



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년06월11일
(11) 등록번호 10-1404642
(24) 등록일자 2014년05월30일

(51) 국제특허분류(Int. Cl.)
H04L 9/30 (2006.01) H04L 9/32 (2006.01)
(21) 출원번호 10-2013-0104107
(22) 출원일자 2013년08월30일
심사청구일자 2013년08월30일
(56) 선행기술조사문헌
KR101382626 B1
KR1020130085491 A
US20070165843 A1

(73) 특허권자
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
김기성
서울 성동구 자동차시장1길 94-15, 606호 (용답동, 오르세오피스텔)
정익래
서울 광진구 긴고랑로4길 53, 203호 (중곡동, 호동아파트)
박소현
서울 성북구 종암로21길 127, 104동 904호 (종암동, 현대아이파크1차아파트)
(74) 대리인
특허법인엠에이피에스

전체 청구항 수 : 총 12 항

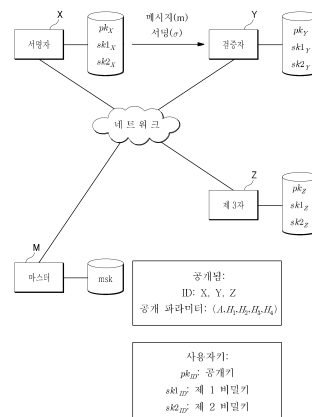
심사관 : 홍기완

(54) 발명의 명칭 래티스 기반 인증서 비사용 서명 시스템 및 방법

(57) 요약

본 발명은 자신의 마스터 키 및 사용자 개체의 공개된 식별 정보를 사용하여, 제 1 비밀키를 생성하여 상기 사용자 개체에게 할당하는 마스터 개체; 및 자신의 공개키 및 제 2 비밀키를 생성하는 사용자 개체;를 포함하되, 상기 사용자 개체는 상기 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체에게 전송하는 서명자 개체; 및 상기 서명자 개체로부터 수신한 서명을 상기 서명자 개체의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 검증자 개체;를 포함하되, 상기 마스터 키, 제 1 비밀키, 및 상기 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용되는 전자 서명 시스템을 제공한다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호	1415128860
부처명	미래창조과학부
연구사업명	정보통신기술인력양성
연구과제명	스마트 그리드 기술연구
기 여 율	1/1
주관기관	고려대학교 산학협력단
연구기간	2013.01.01 ~ 2013.12.31

특허청구의 범위

청구항 1

전자 서명 시스템에 있어서,

자신의 마스터 키 및 사용자 개체의 공개된 식별 정보를 사용하여, 제 1 비밀키를 생성하여 상기 사용자 개체에 할당하는 마스터 개체; 및

자신의 공개키 및 제 2 비밀키를 생성하는 사용자 개체;를 포함하되,

상기 사용자 개체는

상기 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체에게 전송하는 서명자 개체; 및

상기 서명자 개체로부터 수신한 서명을 상기 서명자 개체의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 검증자 개체;를 포함하되,

상기 마스터 키, 제 1 비밀키, 및 상기 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용되는 전자 서명 시스템.

청구항 2

제 1 항에 있어서,

상기 마스터 개체는

균일한 분포(uniform distribution)에서 선택된 임의의 행렬로부터 생성되는 래티스의 짧은 기저(short basis)인 트랩도어를 생성하여,

상기 트랩도어를 자신의 비밀키인 마스터 키로 사용하고, 상기 임의의 행렬을 공개하는 전자 서명 시스템.

청구항 3

제 1 항에 있어서,

상기 마스터 키의 래티스를 상기 사용자 개체의 공개된 식별 정보를 사용하여 확장된 래티스로 확장하고 상기 확장된 래티스의 짧은 기저인 확장된 트랩도어를 생성하여,

상기 확장된 트랩도어를 상기 사용자 개체의 제 1 비밀키로 발급하는 전자 서명 시스템.

청구항 4

제 1 항에 있어서,

상기 사용자 개체는

균일한 분포에서 선택된 임의의 행렬로부터 생성되는 래티스의 짧은 기저인 트랩도어를 생성하여,

상기 트랩도어를 자신의 제 2 비밀키로 사용하고, 상기 임의의 행렬을 자신의 공개키로 사용하는 전자 서명 시스템.

청구항 5

제 1 항에 있어서,

상기 사용자 개체는

자신의 제 1 비밀키 및 공개키를 사용하여 제 1 서명을 생성하고,

자신의 제 2 비밀키 및 공개키를 사용하여 제 2 서명을 생성하여,

상기 제 1 서명과 제 2 서명을 사용하여 상기 서명을 생성하는 전자 서명 시스템.

청구항 6

제 5 항에 있어서,

상기 사용자 개체는

상기 제 1 비밀키의 가우시안 분포로부터 벡터를 샘플링하여 상기 제 1 서명을 생성하는 전자 서명 시스템.

청구항 7

제 5 항에 있어서,

상기 사용자 개체는

상기 제 2 비밀키의 래티스를 상기 공개키를 사용하여 확장된 래티스로 확장하고 상기 확장된 래티스의 짧은 기저인 확장된 트랩도어를 생성하여,

상기 확장된 트랩도어의 가우시안 분포로부터 벡터를 샘플링하여 상기 제 2 서명을 생성하는 전자 서명 시스템.

청구항 8

전자 서명 방법에 있어서,

(a) 마스터 개체가 보안 파라미터를 설정하고 자신의 마스터 키를 생성하는 단계;

(b) 상기 마스터 개체가 서명자 개체 및 검증자 개체를 포함하는 사용자 개체의 공개된 식별 정보에 기초하여, 제 1 비밀키를 생성하여 상기 사용자 개체에게 할당하는 단계;

(c) 상기 사용자 개체가 자신의 공개키 및 제 2 비밀키를 생성하는 단계;

(d) 상기 서명자 개체가 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체에 전송하는 단계; 및

(e) 상기 검증자 개체가 상기 서명자 개체로부터 수신한 서명을 상기 서명자 개체의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 단계;를 포함하되,

상기 마스터 키, 제 1 비밀키, 및 상기 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용되는 전자 서명 방법.

청구항 9

제 8 항에 있어서,

상기 (a) 단계는

균일한 분포(uniform distribution)에서 선택된 임의의 행렬로부터 생성되는 래티스의 짧은 기저(short basis)인 트랩도어를 생성하여,

상기 트랩도어를 상기 마스터 개체의 비밀키인 마스터 키로 사용하고, 상기 임의의 행렬은 공개되는 전자 서명 방법.

청구항 10

제 8 항에 있어서,

상기 (b) 단계는

상기 마스터 키의 래티스를 상기 사용자 개체의 공개된 식별 정보를 사용하여 확장된 래티스로 확장하고 상기 확장된 래티스의 짧은 기저인 확장된 트랩도어를 생성하여,

상기 확장된 트랩도어를 상기 사용자 개체의 제 1 비밀키로 발급하는 전자 서명 방법.

청구항 11

제 8 항에 있어서,

상기 (c) 단계는

균일한 분포에서 선택된 임의의 행렬로부터 생성되는 래티스의 짧은 기저인 트랩도어를 생성하여

상기 트랩도어를 상기 사용자 개체의 제 2 비밀키로 사용하고, 상기 임의의 행렬을 상기 사용자 개체의 공개키로 사용하는 전자 서명 방법.

청구항 12

제 8 항에 있어서,

상기 (d) 단계는

상기 서명자 개체의 제 1 비밀키 및 공개키를 사용하여, 상기 제 1 비밀키의 가우시안 분포로부터 벡터를 샘플링하여 제 1 서명을 생성하고,

상기 서명자 개체의 제 2 비밀키 및 공개키를 사용하여, 상기 제 2 비밀키의 래티스를 상기 공개키를 사용하여 확장된 래티스로 확장하고 상기 확장된 래티스의 짧은 기저인 확장된 트랩도어를 생성하여, 상기 확장된 트랩도어의 가우시안 분포로부터 벡터를 샘플링하여 제 2 서명을 생성하여,

상기 제 1 서명과 제 2 서명을 사용하여 상기 서명을 생성하는 전자 서명 방법.

명세서

기술분야

[0001] 본 발명은 전자 서명 시스템 및 방법에 관한 것이다.

배경기술

[0002] 전자서명 기술은 데이터에 대한 인증 및 무결성 보장 등에 사용되며 암호 분야에서 가장 널리 사용되는 기술 중 하나이다. 이러한 전자서명 기술에서 서명자는 주어진 메시지에 대해 자신의 비밀키를 사용해 서명을 생성하며, 검증자는 비밀키와 쌍을 이루는 공개키를 사용해 해당 서명의 정당성을 판단한다.

[0003] 이러한 환경에서 특정 공개키가 어떠한 사용자의 것인지를 증명하는 문제는 매우 중요하기 때문에 대부분의 환경에서는 신뢰할 수 있는 제3 기관에서 발급하는 공인인증서를 통해 이러한 문제를 해결하고 있다.

[0004] 그러나 인증서 기반 전자 서명 시스템은 공인인증서 유지 관리를 위해 많은 비용을 감수해야하며, 이를 발급하는 기관은 막대한 트래픽을 처리해야 한다. 또한 인증서 자체에 대한 신뢰 문제도 가지고 있다.

[0005] 이에 따라 S. S. Al-Riyami와 K. G. Paterson가 2003년 ASIACRYPT에서 발표한 “Certificateless Public Key Cryptography” 라는 제목의 논문은 인증서를 사용하지 않는 전자서명 기술을 제안하였다. 인증서를 사용하지 않으므로 전술한 여러 문제가 해결되어 종래의 인증서 기반 전자 서명 시스템보다 높은 효율성 및 안전성을 제공한다.

[0006] 한편, 서명의 안전성을 높이기 위해 기반이 되는 수학적 난제에 대한 연구도 많이 이루어졌다. 예를 들어, 이산 대수 문제의 어려움, 곱셈형 함수의 어려움 등 평균적인 경우 어려운 문제들에서 최악의 경우에도 어려운 문제로 알려진 래티스(lattice) 상에서의 SIS(small integer solution) 문제와 LWE(learning with errors) 문제의 어려움에 기반을 두는 서명 기법들로 발전하고 있다. 이러한 래티스 기반 암호화 기술은 높은 안전성 뿐 아니라 연산 효율성 또한 제공한다. 특히 양자 컴퓨터 분석에 대해 내성이 있는 것으로 알려져 최근 이에 대한 연구가 크게 늘고 있다.

[0007] 따라서 래티스 기반의 암호 기술을 인증서 비사용 전자 서명 기술에 적용시킨다면, 더욱 높은 안전성과 효율성을 갖는 전자 서명 기술을 제공할 수 있을 것이다.

[0008] 전자 서명과 관련하여 한국등록특허 제10-0369408호("이동 통신에서의 수신자 지정 대리 서명 방법")에는 이산 대수 문제의 어려움에 기반을 둔 서명 생성 방법이 개시되어 있다.

[0009] 또한, 한국공개특허 제10-2005-0128506호("곱셈형 사상을 이용한 한정적 검증자 서명 방법")에는 곱셈형 사상을

이용한 지정된 검증자 서명 생성 방법이 개시되어 있다.

발명의 내용

해결하려는 과제

[0010] 본 발명은 전술한 문제를 해결하기 위한 것으로서, 그 목적은 높은 안전성 및 효율성을 갖는 전자 서명 시스템 및 방법을 제공하는 것이다.

과제의 해결 수단

[0011] 상기와 같은 목적을 달성하기 위한 본 발명의 제 1 측면에 따른 전자 서명 시스템은 자신의 마스터 키 및 사용자 개체의 공개된 식별 정보를 사용하여, 제 1 비밀키를 생성하여 상기 사용자 개체에게 할당하는 마스터 개체; 및 자신의 공개키 및 제 2 비밀키를 생성하는 사용자 개체;를 포함하되, 상기 사용자 개체는 상기 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체에게 전송하는 서명자 개체; 및 상기 서명자 개체로부터 수신한 서명을 상기 서명자 개체의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 검증자 개체;를 포함하되, 상기 마스터 키, 제 1 비밀키, 및 상기 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용되는 것을 특징으로 한다.

[0012] 상기와 같은 목적을 달성하기 위한 본 발명의 제 2 측면에 따른 전자 서명 방법은 (a) 마스터 개체가 보안 파라미터를 설정하고 자신의 마스터 키를 생성하는 단계; (b) 상기 마스터 개체가 서명자 개체 및 검증자 개체를 포함하는 사용자 개체의 공개된 식별 정보에 기초하여, 제 1 비밀키를 생성하여 상기 사용자 개체에게 할당하는 단계; (c) 상기 사용자 개체가 자신의 공개키 및 제 2 비밀키를 생성하는 단계; (d) 상기 서명자 개체가 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체에게 전송하는 단계; 및 (e) 상기 검증자 개체가 상기 서명자 개체로부터 수신한 서명을 상기 서명자 개체의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 단계;를 포함하되, 상기 마스터 키, 제 1 비밀키, 및 상기 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용되는 것을 특징으로 한다.

발명의 효과

[0013] 본 발명은 높은 안전성 및 효율성을 갖는 전자 서명 시스템 및 방법을 제공한다.

[0014] 구체적으로, 본 발명은 인증서를 사용하지 않기 때문에 인증서 기반 전자 서명 시스템이 갖는 인증서 관리 비용 및 안전성 문제가 없으며, 공인 인증 기관을 전적으로 신뢰하지 않아도 된다.

[0015] 또한, 본 발명은 최악의 경우에도 어려운 수학적 난제인 래티스 상에서의 SIS 문제의 어려움에 기반을 두고 있으므로, 안전성이 높다.

[0016] 래티스 기반으로 설계되었기 때문에 래티스 기반 암호 시스템이 갖는 연산 효율성, 높은 안전성, 양자 컴퓨터 분석에 대한 내성의 장점을 나타낸다. 따라서 현재까지 알려진 무인증서 기반, 즉, 인증서 비사용 전자 서명 기법 중 가장 높은 안전성을 갖는다.

도면의 간단한 설명

[0017] 도 1은 본 발명의 일실시예에 따른 전자 서명 시스템의 구조를 도시함.

도 2는 도 1의 시스템의 구조를 다른 관점에서 도시함.

도 3은 본 발명의 일실시예에 따른 전자 방법의 흐름을 도시함.

도 4은 본 발명의 일실시예에 따른 시스템 설정 방법의 흐름을 도시함.

도 5는 본 발명의 일실시예에 따른 제 1 비밀키 발급 방법의 흐름을 도시함.

도 6은 본 발명의 일실시예에 따른 공개키 및 제 2 비밀키 생성 방법의 흐름을 도시함.

도 7은 본 발명의 일실시예에 따른 서명 방법의 흐름을 도시함.

도 8은 본 발명의 일실시예에 따른 검증 방법의 흐름을 도시함.

도 9는 본 발명의 일실시예에 따른 트랩도어 생성 방법의 흐름을 도시함.

도 10은 본 발명의 일실시예에 따른 트랩도어 확장 방법의 흐름을 도시함.

도 11은 본 발명의 일실시예에 따른 가우시안 샘플링 방법의 흐름을 도시함.

발명을 실시하기 위한 구체적인 내용

- [0018] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0019] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.
- [0020] 전술한 바와 같이, 무인증서 기반, 즉, 인증서 비사용 전자 서명 시스템은 종래의 인증서 기반 전자 서명 시스템이 갖고 있는 인증서를 유지, 관리하는 비용이 크다는 문제, 유출이나 조작 등 인증서 관련 사고가 발생한다는 문제 등을 해결하였다.
- [0021] 이제까지 이러한 무인증서 기반 전자 서명 시스템은 인수 분해, 이산 대수, 페어링(pairing)을 기반으로 설계되었다. 하지만 풀기 어렵다고 여겨지던 이러한 문제들은 최근 양자 알고리즘의 발전으로 더 이상 안전성을 보장할 수 없다.
- [0022] 이에 따라 양자 분석에 내성이 있는 것으로 알려진 래티스 기반 암호 시스템에 대한 관심이 점차 높아지고 있다.
- [0023] 기존의 래티스 기반의 암호 시스템의 경우 안전성 증명이 결여되어 많은 문제점이 있었으나, 최근 최악의 경우(worst-case) 문제에 기반하여 증명되는 래티스 기반의 암호 시스템들이 등장하고 있다. 기존의 암호 시스템이 평균적인 경우(average-case) 문제에 기반하여 설계된 점을 비추어 볼 때 매우 높은 안전성을 제공한다는 것을 알 수 있다.
- [0024] 이 뿐만 아니라 래티스 기반 암호 시스템은 종래 기법에 비해 매우 효율적이다. 인수 분해, 이산 대수, 페어링 등의 연산 복잡도는 $O(n^3)$ 이지만, 래티스에서의 연산 복잡도는 $O(n^2)$ 이다.
- [0025] 따라서 본 발명은 래티스 환경에서 안전성 증명이 가능한 무인증서 기반 전자 서명 기술을 제공하고자 한다.
- [0026] 도 1 및 도 2는 본 발명의 일실시예에 따른 전자 서명 시스템의 구조를 도시하고 있다.
- [0027] 본 발명의 일실시예에 따른 전자 서명 시스템(10)에는 네트워크를 통해 연결된 서명자(X), 검증자(Y), 및 제3자(Z)를 포함하는 사용자와 마스터(M) 등의 개체(entity)가 참여한다.
- [0028] 사용자 중, 서명자(X)는 메시지에 서명하여 검증자(Y)에게 전송하는 개체이며, 검증자(Y)는 서명자(X)로부터 수신한 서명을 검증하는 개체이다. 이때, 검증이란 전술한 바와 같이, 수신한 서명이 서명자(X)에 의해 생성된, 유효한 서명임을 확인하는 것이다.
- [0029] 서명을 생성하여 전송한 서명자(X)와 서명을 수신한 검증자(Y) 이외의 다른 사용자인 제3자(Z)는 해당 서명을 검증할 수 없다.
- [0030] 도면에는 각각 하나의 서명자(X), 검증자(Y), 및 제3자(Z) 개체만이 도시되어 있고, 이들이 서로 다른 개체인 것처럼 도시되어 있다. 그러나 서명자(X), 검증자(Y), 및 제3자(Z)는 모두 사용자 개체로서, 이들의 역할은 고정적이지 않으며, 전자 서명 시스템(10)에 참여하는 사용자 개체의 수에도 제한이 없을 것을 쉽게 이해할 수 있을 것이다. 또한, 이들 각 사용자 개체는 사용자가 사용하는 기기나 소프트웨어 등의 실체를 의미할 수 있다. 이는 통용되는 개념이므로 본 명세서에서 이에 대한 명시적 기재는 생략한다. 예를 들어, 이하 기술에서 사용자라 함은 서명자(X), 검증자(Y), 제3자(Z) 등의 사용자 개체를 나타낼 수 있다.

- [0031] 전술한 바와 같이, 본 발명의 일실시예에 따른 전자 서명 시스템(10)에는 각 사용자의 제 1 비밀키를 생성하여 해당 사용자에게 배포하는 마스터(M) 개체가 참여한다. 여기에서 마스터(M)는 키 생성 기관(Key Generation Center), 인증 기관(Certification Authority) 등 다양한 이름으로 불릴 수 있으며, 전자 서명 시스템(10)이 원활하게 운영되도록 관리하는 기관들을 통칭하는 개념으로 사용하였다.
- [0032] 종래의 공개키 기반 환경에서는 전술한 바와 같이, 마스터(M)가 각 사용자에게 공인 인증서를 발급해야 하지만, 본 발명의 일실시예에 따른 전자 서명 시스템(10)은 마스터(M)가 사용자(X, Y, Z)의 공개된 식별정보(ID)를 사용하여 사용자(X, Y, Z)에게 제 1 비밀키를 발급해줄 뿐이기 때문에, 마스터(M)와 사용자(X, Y, Z) 모두 공인 인증서를 관리할 필요가 없어, 효율적이고 안전성이 높다. 공인 인증서 관리에 따른 비용이나 부하가 없고, 공인 인증서 도용 등의 보안 위험이 없기 때문이다. 또한 후술하는 바와 같이, 사용자(X, Y, Z)는 마스터(M)에게 노출되지 않는 또다른 비밀키인 제 2 비밀키를 스스로 생성하므로, 안정성이 더욱 높다.
- [0033] 종래와 달리 공인 인증서를 사용하지 않으나, 본 발명의 일실시예에 따른 전자 서명 시스템(10)은 공개키 기반 방식이다.
- [0034] 공개키 기반 방식은 서로 대응되는 키 쌍을 생성하여, 이중 공개키는 암호화 또는 서명 확인용으로, 비밀키는 복호화 또는 서명용으로 사용하는 비대칭형 암호화 또는 서명 방식이다.
- [0035] 해당 메시지를 보낸 사용자가 누구인가를 증명하기 위한 것이므로, 어떤 메시지에 서명할 때는 비밀키로 서명하고 대응되는 공개키로 이를 확인한다. 즉, 비밀키는 문서에 찍는 도장의 기능을 한다.
- [0036] 한편 전술한 바와 같이, 암호화 및 전자 서명 시스템은 특정 정보가 없으면 풀기 어려운 수학적 난제에 기반하여 구축되는데, 본 발명의 일실시예에 따른 전자 서명 시스템(10)은 래티스(lattice) 상에서의 SIS(small integer solution) 문제와 LWE(learning with errors) 문제의 어려움에 기반하고 있다.
- [0037] 특정 정보가 없으면 풀기 어렵다는 것은 연산이 일방향성이라는 의미이다. 일방향성은 연산은 쉬워도 연산의 역을 구하는 것은 어렵으며, 그러나 특정 정보를 갖고 있는 경우에는 쉽게 역을 구할 수 있음을 나타낸다. 이러한 특정 정보를 트랩도어(trapdoor)라고 부른다.
- [0038] 예를 들어, 공개키 기반 환경에서 암호화 연산은 암호화는 쉽지만 그 역변환인 복호화는 암호화시 사용한 암호화 키에 대응되는 복호화 키가 없는 경우 매우 어려운 일방향성 연산을 사용하여 수행된다. 즉, 암호화시 사용한 암호화 키에 대응되는 복호화 키가 트랩도어이다. 일단 메시지를 암호화하고 나면 암호화 키가 있어도 이를 복호화할 수 없으므로, 암호화 키는 공개되어도 된다. 반면, 트랩도어 또는 복호화 키는 공개되면 안된다. 따라서, 공개키 기반 환경에서 트랩도어를 생성하는 데 사용된 정보는 공개키로서 공개되고, 트랩도어는 비밀키로서 공개되지 않는다.
- [0039] 래티스에 기반한 SIS 문제와 LWE 문제는 전술한 바와 같이 최악의 경우에서도 풀기 어려운 수학적 난제이므로, 최근 암호화 및 전자 서명 시스템에 이를 적용하려는 시도가 많이 이루어지고 있는 추세이다. 래티스는 공간에서 2개의 기저의 선형 결합을 통해 만들어지는 모든 점의 집합이다. 기저를 모르면 제3자(Z)가 어떤 점을 알아내었다고 해도 해당 점으로부터 가장 가까운 래티스 점이 어딘지 알아내는 것은 매우 어렵다는 장점이 있다. 또한, 기저를 모르면 제3자(Z)는 특정 길이 이내의 래티스 점을 찾는 것이 매우 어렵다는 장점이 있다.
- [0040] 따라서 본 발명의 일실시예에 따른 전자 서명 시스템(10)은 자신의 마스터 키 및 사용자 개체(X, Y, Z)의 공개된 식별 정보를 사용하여, 제 1 비밀키를 생성하여 사용자 개체(X, Y, Z)에게 할당하는 마스터 개체(M) 및 자신의 공개키 및 제 2 비밀키를 생성하는 사용자 개체(X, Y, Z)를 포함하며, 사용자 개체(X, Y, Z)는 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체(Y)에게 전송하는 서명자 개체(X) 및 서명자 개체(X)로부터 수신한 서명을 서명자 개체(X)의 공개된 식별 정보, 및 공개키를 사용하여 검증하는 검증자 개체(Y)를 포함한다. 이때 마스터 키, 제 1 비밀키, 및 제 2 비밀키 생성에 래티스(lattice) 상에서의 SIS(small integer solution) 문제의 어려움에 기반하여 생성한 트랩도어(trapdoor)가 사용된다.
- [0041] 도 2에 도시되어 있는 바와 같이, 이를 위해 마스터(M) 및 각 사용자(A, B, C)는 균일한 분포(uniform distribution)에서 선택된 임의의 행렬로부터 생성되는 래티스의 짧은 기저(short basis)인 트랩도어를 생성하는 트랩도어 생성부(L100), 트랩도어의 래티스를 또다른 래티스로 확장하고 또다른 래티스의 짧은 기저인 확장된 트랩도어를 생성하는 트랩도어 확장부(L200), 및 트랩도어의 가우시안 분포로부터 벡터를 샘플링하는 샘플링부(L300)를 포함하는 라이브러리(L)를 포함하거나 이에 연결될 수 있다.
- [0042] 다시 말해 라이브러리(L)는 래티스 상에서의 SIS 문제의 어려움에 기반하여 트랩도어를 생성, 확장, 및 사용하

기 위한 기본 기능을 본 발명의 일실시예에 따른 전자 서명 시스템(10)의 각 개체(M, X, Y, Z)에 제공한다.

- [0043] 이들 트랩도어 생성부(L100), 트랩도어 확장부(L200), 및 샘플링부(L300)가 수행하는 동작은 도 9 내지 도 11에 도시되어 있다.
- [0044] 마스터(M)의 시스템 설정부(100)는 보안 파라미터를 설정하고, 일련의 해시 함수를 선택하며, 트랩도어 생성부(L100)를 사용하여 생성한 트랩도어를 자신의 비밀키인 마스터 키로 사용하고, 함께 생성된 대응 행렬을 해시 함수와 함께 공개한다.
- [0045] 마스터의 키 발급부(200)는 트랩도어 확장부(L200)를 사용하여 사용자의 공개된 식별 정보로부터 생성된 확장된 트랩도어를 사용자의 제 1 비밀키로 발급한다.
- [0046] 사용자의 키 생성부(300)는 트랩도어 생성부(L100)를 사용하여 생성한 트랩도어를 자신의 제 2 비밀키로 사용하고, 함께 생성된 대응 행렬을 자신의 공개키로 사용한다.
- [0047] 그러므로, 전술한 바와 같이, 사용자의 제 2 비밀키는 마스터(M)와 전혀 무관하게 사용자 자신이 생성하므로, 본 발명의 일실시예에 따른 전자 서명 시스템(10)은 마스터(M)를 전적으로 신뢰하지 않아도 되는 장점을 가진다.
- [0048] 사용자 중 서명자(X)는 서명부(400)를 사용하여 자신의 제 1 비밀키 및 공개키를 사용하여 제 1 서명을 생성하고, 자신의 제 2 비밀키 및 공개키를 사용하여 제 2 서명을 생성하여, 제 1 서명과 제 2 서명을 사용하여 서명을 생성하며, 이때 트랩도어 생성부(L100), 트랩도어 확장부(L200), 샘플링부(L300)를 사용한다.
- [0049] 서명자(X)는 제 1 비밀키의 가우시안 분포로부터 벡터를 샘플링하여 제 1 서명을 생성하며, 제 2 비밀키의 래티스를 공개키를 사용하여 확장된 래티스로 확장하고 확장된 래티스의 짧은 기저인 확장된 트랩도어를 생성하여, 확장된 트랩도어의 가우시안 분포로부터 벡터를 샘플링하여 제 2 서명을 생성한다.
- [0050] 사용자 중 검증자(Y)는 검증부(500)를 사용하여 서명자(X)의 공개된 식별 정보, 및 공개키를 사용하여 제 1 서명 및 제 2 서명을 각각 검증하고, 제 1 서명 및 제 2 서명이 모두 정당한 서명이라고 판단되었을 때 수신한 서명을 받아들이고 그렇지 않으면 기각한다.
- [0051] 따라서 본 발명의 일실시예에 따른 전자 서명 시스템(10)의 각 개체는 도 1의 각 개체에 연결된 키 저장소에 도시되어 있는 것과 같은 키들을 가진다.
- [0052] 마스터(M)는 마스터 키(msk)를 가지며, 이는 외부에 공개되어서는 안되는 비밀키이다. 반면 대응하는 행렬은 해시 함수들과 함께 공개 파라미터 (A, H_1, H_2, H_3, H_4) 로서 공개된다.
- [0053] 각 사용자(X, Y, Z)의 식별 정보와 공개키도 공개된다. 반면 각 사용자(X, Y, Z)의 제 1 비밀키 및 제 2 비밀키는 공개되지 않는다.
- [0054] 도 3는 본 발명의 일실시예에 따른 전자 방법의 흐름을 도시하고 있다.
- [0055] 마스터 개체(M)가 보안 파라미터를 설정하고 자신의 마스터 키를 생성한다(S100).
- [0056] 마스터 개체(M)가 서명자 개체(X) 및 검증자 개체(Y)를 포함하는 사용자 개체(X, Y, Z)의 공개된 식별 정보에 기초하여, 제 1 비밀키를 생성하여 사용자 개체(X, Y, Z)에게 할당한다(S200).
- [0057] 사용자 개체(X, Y, Z)가 자신의 공개키 및 제 2 비밀키를 생성한다(S300).
- [0058] 서명자 개체(X)가 자신의 공개키, 제 1 비밀키, 및 제 2 비밀키를 사용하여 생성한 서명을 검증자 개체(Y)에게 전송한다(S400).
- [0059] 검증자 개체(Y)가 서명자 개체(X)로부터 수신한 서명을 서명자 개체(X)의 공개된 식별 정보, 및 공개키를 사용하여 검증한다(S500).
- [0060] 이하 도 4 내지 도 8을 통해 각 단계를 좀더 자세히 설명한다.
- [0061] 도 4은 본 발명의 일실시예에 따른 시스템 설정 방법의 흐름을 도시하고 있다.
- [0062] 마스터(M)가 해시 함수를 선택한다(S110).
- [0063] 트랩도어 생성부(L100)를 사용하여, 행렬 A 및 대응하는 트랩도어를 생성한다(S120). 트랩도어는 행렬 A에 기초

하여 생성되는 래티스의 짧은 기저(short basis)이다.

[0064] 공개 파라미터로 행렬 A 및 해시 함수를 출력한다(S130).

[0065] 생성된 트랩도어를 마스터 키로 사용한다(S140).

[0066] 다시 말해, 시스템 설정 단계(S100)는 시스템에서 사용할 다양한 파라미터를 정하고 키를 발급하는 기관인 마스터(M)의 마스터 키를 정하는 단계이다.

[0067] 먼저, 다음과 같은 성질을 만족하는 충돌내성 해시 함수(collision-resistant hash function)를 선택한다.

[0068] $H_1: \{0,1\}^* \rightarrow \mathbb{Z}_q^{n \times m}$

[0069] $H_2: \mathbb{Z}_q^{n \times m} \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^n$

[0070] $H_3: \{0,1\}^* \times \mathbb{Z}^m \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^{n \times m}$

[0071] $H_4: \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^n$

[0072] 그리고 마스터(M)는 $(A \in \mathbb{Z}_q^{n \times m}, S_A \in \mathbb{Z}_q^{m \times m}) \leftarrow GBasis(1^n, 1^m, q)$ 을 생성하여 공개 파라미터 $param = (A, H_1, H_2, H_3, H_4)$ 와 마스터키 $msk = S_A$ 를 정한다.

[0073] 도 5는 본 발명의 일실시예에 따른 제 1 비밀키 발급 방법의 흐름을 도시하고 있다.

[0074] 마스터(M)가 사용자 ID를 입력받는다(S210).

[0075] 사용자 ID를 해시하여 행렬 B를 산출한다(S220).

[0076] 트랩도어 확장부(L200)를 사용하여, 마스터 키를 행렬 B에 기초하여 확장한다(S230).

[0077] 확장된 마스터 키를 ID에 대한 제 1 비밀키로서 사용자(X, Y, Z)에게 전달한다(S240).

[0078] 다시 말해, 제 1 비밀키 발급 단계(S200)는 마스터(M)가 자신의 마스터키를 이용해 주어진 ID에 대한 제 1 비밀키를 발급해주는 단계이다.

[0079] 마스터(M)는 $B \in \mathbb{Z}_q^{n \times m} = H_1(ID)$ 을 계산하고, $S_{AllB} \in \mathbb{Z}^{m \times m} \leftarrow BasisDel(A, B, S_A)$ 알고리즘을 실행한 후, S_{AllB} 을 ID에 대한 제 1 비밀키로 정하고, 안전한 방법으로 해당 ID의 서명자에게 키를 전달한다.

[0080] 도 6은 본 발명의 일실시예에 따른 공개키 및 제 2 비밀키 생성 방법의 흐름을 도시하고 있다.

[0081] 후술하는 바와 같이, 사용자(X, Y, Z)는 마스터(M)와 완전히 독립적으로 자신의 공개키 및 제 2 비밀키를 생성한다. 그러므로 전술한 바와 같이 사용자(X, Y, Z)는 마스터(M)를 전적으로 신뢰하지 않아도 되며 제 2 비밀키가 마스터(M)에게 알려지지 않으므로, 본 발명의 일실시예에 따른 전자 서명 방법은 안전성이 높다.

[0082] 사용자(X, Y, Z)가 트랩도어 생성부(L100)를 사용하여, 행렬 C 및 대응하는 트랩도어를 생성한다(S310). 트랩도어는 행렬 C에 기초하여 생성되는 래티스의 짧은 기저(short basis)이다.

[0083] 행렬 C를 자신의 공개키로 사용한다(S320).

[0084] 생성된 트랩도어를 제 2 비밀키로 사용한다(S330).

[0085] 다시 말해, 공개키 및 제 2 비밀키 생성 단계(S300)는 ID에 해당하는 사용자(X, Y, Z)가 자신의 공개키와 제 2 비밀키를 생성하는 단계이다.

[0086] 사용자(X, Y, Z)는 $(C \in \mathbb{Z}_q^{n \times m}, S_C \in \mathbb{Z}_q^{m \times m}) \leftarrow GBasis(1^n, 1^m, q)$ 알고리즘을 실행한 후, C를 자신의 ID에 대한 공개키로 s_k 를 ID에 대한 제 2 비밀키로 출력한다.

[0087] 주목할 점은 전술한 바와 같이 마스터(M)는 사용자(X, Y, Z)의 제 1 비밀키는 직접 생성하였으므로 알 수 있으나, 제 2 비밀키는 사용자(X, Y, Z)가 직접 생성하였기 때문에 알 수 없다. 따라서 아무리 마스터(M)라고 해도 제 2 비밀키를 모르는 이상 서명자(X)의 서명을 생성할 수 없다. 이와 같이 본 발명의 일실시예에 따른 인증서 비사용 전자 서명 시스템(10)은 마스터(M)를 전적으로 신뢰하지 않아도 되는 장점을 가진다.

- [0088] 도 7은 본 발명의 일실시예에 따른 서명 방법의 흐름을 도시하고 있다.
- [0089] 서명자(X)가 메시지 m을 입력받는다(S410).
- [0090] 샘플링부(L300)를 사용하여 자신의 비밀키로부터 제 1 서명을 생성한다(S420). 이때 자신의 공개키를 메시지, 임의의 문자열과 함께 해시한 값이 사용된다.
- [0091] 트랩도어 확장부(L200)를 사용하여 자신의 제 2 비밀키를 확장한다(S440).
- [0092] 샘플링부(L300)를 사용하여 자신의 확장된 제 2 비밀키로부터 제 2 서명을 생성한다(S450). 이때 메시지와 또 다른 임의의 문자열을 함께 해시한 값이 사용된다.
- [0093] 제 1 서명과 제 2 서명을 사용하여 서명을 생성한다(S460).
- [0094] 생성한 서명을 검증자(Y)에게로 발송한다(S470).
- [0095] 다시 말해, 서명 단계(S400)는 ID에 해당하는 서명자(X)가 자신의 제 1 비밀키 s_{sk1} 와 제 2 비밀키 s_c 를 사용하여 주어진 메시지에 대한 서명을 생성하는 단계이다. 서명자(X)는 메시지 m에 대한 서명을 다음과 같이 생성한다.
- [0096] 먼저, $\sigma_1 \in \mathbb{Z}^m \leftarrow \text{SampleD}(\mathcal{S}_{sk1}, H_2(C, m, r_1), s)$ 을 계산한다. 여기서 r_1 은 서명자(X)가 선택한 임의의 문자열이며, 파라미터 s는 *SampleD* 알고리즘에 쓰이는 파라미터다.
- [0097] 그 후, $D \in \mathbb{Z}_q^{n \times m} = H_3(ID, \sigma_1, m, r_1)$ 을 계산하여, 자신의 제 2 비밀키 s_c 을 이용해 $\mathcal{S}_{sk2} \leftarrow \text{BasisDel}(C, D, \mathcal{S}_C)$ 알고리즘을 실행한 후, 생성한 $\mathcal{S}_{sk2}$ 을 사용하여 $\sigma_2 \in \mathbb{Z}^m \leftarrow \text{SampleD}(\mathcal{S}_{sk2}, H_4(m, r_2), s)$ 을 계산한다. 여기서 r_2 역시 서명자(X)가 선택한 임의의 문자열이며, 동일한 파라미터 s를 사용한다.
- [0098] 최종적으로 전자 서명은 $\sigma = (D, C, m, \sigma_1, \sigma_2, r_1, r_2)$ 으로 구성된다.
- [0099] 도 8은 본 발명의 일실시예에 따른 검증 방법의 흐름을 도시하고 있다.
- [0100] 검증자(Y)가 서명자(X)의 서명을 입력받는다(S510).
- [0101] 서명자의 ID 및 공개키를 사용하여 제 1 서명에 대해 도시되어 있는 바와 같은 기지정된 조건을 만족하는지 판단한다(S520).
- [0102] 서명자의 ID 및 공개키를 사용하여 제 1 서명에 대해 도시되어 있는 바와 같은 기지정된 조건을 만족하는지 판단한다(S530).
- [0103] 제 1 서명 조건 및 제 2 서명 조건을 모두 만족할 경우 서명은 검증되며, 하나라도 만족하지 못할 경우 서명은 기각된다(S540).
- [0104] 다시 말해, 검증 단계(S500)는 ID에 해당하는 서명자(X)가 보내온 서명을 검증자(Y)가 검증하는 단계이다.
- [0105] 주어진 서명 $\sigma = (D, m, \sigma_1, \sigma_2, r_1, r_2, C)$ 은 다음과 같은 조건을 만족할 때 정당하다.
- [0106] $\sigma_1 \neq 0, \|\sigma_1\| \leq s \sqrt{2m}$ 그리고 $[A \| H_1(ID)]\sigma_1 = H_2(C, m, r_1)$
- [0107] $\sigma_2 \neq 0, \|\sigma_2\| \leq s \sqrt{2m}$ 그리고 $[C \| H_3(ID, \sigma_1, m, r_1)]\sigma_2 = H_4(m, r_2)$
- [0108] 이하 도 9 내지 도 11을 통해 본 발명의 일실시예에 따른 전자 서명 방법이 키 발급 및 서명에 사용하기 위해 래티스 상에서의 SIS 문제의 어려움에 기반하여 트랩도어를 생성, 확장, 및 샘플링하는 방법을 구체적으로 설명한다.
- [0109] 본 발명은 m차원의 풀-랭크(full-rank) 정수 래티스를 사용하며, 이것은 유한개의 지수(finite index)를 가지는 $\mathbb{Z}^m$ 의 이산 가산 서브그룹(discrete additive subgroup)이다. 즉, 몫군(quotient group) $\mathbb{Z}^m/A\mathbb{Z}$ 이 유한하다.
- [0110] 하나의 래티스 $A \subseteq \mathbb{Z}^n$ 은 m개의 선형 독립 기저(basis) 벡터 $B = \{b_1, \dots, b_m\} \subset \mathbb{Z}^n$ 의 모든 정수 선형 결합의 집합과 동

일하게 정의된다.

$$A = L(B) = \{Bc = \sum_{i \in \{1, \dots, m\}} c_i b_i : c \in \mathbb{Z}^m\}$$

여기서 $m \geq 2$ 일 경우, 동일한 래티스를 생성하는 기저들은 무수히 많다.

본 발명은 다음과 같은 정수 래티스의 특정한 형태를 사용한다. 여기서 $n \geq 1$ 과 $\sigma \geq 2$ 는 정수이며, 차원 n 은 본 발명에서 사용되는 시큐리티 파라미터이고, 모든 다른 파라미터들은 n 의 함수들로 내포되었다고 가정한다. 여기서 m 차원 하드 래티스는 패리티 검사 행렬(parity check matrix) $A \in \mathbb{Z}_q^{n \times m}$ 에 의해 생성되며, 다음과 같이 정의된다.

$$A^\perp(A) = \{x \in \mathbb{Z}^m : Ax = \sum_{i \in \{1, \dots, m\}} x_i \cdot a_i = 0 \in \mathbb{Z}_q^n\} \subseteq \mathbb{Z}^m$$

어떤 y 에 대해서, 패리티 검사 행렬 $A \in \mathbb{Z}_q^{n \times m}$ 에 의해 생성되는 코셋(coset)은 다음과 같이 정의된다.

$$A_y^\perp(A) = \{x \in \mathbb{Z}^m : Ax = y \in \mathbb{Z}_q^n\}$$

어떤 고정된 상수 $\sigma > 1$ 과 어떤 $m \geq \Omega(n \log \sigma)$ 에 대해, 균등하게 랜덤한(uniformly random) $A \in \mathbb{Z}_q^{n \times m}$ 의 열 벡터는 $\mathbb{Z}_q^n$ 상의 모두를 ($2^{-\Omega(n)} = \text{negl}(n)$ 확률을 제외하고) 생성할 수 있다. 따라서 본 발명은 균등하게 랜덤한 A 를 사용한다.

본 발명이 기반하는 하드 래티스의 SIS(short integer solution) 문제는 평균적인 경우에 어려운 문제(average-case hardness problems)에 속해있으며, Miklos Ajtai는 이 문제를 최악의 경우에 어려운 문제(worst-case hardness problems)로 커넥션(connection)하는 방법을 발견하였다.

래티스의 SIS 문제는 (ℓ_2 norm에서의 $SIS_{\beta, \sigma}$ 문제)는 어떤 $m = \text{poly}(n)$ 에 대해 균등하게 랜덤한 행렬 $A \in \mathbb{Z}_q^{n \times m}$ 을 입력으로 받아 $\|v\|_2 \leq \beta$ 와 $Av = 0 \in \mathbb{Z}_q^n$ (즉, $v \in A^\perp(A)$) 을 만족하는 영이 아닌(nonzero) 정수 벡터 $v \in \mathbb{Z}^m$ 을 찾는 것으로 정의된다.

도 9는 본 발명의 일실시예에 따른 트랩도어 생성 방법의 흐름을 도시하고 있다.

보안 파라미터를 입력받는다(SL110).

균등한 분포에서 임의의 행렬을 선택한다(SL120).

행렬로부터 래티스를 생성한다(SL130).

래티스의 짧은 기저를 산출한다(SL140).

산출된 짧은 기저를 트랩도어로서 행렬과 함께 출력한다(SL150).

다시 말해, 트랩도어 생성부(L100)는 균등한 분포에서 임의로 선택된 매트릭스 $A \in \mathbb{Z}_q^{n \times m}$ 와 이러한 매트릭스로 생성되는 래티스 $A^\perp(A)$ 의 짧은 기저 S 를 생성하는 알고리즘 $(A \in \mathbb{Z}_q^{n \times m}, S \in \mathbb{Z}_q^{m \times m}) \leftarrow \text{GBasis}(1^n, 1^m, q)$ 을 제공한다.

임의의 래티스의 짧은 기저를 찾는 문제는 결국 SIS 문제이기 때문에 알고리즘 $\text{GBasis}(1^n, 1^m, q)$ 은 임의의 래티스와 그것의 트랩도어(SIS 문제에 대한) 즉, 짧은 기저를 생성하게 된다.

이때 파라미터들은 $m = O(n \log \sigma)$ 을 만족하며, 기저 S 의 길이는 $\|S\| = O(\sqrt{n \log \sigma})$ 만큼 짧다.

도 10은 본 발명의 일실시예에 따른 트랩도어 확장 방법의 흐름을 도시하고 있다.

래티스를 나타내는 행렬, 즉, 래티스를 생성하는 행렬 및 래티스의 짧은 기저를 입력받는다(SL210).

균등한 분포에서 임의의 행렬을 선택한다(SL220).

확장된 래티스를 생성한다(SL230).

확장된 래티스의 짧은 기저를 산출한다(SL240).

- [0134] 산출된 짧은 기저를 확장된 트랩도어로 출력한다(SL250).
- [0135] 다시 말해, 트랩도어 확장부(L200)는 래티스의 짧은 기저를 위임하는 알고리즘을 제공한다.
- [0136] 래티스 $A^\perp(A)$ 와 그것의 짧은 기저 $\mathcal{S}_A$, $B \in \mathbb{Z}_q^{n \times m}$ 를 입력값으로 하는 알고리즘 $BasisDel(A, B, \mathcal{S}_A)$ 은 출력값으로 래티스 $A^\perp(AB)$ 와 그것의 짧은 기저 $\mathcal{S}_{AB}$ 을 출력한다.
- [0137] 도 11은 본 발명의 일실시예에 따른 가우시안 샘플링 방법의 흐름을 도시하고 있다.
- [0138] 래티스 의 짧은 기저, 샘플링 입력값, 샘플링 파라미터를 입력받는한다(SL310).
- [0139] 가우시안 분포를 생성한다(SL320).
- [0140] 분포로부터 벡터를 샘플링한다(SL330).
- [0141] 벡터를 출력한다(SL340).
- [0142] 다시 말해, 샘플링부(L300)는 래티스의 가우시안 분포로부터 벡터를 샘플링하는 알고리즘을 제공한다.
- [0143] 어떤 $\varepsilon > 0$ 과 차원 $m \geq 1$ 에 대해, 가우시안 함수(Gaussian function) $\rho_\varepsilon: \mathbb{R}^m \rightarrow [0, 1]$ 는 $\rho_\varepsilon(\mathbf{x}) = \exp(-\pi \|\mathbf{x}\|^2 / \varepsilon^2)$ 로 정의된다.
- [0144] 어떤 코셋 $A_0^\perp(A)$ 에 대해, 코셋 상의 (중심이 0인) 이산 가우시안 분포(discrete Gaussian distribution) $\mathcal{D}_{A_0^\perp(A), \varepsilon}$ 는 각각의 $\mathbf{x} \in A_0^\perp(A)$ 에서 $\rho_\varepsilon(\mathbf{x})$ 에 비례하는 확률을 가진다.
- [0145] 래티스에서의 가우시안 분포와 관련되는 성질을 살펴보면 다음과 같다. 여기에서 ε 는 어떤 $A \in \mathbb{Z}_q^{n \times m}$ 에 대한 $A^\perp(A)$ 의 기저를 말하며, $\varepsilon \geq \|\tilde{\mathbf{g}}\| \cdot \omega(\sqrt{\log n})$ 이다.
- [0146] $\Pr_{\mathbf{x} \sim \mathcal{D}_{A_0^\perp(A), \varepsilon}}[\|\mathbf{x}\| > \varepsilon \cdot \sqrt{m}] \leq \text{negl}(n)$
- [0147] $\Pr_{\mathbf{x} \sim \mathcal{D}_{A_0^\perp(A), \varepsilon}}[\mathbf{x} = 0] \leq \text{negl}(n)$
- [0148] 이에 따라 $\text{SampleD}(\mathcal{S}, y, \varepsilon)$ 은 ($\text{negl}(n)$ 통계적인 거리를 가지는) $\mathcal{D}_{A_0^\perp(A), \varepsilon}$ 로부터 샘플링을 할 수 있다.
- [0149] 전술한 바와 같이, 가우시안 샘플링은 서명자(X)가 자신의 비밀키, 즉 확장된 트랩도어에서 서명에 사용할 래티스 벡터를 추출하기 위해 사용한다.
- [0150] 본 발명의 일실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독 가능 매체는 컴퓨터 저장 매체 및 통신 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 반송파와 같은 변조된 데이터 신호의 기타 데이터, 또는 기타 전송 메커니즘을 포함하며, 임의의 정보 전달 매체를 포함한다.
- [0151] 본 발명의 방법 및 시스템은 특정 실시예와 관련하여 설명되었지만, 그것들의 구성 요소 또는 동작의 일부 또는 전부는 범용 하드웨어 아키텍처를 갖는 컴퓨터 시스템을 사용하여 구현될 수 있다. 본 발명의 일실시예의 하나 이상의 구성 요소 또는 동작을 실시하기 위하여 사용될 수 있는 컴퓨터 시스템 아키텍처의 일례를 설명하면, 하드웨어 시스템은 프로세서, 캐쉬, 메모리 및 상술한 기능에 관련된 하나 이상의 소프트웨어 어플리케이션 및 드라이버를 포함할 수 있다.
- [0152] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

[0153] 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

[0154] 10: 전자 서명 시스템

M: 마스터

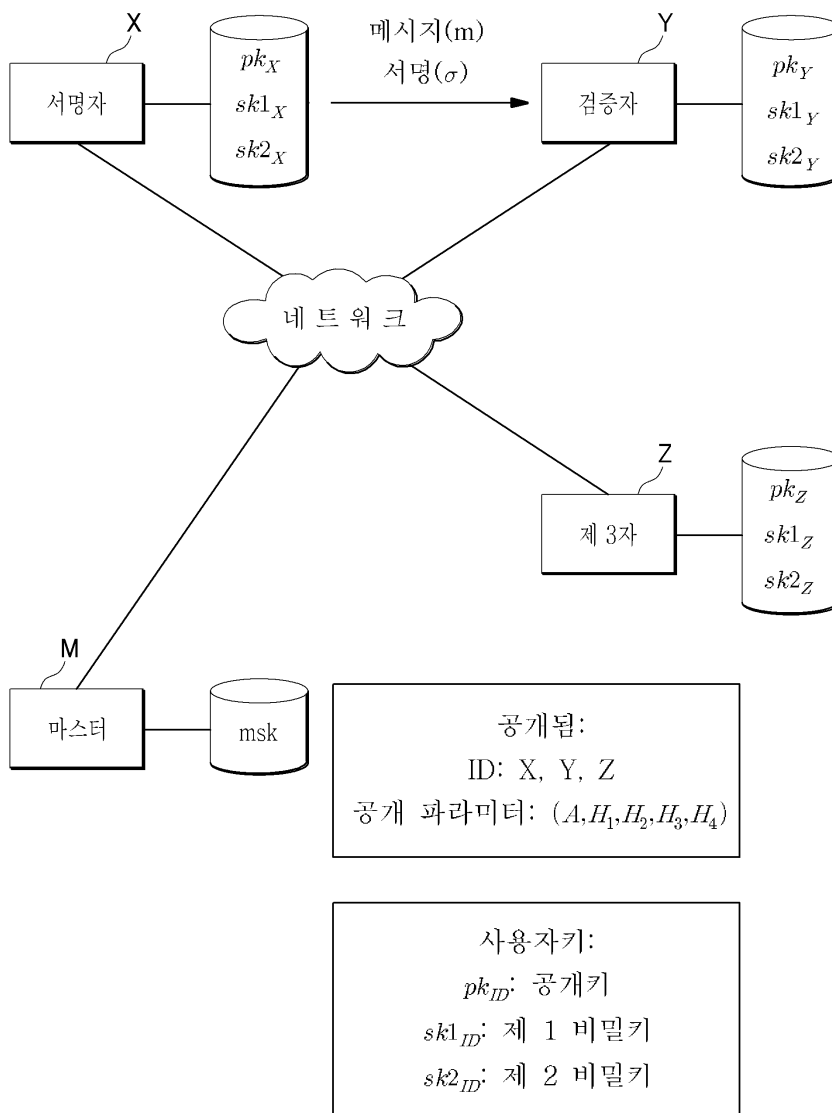
X: 서명자

Y: 검증자

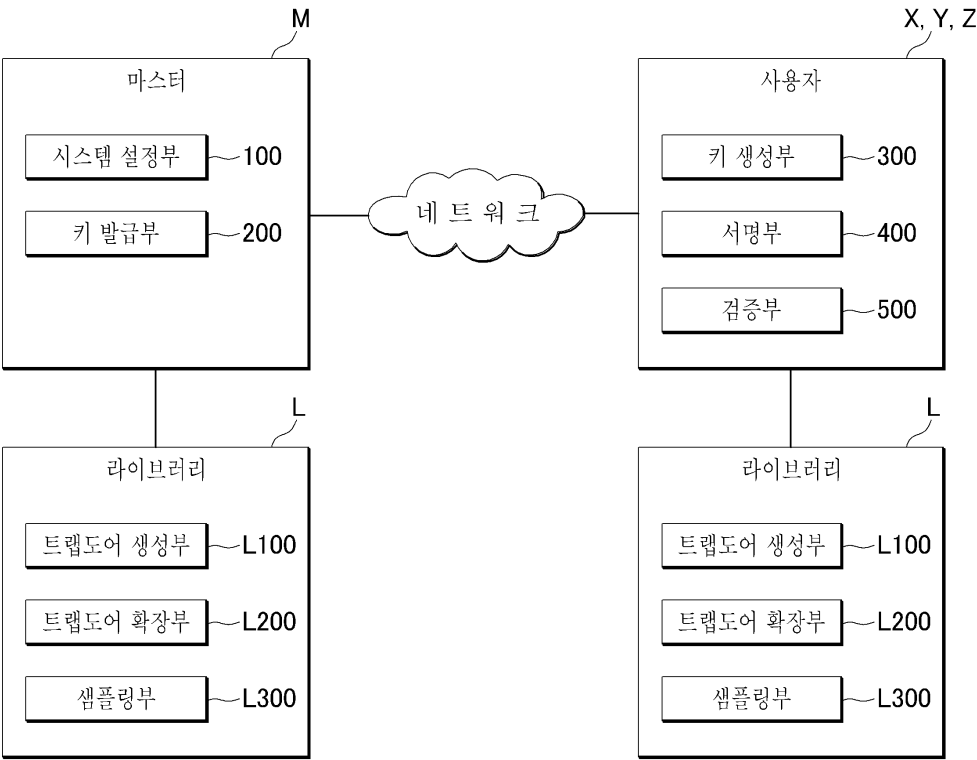
Z: 제3자

도면

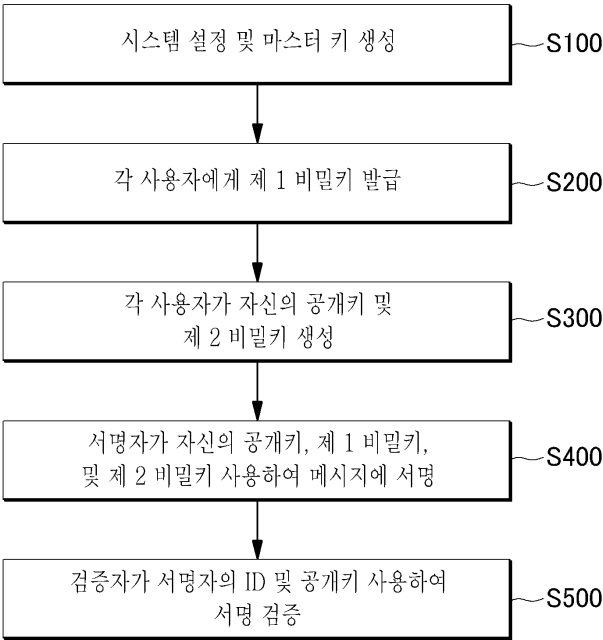
도면1



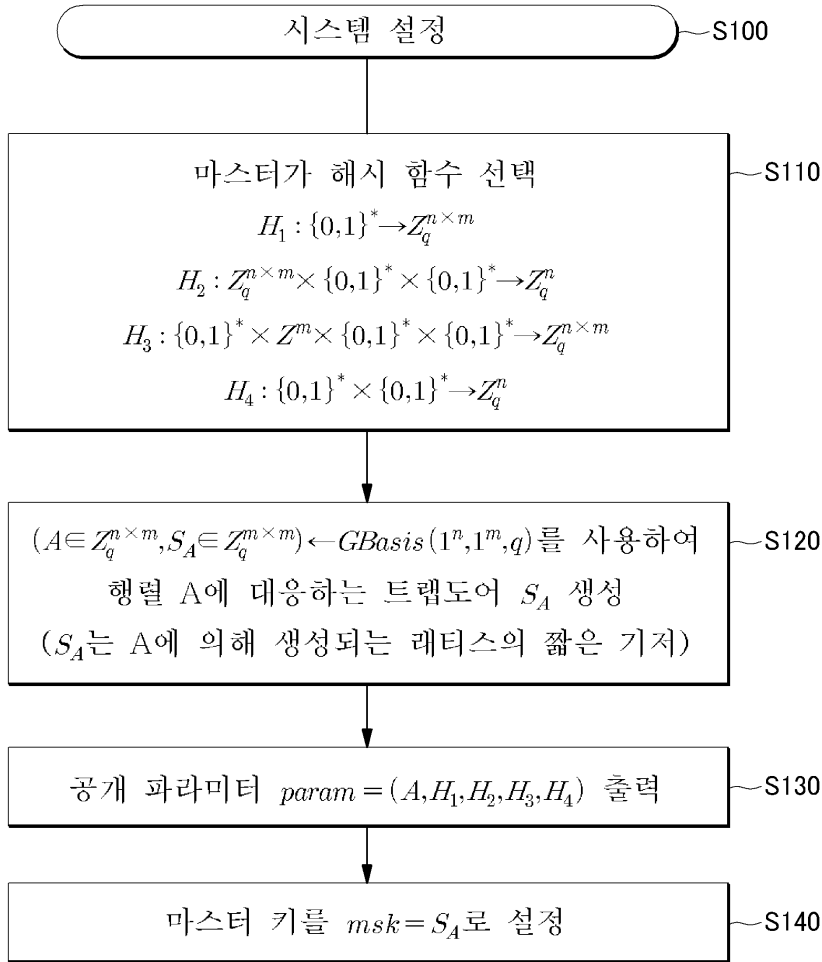
도면2



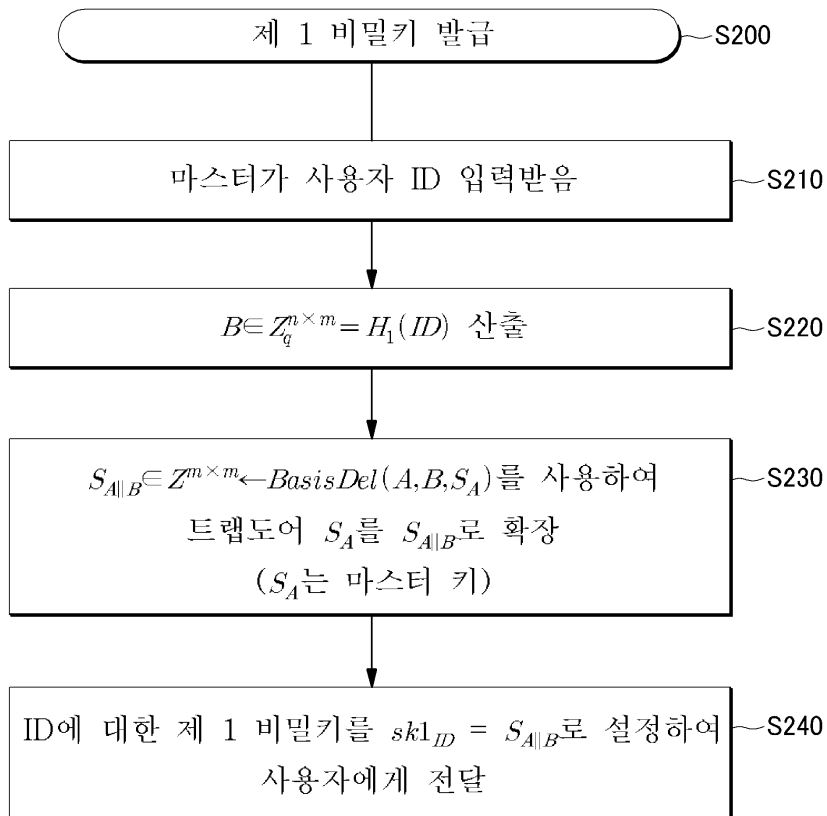
도면3



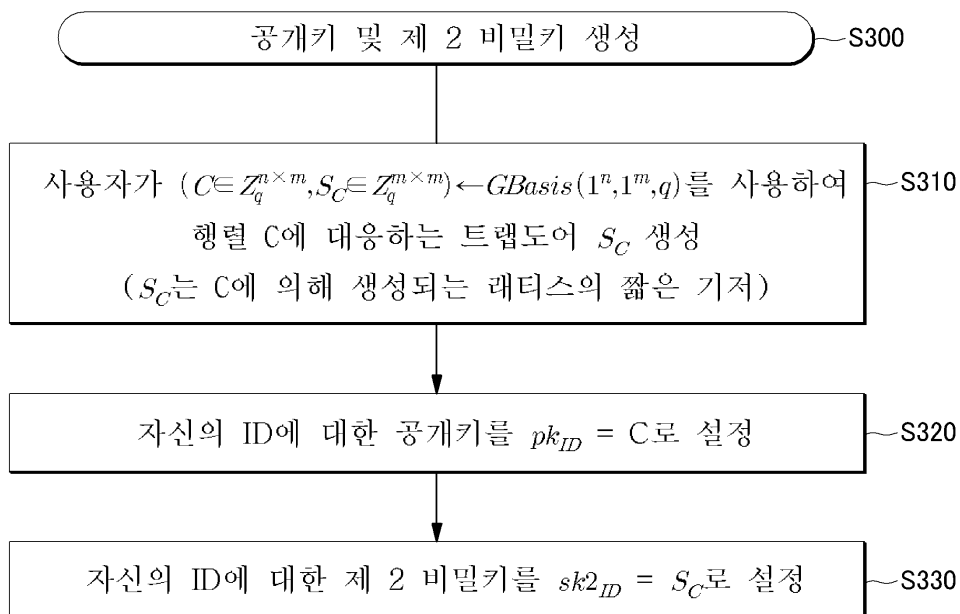
도면4



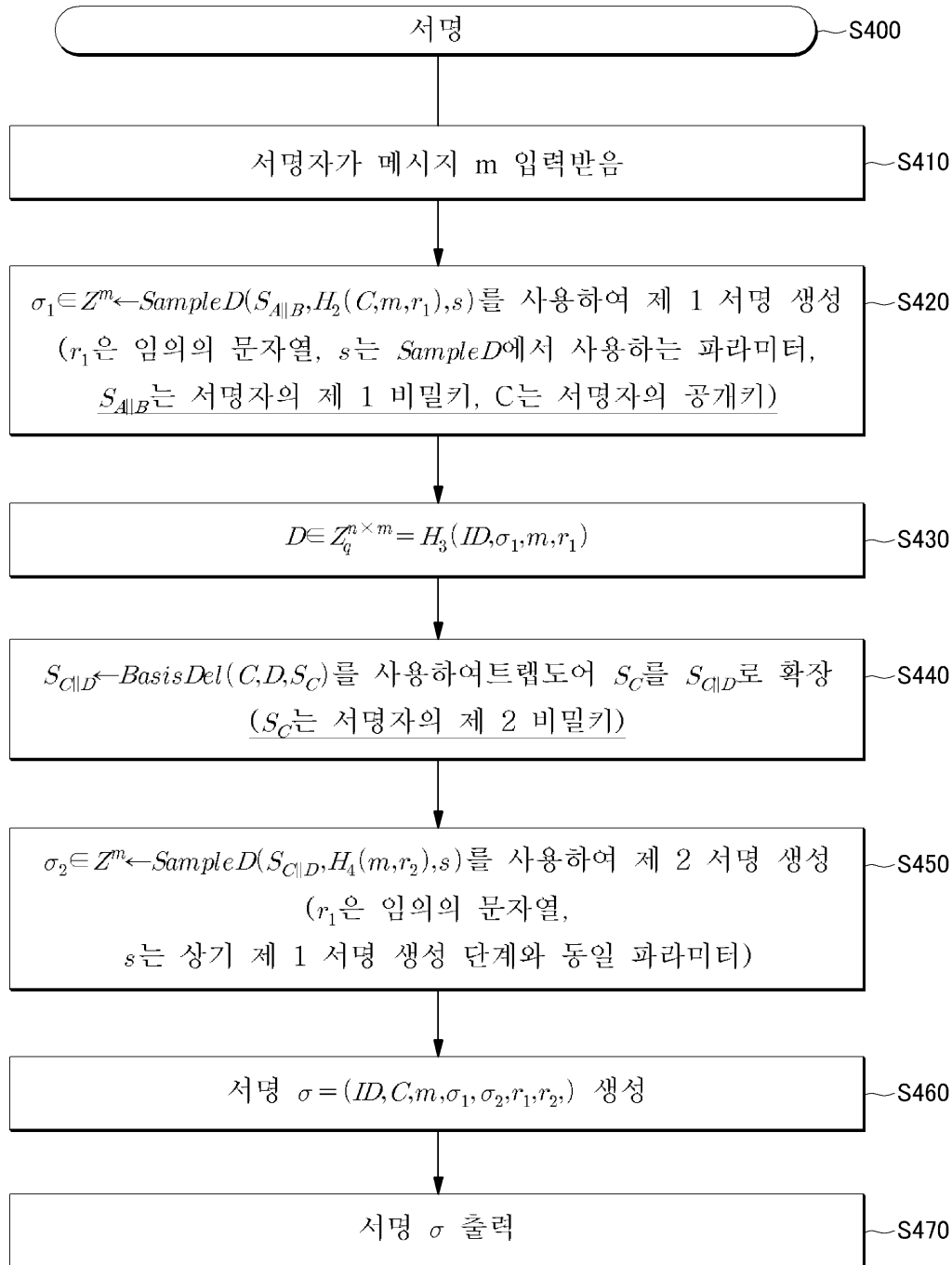
도면5



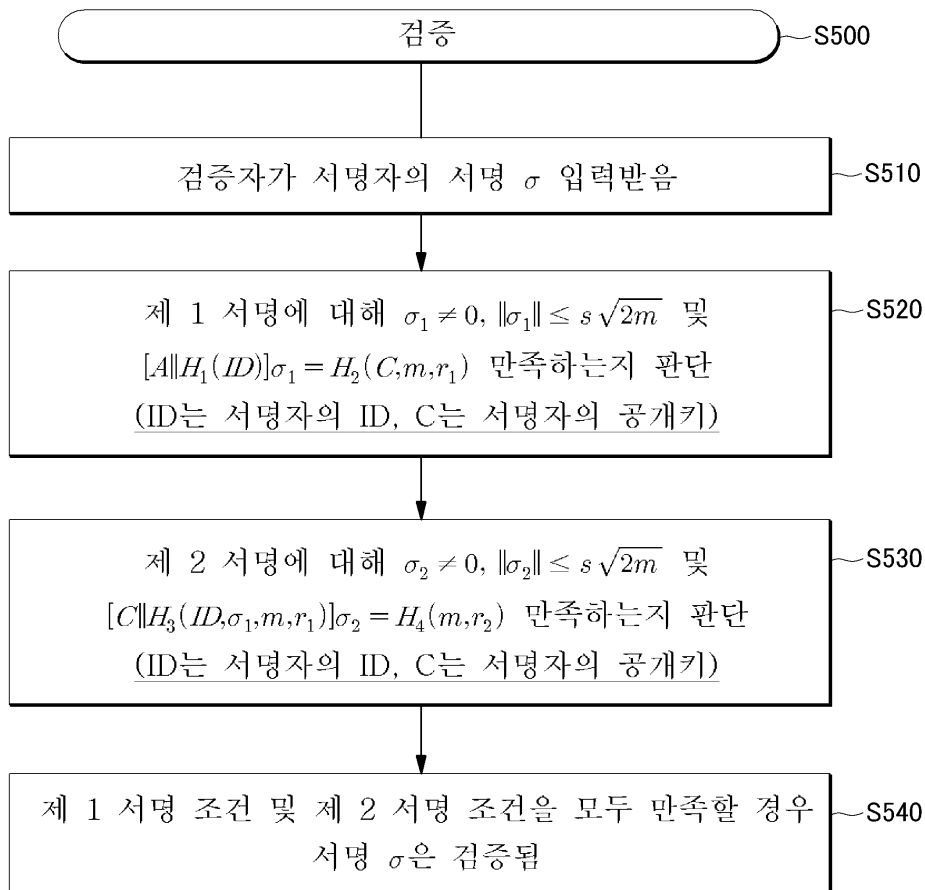
도면6



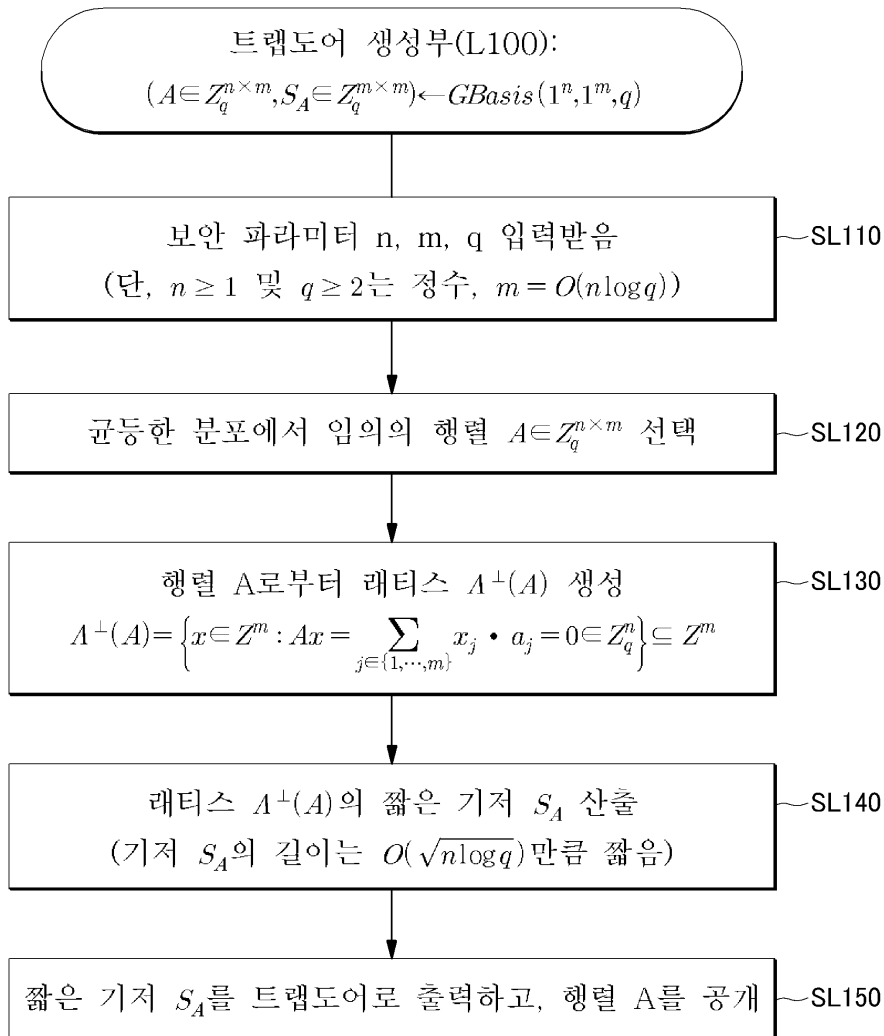
도면7



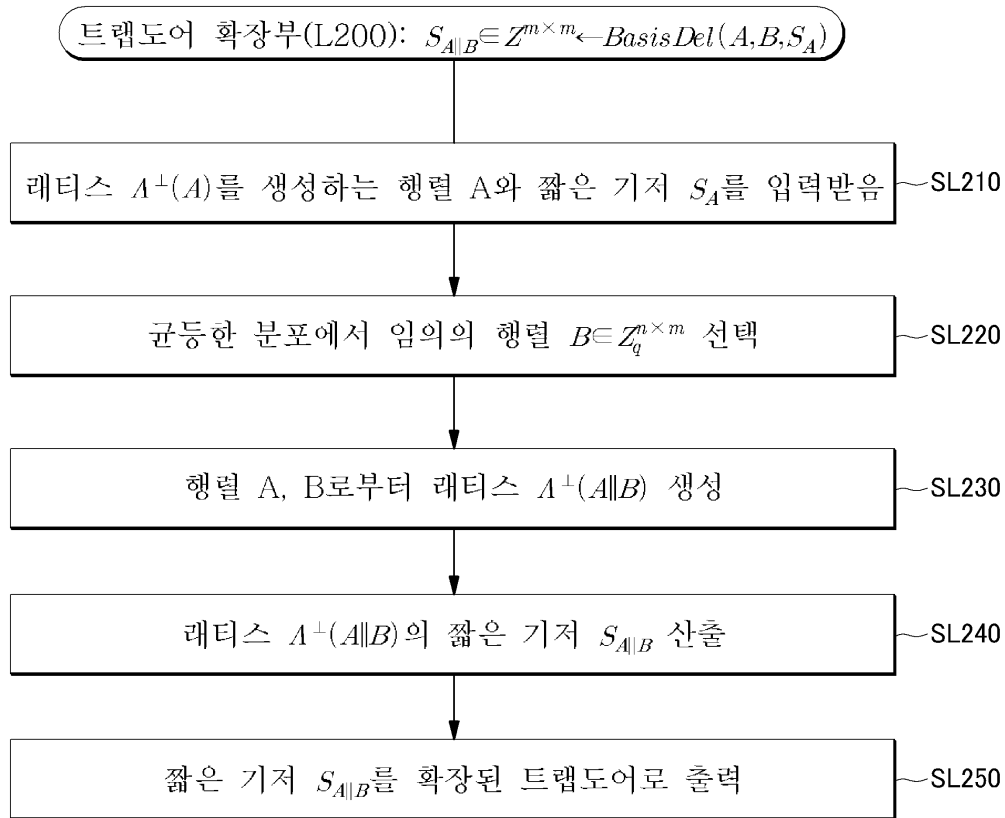
도면8



도면9



도면10



도면11

