*Journal of The Korea Institute of Information Security & Cryptology* ISSN 1598-3986(Print)
VOL.31, NO.3, Jun. 2021 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2021.31.3.509

# 래티스에서 ID 기반의 이행성 서명 기법*

노 건 태,† 천 지 영‡
서울사이버대학교 (교수)

# Identity-Based Transitive Signature Scheme from Lattices*

Geontae Noh,† Ji Young Chun‡
Seoul Cyber University (Professor)

요 약

이행성 서명 기법은 이행적으로 닫혀있는 그래프에서 간선들을 인증하는 경우 매우 유용하게 사용될 수 있는 기법이다. 다시 말하면, 간선 (i, j)에 대한 인증 값이 있고, 간선 (j, k)에 대한 인증 값이 있을 때, 해당 인증 값을 이행성 서명을 통해 생성하게 되면 간선 (i, k)에 대한 인증 값도 별도의 인증 절차 없이 바로 계산할 수 있게 된다. 본 논문에서 우리는 공개키가 난수 형태가 아닌, ID 구조를 가지는 이행성 서명 기법을 처음으로 설계한다. 우리가 설계한 기법은 래티스의 어려운 문제에 기반을 두고 설계되었다.

ABSTRACT

The transitive signature scheme is a technique that can be very useful when authenticating edges in a graph that is transitively closed. In other words, when there is an authentication value for an edge (i, j) and an authentication value for an edge (j, k), the authentication value for the edge (i, k) can also be calculated immediately without any separate authentication procedure through a transitive signature. In this paper, we propose the first identity-based transitive signature scheme. Our scheme is based on the lattice problem.

**Keywords:** Lattice-based cryptography, Identity-based, Transitive signature

## I. 서 론

이행성 서명(transitive signature) 기법은 2002년, S. Micali와 R.L. Rivest에 의해 처음으로 소개되었다[1]. 이행성 서명은 이행적으로 닫혀있는 그래프에서 간선(edge) 정보를 서명함으로써 인증하는 경우 유용하게 사용된다. 간단히 예를 들면, 이행적으로 닫혀있는 그래프에서 $i$번째 정점(node)과 $j$번째 정점이 연결되어 있을 때, 두 정점이 연결되었음을 인증하기 위해 $(i,j)$ 간선을 전자서명한다. 이러한 간선의 인증 절차를 기존의 전자서명이 아니라 이행성 서명을 사용하게 되면, $i$번째 정점과 $j$번째 정점이 연결되어 있음을 나타내는 $(i,j)$ 간선의 서명 값과, $j$번째 정점과 $k$번째 정점이 연결되어 있음을 나타내는 $(j,k)$ 간선의 서명 값을 사용하여 어떤 비밀 정보 없이 누구나 $i$번째 정점과 $k$번째 정점이 연결되어 있다는 $(i,k)$ 간선의 서명 값을 계산해낼 수 있다.

$n$개의 노드들로 구성된 무방향 그래프(undirected graph)가 있을 때 전자 서명 기법을 사용하면 최대 $\frac{n(n-1)}{2}$개의 인증 값을 생성해야 하나, 이행성 서명 기법을 사용하면 최대 $n-1$개의 인증 값만 생성하면 된다. 이러한 이행성 서명 기법

Received(04. 26. 2021), Modified(05. 21. 2021), Accepted(05. 23. 2021)
* 이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2021R1F1A1060543).
† 주저자, gnoh@iscu.ac.kr
‡ 교신저자, jychun@iscu.ac.kr(Corresponding author)

을 활용하면 애드혹 네트워크(ad hoc network)에서 안전한 라우팅 인증에 사용될 수 있다[2].

이행성 서명에 대한 연구는 2002년을 기점으로 꾸준히 연구되었으며, 지금까지 무방향 그래프에서 이행성 서명 기법을 설계하는 연구 위주로 진행되었다. 이러한 이유로는 방향 그래프(directed graph)에서 이행성 서명 기법을 설계하는 것의 어려움 때문인데, 실제로 2003년, S.R. Hohenberger는 방향 그래프에서의 이행성 서명 기법을 설계하는 것이 어려움을 자신의 논문을 통해 보인 바 있다[3]. 따라서 본 논문에서도 무방향 그래프인 경우로 한정한다.

또한, 지금까지 설계된 이행성 서명 기법들은 공개키가 난수 형태를 기본으로 하여 설계되었으며, 아직 공개키가 ID 형태인 이행성 서명 기법은 제안되지 않았다. 다시 말하면, 현재까지의 이행성 서명 기법의 공개키는 난수 형태여서 사용자의 식별을 위해서는 추가적인 절차가 필요하지만, ID 기반의 이행성 서명 기법이 존재한다면 사용자의 이메일과 같이 식별이 용이한 정보가 공개키로 바로 사용되기 때문에 절차가 간결해진다는 장점이 있다.

### 1.1 관련 연구

2002년, S. Micali와 R.L. Rivest에 의해 처음으로 소개되었으며[1], 지금까지 다양한 구조에서 설계된 이행성 서명 기법들이 다수 제안되었다. 2004년에는 S.F. Shahandashti 등에 의해 겹선형 맵(bilinear map)에 기반을 두고 이행성 서명 기법이 설계되었다[4]. 이후 M. Bellare와 G. Neven에 의해 RSA 가정, 인수 분해 문제, 이산대수 문제, GDH(Gap Diffie-Hellman) 가정에 기반을 둔 이행성 서명 기법들이 설계되었다[5,6]. 2010년에는 Z. Gong 등에 의해 LFSR(Linear Feedback Shift Register)에 기반을 둔 이행성 서명 기법이 설계되었고[7]. 2016년에는 C. Lin 등에 의해 겹선형 맵과 암호학적 해시 함수를 사용한 이행성 서명 기법이 설계되었다[8]. 2019년에는 G. Noh와 I.R. Jeong에 의해 래티스의 어려운 문제에 기반을 둔 이행성 서명 기법이 처음 설계되었다[9].

### 1.2 공헌

지금까지 다수의 이행성 서명 기법들이 설계되었으나, 모두 공개키가 난수 형태를 가지는 형태의 설계들만 이루어져 왔다. 본 논문에서 우리는 공개키가 난수 형태가 아닌, ID 형태인 이행성 서명 기법을 처음으로 설계한다. 본 논문에서 설계된 ID 기반의 이행성 서명 기법은 래티스에서 SIS(Short Integer Solution) 문제의 어려움에 기반을 두고 설계되며, 안전성 또한 증명된다.

### 1.3 설계 아이디어

2019년, G. Noh와 I.R. Jeong에 의해 래티스에서 난수 형태의 공개키 기반으로 설계된 이행성 서명 기법[9]과는 달리, 이행성 서명 기법의 공개키를 ID 기반으로 설계하기 위해서는 ID 기반 키 구조를 추가로 설계해야 한다. ID 기반 키 구조를 가지는 이행성 서명 기법을 설계하기 위해 우리는 G. Noh와 I.R. Jeong의 이행성 서명 기법의 설계 구조를 기본으로 하고, D. Cash 등이 설계한 트랩도어 확장 알고리즘을 적용하여 공개키 구조를 ID 기반으로 설계한다[9,10]. D. Cash 등의 트랩도어 확장 알고리즘을 통해 래티스 행렬과 ID 구조를 결합하여 확장된 래티스를 구성할 수 있게 되며, ID 구조에 대응되는 트랩도어를 통해 확장된 래티스의 기저를 생성하는 것이 가능해진다.

제안하는 기법에서 간선 $(i,j)$의 서명이 이행성을 가지도록 하기 위해 정점 $i$와 $j$ 각각에 대한 해시 값을 계산할 수 있는 벡터를 각각 샘플링하였으며, 이렇게 샘플링된 두 벡터의 차이를 서명으로 구성하였다. 이러한 구조를 기본 아이디어로 하여 제안하는 서명 기법은 이행성을 가진다.

### 1.4 구성

본 논문은 다음과 같이 구성된다. 2장에서는 본 논문을 이해하기 위한 사전 지식을 살펴보고, 3장에서는 ID 기반의 이행성 서명 기법을 설계한다. 4장에서는 설계된 ID 기반의 이행성 서명 기법의 안전성을 증명하고, 5장에서 결론을 맺는다.

## II. 배경 지식

### 2.1 표기법

본 논문은 다음의 표기를 사용한다. 시큐리티 파라미터는 $n$이다. 빅오((big-$O$) 표기법을 사용한다. 벡터나 행렬의 접합은 $\|$, 유클리디안 거리는 $\| \ \|$로 표기한다. 행렬 $A$의 그램-슈미트 (Gram-Schmidt) 직교화는 $\widetilde{A}$이다. $m = poly(n)$은 어떤 양의 정수 $c$에 대해 $m \in \Theta(n^c)$임을 의미한다. 충분히 큰 $n$과 어떤 $c > 0$에 대해 $|negl(n)| < 1/n^c$이면 $negl(n)$는 무시할만하다.

### 2.2 래티스

본 절에서 우리는 패리티 검사 행렬(parity check matrix) $A \in Z_q^{n \times m}$로부터 생성되는 $m$차원 풀-랭크 정수 래티스(full-rank integer lattice)의 특정 형태를 다음과 같이 정의한다.

$$\Lambda^{\perp}(A) = \left\{ Ax = \sum_{i \in \{1,\cdots,m\}} x_i a_i = 0 \in Z_q^n : x \in Z^m \right\} \quad (1)$$

식 (1)에서 $n \geq 1$과 $q \geq 2$는 정수이고, $m = O(n \log q)$에 대해 유니폼하게 랜덤인 행렬 $A \in Z_q^{n \times m}$의 열벡터들은 선형 결합을 통해 $Z_q^n$를 모두 표현할 수 있다[11].

#### 2.2.1 SIS 문제

본 항에서는 우리가 설계한 기법의 안전성을 증명하기 위해 사용하는 어려운 문제인 SIS 문제를 살펴본다.

$SIS_{n,m,q,\beta}$는 어떤 $m = poly(n)$과 $q \geq 2$에 대해 유니폼하게 랜덤인 행렬 $A \in Z_q^{n \times m}$를 입력받아 $\| v \| \leq \beta$를 만족하는 래티스 $\Lambda^{\perp}(A)$의 0이 아닌 벡터 $v \in Z^m$를 찾는 문제이다.

#### 2.2.2 기본 알고리즘

본 항에서는 우리가 3장에서 설계한 기법에서 사용된 기본 알고리즘들을 살펴본다.

- $TrapGen(1^n, 1^m, q)$[12]: 트랩도어 생성 알고리즘 $TrapGen(1^n, 1^m, q)$은 시큐리티 파라미터 $n$, 래티스 차원 $m = O(n \log q)$, 자연수 $q$를 입력받아 유니폼하게 랜덤인 행렬 $A \in Z_q^{n \times m}$와 래티스 $\Lambda^{\perp}(A)$의 기저 $TINZ^{m \times m}$를 생성한다. 여기서 $T$는 $\| \widetilde{T} \| = O(\sqrt{n \log q})$와 $AT = 0 \pmod q$를 만족한다.
- $SampleD(A, T, y, s)$[13]: 가우시안 샘플링 알고리즘 $SampleD(A, T, y, s)$는 유니폼하게 랜덤인 행렬 $A \in Z_q^{n \times m}$, 래티스 $\Lambda^{\perp}(A)$의 기저 $T \in Z^{m \times m}$, 벡터 $y \in Z_q^n$, 가우시안 파라미터 $s$를 입력받아 $Av = y \pmod q$와 $\|v\| \leq s\sqrt{m}$를 만족하는 벡터 $v \in Z^m$를 가우시안 샘플링한다.
- $SampleDom(1^{2m}, s)$[13]: $SampleDom(1^{2m}, s)$ 알고리즘은 가우시안 샘플링 알고리즘 $SampleD(A, T_A, y, s)$의 정의역 벡터 $x \in Z^m$를 가우시안 샘플링하며, 여기서 $x$는 $\| x \| \leq s\sqrt{m}$를 만족한다.
- $ExtBasis(T, A \| A')$[10]: 트랩도어 확장 알고리즘 $ExtBasis(T, A \| A')$는 유니폼하게 랜덤인 행렬 $A \in Z_q^{n \times m}$, 래티스 $\Lambda^{\perp}(A)$의 기저 $T \in Z^{m \times m}$, 유니폼하게 랜덤인 행렬 $A' \in Z_q^{n \times m'}$을 입력받아 래티스 $\Lambda^{\perp}(A \| A')$의 기저 $T' \in Z^{(m+m') \times (m+m')}$를 생성한다. 여기서 $T'$는 $\| \widetilde{T} \| = \| \widetilde{T'} \|$를 만족한다.

### 2.3 ID 기반의 이행성 서명

본 장에서는 ID 기반의 이행성 서명을 정의한다. ID 기반의 이행성 서명 기법 $IBTS = \{Setup, Extract, Sign, Vrfy, Comp\}$는 다음과 같이 구성된다.

- $Setup$: 마스터 비밀키 $msk$와 공개 파라미터 $params$를 생성한다.
- $Extract(msk, params, ID)$: 마스터 비밀키 $msk$, 공개 파라미터 $params$, ID $ID$를 입력받아 $ID$에 대응되는 비밀키 $sk_{ID}$를 생성한다.
- $Sign(params, ID, sk_{ID}, (i,j))$: 공개 파라미터 $params$, ID $ID$, $ID$에 대응되는 비밀키 $sk_{ID}$, 간선 $(i,j)$를 입력받아 간선 $(i,j)$에 대한 서명 $\sigma_{(i,j)}$을 생성한다.
- $Vrfy(params, ID, (i,j), \sigma_{(i,j)})$: 공개 파라미터 $params$, ID $ID$, 간선 $(i,j)$, 서명 $\sigma_{(i,j)}$를 입력

받아 서명 $\sigma_{(i,j)}$가 간선 $(i,j)$에 대한 서명이 맞는지 검증한다.

- $Comp(params, ID, (i,j), (j,k), \sigma_{(i,j)}, \sigma_{(j,k)})$: 공개 파라미터 $params$, ID $ID$, 간선 $(i,j)$와 이것에 대한 서명 $\sigma_{(i,j)}$, 간선 $(j,k)$와 이것에 대한 서명 $\sigma_{(j,k)}$를 입력받아 간선 $(i,k)$에 대한 서명 $\sigma_{(i,k)}$를 생성한다.

ID 기반의 이행성 서명은 다음의 세 가지 성질들을 만족해야 한다.

- 정확성(correctness): 정당한 사용자가 자신의 ID $ID$와 비밀키 $sk_{ID}$를 사용하여 $Sign(params, ID, sk_{ID}, (i,j))$ 알고리즘으로부터 정당하게 간선 $(i,j)$에 대한 서명 $\sigma_{(i,j)}$를 생성하였다면, $Vrfy(params, ID, (i,j), \sigma_{(i,j)})$ 알고리즘을 사용하여 이 서명 $\sigma_{(i,j)}$가 간선 $(i,j)$에 대한 서명이 맞는지 검증되어야 한다.
- 이행성(transitivity): 정당한 절차를 거쳐 생성된 두 개의 간선 $(i,j)$, $(j,k)$에 대한 서명 $\sigma_{(i,j)}$와 $\sigma_{(j,k)}$가 있다면, 어떠한 비밀 정보 없이 누구나 간선 $(i,k)$에 대한 서명 $\sigma_{(i,k)}$를 만들어낼 수 있어야 한다.
- 이행적 위조 불가능성(transitive unforgeability): 정당한 사용자만이 각 간선에 대한 정당한 서명을 생성할 수 있어야 한다. 단, 이행적 성질로 생성할 수 있는 경우는 제외한다.

위의 세 가지 성질 중 이행적 위조 불가능성은 다음과 같은 챌린저와 공격자 사이의 게임으로 표현할 수 있다. 게임을 시작하기 전, $Setup$ 단계에서 챌린저는 공격자에게 공개 파라미터 $params$를 제공한다. 이후, 공격자는 다음의 두 가지 오라클에 접근할 수 있고, 마지막에는 $Output$ 단계가 존재한다.

- $Extract(msk, params, \bullet)$: 공격자는 자신이 선택한 ID $ID$에 대응되는 비밀키 $sk_{ID}$를 얻을 수 있다.
- $Sign(params, \bullet, sk_{ID}, \bullet)$: 공격자는 자신이 선택한 ID $ID$와 간선 $(i,j)$에 대한 서명 $\sigma_{(i,j)}$를 얻을 수 있다.
- $Output$: 공격자는 자신이 선택한 ID $ID^*$와 간선 $(i^*, j^*)$에 대한 서명 $\sigma_{(i^*,j^*)}$를 챌린저에게 보낸다. 만약 $ID^*$가 $Extract(msk, params, \bullet)$ 오라클에 질의된 적이 없으며, $\sigma_{(i^*,j^*)}$가 $ID^*$와 $(i^*, j^*)$에 대한 정당한 서명이고, 이전의 서명들로부터 이행적으로 생성한 것이 아니면 공격자는 이행적 위조 불가능성 게임에서 승리한다.

## III. 제안하는 기법

우리는 본 장에서 ID 기반의 이행성 서명 기법을 처음으로 제안한다. 우리가 제안하는 기법은 SIS 문제의 어려움에 기반을 두고 설계되었다. 우리가 제안한 기법은 스테이트(state)를 사용하는 기법이며, M. Bellare와 G. Neven이 구성한 일반적인 변환을 사용하여 스테이트 없는 기법으로 변환할 수 있다[6].

제안하기에 앞서, 우리의 기법에서 사용될 파라미터들을 아래의 Table 1.과 같이 먼저 정의한다.

Table 1. Parameters

| Parameters | |
|---|---|
| $m$ | $O(n \log q)$ |
| $q$ | $poly(n)$ |
| $s$ | $O(\sqrt{n \log q}) \bullet \omega(\sqrt{\log n})$ |

제안하는 ID 기반의 이행성 서명 기법 $IBTS = \{Setup, Extract, Sign, Vrfy, Comp\}$는 다음과 같이 구성된다.

- $Setup$: 마스터 비밀키 $msk$와 공개 파라미터 $params$를 다음과 같이 생성한다.
  1) 트랩도어 생성 알고리즘 $TrapGen(1^n, 1^m, q)$을 사용하여 $(A_0, T_0)$를 생성한다. 여기서 $A_0 \in Z_q^{n \times m}$이고, $T_0 \in Z^{m \times m}$이다.
  2) 두 개의 해시함수 $H_0 : \{0,1\}^* \rightarrow Z_q^{n \times m}$, $H_1 : \{0,1\}^* \rightarrow Z_q^n$를 선택한다. 여기서 $H_0(\bullet)$와 $H_1(\bullet)$은 모두 안전성 분석 시 랜덤 오라클로 사용된다.
  3) 마스터 비밀키는 $msk = T_0$이고, 공개 파라미터는 $params = \{A_0, H_0, H_1\}$이다.
- $Extract(msk, params, ID)$: 마스터 비밀키 $msk$, 공개 파라미터 $params$, ID $ID$를 입력받아 $ID$에 대응되는 비밀키 $sk_{ID}$를 다음과 같이

생성한다.

1) 트랩도어 확장 알고리즘 $ExtBasis(T_0, A_0 \| H_0(ID))$를 사용하여 $T_{ID} \in Z^{2m \times 2m}$를 생성한다.

2) $ID$에 대응되는 비밀키는 $sk_{ID} = T_{ID}$이다.

- $Sign(params, ID, sk_{ID}, (i,j))$: 공개 파라미터 $params$, ID $ID$, $ID$에 대응되는 비밀키 $sk_{ID}$, 간선 $(i,j)$를 입력받아 간선 $(i,j)$에 대한 서명 $\sigma_{(i,j)}$을 다음과 같이 생성한다.
  1) 만약 스테이트 $St(i)$가 비어있다면, 가우시안 샘플링 알고리즘 $SampleD(A_0 \| H_0(ID), T_{ID}, H_1(i), s)$를 사용하여 벡터 $v_i$를 샘플링하고, $St(i)$에 $v_i$를 추가한다.
  2) 만약 스테이트 $St(j)$가 비어있다면, 가우시안 샘플링 알고리즘 $SampleD(A_0 \| H_0(ID), T_{ID}, H_1(j), s)$를 사용하여 벡터 $v_j$를 샘플링하고, $St(j)$에 $v_j$를 추가한다.
  3) 스테이트 $St(i)$와 $St(j)$에서 $v_i$와 $v_j$를 가져와서 서명 $\sigma_{(i,j)} = v_i - v_j$를 계산한다.
- $Vrfy(params, ID, (i,j), \sigma_{(i,j)})$: 공개 파라미터 $params$, ID $ID$, 간선 $(i,j)$, 서명 $\sigma_{(i,j)}$를 입력받아 서명 $\sigma_{(i,j)}$가 간선 $(i,j)$에 대한 서명이 맞는지 다음과 같이 검증한다.
  1) $\| \sigma_{(i,j)} \| \leq 2s \cdot \sqrt{m}$를 만족하는지 확인한다.
  2) $(A_0 \| H_0(ID)) \cdot \sigma_{(i,j)} = H_1(i) - H_1(j)$를 만족하는지 확인한다.
  3) 위의 두 조건을 모두 만족하면 서명 $\sigma_{(i,j)}$는 간선 $(i,j)$에 대한 정당한 서명이고, 그렇지 않다면 정당한 서명이 아니다.
- $Comp(params, ID, (i,j), (j,k), \sigma_{(i,j)}, \sigma_{(j,k)})$: 공개 파라미터 $params$, ID $ID$, 간선 $(i,j)$와 이것에 대한 서명 $\sigma_{(i,j)}$, 간선 $(j,k)$와 이것에 대한 서명 $\sigma_{(j,k)}$를 입력받아 간선 $(i,k)$에 대한 서명 $\sigma_{(i,k)} = \sigma_{(i,j)} + \sigma_{(j,k)}$를 생성한다.

제안하는 기법은 공개 정보로 $ID$와 $params$, 비밀 정보로는 $msk$와 $sk_{ID}$로 구성된다. $ID$는 사용자의 ID이고, $params$는 두 개의 해시 함수 $H_0(\cdot)$와 $H_1(\cdot)$, 그리고 $Z_q$ 상에서 $n \times m$ 차원 행렬로 구성된다. $msk = T_0$와 $sk_{ID} = T_{ID}$는 각각 $\| \widetilde{T_0} \| = O(\sqrt{n \log q})$와 $\| \widetilde{T_{ID}} \| = O(\sqrt{n \log q})$를 만족하며, 각각 $Z$ 상에서 $m \times m$ 차원 행렬과 $2m \times 2m$ 차원 행렬로 구성된다. 제안하는 기법을 통해 생성된 서명 $\sigma_{(i,j)}$은 $\| \sigma_{(i,j)} \| \leq 2s \cdot \sqrt{m}$를 만족하는 $2m$ 차원 벡터이다.

## IV. 안전성 분석

본 논문의 3장에서 우리가 제안한 기법은 정확성, 이행성, 이행적 위조 불가능성을 모두 만족한다. 본 장에서는 우리가 제안한 기법이 정확성, 이행성, 이행적 위조 불가능성을 만족하는지 분석한다.

처음으로, 우리의 기법이 정확성을 만족하는지 분석한다.

정리 1. 〔정확성〕 우리의 기법은 정확성을 만족한다.

증명. $Sign(params, ID, sk_{ID}, (i,j))$ 알고리즘에서 가우시안 샘플링 알고리즘 $SampleD(A_0 \| H_0(ID), T_{ID}, H_1(i), s)$와 $SampleD(A_0 \| H_0(ID), T_{ID}, H_1(j), s)$을 통해 $\| v_i \| \leq s \cdot \sqrt{2m}$, $\| v_j \| \leq s \cdot \sqrt{2m}$, $(A_0 \| H_0(ID)) \cdot v_i = H_1(i)$, $(A_0 \| H_0(ID)) \cdot v_j = H_1(j)$를 만족하는 벡터 $v_i$와 $v_j$를 샘플링한다. 즉, $(A_0 \| H_0(ID)) \cdot \sigma_{(i,j)} = (A_0 \| H_0(ID)) \cdot (v_i - v_j) = H_1(i) - H_1(j)$와 $\| \sigma_{(i,j)} \| = \| v_i - v_j \| \leq 2s \cdot \sqrt{m}$이다. 따라서 제안하는 기법은 정확성을 만족한다. □

다음으로, 우리의 기법이 이행성을 만족하는지 분석한다.

정리 2. 〔이행성〕 우리의 기법은 이행성을 만족한다.

증명. $Comp(params, ID, (i,j), (j,k), \sigma_{(i,j)}, \sigma_{(j,k)})$ 알고리즘은 $\sigma_{(i,k)} = \sigma_{(i,j)} + \sigma_{(j,k)} = (v_i - v_j) + (v_j - v_k) = v_i - v_k$을 계산하는데, $\| v_i \| \leq s \cdot \sqrt{2m}$, $\| v_j \| \leq s \cdot \sqrt{2m}$, $(A_0 \| H_0(ID)) \cdot v_i = H_1(i)$, $(A_0 \| H_0(ID)) \cdot v_j = H_1(j)$를 만족한다. 즉,

$(A_0 \| H_0(ID)) \cdot \sigma_{(i,k)} = (A_0 \| H_0(ID)) \cdot (v_i - v_k)$
$= H_1(i) - H_1(k)$이고, $\| \sigma_{(i,k)} \| \le 2s \cdot \sqrt{m}$이다. 따라서 제안하는 기법은 이행성을 만족한다. □

마지막으로, 우리의 기법이 이행적 위조 불가능성을 만족하는지 분석한다.

정리 3. 〔이행적 위조 불가능성〕 만약 SIS 문제가 어렵다고 가정하면, 우리의 기법은 이행적 위조 불가능성을 만족한다.

증명. 챌린저와 공격자 사이의 게임을 사용하여 제안하는 기법이 이행적 위조 불가능성을 만족하는지를 증명한다. 챌린저는 SIS 문제를 풀기를 원하고, 공격자는 제안하는 기법의 이행적 위조 불가능성을 공격하려고 한다. 제안하는 기법에서 사용되는 두 개의 해시 함수 $H_0 : \{0,1\}^* \rightarrow Z_q^{n \times m}$와 $H_1 : \{0,1\}^* \rightarrow Z_q^n$은 모두 랜덤 오라클로 사용된다. *Setup* 단계에서 챌린저는 $SIS_{n,2m,q,\beta}$ 문제의 입력으로 행렬 $A \in Z_q^{n \times 2m}$를 받는다. 챌린저는 행렬 $A$를 $A = (A_0 \| A_1)$로 분해하는데, 여기서 $A_0 \in Z_q^{n \times m}$이고 $A_1 \in Z_q^{n \times m}$이다. 챌린저는 공격자에게 공개 파라미터 $params = \{A_0\}$를 제공하고, 공격자는 $H_0(\cdot)$와 $H_1(\cdot)$에 대한 해시 오라클과 $Extract(msk, params, \cdot)$ 오라클, $Sign(params, \cdot, sk_{ID}, \cdot)$ 오라클을 사용할 수 있다.

- $H_0(\cdot)$: 공격자는 자신이 선택한 ID $ID'$를 최대 $q_{H_0}$번 질의할 수 있다. 챌린저는 $q_{H_0}$번의 질의 중 1번에 한해 $ID'$에 대한 리턴으로 $A_1$을 공격자에게 돌려주고, 나머지 경우에는 트랩도어 생성 알고리즘 $TrapGen(1^n, 1^m, q)$을 사용하여 $(A', T')$를 생성하고 $A'$를 공격자에게 돌려준다. 여기서 $A' \in Z_q^{n \times m}$이고, $T' \in Z^{m \times m}$이다. 챌린저는 생성된 모든 정보들을 저장하여 동일한 질의에 대해 항상 동일한 값을 공격자에게 돌려준다. $H_0(\cdot)$ 오라클은 $Extract(msk, params, \cdot)$ 오라클과 $Sign(params, \cdot, sk_{ID}, \cdot)$ 오라클을 통해호출 가능하며, 이러한 경우에는 $A_1$를 공격자에게 돌려주지 않는다.
- $H_1(\cdot)$: 공격자는 자신이 선택한 정점 $i$에 대해 질의할 수 있다. 챌린저는 $SampleDom(1^{2m}, s)$ 알고리즘을 사용하여 $v_i \leftarrow Z^{2m}$를 샘플링하고, $Av_i = h_i \pmod q$를 계산하여 공격자에게 $h_i$를 돌려준다. 챌린저는 생성된 모든 정보들을 저장하여 동일한 질의에 대해 항상 동일한 값을 공격자에게 돌려준다.
- $Extract(msk, params, \cdot)$: 공격자는 자신이 선택한 ID $ID'$를 질의할 수 있다. 만약 공격자가 질의한 $ID'$가 $H_0(\cdot)$ 오라클에 질의한 적이 있고, 반환값으로 공격자에게 $A_1$를 돌려줬다면, 챌린저는 즉시 게임을 종료한다. 만약 공격자가 $H_0(\cdot)$ 오라클에 $ID'$를 질의했었다면, 챌린저는 저장되어 있는 정보들을 기반으로 트랩도어 확장 알고리즘 $ExtBasis(T', A_0 \| A')$를 수행하고, 이것에 대한 결과값인 $T''$를 공격자에게 돌려준다. 만약 공격자가 선택한 $ID'$가 $Extract(msk, params, \cdot)$ 오라클 또는 $H_0(\cdot)$ 오라클에 질의한 적이 없다면, 챌린저는 이 $ID'$를 $H_0(\cdot)$ 오라클에 질의하여 $ID'$에 대응되는 $(A', T')$를 검색해오고, 트랩도어 확장 알고리즘 $ExtBasis(T', A_0 \| A')$를 수행하여 이것에 대한 결과값인 $T''$를 공격자에게 돌려준다. 챌린저는 생성된 모든 정보들을 저장한다.
- $Sign(params, \cdot, sk_{ID}, \cdot)$: 공격자는 자신이 선택한 $(ID', (i,j))$를 질의할 수 있다. 만약 공격자가 $ID'$를 $H_0(\cdot)$ 오라클에 질의했었고 ($Extract(msk, params, \cdot)$ 오라클에 질의하여 $H_0(\cdot)$ 오라클을 호출한 경우도 포함), 그것에 대한 결과값으로 $A_1$를 공격자에게 돌려줬었다면, 챌린저는 여기서 게임을 종료한다. 챌린저는 $ID'$를 $H_0(\cdot)$ 오라클에 질의하거나 기존에 저장된 정보들을 사용하여 $A'$와 $T'$를 얻는다. 마찬가지로 챌린저는 $i$와 $j$를 $H_1(\cdot)$ 오라클에 질의하거나, 이미 저장된 정보들을 사용하여 $(v_i, h_i)$와 $(v_j, h_j)$를 얻는다. 챌린저는 서명 $\sigma_{(i,j)} = v_i - v_j$를 계산하여 공격자에게 돌려준다.
- *Output*: 공격자는 $(ID^*, (i^*, j^*), \sigma_{(i^*,j^*)})$를 챌린저에게 보낸다. 만약 공격자가 $ID^*$를 $H_0(\cdot)$ 오라클에 질의($Extract(msk, params, \cdot)$ 오라클에 질의하여 $H_0(\cdot)$ 오라클을 호출한 경우도 포함)한 결과로 $A_1$를 반환하지 않았다면, 챌린저는 여

기서 게임을 종료한다. 챌린저는 $H_1(\bullet)$ 오라클을 통해 저장된 정보들을 사용하여 $(i^*, v_{i^*}, h_{i^*})$와 $(j^*, v_{j^*}, h_{j^*})$를 얻는다. 챌린저가 대답할 $SIS_{n,2m,q,\beta}$ 문제의 답은 $z = \sigma_{(i^*,j^*)} - v_{i^*} + v_{j^*}$이다.

위의 챌린저와 공격자 사이의 게임에서 중간에 종료되는 경우는 $Extract(msk, params, \bullet)$ 오라클, $Sign(params, \bullet, sk_{ID}, \bullet)$ 오라클, $Output$ 단계에서 존재하며, 제안하는 기법의 이행적 위조 불가능성에 대한 공격자의 성공 확률이 $\epsilon$일 때, 챌린저는 $\frac{\epsilon}{q_{H_0}} - negl(n)$의 확률로 $SIS_{n,2m,q,\beta}$ 문제를 풀 수 있다. □

## V. 결 론

본 논문에서 우리는 ID 기반의 이행성 서명 기법을 처음으로 제안하였다. 우리가 제안한 기법은 래티스에서 SIS 문제의 어려움에 기반을 두고 설계되었고, 랜덤 오라클 모델에서 안전성을 증명하였다.

## References


[1] S. Micali and R.L. Rivest, "Transitive signature schemes," Proceedings of the Cryptographers' Track, RSA Conference, CT-RSA'02, LNCS 2271, pp. 236-243, Feb. 2002.

[2] S. Xu, Y. Mu, and W. Susilo, "Authenticated AODV Routing Protocol using one-time signature and transitive signature schemes," Journal of Networks, vol. 1, no. 1, pp. 47-53, May 2006.

[3] S.R. Hohenberger, "The cryptographic impact of groups with infeasible inversion," Master's Thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2003.

[4] S.F. Shahandashti, M. Salmasizadeh, and J. Mohajeri, "A Provably Secure Short Transitive Signature Scheme from Bilinear Group Pairs," Proceedings of the 4th International Conference on Security in Communication Networks, SCN'04, LNCS 3352, pp. 60-76, Sep. 2004.

[5] M. Bellare and G. Neven, "Transitive signatures based on factoring and RSA," Advances in Cryptology, ASIACRYPT'02, LNCS 2501, pp. 397-414, Dec. 2002.

[6] M. Bellare and G. Neven, "Transitive signatures: new schemes and proofs," IEEE Transactions on Information Theory, vol. 51, no. 6, pp. 2133-2151, May 2005.

[7] Z. Gong, Z. Huang, W. Qiu, and K. Chen, "Transitive Signature Scheme from LFSR," Journal of Information Science and Engineering, vol. 26, no. 1, pp. 131-143, Jan. 2010.

[8] C. Lin, F. Zhu, W. Wu, K. Liang, K-K.R. Choo, "A New Transitive Signature Scheme," Proceedings of the International Conference on Network and System Security, NSS'16, LNCS 9955, pp. 156-167, Sep. 2016.

[9] G. Noh and I.R. Jeong, "Transitive Signature Schemes for Undirected Graphs from Lattices," KSII Transcations on Internet and Information Systems, vol. 13, no. 6, pp. 3316-3332, Jun. 2019.

[10] D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert, "Bonsai Trees, or How to Delegate a Lattice Basis," Advances in Cryptology, EUROCRYPT'10, LNCS 6110, pp. 523-552, Jun. 2010.

[11] O. Regev, "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography," Journal of the ACM, vol. 56, no. 6, pp. 34:1-34:40, Sep. 2009.

[12] J. Alwen and C. Peikert, "Generating

Shorter Bases for Hard Random Lattice," Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science, pp. 75-86, Feb. 2009.
[13] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for Hard Lattices and New Cryptographic Constructions," Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 197-206, May 2008.

## 〈저자소개〉

노 건 태 (Geontae Noh) 종신회원
2008년 2월: 고려대학교 산업시스템정보공학과 졸업
2010년 2월: 고려대학교 정보경영공학과 석사
2014년 8월: 고려대학교 정보보호학과 박사
2014년 9월~2017년 2월: 고려대학교 정보보호연구원 박사후 연구원, 연구교수
2017년 2월~현재: 서울사이버대학교 빅데이터·정보보호학과 조교수
2020년 3월~현재: 서울사이버대학교 빅데이터·AI센터 센터장
〈관심분야〉 암호 이론, 데이터 보안, 프라이버시 향상 기술

천 지 영 (Ji Young Chun) 종신회원
1997년 2월: 이화여자대학교 수학과 졸업
2006년 2월: 고려대학교 정보보호학과 석사
2011년 8월: 고려대학교 정보경영공학과 박사
2011년 9월~2019년 12월: 고려대학교 정보보호연구원 연구교수, 시간강사
2012년 8월~2014년 3월: University of Illinois at Urbana-Champaign 박사후 연구원
2020년 1월~2021년 1월: 이화여자대학교 엘텍공과대학 컴퓨터공학전공 특임교수
2021년 2월~현재: 서울사이버대학교 빅데이터·정보보호학과 조교수
〈관심분야〉 프라이버시 향상 기술, 빅데이터 보안, 연합 학습