = r_2 Q_A$$
$$U_3 = r_1 r_3 Q_B$$
$$H = H_2(m, e(r_2 Q_B, S_A))$$
$$V = r_3 H + r_1^{-1} S_A$$

(4) SigVer : 지정된 검증자는 메시지와 서명의 쌍 (m, σ)을 받으면, $H = H_2(m, e(U_2, S_B))$를 계산한 후, 다음의 등식이 만족되면 서명을 받아들인다.

$$e(U_1, V) = e(U_3, H) e(S_B, Q_A)$$

(5) SigSimul : 지정된 검증자는 임의로 세개의 수 $r_1', r_2', r_3' \in Z_q^*$를 선택하고 서명 $\sigma' = (U_1', U_2', U_3', V')$ 을 다음과 생성한다.

$$U_1' = r_1' Q_A$$
$$U_2' = r_2' Q_B$$
$$U_3' = r_1' r_3' Q_A$$
$$H' = H_2(m, e(r_2' Q_A, S_B))$$
$$V' = r_3' H' + (r_1')^{-1} S_B$$

Kumar et al.의 기법에서는 서명을 검증하기 위하여 $H = H_2(m, e(U_2, S_B))$를 계산해야 하므로 반드시 검증자의 비밀키를 알아야 한다. 즉, 서명자의 비밀키만을 갖고는 검증을 할 수가 없다. 따라서 서명자의 키 노출 공격에 안전하다.

2. Zhang-Mao의 기법

Zhang-Mao 서명 기법도 다음과 같이 Setup, KeyExtract, SigGen, SigVer, SigSimul 알고리즘으로 구성된다.

(1) Setup : KGC는 그룹 G_1과 G_2를 선택하고, 마스터 비밀값 $s \in Z_q^*$를 선택한 후 공개키 $P_{pub} = sP$를 계산한다. 또한 암호학적 일방향 해쉬 함수

Signer A		Designated Verifier B
$r_1, r_2 \in Z_q^*$ $U_1 = r_1 Q_B$ $U_2 = r_1 r_2 Q_B$ $H = H_2(m, U_1, U_2)$ $V = r_2 H + r_1^{-1} S_A$	$\xrightarrow{\{U_1, U_2, V\}}$	——— Verification ——— $H = H_2(m, U_1, U_2)$ $e(U_1, V) \stackrel{?}{=} e(U_2, H) e(S_B, Q_A)$ ——— Simulation ——— $r_1', r_2' \in Z_q^*$ $U_1' = r_1' Q_A$ $U_2' = r_1' r_2' Q_A$ $H' = H_2(m, U_1', U_2')$ $V' = r_2' H' + (r_1')^{-1} S_A$

그림 2. Zhang-Mao's Scheme

$H_1 : \{0,1\}^* \rightarrow G_1$과

$H_2 : \{0,1\}^* \times G_1 \times G_1 \rightarrow G_1$ 를 선택한다. 또한 KGC는 마스터 비밀키 s를 비밀로 하고 공개 시스템 파라미터

$param = (G_1, G_2, e, q, P, P_{pub}, H_1, H_2)$

를 공개한다.

(2) KeyExtract : 공개 확인자가 ID_U인 유저는 비밀키를 얻기 원할 때, KGC로부터 비밀키 $S_{ID_U} = s Q_{ID_U}$를 받는다. 여기에서 공개 확인자 ID_A를 갖는 서명자 A는 자신의 키 쌍 (S_A, Q_A)를 갖고 지정된 검증자 B는 (S_B, Q_B)를 갖는다.

(3) SigGen : 서명자 A는 메시지 m에 대해서 임의의 두 수 $r_1, r_2 \in Z_q^*$를 선택하여 다음과 같이 서명 $\sigma = (U_1, U_2, V)$를 생성한다.

$U_1 = r_1 Q_B$
$U_2 = r_1 r_2 Q_B$
$H = H_2(m, U_1, U_2)$
$V = r_2 H + r_1^{-1} S_A$

(4) SigVer : 지정된 검증자는 메시지와 서명의 쌍 (m, σ)을 받으면, $H = H_2(m, U_1, U_2)$를 계산한 후, 다음의 등식이 만족되면 서명을 받아들인다.

$e(U_1, V) = e(U_2, H) e(S_B, Q_A)$

(5) SigSimul : 지정된 검증자는 임의로 두 개의 수 $r_1', r_2' \in Z_q^*$를 선택하고 서명 $\sigma' = (U_1', U_2', V')$을 다음과 같이 생성한다.

$U_1' = r_1' Q_A$
$U_2' = r_1' r_2' Q_A$
$H' = H_2(m, U_1', U_2')$
$V' = r_2' H' + (r_1')^{-1} S_A$

Zhang-Mao 기법은 그들이 주장한대로 소스 하이딩 성질을 만족한다. 즉, 메시지 m에 대한 서명 $\sigma = (U_1, U_2, V)$가 주어졌을 때, 공격자에게 서명자와 검증자의 비밀키가 주어지더라도 서명자와 검증자 중에 누가 실제로 서명을 생성했는지는 알 수가 없다. 왜냐하면, 서명자와 검증자의 비밀키를 모두 갖고 있는 공격자가 $\sigma = (U_1, U_2, V)$를 얻었을 때, $H = H_2(m, U_1, U_2)$를 계산한 후에 다음과 같이 서명을 검증할 수 있기 때문이다.

$$\begin{aligned} e(U_1, V) &= e(U_2, H) e(S_B, Q_A) \\ &= e(U_2, H) e(Q_B, S_A) \end{aligned}$$

그러나 Zhang-Mao 기법은 키 노출 공격에 안전하지 않다. 만약 서명자의 비밀키가 어떤 이유에서든 노출된다면, 공격자는 $H = H_2(m, U_1, U_2)$ 를 계산한 후 등식 $e(U_1, V) = e(U_2, H) e(Q_B, S_A)$이 성립하는지를 확인할 수 있다. 따라서 만약에 서명자의 비밀키

Signer A		Designated Verifier B
$k \in Z_q^*$ $U = kP$ $w = k + H_2(e(Q_B, S_A))$ $V = H_3(m, e(Q_B, P_{pub})^w)$	$\xrightarrow{\{U,V\}}$	Verification $h = H_2(e(S_B, Q_A))$ $V \stackrel{?}{=} H_3(m, e(S_B, U)e(hQ_B, P_{pub}))$ Simulation $k' \in Z_q^*$ $U' = k'P$ $w' = k' + H_2(e(Q_B, S_A))$ $V' = H_3(m, e(Q_B, P_{pub})^{w'})$

그림 3. Proposed Scheme

를 갖고 있는 공격자가 아직 지정된 검증자가 서명을 받지 않았다는 것을 확신할 수 있으면 실제 서명자가 생성한 서명이라는 것을 확신할 수 있다.

IV. 키 노출 공격에 안전한 ID-기반의 강한 지정된 검증자 서명 기법

이번 장에서는 새로운 ID 기반의 강한 지정된 검증자 서명을 제안한다. 제안하는 기법은 Kumar et al.의 기법처럼 키 노출 공격에는 안전하나 소스 하이딩 성질은 만족하지 않는다. 하지만 Kumar et al.의 기법보다 적은 계산량을 갖는다.

1. 제안하는 기법

(1) Setup : KGC는 공개 시스템 파라미터 $param = (G_1, G_2, e, q, P, P_{pub}, H_1, H_2, H_3)$를 공개한다. 여기에서 일방향 해쉬 함수 H_1은 Zhang-Mao 기법에서와 같고 H_2와 H_3는 각각 $H_2 : G_2 \rightarrow Z_q^*$이고 $H_3 : \{0,1\}^* \times G_2 \rightarrow G_2$이다.

(2) KeyExtract : 서명자 A는 자신의 키 쌍 (S_A, Q_A)를 갖고 지정된 검증자 B는 (S_B, Q_B)를 갖는다.

(3) SigGen : 서명자 A는 메시지 m에 대해서 임의의 수 $k \in Z_q^*$를 선택하여 다음과 같이 서명 $\sigma = (U, V)$를 생성한다.

$$U = kP$$
$$w = k + H_2(e(Q_B, S_A))$$
$$V = H_3(m, e(Q_B, P_{pub})^w)$$

(4) SigVer : 지정된 검증자는 메시지와 서명의 쌍 (m, σ)을 받으면, $h = H_2(e(S_B, Q_A))$를 계산한 후, 다음의 등식이 만족되면 서명을 받아들인다.

$$V = H_3(m, e(S_B, U)e(hQ_B, P_{pub}))$$

(5) SigSimul : 지정된 검증자는 임의의 수 $k' \in Z_q^*$을 선택하고 서명 $\sigma' = (U', V')$을 다음과 같이 생성한다.

$$U' = k'P$$
$$w' = k' + H_2(e(Q_B, S_A))$$
$$V' = H_3(m, e(Q_B, P_{pub})^{w'})$$

2. 제안한 기법 분석

이번 절에서는 제안하는 기법이 정확성, 위조 불가능성, 키 노출 공격에 대한 안전성을 제공함을 보인다.

2.1 정확성 (Correctness)

제안한 ID-기반의 강한 지정된 검증자 서명은 서명 생성 알고리즘을 제대로 수행했을 때 다음과 같이 서명 검증식에 따라 올바르게 검증될 수 있다.

$$
\begin{aligned}
h &= H_2(e(Q_B, S_A)) \\
V &= H_3(m, e(Q_B, P_{pub})^w) \\
&= H_3(m, e(wQ_B, P_{pub})) \\
&= H_3(m, e((k+h)Q_B, P_{pub})) \\
&= H_3(m, e(kQ_B, P_{pub})e(hQ_B, P_{pub})) \\
&= H_3(m, e(sQ_B.kP)e(hQ_B, P_{pub})) \\
&= H_3(m, e(S_B, U)e(hQ_B, P_{pub}))
\end{aligned}
$$

2.2 위조불가능성 (Unforgeability)

제 3자가 검증식을 통과하는 서명을 위조해내기 위해서는 w (또는 w')을 계산할 수 있어야 한다. 하지만, w (또는 w')는 서명자(또는 지정된 검증자)의 비밀키가 없이는 구할 수가 없다. 따라서 서명자의 비밀키(또는 지정된 검증자의 비밀키)를 가지고 있어야만 서명을 생성할 수 있다.

2.3 키 노출 공격에 대한 안전성 (Security against key-compromise attack)

제안한 기법에서 서명 검증을 위해서는 $e(S_B, U)$를 계산할 수 있어야 한다. 즉, 서명 검증에는 지정된 검증자의 비밀키가 반드시 있어야 한다. 따라서, 제 3자가 서명자의 비밀키를 가지고 있고, 서명자로부터 지정된 검증자로 전달되는 서명을 가로챈다고 하더라도 지정된 검증자의 비밀키가 없이는 서명을 검증할 수 없다.

3. 세 기법이 제공하는 성질과 계산량 비교

[표 1]에서는 지금까지 기술한 세 기법의 성질을 비교하고 있다. 그리고 Kumar et al. 기법과 우리가 제안하는 기법의 계산량 비교는 [표 2]와 같다. 본 논문의 목적이 키 노출 공격에 안전한 ID-기반의 강한 지정된 검증자 서명을 제안하는 것이므로 Kumar et al.의 기법과 우리가 제안하는 기법만을 비교하겠다. 두 기법의 계산량 비교에 있어서 C_p는 pairing 연산에 필요한 계산량, C_m은 G_1 상에서 곱셈 연산에 필요한 계산량, 그리고 C_e는 G_2 상에서 지수 연산에 필요한 계산량이라고 표현하겠다.

우선 서명 길이를 비교하는데, Kumar et al.의 기법의 서명 $\sigma = (U_1, U_2, U_3, V)$에서 U_1, U_2, U_3, V 모두 G_1의 원소이므로 서명 길이는 $4|G_1|$이고, 우리가 제안하는 기법의 서명은 $\sigma = (U, V)$이므로 서명 길이는 $|G_1| + |G_2|$이다.

다음으로 서명 생성과 서명 검증에 드는 계산량을 비교해 보겠다. 서명 생성에 있어서는, Kumar et al.의 기법에서는 6번의 G_1 상에서의 곱셈 연산이 필요하고 한 번의 pairing 연산이 필요하다. 이에 반해 우리가 제안하는 기법에서는 $e(Q_B, S_A)$와 $e(Q_B, P_{pub})$는 미리 계산되어질 수 있으므로 한 번의 G_1 상에서의 곱셈 연산과 한 번의 G_2 상에서 지수 연산이 필요하다. 서명 검증에 있어서는, Kumar et al.의 기법에서는 3번의 pairing 연산이 필요하고 우리가 제안한 기법에서 h와 $e(hQ_B, P_{pub})$는 미리 계산되어질 수 있으므로 한 번의 pairing 연산이 필요하다. 따라서 우리가 제안한 기법이 Kumar et al.이 제안한 기법보다 효율적임을 알 수 있다.

표 1. 세 기법의 성질 비교

기법	위조불가능성	소스 하이딩	키 노출 공격에 대한 안전성
Kumar et al	O	X	O
Zhang-Mao	O	O	X
제안하는 법	O	X	O

표 2. Kumar et al. 기법과 제안하는 기법의 계산량 비교

기법	서명 길이	서명 생성 계산량	서명 검증 계산량
Kumar et al	$4\|G_1\|$	$6C_m + 1C_p$	$3C_p$
제안하는 기법	$\|G_1\| + \|G_2\|$	$1C_m + 1C_e$	$1C_p$

V. 결 론

본 논문에서는 최근에 발표된 Zhang-Mao의 강한 지정된 검증자 서명이 키 노출 공격에 안전하지 않음을 보이고, 소스 하이딩 성질을 만족하는 강한 지정된 검증자 서명이 키 노출 공격에는 안전하지 않음을 보였다. 강한 지정된 검증자 서명의 중요한 성질은 서명자

와 검증자 중에 누가 서명을 생성했는지 모르도록 하는 것이기 때문에 비밀키 노출 공격에서도 서명자의 신분이 노출되지 않는 것이 중요하다. 따라서 Zhang-Mao 기법보다 효율적이면서 키 노출 공격에 안전한 ID 기반의 강한 지정된 검증자 서명을 제안하였다.

참 고 문 헌

[1] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated Verifier Proofs and Their Applications," Eurocrypt'96, LNCS Vol.1070, pp.145-154, 1996.

[2] S. Saeednia, S. Kremer, and O. Markovitch, "An Efficient Strong Designated Verifier Signature Scheme," ICISC'03, LNCS Vol.2971, pp.40-54, 2003.

[3] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," Crypto'01, LNCS Vol.2139, pp.213-229, 2001.

[4] K. Kumar, G. Shailaja and A. Saxena, "Identity Based Strong Designated Verifier Signature Scheme," Informatica, Vol.18, No.2, pp.239-252, 2007.

[5] J. Zhang and J. Mao, "A Novel ID-based Designated Verifier Signature Scheme," Information Sciences, Vol.178, Issue3, pp.766-773, 2008.

[6] Ji-Seon Lee and Jik Hyun Chang, "Comment on Saeednia et al.'s Strong Designated Verifier Signature Scheme," Computer Standards & Interfaces, Vol.31, Issue1, pp.258-260, 2009.

저 자 소 개

이 지 선(Ji-Seon Lee) 정회원

- 1991년 2월 : 서강대학교 전산학과(공학사)
- 1998년 8월 : 서강대학교 컴퓨터공학과(공학석사)
- 2008년 2월 : 서강대학교 컴퓨터공학과(공학박사)
- 2008년 3월 ~ 현재 : 고려대학교 BK21 유비쿼터스 정보보호 사업단 연구교수

<관심분야> : 암호학, 네트워크 보안, 콘텐츠 보안

장 직 현 (Jik-Hyun Chang) 정회원

- 1972년 2월 : 서울대학교 수학과(이학사)
- 1977년 8월 : 서울대학교 수학과(이학석사)
- 1986년 8월 : Univ. of Minnesota 전산학과(공학박사)
- 1986년 9월 ~ 현재 : 서강대학교 컴퓨터공학과 교수

<관심분야> : 알고리즘 설계와 분석, 암호 알고리즘

이 동 훈(Dong-Hoon Lee) 정회원

- 1983년 8월 : 고려대학교 경제학과 학사
- 1987년 12월 : Univ. of Oklahoma 전산학과 (공학석사)
- 1992년 5월 : Univ. of Oklahoma 전산학과 (공학박사)
- 1993년 3월 ~ 2001년 2월 : 고려대학교 전산학과 교수
- 2001년 3월 ~ 현재 : 고려대학교 정보경영공학부 교수

<관심분야> : 암호이론, 암호프로토콜, RFID/USN보안, 프라이버시 보호 기술