JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 28 No. 7, pp. 77-84, July 2023
https://doi.org/10.9708/jksci.2023.28.07.077

Efficient and Secure Signature Scheme applicable to Secure multi-party Computation

Myoungin Jeong*

*Assistant Professor, Dept. of Mathematics, Korea Military Academy, Seoul, Korea

[Abstract]

This research originated from the need to enhance the security of secure multiparty computation by ensuring that participants involved in multiparty computations provide truthful inputs that have not been manipulated. While malicious participants can be involved, which goes beyond the traditional security models, malicious behaviors through input manipulation often occur in real-world scenarios, leading to privacy infringements or situations where the accuracy of multiparty computation results cannot be guaranteed. Therefore, in this study, we propose a signature scheme applicable to secure multiparty technologies, combining it with secret sharing to strengthen the accuracy of inputs using authentication techniques. We also investigate methods to enhance the efficiency of authentication through the use of batch authentication techniques. To this end, a scheme capable of input certification was designed by applying a commitment scheme and zero-knowledge proof of knowledge to the CL signature scheme, which is a lightweight signature scheme, and batch verification was applied to improve efficiency during authentication.

▸ **Key words**: Signature scheme, Batch verification, Zero Knowledge Proof of Knowledge, Commitment scheme, SMC

[요 약]

본 연구는 다자간 계산에 참여하는 참여자가 조작되지 않은 진실한 입력을 입력하도록 보장하는 기능을 통해 안전한 다자간 프로토콜의 보안을 강화해야 할 필요성으로부터 시작되었다. 이것은 악의적인 참가자가 참여하는 경우이지만 전통적인 보안 모델을 벗어나는 것으로, 실제 상황에서 입력 조작을 통한 악의적인 행동은 종종 일어나며 이를 통해 프라이버시를 침해받거나 다자간 계산 결과의 정확성이 보장받을 수 없는 경우가 발생한다. 따라서 본 연구에서는 인증 기술을 이용하여 입력의 정확성을 강화하기 위해 비밀 공유를 기반으로 하는 안전한 다자간 기술에 결합하여 적용이 가능한 서명 체계를 제안하며 배치인증 기술을 이용하여 인증의 효율성을 강화하기 위한 방법에 대해 연구한다. 이를 위해 경량화된 서명 체계인 CL signature scheme에 commitment scheme과 영지식증명을 적용하여 입력 인증이 가능한 스킴을 설계하였으며, 인증 간에 효율성을 향상시키기 위해 배치인증을 적용하였다.

▸ **주제어**: 서명체계, 배치인증, 영지식증명, Commitment scheme, 안전한 다자간 계산

• First Author: Myoungin Jeong, Corresponding Author: Myoungin Jeong
*Myoungin Jeong (mangjj@kma.ac.kr), Dept. of Mathematics, Korea Military Academy
• Received: 2023. 06. 27, Revised: 2023. 07. 17, Accepted: 2023. 07. 17.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

안전한 다자간 계산(Secure Multi-party Computation, SMC)은 1980년대에 Andrew Yao[1]에 의해 2자 간 계산으로 기반이 마련되었고, 그 후 Goldreich[2], Micali 등에 의해 다자간 연산으로 확장되었다. 최근 클라우드 컴퓨팅 환경이 대중화되고 개인 정보 보호의 중요성이 강조되면서 다수의 참여자 간에 정보를 공개하지 않으면서 필요한 정보들만 공유하게 할 수 있게 하는 안전한 다자간 계산에 관한 연구가 활발히 진행되고 있다. 예를 들면, 금융기관, 병원, 신용정보회사 등 민감한 정보를 취급하는 기관에서 타 기관과 협업을 할 경우에 데이터 전체를 공유하지 않고 필요한 정보만을 공유하기 위해 안전한 다자간 계산을 이용할 수 있다. 또한, 강화된 보안의 수단으로 생체정보(지문, 홍채, 혈관, DNA 등)를 사용하는 장치들의 개발이 가속화됨에 따라 프라이버시를 유지하면서 작업을 수행하는 방법에 관한 관심이 더욱 높아지고 있다, 이러한 생체정보들은 한번 유출되어 버리면 초기화하거나 다시 설정할 수 없으므로 그에 대한 보안은 매우 중요하게 취급되어야 하며, 안전한 다자간 계산은 생체정보의 보안을 유지하기 위한 하나의 해결책이 될 수 있다. 그 밖에도 클라우드 컴퓨팅 환경을 이용하여 여러 기관에서 신용정보, 의료정보, 금융정보 등을 가공할 때 원본 데이터를 모두 노출하지 않으면서 원하는 데이터만 추출할 수 있도록 할 수 있게 하는데 안전한 다자간 계산이 핵심적인 역할을 한다. 이렇게 여러 기관의 민감한 정보를 다룸에 있어서 다자간 프라이버시 보호가 가능한 안전한 계산 방법에 관한 연구가 선행되어야 하고, 더불어 그 결과로 제공되는 산출물이 조작되거나 변형되지 않은 원래 계산하고자 했던 데이터와 같은 데이터라는 확신을 참여자에게 줄 수 있어야 한다. 이것은 악의적인 참여자가 있는 경우의 전통적인 보안 모델을 벗어난 것으로 이러한 경우 심각한 보안 위험이 될 수 있으며 결과의 정확성에 대해 보장받을 수 없다. 본 논문에서는 SMC 프로토콜에 적용할 수 있는 CL 서명 체계와 영지식증명(Zero Knowledge Proof of Knowledge, ZKPK), Commitment scheme 등을 통해 안전한 다자간 계산 과정에서 데이터의 변조 방지할 수 있게 하는 입력 인증 프로토콜을 제안하고자 한다.

II. Related Work

안전한 다자간 계산이 처음 제안된 이후 이와 관련된 많은 연구가 이루어졌다. 최근의 기술의 발전으로 SMC 계산의 오버헤드 문제가 해결되어 실생활의 다양한 경우에 사용되고 있다[3, 4, 5].

입력 인증에 관한 연구는 2004년 Halpern and Teague[6], 2013년 Wallrabenstein and Clifton[7]이 게임이론을 이용하여 합리적인 참여자가 자신의 입력값을 진실하게 입력하도록 하는 방법에 관해 연구하였다. 또한 2003년에는 Camenisch et al.에 의해 anonymous credential[8]과 set operation[9, 10] 등 몇 가지 특정한 SMC 응용 분야에서의 연구가 이루어졌고, 최근에는 일반적인 함수들에 연구가 진행 중이다[11~13]. 이러한 연구의 대부분은 Garbled Circuit(GC)에 기반을 둔 2자 간의 안전한 계산에 초점을 맞추고 있다. 그러나 GC는 서명 체계나 인증 기법과 자연스럽게 결합할 수 없으므로 GC에만 국한되지 않는 확장된 연구가 필요하다. 따라서 본 논문에서는 일반적인 비밀 공유 기법을 기반으로 한 다자간 계산이라는 설정에서의 입력 인증 문제에 관한 연구를 진행하였다. 비밀 공유 기법과 잘 알려진 서명 체계들은 모두 대수적인 구조 위에서 고안되었기 때문에 비밀 공유 기반의 SMC에서 입력 인증에 서명 체계를 사용하면 자연스럽게 결합할 수 있다는 장점이 있다. 그러나 일반적으로 서명을 사용하는 것과는 다르게 안전성 측면에서 추가로 고려해야 하는 사항이 있다. 그것은 서명을 검증할 때 verifier에게 서명이 된 메시지에 대한 정보를 노출해서는 안 된다는 것이다. 또한 일반적으로 SMC를 사용하는 데이터는 용량이 매우 큰 데이터이고, 서명은 공개키를 기반으로 하고 인증 간의 메시지에 대한 비밀을 유지하기 위해 ZKPK 방법을 주로 사용하기 때문에 서명 검증 단계에서 계산량이 상당하여 속도 향상을 위한 방안을 마련해야 한다.

III. Preliminaries

1. Standard form of SMC

SMC는 일반적으로 여러 명($k \geq 1$)의 참여자가 함수 f에 각각 비밀 입력 in_1, $\cdots in_k$을 입력하며, 이 함수의 계산에는 $m(\geq 2)$개의 computation party가 참여하여 $s(\geq 1)$개의 결괏값을 생성하여 미리 약속된 참여자에게만 결과를 공개한다. 각 참여자의 입력값은 다른 참여자들에게는 비밀로 유지되어야 하고, 미리 합의된 참여자가 아닌 경우 결과를 알 수 없다. 전통적인 보안 모델은 악의적인 참여자들에 따라 semi-honest 모델과 malicious 모델로 정의한다. semi-honest 모델은 악의를 가진, 정직하지 않은 참가자들도 규정된 계산 과정은 올바르게 따른다. 그

러나 malicious 모델에서 악의적인 참여자는 다른 참여자의 입력에 대한 정보를 무단으로 얻으려 시도하거나 의도한 정보를 알아내기 위해 정해진 계산 과정을 임의로 벗어날 수 있다. semi-honest 모델과 malicious 모델 모두 입력된 입력값에 대한 함수 출력의 정확성은 보장된다. 그러나 입력된 입력값이 변조되지 않은 참가자들이 입력한 입력이 맞는지는 확인할 수 없다. 따라서 악의적인 참가자가 입력을 조작하여 프로토콜의 안전성이나 정확성을 해치려고 시도할 수 있다. 예를 들면, 악의적인 참가자는 자신의 입력을 조작하여 모든 출력 수신자가 잘못된 정보를 수신하게 하고 자신은 본인이 의도한 오류를 보완하여 올바른 결괏값을 알아낼 수 있다. 또는, 악의적인 참가자는 자신의 입력값을 조작하지 않았을 경우 정상적인 과정을 통해 함수를 계산할 때 알 수 있었던 것 이상으로 다른 사람의 데이터에 관한 정보를 알아낼 수 있게 자신의 입력을 수정할 수 있다[6]. 이러한 공격은 일반적인 SMC 보안 모델의 범위를 벗어나며 일반적인 SMC 프로토콜을 통해서는 이러한 위험성을 방지할 수 없다.

2. Signature Scheme

서명 체계는 키를 생성하는 키 생성(Key generation), 서명을 생성하는 서명(Sign), 서명의 검증(Verification) 이렇게 세 개의 알고리즘으로 구성된다.

Definition 1. (Signature Scheme)

KeyGen: 확률적 다항시간(probabilistic polynomial-time, PPT) 알고리즘으로 공개키, 개인 키 쌍 (pk, sk)을 생성한다.

Sign: PPT 알고리즘으로 개인 키 sk와 메시지 m을 입력으로 하여 서명 σ을 생성한다.

Verify: 결정적 다항시간(deterministic polynomial-time) 알고리즘으로 공개키 pk와 메시지 m, 서명 σ을 입력으로 하여 한 비트를 출력한다.

$\Pi = (\mathrm{KeyGen}, \mathrm{Sign}, \mathrm{Verify})$을 서명 체계라 한다.

3. Zero Knowledge Proof of Knowledge

영지식증명(Zero Knowledge Proof of Knowledge)은 prover와 verifier 간의 양자 간 상호작용 프로토콜을 말한다. 이 과정에서 prover는 verifier에게 자신이 사용한 어떠한 정보도 공개하지 않으면서 자신이 진실한 진술(statement)을 하고 있다는 것을 확신시킬 수 있다. 특정한 변수(variables)와 진술을 사용한 ZKPK의 표기는 $PK\{(\mathrm{variables}): \mathrm{statement}\}$로 하기로 하며, 변수는 prover만이 알고 있는 정보로 verifier에게 공개하지 않으며 진술은 prover와 verifier 모두에게 공개된다. 이 프로토콜이 성공하면 verifier는 변수에 대한 정보 없이도 prover가 변수를 알고 있다는 사실을 받아들인다.

4. Commitment Scheme

commitment scheme은 메시지 m에 대한 정보를 공개하지 않으면서 그 메시지 m에 커밋이 이루어지며, m에 대한 커밋이 주어지면 m 이외의 값으로 공개할 수 없다. 즉, 값 m이 커밋되면 사용자가 공개할 때까지 이를 변경할 수 없으며 비공개로 유지된다. commitment scheme이 갖는 이러한 속성을 숨기기(hiding)와 묶기(binding)라고 한다. commitment scheme은 Commit 알고리즘과 Open 알고리즘으로 구성되며, Commit 알고리즘을 이용하여 메시지 m에 대한 커밋이 이루어지며 Open 알고리즘으로 이를 공개할 수 있다. Commit 알고리즘은 랜덤 넘버(r)를 이용한 랜덤화된 알고리즘을 사용할 것이며 $com(m, r)$로 표시한다. 본 논문에서는 이산로그에 기반한 잘 알려진 Pedersen commitment scheme[14]을 사용한다. prime order q를 가진 그룹 G와 두 개의 generator g, h를 이용하며 메시지 $m \in \mathbb{Z}_q$에 커밋하기 위해서 랜덤 넘버 $r \in \mathbb{Z}_q$를 선택하며 $com(m, r) = g^m h^r$로 정의한다. 이 commitment를 공개(open)하기 위해서는 r을 밝혀야 한다.

5. CL Signature Scheme

서명 체계를 설명하는 데 필요한 bilinear map을 먼저 정의하면 다음과 같다.

Definition 2. (Bilinear map) 아래의 조건들을 만족하는 일방향함수 $e : G \times G \rightarrow \mathbf{G}$를 bilinear map이라 한다.
- Efficient: G와 $\mathbf{G}$는 같은 prime order q를 갖는 그룹이며 e를 계산하는 효율적인 알고리즘이 존재한다.
- Bilinear: 모든 $g, h \in G$와 $a, b \in \mathbb{Z}_q$에 대해 $e(g^a, h^b) = e(g, h)^{ab}$이다.
- Non-degenerate: g가 G를 생성하면 $e(g, g)$가 $\mathbf{G}$를 생성한다.

준비단계에서 prime order q를 가지며 bilinear map e이 정의되는 그룹 $G = \langle g \rangle$와 마찬가지로 prime order q를 가지며 $e(g, g)$에 의해 생성되는 그룹 $\mathbf{G}$가 결정된다고 가정한다. 따라서 준비단계에서는

(q, G, G, g, e)이 결정된다.

Camenisch-Lysyanskaya signature Scheme A(CL signature scheme A)[15, 16]를 사용하기 위해 다음과 같이 정의한다.

Key generation: 임의로 $x, y \in \mathbb{Z}_q$를 선택하여 $X = g^x$, $Y = g^y$을 계산한다. 개인 키는 $sk = (x, y)$이며 공개키는 $pk = (q, g, \mathsf{G}, g, e, X, Y)$이다.

Signing: 입력된 메시지 $m \in \mathbb{Z}_q$, 개인 키 $sk = (x, y)$, 공개키 $pk = (q, g, \mathsf{G}, g, e, X, Y)$에 대해 랜덤 $a \in G$를 골라 서명 $\sigma = (a,b,c) = (a, a^y, a^{x+mxy})$를 생성한다.

Verification: 입력 메시지 m, 서명 $\sigma = (a, b, c)$, 공개키 $pk = (q, g, \mathsf{G}, g, e, X, Y)$에 대해 $e(a, Y) = e(g, b)$와 $e(X, a) \bullet e(X, b)^m = e(g, c)$가 성립하는지 확인한다. 두 개의 식이 모두 성립하면 결과는 1이며, 그렇지 않을 경우 결과는 0이다.

Proof of signature: prover와 verifier는 모두 공개키를 알고 있으며 prover는 $m \in \mathbb{Z}_q$와 그것을 이용하여 생성한 서명 $\sigma = (a,b,c) = (a, a^y, a^{x+mxy})$을 가지고 있다.

1. Prover는 두 개의 랜덤넘버 $r', r'' \in \mathbb{Z}_q$를 선택하여 감춰진(blinded) 서명 $\tilde{\sigma} = (a^{r''}, b^{r''}, c^{r''r'}) = \left(\tilde{a}, \tilde{a}^y, \left(\tilde{a}^{x+mxy}\right)^{r'}\right) = (\tilde{a}, \tilde{b}, \hat{c})$를 계산하여 verifier에게 보낸다.

2. $\mathbf{v}_x = e(X, \tilde{a}), \mathbf{v}_{xy} = e(X, \tilde{b}), \mathbf{v}_s = e(g, \hat{c})$라 하자. prover와 verifier는 $PK\{(\mu, \rho) : \mathbf{v}_x^{-1} = \mathbf{v}_{xy}^{\mu} \mathbf{v}_x^{\rho}\}$ 영지식증명을 시행한다.

3. verifer는 위의 영지식증명이 성공하고 $e(\tilde{a}, Y) = e(g, \tilde{b})$이면 서명을 수용한다.

6. Batch Verification

Batch Verification은 다른 메시지들에 한 명 또는 각각의 서명자들이 시행한 서명들을 한 번에 인증하는 방법이다. 본 논문에서는 생체정보 등 한 명이 대량의 메시지에 서명한 경우(같은 키 사용)에 대해 다루었다. 이것은 각각의 서명들을 하나씩 인증하는 것보다 효율적이다.

Definition 3. (Batch verification of signatures) 서명 체계 $\Pi = (\mathrm{KeyGen}, \mathrm{Sign}, \mathrm{Verify})$와 보안 파라미터 κ에 대해 n명의 서명자 $P_1, P_2, \cdots, P_n$이 $\mathrm{KeyGen}(1^{\kappa})$ 알고리즘과 공개키 $PK = \{pk_1, \cdots, pk_n\}$을 이용하여 만든 키 쌍들을 $(pk_1, sk_1), \cdots, (pk_n, sk_n)$라 하자. batch verification 알고리즘 Batch는 (pk_i, m_i, σ_i)들을 입력으로 하고 한 비트를 출력하는 PPT 알고리즘이며 다음 성질을 만족한다.

- 모든 $i \in [1, n]$에 대해 $pk_i \in PK$이고 $\mathrm{Verify}(pk_i, m_i, \sigma_i) = 1$이면 $\mathrm{Batch}((pk_1, m_1, \sigma_1), \cdots, (pk_n, m_n, \sigma_n)) = 1$ 이다.
- 모든 $i \in [1, n]$에 대해 $pk_i \in PK$이고 적어도 하나의 $i \in [1, n]$에 대해 $\mathrm{Verify}(pk_i, m_i, \sigma_i) = 0$이면 최대확률 $2^{-\kappa}$으로 $\mathrm{Batch}((pk_1, m_1, \sigma_1), \cdots, (pk_n, m_n, \sigma_n)) = 1$ 이다.

Batch verification을 사용하면 단 하나의 서명 인증이 실패하더라도 verifier는 적어도 $1 - 2^{-\kappa}$의 확률로 이를 발견할 수 있으며, Batch verification은 서명들을 각각 인증하는 것보다 속도가 빠르다.

IV. The Proposed Protocol

1. Constructions based on CL Signature Scheme A

SMC에 활용할 수 있는 입력 인증을 활용한 서명 체계 중 하나는 Camenisch와 Lysyanskaya가 제안한 CL 서명 체계가 있다[15, 16]. CL 서명은 익명성을 필요로 하는 응용체계를 위해 디자인된 서명 체계로, 같은 서명을 여러 번 사용했을 때 메시지 프라이버시와 불연계성(unlinkability)을 달성할 수 있는 서명 체계이다. 다음과 같이 CL 서명 체계에 기반하여 프라이버시를 달성할 수 있도록 프로토콜을 디자인한다.

프라이버시를 달성할 수 있는 서명 체계 $\Pi = (\mathrm{KeyGen}, \mathrm{Sign}, \mathrm{PrivVerify})$를 다음과 같이 정의한다. **KeyGen, Sign** 알고리즘은 **KeyGen** 단계에서 $h = g^u, u \in \mathbb{Z}_q$를 추가로 계산하여 공개키에 추가하는 것을 제외하면 나머지는 그대로 사용하며(즉, $pk = (q, G, \mathrm{G}, g, h, e, X, Y)$), 인증 알고리즘 **PrivVerify**는 다음과 같이 수정하여 정의한다.

PrivVerify: prover는 서명 $\sigma = (a, b, c)$와 비공개인 메시지 $m \in \mathbb{Z}_q$를 알고, prover와 verifier 모두 공개키 pk를 알고 있다. prover는 랜덤넘버 $r \in \mathbb{Z}_q$를 이용하여 commitment인 $x_m = com(m, r) = g^m h^r$를 계산하여

x_m을 verifier에게 보낸다. 나머지 부분은 앞에서 살펴본 CL signature scheme의 **Proof of signature** 단계와 2번의 ZKPK가 $PK\{(\mu, \rho, \gamma) : x_m = g^{\mu}h^{\gamma} \wedge \mathbf{v}_x^{-1} = \mathbf{v}_{xy}^{\mu}\mathbf{v}_s^{\rho}\}$ 로 수정되는 것을 제외하면 동일하다.

제안된 서명 체계 $\Pi = (\mathrm{KeyGen}, \mathrm{Sign}, \mathrm{PrivVerify})$는 기존의 CL scheme과 키 생성, 서명 단계까지는 같으며 인증단계에서의 영지식증명만 기존의 서명보다 확장되었다. 따라서 CL scheme의 위조 불가능성(unforgeability)을 동일하게 만족한다.

기존 알고리즘과 수정된 알고리즘에 소요되는 연산을 자세히 비교하기 위해 PrivVerify의 ZKPK에서 사용된 정확한 연산(지수승과 페어링 함수 계산)을 살펴보면 다음과 같다. ZKPK에서 prover는 랜덤넘버 $v_1, v_2, v_3 \in \mathbb{Z}_q$를 선택하고 $T_1 = g^{v_1}h^{v_3}$, $T_2 = \mathbf{v}_{xy}^{v_1}\mathbf{v}_s^{v_2}$를 계산하여 T_1, T_2를 verifier에게 전송한다. verifier는 랜덤으로 $e \in \mathbb{Z}_q$를 선택하여 그것을 prover에게 보낸다. prover는 $r_1 = v_1 + em \bmod q$, $r_2 = v_2 + er' \bmod q$와 $r_3 = v_3 + er \bmod q$를 계산하여 verifier에게 전송한다. 마지막으로 verifier는 $g^{r_1}h^{r_3} = T_1 x_m^e$과 $\mathbf{v}_{xy}^{r_1}\mathbf{v}_s^{r_2} = T_2\mathbf{v}_x^{-e}$가 성립하면 ZKPK는 성공(statement가 진실)한다.

2. Modified CL Scheme A

다음은 CL Scheme A 좀 더 단순화하여 SMC에 활용할 때 인증단계에서의 효율성을 높일 수 있도록 수정하였다. CL scheme A에서와 같이 **KeyGen** 단계에서 공개키에 h를 추가하고 **KeyGen, Sign** 알고리즘은 같으며 **PrivVerify** 알고리즘은 아래와 같이 수정한다.

PrivVerify: prover는 비공개인 메시지 $m \in \mathbb{Z}_q$와 해당 메시지의 서명 $\sigma = (a, b, c) = (a, a^y, a^{x+mxy})$를 알고, prover와 verifier 모두 공개키 $pk = (q, G, \mathrm{G}, g, h, e, X, Y)$ 를 알고 있다.

1. prover는 랜덤넘버 $r \in \mathbb{Z}_q$를 선택하여 commitment인 $x_m = com(m, r) = g^m h^r$를 계산하여 x_m을 verifier에게 보낸다.

2. prover는 랜덤넘버 $r' \in \mathbb{Z}_q$를 선택하여 랜덤화한 서명 $\tilde{\sigma} := (a, b, c^{r'}) = (a, b, \tilde{c})$을 계산하여 verifier에게 전송한다.

3. $\mathbf{v}_x = e(X, a)$, $\mathbf{v}_{xy} = e(X, b)$, $\mathbf{v}_s = e(g, \tilde{c})$라고 하고 verifier는 영지식증명 $PK\{(\mu, \rho, \gamma) : x_m = g^{\mu}h^{\gamma} \wedge \mathbf{v}_x^{-1} = \mathbf{v}_{xy}^{\mu}\mathbf{v}_s^{\rho}\}$을 시행한다.

4. verifier가 3번의 영지식증명에 성공하고 $e(a, Y) = e(g, b)$이면 **PrivVerify**의 결과는 1, 아닌 경우 결과는 0이다.

수정된 CL scheme A 서명에서는 인증단계에서 원래 CL scheme A에 포함되어 있던 랜덤 넘버 r', r''을 이용하여 감춰진(blinded) 서명을 생성하는 랜덤화 단계를 단순화하였다. 이 경우 같은 서명을 여러 번 사용할 경우 불연계성(unlinkability) 달성이 힘들어진다는 것을 의미한다. 그러나 이것은 서명 체계의 위조 불가 속성(unforgeablility)에는 영향을 미치지 않으므로 우리가 가정한 SMC 상황, 즉 n명의 참여자의 n개의 서명을 인증하는 경우의 안전성에 위협이 되지 않는다.

3. Batch Verification of Modified CL Scheme A

다음은 n개의 서명을 인증하기 위한 batch verification에 관해 설명한다. 본 논문에서는 알려진 다양한 batch verification 방법 중 verifier가 보안 파라미터 l_b(유효하지 않은 서명이 포함되었을 때 batch verification을 통과할 확률이 최대 2^{-l_b}, $l_b = 60$ 또는 80으로 설정)를 사용한 small exponent test[17] 버전을 사용하여 한 명의 서명자가 같은 키를 사용하여 생성한 여러 개의 서명을 검증하는 상황을 특정하여 연구를 시행하였다. batch verification을 시행하는 **Batch** 알고리즘을 다음과 같이 정의하였다.

Batch: prover는 n개의 메시지 $m_i \in \mathbb{Z}_q$ $(i = 1, \cdots, n)$에 대한 서명 $\sigma_i = (a_i, b_i, c_i)$를 알고 있으며 prover와 verifier는 모두 공개키 $pk = (q, G, \mathrm{G}, g, h, e, X, Y)$를 알고 있다.

1. prover는 랜덤넘버 $r_i \in \mathbb{Z}_q$ commitment $x_{m_i} = com(m_i, r_i) = g^{m_i}h^{r_i}$를 생성하여 verifier에게 보낸다.

2. prover는 랜덤넘버 $r' \in \mathbb{Z}_q$를 선택하여 감춰진(blinded) 서명 $\tilde{\sigma}_i = (a_i, b_i, c_i^{r'_i}) = (a_i, b_i, \tilde{c}_i)$,

$(i=1, \cdots, n)$들을 계산하여 verifier에게 보낸다.

3. verifier는 랜덤넘버 $\delta_1, \cdots, \delta_n \in \{0,1\}^{l_b}$를 선택하여 prover에게 보낸다.

4. 각 참여자는 $\hat{\mathbf{v}}_x = e\left(X, \prod_{i=1}^{n} a_i^{\delta_i}\right)$, $\hat{\mathbf{v}}_{xy_i} = e\left(X, b_i^{\delta_i}\right)$ $\hat{\mathbf{v}}_{s_i} = e\left(g, \tilde{c}_i^{\delta_i}\right)$, $(i=1, \cdots, n)$를 계산하여 영지식증명 $PK\left\{(\mu_1, \cdots, \mu_n, \rho_1, \cdots, \rho_n, \gamma_1, \cdots, \gamma_n): \hat{\mathbf{v}}_x^{-1} = \prod_{i=1}^{n} \hat{\mathbf{v}}_{xy_i}^{\mu_i} \hat{\mathbf{v}}_{s_i}^{\rho_i} \wedge x_{m_1} = g^{\mu_1} h^{\gamma_1} \wedge x_{m_n} = g^{\mu_n} h^{\gamma_n}\right\}$를 시행한다.

5. 4번의 영지식증명을 통과하고 $e\left(\prod_{i=1}^{n} a_i^{\delta_i}, Y\right) = e\left(g, \prod_{i=1}^{n} b_i^{\delta_i}\right)$이면 verifier는 결과로 1을, 그렇지 않으면 0을 출력한다.

위에서 정의한 Batch 알고리즘은 수정된 CL scheme A의 batch verifier이다. 그리고 4번의 영지식증명 단계를 설명하면 다음과 같다. prover는 랜덤넘버 $v_i, v'_i, v''_i \in \mathbb{Z}_q$를 선택하고 $T_i = g^{v_i} h^{v''_i}$, $(i=1, \cdots, n)$와 $T = \prod_{i=1}^{n} \left(\mathbf{v}_{xy_i}^{v_i}, \mathbf{v}_{s_i}^{v'_i}\right)$를 계산하여 T_i들과 T를 verifier에게 전송한다. verifier는 랜덤으로 $e \in \mathbb{Z}_q$를 선택하여 그것을 prover에게 보낸다. 챌린지 $e \in \mathbb{Z}_q$를 verifier로부터 수신한 후 prover는 $u_i = v_i + em_i \bmod q$, $u'_i = v'_i + er'_i \bmod q$와 $u''_i = v''_i + er_i \bmod q$를 계산하여 verifier에게 전송한다. verifier는 $g^{u_i} h^{u''_i} = T_i x_{m_i}^{e} (i=1, \cdots, n)$와 $\prod_{i=1}^{n} \left(\mathbf{v}_{xy_i}^{u_i} \mathbf{v}_{s_i}^{u'_i}\right) = T\mathbf{v}_x^{-e}$가 성립하면 ZKPK는 성공한 것이다.

4. Comparison of computation Cost

SMC에 입력 인증 프로토콜을 사용하여 비용을 분석하기 위해서는 서명 인증과 이를 SMC 프로토콜에 접목하는데 걸리는 시간을 평가해야 한다. 따라서 서명 체계에서 서명이 한 번 이루어졌을 때 인증에 필요한 계산량을 정확하게 평가해야 한다. 위에서 살펴본 CL scheme A와 수정된 CL scheme A, 그리고 batch verification에 필요한 연산량을 분석하면 다음과 같다.

우선 CL scheme A를 안전한 다자간 계산에 사용하기 위해서는 참여자들이 n개의 서명을 인증하는 데에 서명 랜덤화를 위해 $3n$번의 모듈로 지수 연산(mod exp)과 commitment 생성을 위한 $2n$번의 mod exp, ZKPK를 위해 $10n$번의 mod exp와 $5n$번의 페어링 연산이 필요하다. 따라서 CL scheme A를 사용하여 n개의 서명을 인증하는 데에 $15n$ 번의 mod exp와 $5n$번의 페어링 연산이 필요하다.

인증단계에서 랜덤화를 제외한 수정된 CL scheme A의 계산을 위해서는 랜덤화에 필요한 $2n$번의 mod exp 연산이 감소했다. 따라서 n개의 서명을 인증하기 위해서는 $13n$번의 mod exp와 $5n$번의 페어링 연산이 필요하다.

마지막으로 SMC에서 n개의 인증된 입력을 수정된 CL scheme A의 batch verification에 적용한 경우에는 서명 랜덤화에 n번의 mod exp, commitment 생성에 $2n$번의 mod exp, 영지식증명에 $12n+1$번의 mod exp와 $2n+3$번의 페어링이 필요하다. 따라서 총 $15n+1$번의 mod exp와 $2n+3$의 페어링 연산이 필요하다. 이것은 n개의 메시지에 대한 서명을 각각 인증하는 것에 비해 계산량이 가장 큰 페어링 연산을 상당히 감소시켰음을 알 수 있다.

수정된 CL scheme A에 batch verification을 적용하는 구조는 각각의 메시지에 독립된 commitment를 생성하도록 설계되어 있다. SMC의 구조에 따라 단일 commitment를 허용하는 경우가 있으므로 n개의 메시지를 이용하여 하나의 commitment를 생성할 때의 영지식증명의 계산량이 어떻게 되는지 알아보자. n개의 메시지에 하나의 commitment를 생성하는 것은 $com(m_1, \cdots, m_n, r) = g_1^{m_1} \cdots g_n^{m_n} h^r$로 정의할 수 있다. 이 경우 영지식증명 과정에서 $3n-3$번의 mod exp 계산이 감소한다. 따라서 서명 랜덤화에 n번의 mod exp, commitment 생성에 $n+1$번의 mod exp, 영지식증명에 $9n+4$번의 mod exp와 $2n+3$번의 페어링 연산이 필요하다.

본 논문에서 제안한 랜덤화를 제거하여 효율성이 향상된 수정된 CL scheme A의 경우 commitment가 사전에 계산되어 저장되어 있다고 가정하면 메시지 당 하나의 서명을 한 경우, n개의 메시지에 n개의 commitment를 생성하여 batch verification을 사용할 경우, n개의 메시지에 단일 commitment를 생성할 경우의 인증에 필요한 계산은 다음 표와 같다.

Table 1. Performance of private verification for a single signature and a batch of size n. It is assumed that commitments are stored pre-computed.

case	Modified CL scheme A
single signature	$11n$ mod exp and $5n$ pairings
Batch with n commitments	$13n+1$ mod exp and $2n+3$ pairings
Batch with 1 commitment	$9n+4$ mod exp and $2n+3$ pairings

따라서 안전한 다자간 계산 과정에서 서명과 인증을 이용하면 각각의 메시지에 서명하는 경우보다는 commitment를 이용하여 batch verification을 하는 경우의 계산량이 상당히 감소하며(가장 연산량이 큰 페어링 계산이 $3n$번 감소), n 개의 commitment를 생성하는 경우보다는 하나의 commitment를 생성하는 SMC 구조를 이용하는 경우의 계산량이 더 작아짐을 확인할 수 있다.

V. Conclusions

본 연구에서는 전통적인 보안 모델을 벗어나는 안전성의 위협이 되는 경우인 악의적인 입력의 조작을 방지하기 위해 입력인증의 한 방법인 수정된 CL signature를 SMC에서 사용하기 위해 효율적이고 안전한 batch verification을 하는 방법을 알아보았다. 수정된 CL signature를 이용하면 검증에서의 계산량이 획기적으로 줄어들고, 특히 batch verification과 commitment scheme을 활용하면 계산량을 더욱 줄어든다. 따라서 본 연구를 통해 입력값의 용량이 큰 경우에도 효율적인 검증이 가능함을 확인하였으며, 비밀을 유지하며 검증하게 하는 아이디어가 다른 서명 체계에도 적용될 수 있음을 알 수 있었다. 차후 연구에서는 일반적인 서명 체계에서 입력값의 정확성을 강제할 수 있게 하는 방법으로 연구를 확장하고자 하며, 더불어 효율성을 향상하여 SMC에 적용할 수 있도록 하는 것을 목표로 한다.

ACKNOWLEDGEMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2020R1G1A1A01100862)

REFERENCES

[1] Yao, Andrew C. "Protocols for secure computations." In 23rd annual symposium on foundations of computer science (sfcs 1982), pp. 160-164. IEEE, 1982. https://doi.org/10.1109/SFCS.1982.38

[2] Goldreich, Oded. "Secure multi-party computation." Manuscript. Preliminary version 78, no. 110 (1998).

[3] D. Bogdanov, M. Joemets, S. Siim, and M. Vaht. How the Estonian tax and customs board evaluated a tax fraud detection system based on secure multi-party computation. In Financial Cryptography and Data Security, pages 227-234, 2015. https://doi.org/10.1007/978-3-662-47854-7_14

[4] P. Bogetoft, D. Christensen, I. Damgard, M. Geisler, T. Jakobsen, M. Kroigaard, J. D. Nielsen, J. B. Nielsen, K. Nielsen, J. Pagter, and M. Schwartzbach. Secure multiparty computation goes live. In Financial Cryptography and Data Security, pages 325-343, 2009. https://doi.org/10.1007/978-3-642-03549-4_20

[5] Kreuter, Benjamin. "Secure multiparty computation at Google." In Real World Crypto Conference (RWC). 2017.

[6] Halpern, Joseph, and Vanessa Teague. "Rational secret sharing and multiparty computation." In Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, pp. 623-632. 2004. https://doi.org/10.1145/1007352.1007447

[7] Wallrabenstein, John Ross, and Chris Clifton. "Equilibrium concepts for rational multiparty computation." In Decision and Game Theory for Security: 4th International Conference, GameSec 2013, Fort Worth, TX, USA, November 11-12, 2013. Proceedings 4, pp. 226-245. Springer International Publishing, 2013. https://doi.org/10.1007/978-3-319-02786-9_14

[8] Camenisch, Jan, Dieter Sommer, and Roger Zimmermann. "A general certification framework with applications to privacy-enhancing certificate infrastructures." In Security and Privacy in Dynamic Environments: Proceedings of the IFIP TC-11 21st International Information Security Conference (SEC 2006), 22–24 May 2006, Karlstad, Sweden 21, pp. 25-37. Springer US, 2006. https://doi.org/10.1007/0-387-33406-8_3

[9] Camenisch, Jan, and Gregory M. Zaverucha. "Private intersection of certified sets." In Financial Cryptography and Data Security: 13th International Conference, FC 2009, Accra Beach, Barbados, February 23-26, 2009. Revised Selected Papers 13, pp. 108-127. Springer Berlin Heidelberg, 2009. https://doi.org/10.1007/978-3-642-03549-4_7

[10] De Cristofaro, Emiliano, and Gene Tsudik. "Practical private set intersection protocols with linear complexity." In Financial Cryptography and Data Security: 14th International Conference, FC 2010, Tenerife, Canary Islands, January 25-28, 2010, Revised Selected Papers 14, pp. 143-159. Springer Berlin Heidelberg, 2010. https://doi.org/10.1007/978-3-642-14577-3_13

[11] Blanton, Marina, and Fattaneh Bayatbabolghani. "Efficient server-aided secure two-party function evaluation with applications to genomic computation." Cryptology ePrint Archive (2015).

[12] Katz, Jonathan, Alex J. Malozemoff, and Xiao Wang. "Efficiently enforcing input validity in secure two-party computation." Cryptology ePrint Archive (2016).

[13] Zhang, Y., Blanton, M., Bayatbabolghani, F. (2017). Enforcing Input Correctness via Certification in Garbled Circuit Evaluation. In: Foley, S., Gollmann, D., Snekkenes, E. (eds) Computer Security – ESORICS 2017. ESORICS 2017. Lecture Notes in Computer Science(), vol 10493. Springer, Cham. https://doi.org/10.1007/978-3-319-66399-9_30

[14] Pedersen, T.P. (1992). Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. In: Feigenbaum, J. (eds) Advances in Cryptology — CRYPTO '91. CRYPTO 1991. Lecture Notes in Computer Science, vol 576. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-46766-1_9

[15] Camenisch, J., Lysyanskaya, A. (2003). A Signature Scheme with Efficient Protocols. In: Cimato, S., Persiano, G., Galdi, C. (eds) Security in Communication Networks. SCN 2002. Lecture Notes in Computer Science, vol 2576. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-36413-7_20

[16] Camenisch, J., Lysyanskaya, A. (2004). Signature Schemes and Anonymous Credentials from Bilinear Maps. In: Franklin, M. (eds) Advances in Cryptology – CRYPTO 2004. CRYPTO 2004. Lecture Notes in Computer Science, vol 3152. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-28628-8_4

[17] Bellare, M., Garay, J.A., Rabin, T. (1998). Fast batch verification for modular exponentiation and digital signatures. In: Nyberg, K. (eds) Advances in Cryptology — EUROCRYPT'98. EUROCRYPT 1998. Lecture Notes in Computer Science, vol 1403. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0054130

Author

Myoungin Jeong received the B.S. degree in Dept. of Mathematics from Korea Military Academy, Korea, in 2004. She received M.S. degree in Dept. of Mathematical Science from Seoul National University, Korea, in 2008. And received Ph.D. degree in Dept. of Mathematics from University at Buffalo, United States, in 2018. Dr. Jeong joined the faculty of the Department of Mathematics at Korea Military Academy, Seoul, Korea, in 2018. She is currently an Assistant Professor in the Department of Mathematics, Korea Military Academy. She is interested in information security, cryptography, and scientific combat training system.