(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년03월09일
(11) 등록번호 10-1601173
(24) 등록일자 2016년03월02일

(51) 국제특허분류(Int. Cl.)
H04L 9/08 (2006.01)
(21) 출원번호 10-2014-0175661
(22) 출원일자 2014년12월09일
심사청구일자 2014년12월09일
(56) 선행기술조사문헌
Kwangsu Lee 외 2인, IACR Cryptology ePrint Archive, "Efficient revocable identity-based encryption via subset difference methods" (2014.02.24. 공개)
Jian Weng 외 3인, Journal of Information Science and Engineering 24, "identity-based Parallel Key-insulated Signature Without Random Oracles" (2008.07. 공개)

(73) 특허권자
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
류금숙
서울 용산구 두텁바위로 96-8, 2층 (후암동)
김종현
서울특별시 강남구 학동로68길 29, 101동 1301호 (삼성동, 삼성동 힐스테이트)
(뒷면에 계속)
(74) 대리인
김등용, 김홍석

전체 청구항 수 : 총 6 항 심사관 : 문형섭

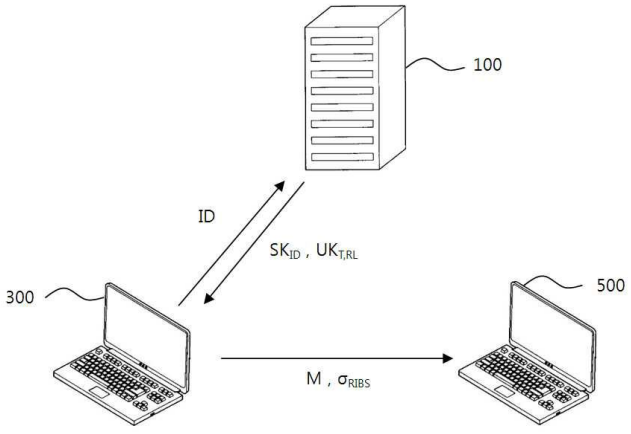
(54) 발명의 명칭 **폐기가능한 ID 기반 서명 방법 및 서버**

(57) 요 약

폐기가능한 ID 기반 서명 방법이 개시된다. 상기 방법은, (a) 서버가 보안 상수(1^{λ})와 사용자의 집합(N)을

(뒷면에 계속)

대 표 도 - 도1



이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 단계, (b) 상기 서버가 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 단계, (c) 상기 서버가 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 단계, (d) 상기 제1 단말기가 상기 업데이트키($UK_{T,RL}$)와 상기 개인키(SK_{ID})를 이용하여 상기 제1 단말기의 서명키($ssk_{ID,T}$)를 생성하는 단계, 및 (e) 상기 제1 단말기가 상기 서명키($ssk_{ID,T}$)를 이용하여 서명(σ_{RIBS})을 생성하는 단계를 포함한다.

(72) 발명자

엄지은

서울 서대문구 가재울로 45, 102동 102호 (남가좌동, 현대아파트)

박승환

서울 동작구 노량진로6마길 8

이동훈

서울 종로구 사직로8길 34, 1404호 (내수동, 경희궁의아침3단지)

이 발명을 지원한 국가연구개발사업

과제고유번호	NIPA-2014-H0301-14-1004
부처명	미래창조과학부
연구관리전문기관	정보통신산업진흥원
연구사업명	ITRC
연구과제명	제어시스템 보안 연구
기 여 율	1/1
주관기관	고려대학교 산학협력단
연구기간	2014.01.01 ~ 2014.12.31

명 세 서

청구범위

청구항 1

(a) 서버가 보안 상수(1^{λ})와 사용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 단계;

(b) 상기 서버가 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 단계;

(c) 상기 서버가 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 단계;

(d) 상기 제1 단말기가 상기 업데이트키($UK_{T,RL}$)와 상기 개인키(SK_{ID})를 이용하여 상기 제1 단말기의 서명키($ssk_{ID,T}$)를 생성하는 단계; 및

(e) 상기 제1 단말기가 상기 서명키($ssk_{ID,T}$)를 이용하여 서명(σ_{RIBS})을 생성하는 단계를 포함하고,

상기 (a) 단계는,

상기 사용자의 집합(N)의 원소가 아닌 임의의 ID(ID_{KGC})를 선택하는 단계;

상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제1 공개 상수(PP_{MS})와 제1 마스터키(MSK_{MS})를 생성하고, 상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제2 공개 상수(PP_{IBS})와 제2 마스터키(MSK_{IBS})를 생성하는 단계;

상기 제1 마스터키(MSK_{MS})와 상기 사용자의 집합(N)에 속하는 ID들을 이용하여 제1 비밀키($SK_{MS,i}$)를 생성하고, 상기 제1 마스터키(MSK_{MS})와 상기 임의의 ID(ID_{KGC})를 이용하여 제2 비밀키(SK_{KGC})를 생성하는 단계; 및

공집합인 상기 폐기 목록(RL)을 생성하는 단계를 포함하고,

상기 공개 상수(PP)는 상기 제1 공개 상수(PP_{MS}), 상기 제2 공개 상수(PP_{IBS}), 및 상기 임의의 ID(ID_{KGC})를 포함하고,

상기 마스터키(MSK)는 상기 제1 마스터키(MSK_{MS}), 상기 제2 마스터키(MSK_{IBS}), 상기 제1 비밀키($SK_{MS,i}$), 및 상기 제2 비밀키(SK_{KGC})를 포함하는,

폐기가능한 ID 기반 서명 방법.

청구항 2

삭제

청구항 3

제1항에 있어서,

제2 단말기가 상기 서명(σ_{RIBS}), 상기 공개 상수(PP), 메시지(M), 상기 ID, 및 상기 시간(T)을 이용하여, 상기 ID가 상기 시간(T)에 폐기되었는지 여부와 상기 메시지(M)가 상기 ID를 이용하여 서명되었는지 여부를 검증하는 단계를 더 포함하는, 폐기가능한 ID 기반 서명 방법.

청구항 4

제3항에 있어서,

상기 서버가 상기 폐기 목록(RL)을 주기적으로 업데이트하는 단계를 더 포함하는, 폐기가능한 ID 기반 서명 방법.

청구항 5

서버 상에서 구현되는 폐기가능한 ID 기반 서명 방법에 있어서,

(a) 보안 상수(1^{λ})와 사용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 단계;

(b) 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 단계; 및

(c) 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 단계를 포함하고,

상기 (a) 단계는,

상기 사용자의 집합(N)의 원소가 아닌 임의의 ID(ID_{KGC})를 선택하는 단계;

상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제1 공개 상수(PP_{MS})와 제1 마스터키(MSK_{MS})를 생성하고, 상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제2 공개 상수(PP_{IBS})와 제2 마스터키(MSK_{IBS})를 생성하는 단계;

상기 제1 마스터키(MSK_{MS})와 상기 사용자의 집합(N)에 속하는 ID들을 이용하여 제1 비밀키($SK_{MS,i}$)를 생성하고, 상기 제1 마스터키(MSK_{MS})와 상기 임의의 ID(ID_{KGC})를 이용하여 제2 비밀키(SK_{KGC})를 생성하는 단계; 및

공집합인 상기 폐기 목록(RL)을 생성하는 단계를 포함하고,

상기 공개 상수(PP)는 상기 제1 공개 상수(PP_{MS}), 상기 제2 공개 상수(PP_{IBS}), 및 상기 임의의 ID(ID_{KGC})를 포함하고,

상기 마스터키(MSK)는 상기 제1 마스터키(MSK_{MS}), 상기 제2 마스터키(MSK_{IBS}), 상기 제1 비밀키($SK_{MS,i}$), 및 상기 제2 비밀키(SK_{KGC})를 포함하는,

폐기가능한 ID 기반 서명 방법.

청구항 6

제5항에 있어서,

상기 폐기 목록(RL)을 업데이트하는 단계를 더 포함하는, 폐기가능한 ID 기반 서명 방법.

청구항 7

보안 상수(1^{λ})와 사용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 셋업 모듈;

상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 개인키 생성 모듈; 및

상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 업데이트키 생성 모듈을 포함하고,

상기 셋업 모듈은,

상기 사용자의 집합(N)의 원소가 아닌 임의의 ID(ID_{KGC})를 선택하고,

상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제1 공개 상수(PP_{MS})와 제1 마스터키(MSK_{MS})를 생성하고, 상기 보안 상수(1^{λ})와 상기 사용자의 집합(N)을 이용하여 제2 공개 상수(PP_{IBS})와 제2 마스터키(MSK_{IBS})를 생성하고,

상기 제1 마스터키(MSK_{MS})와 상기 사용자의 집합(N)에 속하는 ID들을 이용하여 제1 비밀키($SK_{MS,i}$)를 생성하고, 상기 제1 마스터키(MSK_{MS})와 상기 임의의 ID(ID_{KGC})를 이용하여 제2 비밀키(SK_{KGC})를 생성하고,

공집합인 상기 폐기 목록(RL)을 생성하는 단계를 포함하고,

상기 공개 상수(PP)는 상기 제1 공개 상수(PP_{MS}), 상기 제2 공개 상수(PP_{IBS}), 및 상기 임의의

ID(ID_{KGC})를 포함하고,

상기 마스터키(MSK)는 상기 제1 마스터키(MSK_{MS}), 상기 제2 마스터키(MSK_{IBS}), 상기 제1 비밀키($SK_{MS,i}$), 및 상기 제2 비밀키(SK_{KGC})를 포함하는,

폐기가능한 ID 기반 서명 서버.

발명의 설명

기 술 분 야

[0001] 본 발명의 개념에 따른 실시 예는 폐기가능한 ID 기반 서명 방법에 관한 것으로, 특히 주기적으로 업데이트키를 생성하여 사용자의 키 폐기가 가능한 ID 기반 서명 방법 및 서버에 관한 것이다.

배 경 기 술

[0002] ID 기반 서명(Identity-based Signature; IBS)은 사용자가 처음 등록된 후에는 지속적으로 같은 키를 이용하여 서명한다. 따라서, 사용 중에 계약기간이 만료되거나 키가 노출된 경우에는 더 이상 서명할 수 없도록 이 키를 폐기할 수 있는 방법이 필요하다. 이러한 방법을 가진 IBS를 폐기 가능한 ID 기반 서명(Revocable ID-based Signature)이라 한다. ID 기반 서명을 폐기 가능한 ID 기반 서명으로 변형하기 위해서는, 폐기되지 않은 사용자에 대해 시간에 따라 키를 지속적으로 갱신해주는 과정이 필요하고, 이 과정에서 업데이트키를 사용한다. 그러나, 기존의 ID 기반 서명에 단순히 업데이트키 생성 과정을 추가하면 폐기되지 않은 사용자 각각에 대한 업데이트키를 생성해주어야 하므로 폐기되지 않은 사용자 수에 비례하는 크기의 업데이트키가 필요하게 되며, 이는 매우 비효율적이다. 따라서, 본 발명에서는 기존의 ID 기반 서명을 이용하여 업데이트키의 크기가 효율적인 폐기가능한 ID 기반 서명을 설계하는 새로운 방식을 제안하고자 한다.

발명의 내용

해결하려는 과제

[0003] 본 발명이 이루고자 하는 기술적인 과제는 사용자의 키 폐기(Key revocation)가 가능한 ID 기반 서명 방법 및 서버를 제공하는 것이다.

과제의 해결 수단

[0004] 본 발명의 실시 예에 따른 폐기가능한 ID 기반 서명 방법은, (a) 서버가 보안 상수(1^{λ})와 사용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 단계, (b) 상기 서버가 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 단계, (c) 상기 서버가 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 단계, (d) 상기 제1 단말기가 상기 업데이트키($UK_{T,RL}$)와 상기 개인키(SK_{ID})를 이용하여 상기 제1 단말기의 서명키($ssk_{ID,T}$)를 생성하는 단계, 및 (e) 상기 제1 단말기가 상기 서명키($ssk_{ID,T}$)를 이용하여 서명(σ_{RIBS})을 생성하는 단계를 포함한다.

[0005] 본 발명의 다른 실시 예에 폐기가능한 ID 기반 서명 방법은, 서버 상에서 구현되며, (a) 보안 상수(1^{λ})와 사

용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 단계, (b) 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 단계, 및 (c) 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 단계를 포함한다.

[0006] 본 발명의 실시 예에 의한 폐기가능한 ID 기반 서명 서버는, 보안 상수(1^{λ})와 사용자의 집합(N)을 이용하여 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성하는 셋업 모듈, 상기 마스터키(MSK)와 제1 단말기의 ID를 이용하여 상기 제1 단말기의 개인키(SK_{ID})를 생성하고, 상기 개인키(SK_{ID})를 상기 제1 단말기로 송신하는 개인키 생성 모듈, 및 상기 마스터키(MSK), 시간(T), 및 상기 폐기 목록(RL)을 이용하여 업데이트키($UK_{T,RL}$)를 생성하고, 상기 업데이트키($UK_{T,RL}$)를 상기 제1 단말기로 송신하는 업데이트키 생성 모듈을 포함한다.

발명의 효과

[0007] 본 발명의 실시 예에 따른 폐기가능한 ID 기반 서명 방법 및 서버는 기존의 ID 기반 서명 기법을 이용하여 폐기가능한 ID 기반 서명 기법을 쉽게 설계할 수 있다.

[0008] 또한, 업데이트키의 길이와 서명키의 길이가 짧아 효율적인 기법을 설계할 수 있는 효과가 있다.

도면의 간단한 설명

[0009] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 본 발명의 일 실시예에 따른 폐기 가능한 ID 기반 서명 시스템을 도시한다.

도 2는 도 1에 도시된 서버의 기능 블럭도이다.

도 3은 도 1에 도시된 제1 단말기의 기능 블럭도이다.

발명을 실시하기 위한 구체적인 내용

[0010] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시 예들에 대해서 특정한 구조적 또는 기능적 설명은 단지 본 발명의 개념에 따른 실시 예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시 예들은 다양한 형태들로 실시될 수 있으며 본 명세서에 설명된 실시 예들에 한정되지 않는다.

[0011] 본 발명의 개념에 따른 실시 예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시 예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시 예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.

[0012] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.

[0013] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어"

있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0014] 본 명세서에서 사용한 용어는 단지 특정한 실시 예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0015] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0016] 본 발명에 대한 구체적인 설명에 들어가기에 앞서 본 발명에 이용되는 이론들에 대해 설명하면 다음과 같다.

[0017] ID 기반 암호 기법(Identity-based Cryptography)은 공개키 암호 기법의 한 종류이다. 기존의 공개키 암호 기법은 공개키의 소유자를 확인하기 위해 공개키 기반 구조(Public Key Infrastructure: PKI)를 사용하였는데 이를 관리하는데 많은 노력이 필요하다. 이러한 문제점을 가지고 있는 PKI의 대안으로 ID 기반 암호 기법이 제안되었다. ID 기반 암호 기법은 1984년 Shamir에 의해 처음 제시된 개념으로, 이메일 주소, 전화번호와 같이 사용자를 식별할 수 있는 정보(Identity; ID)를 공개키로 사용하는 암호 기법이다. 사용자의 ID를 공개키로 사용하면 공인인증서가 필요하지 않아 PKI의 관리의 문제를 해결할 수 있다.

[0018] 그러나, ID 기반 암호 기법은 키 위탁(Key Escrow) 문제와 효율적인 사용자 폐기의 문제가 있어 이에 대한 많은 연구가 진행되고 있다.

[0019] ID 기반 서명(Identity-based Signature)은 ID 기반 암호 기법의 예로, E-mail 주소, IP 주소, 전화번호 등과 같은 사용자의 ID를 서명에 대한 검증키로 사용하는 서명 기법이다. 서명자는 PKG(Private Key Generator)라는 비밀키 분배기관으로부터 자신의 ID와 대응되는 서명키를 발급받고 이를 이용해 서명을 생성한다. PKI 구조의 서명 기법과 달리, ID 기반 서명은 메시지에 대한 서명을 검증하기 위해 인증서를 따로 필요하지 않다는 장점을 가지고 있다. ID 기반 서명은 $Setup_{IBS}$, $KeyGen_{IBS}$, $Sign_{IBS}$, 및 $Verify_{IBS}$와 같은 네 개의 알고리즘으로 구성되어 있다. 각각의 알고리즘에 대한 설명은 다음과 같다.

[0020] $Setup_{IBS}$ 알고리즘(셋업 알고리즘)은 보안 상수(1^{λ})와 사용자의 집합(N)을 입력으로 받고, 공개 상수(PP)와 마스터키(MSK)를 출력한다. $Setup_{IBS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0021] $$Setup_{IBS}(1^{\lambda},\mathrm{N})\rightarrow PP,MSK$$

[0022] $KeyGen_{IBS}$ 알고리즘(키생성 알고리즘)은 마스터키(MSK)와 ID를 입력으로 받고, ID에 대한 서명키(SK)를 출력한다. $KeyGen_{IBS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0023] $$KeyGen_{IBS}(MSK,ID)\rightarrow SK$$

[0024] $Sign_{IBS}$ 알고리즘(서명 알고리즘)은 메시지(M), ID, 및 서명키(SK)를 입력으로 받고, 서명(σ)을 출력한다. $Sign_{IBS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0025] $$Sign_{IBS}(ID,M,SK)\rightarrow\sigma$$

[0026] $Verify_{IBS}$ 알고리즘(검증 알고리즘)은 메시지(M), ID, 서명(σ), 공개 상수(PP)를 입력으로 받고, 이를 검증한다. $Verify_{IBS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0027] $$Verify_{IBS}(\sigma, PP, M, ID) \rightarrow 0 \text{ or } 1$$

[0028] 다중 서명(Multi-signature)은 동일한 메시지에 대해 여러 서명자가 서명을 하는 기법이다. 기존의 서명기법은 n 명의 서명자가 하나의 동일한 메시지에 서명할 경우, n 개의 서명이 생성되게 된다. 하지만 다중 서명을 이용해 n 명의 서명자가 하나의 동일한 메시지에 서명할 경우 하나의 서명으로 생성하여 서명의 길이를 줄일 수 있다. 이러한 기법은 전력이 제한되는 환경에서 전력이 많이 사용되는 원거리 무선통신을 해야 하는 경우 유용하게 사용될 수 있다.

[0029] ID 기반 다중서명(Identity-baseed Multi-signature)은 ID 기반 서명기법의 확장으로, ID 기반 서명기법의 장점과 다중 서명 기법의 장점을 하나로 합친 서명 기법이다. ID 기반 다중 서명은 $Setup_{MS}$, $KeyGen_{MS}$, $Sign_{MS}$, 및 $Verify_{MS}$와 같은 네 개의 알고리즘으로 구성되어 있다. 각각의 알고리즘에 대한 설명은 다음과 같다.

[0030] $Setup_{MS}$ 알고리즘(셋업 알고리즘)은 보안 상수(1^{λ})와 사용자의 집합($\mathbb{N}$)을 입력으로 받고, 공개 상수(PP)와 마스터키(MSK)를 출력한다. $Setup_{MS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0031] $$Setup_{MS}(1^{\lambda}, \mathbb{N}) \rightarrow PP, MSK$$

[0032] $KeyGen_{MS}$ 알고리즘(키생성 알고리즘)은 마스터키(MSK)와 ID_i 를 입력으로 받고, ID_i에 대한 서명키(SK_i)를 출력한다. $KeyGen_{MS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0033] $$KeyGen_{MS}(MSK, ID_i) \rightarrow SK_i$$

[0034] $Sign_{MS}$ 알고리즘(서명 알고리즘)은 메시지(M)와 서명할 ID의 집합(S), 집합(S)에 포함된 사용자의 서명키(SK_i)를 입력으로 받고, 집합(S)에 포함된 사용자들 각각의 서명키로 생성된 서명(σ)을 출력한다. $Sign_{MS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0035] $$Sign_{MS}(SK_i, M, S) \rightarrow \sigma$$

[0036] $Verify_{MS}$ 알고리즘(검증 알고리즘)은 서명(σ)과 공개 상수(PP), 서명한 ID의 집합(S), 메시지(M)을 입력으로 받고, 이 서명이 집합(S)의 모든 ID로 서명되었는지 여부를 검증한다. $Verify_{MS}$ 알고리즘은 다음과 같이 표현될 수 있다.

[0037] $$Verify_{MS}(\sigma, PP, M, ID) \rightarrow 0 \text{ or } 1$$

[0038] 폐기 가능한 ID 기반 서명(Revocable Identity-based Signature)은 ID 기반 서명에서 계약 기간이 만료되거나 개인키가 노출된 사용자에 대한 효율적인 폐기 메커니즘을 제공하는 서명 기법이다. 폐기 가능한 ID 기반 서명은 Setup, PriKeyGen, UpKeyGen, DeriveKey, Sign, Verify, 및 Revoke와 같은 7 개의 알고리즘으로 이루어져 있다.각각의 알고리즘에 대한 설명은 다음과 같다.

[0039] Setup 알고리즘(셋업 알고리즘)은 보안 상수(1^{λ})와 사용자의 집합(N)을 입력으로 받고, 폐기 목록(revocation list; RL), 공개 상수(public parameter; PP), 및 마스터키(MSK)를 출력한다. 셋업 알고리즘 단계에서 출력된 폐기 목록(RL)은 비어있을 수 있다. 즉, 폐기 목록(RL)은 공집합일 수 있다. Setup 알고리즘은 다음과 같이 표현될 수 있다.

[0040] $$Setup(1^{\lambda},\mathrm{N})\rightarrow RL, PP, MSK$$

[0041] PriKeyGen 알고리즘(개인키 생성 알고리즘)은 마스터키(MSK)와 ID를 입력으로 받고, ID에 대한 개인키(SK_{ID})를 출력한다. PriKeyGen 알고리즘은 다음과 같이 표현될 수 있다.

[0042] $$PriKeyGen(MSK, ID)\rightarrow SK_{ID}$$

[0043] UpKeyGen 알고리즘(업데이트키 생성 알고리즘)은 마스터키(MSK), 시간(T), 폐기 목록(RL)을 입력으로 받고, 시간(T)과 폐기 목록(RL)에 대한 업데이트키($UK_{T,RL}$)를 출력한다. UpKeyGen 알고리즘은 다음과 같이 표현될 수 있다.

[0044] $$UpKeyGen(MSK, T, RL)\rightarrow UK_{T,RL}$$

[0045] DeriveKey 알고리즘(서명키 생성 알고리즘)은 업데이트키($UK_{T,RL}$)와 개인키(SK_{ID})를 입력으로 받고, ID와 시간(T)에 대한 서명키($ssk_{ID,T}$)를 출력한다. DeriveKey 알고리즘은 다음과 같이 표현될 수 있다.

[0046] $$DeriveKeyGen(UK_{T,RL}, SK_{ID})\rightarrow ssk_{ID,T}$$

[0047] Sign 알고리즘(서명 알고리즘)은 ID, 서명키($ssk_{ID,T}$), 및 메시지(M)를 입력으로 받고, ID,시간(T), 및 메시지(M)에 대한 서명($\sigma_{ID,T,M}$)을 출력한다. Sign 알고리즘은 다음과 같이 표현될 수 있다.

[0048] $$Sign(ID, ssk_{ID,T}, M)\rightarrow \sigma_{ID,T,M}$$

[0049] Verify 알고리즘(검증 알고리즘)은 서명(σ), 메시지(M), ID, 및 시간(T)을 입력으로 받고, 서명(σ)을 검증한다. Verify 알고리즘은 다음과 같이 표현될 수 있다.

[0050] $$Verify(\sigma, M, ID, T)\rightarrow 0 \text{ or } 1$$

[0051] Revoke 알고리즘(폐기 알고리즘)은 폐기될 ID, 시간(T), 및 기존의 폐기 목록(RL_T)을 입력으로 받고, 새로운 폐기 목록(RL_{T+1})을 출력한다. 즉, Revoke 알고리름은 폐기 목록을 업데이트할 수 있다. Revoke 알고리즘은 다음과 같이 표현될 수 있다.

[0052] $$Revoke(ID_{revoked}, T, RL_T) \rightarrow RL_{T+1}$$

[0053] 본 발명에서는 기존의 ID 기반 서명 기법을 이용하여 효율적인 폐기 가능한 ID 기반 서명 기법을 설계하는 방법을 제안한다. 기존의 ID 기반 서명에 단순히 업데이트키 생성 과정을 추가하여 폐기 가능한 ID 기반 서명 기법을 설계하는 경우, 폐기되지 않은 모든 ID에 대한 업데이트키를 각각 생성해주어야 하므로 업데이트키의 크기가 사용자 수에 따라 비효율적으로 늘어나게 된다. 이를 해결하기 위해 폐기되지 않은 모든 사용자에 대해 동일한 업데이트키를 생성하는 과정이 필요하게 된다. 이때, 업데이트키는 결국 ID가 시간(T)에서 폐기되지 않았음을 인증하는 역할을 하는 것이므로 이에 대해 서명기법을 적용하고자 한다. 즉, 키를 생성하는 기관(PKG)이 사용자의 키에 대해 서명하는 것이고, 또한 모든 키에 대해 서명해야 하므로 이 과정에서 ID 기반 다중 서명 기법을 사용하고자 한다.

[0054] ID 기반 다중 서명 기법은 하나의 메시지에 대해 여러 번의 서명을 거듭하여도 일정한 크기의 서명을 생성하기 때문에 ID 기반 다중 서명 기법을 업데이트키 생성 과정에 적용하면 일정한 크기의 업데이트키를 얻을 수 있다. 그러나, 단순히 이 기법을 업데이트키 생성 과정에 적용하여 서명키를 생성하게 되면 모든 ID의 개인키에 대한 서명이 전부 포함되므로 이 서명을 검증하기 위해서는 모든 ID의 개인키가 필요하게 된다. 따라서, 폐기되지 않은 모든 ID에 대한 서명을 업데이트키로 사용한 후 서명키를 생성하는 과정에서 이를 다시 특정 ID에 대한 서명키로 바꿔줄 필요가 있다.

[0055] 즉, 원래의 ID 기반 다중 서명 과정이 Setup, KeyGen, Sign, 및 Verify의 네 단계의 과정을 거치던 것에 추가로 모든 폐기되지 않은 사용자에 대한 업데이트키를 이용하여 특정 사용자의 ID에 대한 서명키를 만드는 ReduSet 알고리즘(서명변형 알고리즘)을 추가하여 해결할 수 있다.

[0056] ReduSet 알고리즘은 집합($\mathcal{S}$)의 모든 ID에 대한 서명을 입력으로 받고, 특정 ID에 대한 서명으로 변형하여 출력하는 알고리즘이다. ReduSet 알고리즘은 다음과 같이 표현될 수 있다.

[0057] $$ReduSet(\sigma_S) \rightarrow \sigma_{ID}$$

[0058] 따라서, 본 발명에서는 기존의 ID 기반 서명의 Setup, KeyGen, Sign, 및 Verify의 네 단계와 ID 기반 다중 서명의 Setup, KeyGen, Sign, 및 Verify의 네 단계, 그리고 새로운 ReduSet 단계를 포함하는 총 아홉 단계를 이용하여 기존의 ID 기반 다중 서명을 이용하여 ID 기반 서명을 폐기 가능한 ID 기반 서명으로 바꾸는 새로운 설계방식을 제안한다.

[0059] 본 발명에서 제안하는 폐기가능한 ID 기반 서명 방법에 포함되는 구체적인 단계는 아래와 같다.

[0060] 1. 셋업 단계($Setup_{RIBS}(1^{\lambda}, \mathrm{N}) \rightarrow RL, PP, MSK$)

[0061] (1) 우선, 보안 상수(1^{λ})와 사용자의 집합(N)을 입력으로 받고, 사용자의 집합(N)에 포함되지 않은, 즉 집합(N)의 원소가 아닌 임의의 아이디(ID_{KGC})를 선택한다.

[0062] (2) 다음으로, ID 기반 다중 서명의 셋업 알고리즘($Setup_{MS}(1^{\lambda}, \mathrm{N}) \rightarrow PP_{MS}, MSK_{MS}$)과 ID 기반 서명의 셋업 알고리즘($Setup_{IBS}(1^{\lambda}, \mathrm{N}) \rightarrow PP_{IBS}, MSK_{IBS}$)을 수행한다.

[0063] (3) 다음으로, ID 기반 다중 서명의 키생성 알고리즘($KeyGen_{MS}(MSK, ID_i) \rightarrow SK_{MS,i}$)을 수행하고, ID 기반 다중 서명의 키생성 알고리즘($KeyGen_{MS}(MSK, ID_{KGC}) \rightarrow SK_{KGC}$)을 수행하여 ID_{KGC} 에 대한 개인키를 생성한다.

[0064] (4) 마지막으로, 비어있는 집합, 즉 폐기 목록(RL)을 생성한다.

[0065] 출력되거나 생성된 값들을 이용하여, 폐기 목록(RL), 공개 상수 ($PP=< PP_{MS}, PP_{IBS}, ID_{KGC} >$), 마스터키 ($MSK=< MSK_{MS}, MSK_{IBS}, \{SK_{MS,i}\}, SK_{KGC} >$)를 출력한다.

[0066] 2. 개인키 생성 단계($PriKeyGen_{RIBS}(MSK_{IBS}, ID) \rightarrow SK_{ID}$)

[0067] 마스터키(MSK_{IBS})와 ID를 입력으로 받고, ID 기반 서명의 키생성 알고리즘 ($KeyGen_{IBS}(MSK_{IBS}, ID) \rightarrow SK_{IBS,ID}$)을 수행하여, 개인키($SK_{ID} = < SK_{IBS,ID} >$)를 출력한다.

[0068] 3. 업데이트키 생성 단계($UpKeyGen_{RIBS}(MSK, T, RL) \rightarrow UK_{T,RL}$)

[0069] 마스터키((MSK), 시간(T), 및 폐기 목록(RL)을 입력으로 받고, 전체 사용자의 집합(N)을 이용하여 폐기되지 않은 사용자의 집합($S=\mathrm{N}-RL$)을 생성한 후, ID 기반 다중 서명의 서명 알고리즘 ($Sign_{MS}(\{SK_{MS,i}\}, T, S) \rightarrow \sigma_{MS,T,S}$)을 수행하여, 업데이트키($UK_{T,RL} =< \sigma_{MS,T,S} >$)를 출력한다.

[0070] 4. 서명키 생성 단계($DeriveKeyGen_{RIBS}(UK_{T,RL}, SK_{ID}) \rightarrow ssk_{ID,T}$)

[0071] 업데이트키($UK_{T,RL}$)와 개인키(SK_{ID})를 입력으로 받고, 서명변환 알고리즘 ($ReduSet(\sigma_{MS,T,S}) \rightarrow \sigma_{MS,T,ID}$)을 수행하여 서명키($ssk_{ID,T} =< \sigma_{MS,T,ID}, SK_{ID} >$)를 출력한다.

[0072] 5. 서명 단계($Sign_{RIBS}(ID, ssk_{ID,T}, M) \rightarrow \sigma_{RIBS}$)

[0073] ID, 서명키($ssk_{ID,T}$), 및 메시지(M)을 입력으로 받고, ID 기반 서명의 서명 알고리즘 ($Sign_{IBS}(ID, M, ssk_{ID,T}) \rightarrow \sigma_{IBS,ID,M}$)을 수행하여, 서명 ($\sigma_{RIBS} =< \sigma_{IBS,ID,M}, \sigma_{MS,T,ID} >$)을 출력한다.

[0074] 6. 검증 단계($Verify_{RIBS}(\sigma_{RIBS}, PP, M, ID, T) \rightarrow 0 \text{ or } 1$)

[0075] 서명(σ_{RIBS}), 메시지(M), 공개 상수(PP), ID, 시간(T)을 입력으로 받고, ID 기반 다중 서명의 검

증 알고리즘($Verify_{MS}(\sigma_{MS,T,ID}, PP_{MS}, ID, T) \rightarrow 0 \text{ or } 1$)과 ID 기반 서명의 검증 알고리즘($Verify_{IBS}(\sigma_{IBS,ID,M}, PP_{IBS}, M, ID) \rightarrow 0 \text{ or } 1$)을 수행한다. 즉, 검증 알고리즘($Verify_{MS}(\sigma_{MS,T,ID}, PP_{MS}, ID, T) \rightarrow 0 \text{ or } 1$)을 수행함으로써, ID가 시간(T)에서 폐기되지 않은 ID 인지 여부가 검증되고, 검증 알고리즘($Verify_{IBS}(\sigma_{IBS,ID,M}, PP_{IBS}, M, ID) \rightarrow 0 \text{ or } 1$)을 수행함으로써, 메시지(M)가 ID로 서명된 것인지 검증될 수 있다. 이때, 두 개의 알고리즘의 출력이 모두 1인 경우 1을 출력하고, 그렇지 않은 경우 0을 출력한다.

[0076] 7. 폐기 단계($Revoke_{RIBS}(ID_{revoked}, T, RL_T) \rightarrow RL_{T+1}$)

[0077] 폐기될 ID, 시간(T), 및 기존의 폐기 목록(RL_T)을 입력으로 받고, 새로운 폐기 목록(RL_{T+1})을 출력한다.

[0078] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시 예들을 상세히 설명한다.

[0079] 도 1은 본 발명의 일 실시예에 따른 폐기 가능한 ID 기반 서명 시스템을 도시한다. 도 1을 참조하면 시스템(10)은 서버(100), 제1 단말기(300), 및 제2 단말기(500)를 포함한다.

[0080] 서버(100)는 제1 단말기(300)로부터 ID를 수신하고, 수신된 ID에 대응하는 개인키(SK_{ID})를 생성하여 생성된 개인키(SK_{ID})를 제1 단말기(300)로 송신한다. 또한, 서버(100)는 업데이트키($UK_{T,RL}$)를 생성하고, 생성된 업데이트키($UK_{T,RL}$)를 제1 단말기(300)로 송신한다. 서버(100)는 업데이트키($UK_{T,RL}$)를 주기적으로 생성할 수 있다.

[0081] 제1 단말기(300)는 서버(100)로부터 수신된 업데이트키($UK_{T,RL}$)와 개인키(SK_{ID})를 이용하여 서명키($ssk_{ID,T}$)를 생성하고, 생성된 서명키($ssk_{ID,T}$)를 이용하여 서명(σ_{RIBS})을 생성한다.

[0082] 제2 단말기(500)는 제1 단말기(300)로부터 서명(σ_{RIBS})과 메시지(M)를 수신하고, 제1 단말기(300)의 ID가 시간(T)에서 폐기되었는지 여부를 검증하며, 서명(σ_{RIBS})이 ID를 이용하여 서명된 것인지 여부를 검증한다.

[0083] 제1 단말기(300)와 제2 단말기(500)는 PC(personal computer), 태블릿 PC, 노트북(notebook), 넷-북(net-book), e-리더(e-reader), PDA(personal digital assistant), PMP(portable multimedia player), MP3 플레이어, 또는 MP4 플레이어와 같은 데이터 처리 장치로 구현되거나, 모바일폰(mobile phone), 스마트폰(smart phone) 등과 같은 핸드헬드 장치(handheld device)로 구현될 수 있다.

[0084] 도 2는 도 1에 도시된 서버의 기능 블럭도이다. 도 1과 도 2를 참조하면, 서버(100)는 셋업 모듈(110), 개인키 생성 모듈(130), 및 업데이트키 생성 모듈(150)을 포함한다. 실시 예에 따라, 서버(100)는 폐기 모듈(170)을 더 포함할 수 있다.

[0085] 셋업 모듈(110)은 셋업 알고리즘($Setup_{RIBS}(1^{\lambda}, N) \rightarrow RL, PP, MSK$)을 수행할 수 있다. 즉, 셋

업 모듈(100)은 보안 상수(1^{λ})와 사용자의 집합(N)을 입력으로 하는 셋업 알고리즘을 수행하여, 폐기 목록(RL), 공개 상수(PP), 및 마스터키(MSK)를 생성할 수 있다.

[0086] 개인키 생성 모듈(130)은 개인키 생성 알고리즘($PriKeyGen_{RIBS}(MSK_{IBS}, ID) \rightarrow SK_{ID}$)을 수행할 수 있다. 즉, 개인키 생성 모듈(130)은 마스터키(MSK_{IBS})와 ID를 입력으로 하는 개인키 생성 알고리즘을 수행하여, 개인키(SK_{ID})를 생성하고, 생성된 개인키(SK_{ID})를 제1 단말기(300)로 송신할 수 있다.

[0087] 업데이트키 생성 모듈(150)은 업데이트키 생성 알고리즘($UpKeyGen_{RIBS}(MSK, T, RL) \rightarrow UK_{T,RL}$)을 수행할 수 있다. 즉, 업데이트키 생성 모듈(150)은 마스터키(MSK), 시간(T), 및 폐기 목록(RL)을 입력으로 하는 업데이트키 생성 알고리즘을 수행하여, 업데이트키($UK_{T,RL}$)를 생성하고, 생성된 업데이트키($UK_{T,RL}$)를 제1 단말기(300)로 송신할 수 있다.

[0088] 폐기 모듈(170)은 폐기 알고리즘($Revoke_{RIBS}(ID_{revoked}, T; RL_T) \rightarrow RL_{T+1}$)을 수행할 수 있다. 구체적으로, 폐기 모듈(170)은 폐기될 ID, 시간(T), 폐기 목록(RL_T)을 입력으로 하는 폐기 알고리즘을 수행하여, 새로운 폐기 목록(RL_{T+1})을 생성할 수 있다. 즉, 폐기 모듈(170)은 폐기 목록을 주기적으로 갱신하거나 업데이트할 수 있다.

[0089] 도 3은 도 1에 도시된 제1 단말기의 기능 블럭도이다. 도 1 내지 도 3을 참조하면, 제1 단말기(300)는 서명키 생성 모듈(310)과 서명 모듈(330)을 포함한다.

[0090] 서명키 생성 모듈(310)은 서명키 생성 알고리즘($DeriveKeyGen_{RIBS}(UK_{T,RL}, SK_{ID}) \rightarrow ssk_{ID,T}$)을 수행할 수 있다. 즉, 서명키 생성 모듈(310)은 서버(100)로부터 수신된 업데이트키($UK_{T,RL}$)와 개인키(SK_{ID})를 입력으로 하는 서명키 생성 알고리즘을 수행하여, 서명키($ssk_{ID,T}$)를 생성할 수 있다.

[0091] 서명 모듈(330)은 서명 알고리즘($Sign_{RIBS}(ID, ssk_{ID,T}, M) \rightarrow \sigma_{RIBS}$)을 수행할 수 있다. 즉, 서명 모듈(330)은 ID, 서명키($ssk_{ID,T}$), 메시지(M)를 입력으로 하는 서명 알고리즘을 수행하여, 서명(σ_{RIBS})을 생성할 수 있다.

[0092] 도 2와 도 3에 도시된 서버(100)와 제1 단말기(300)의 구성들 각각은 기능 및 논리적으로 분리될 수 있음으로 나타내는 것이며, 반드시 각각의 구성이 별도의 물리적 장치로 구분되거나 별도의 코드로 작성됨을 의미하는 것이 아님을 본 발명의 기술분야의 평균적 전문가는 용이하게 추론할 수 있을 것이다.

[0093] 또한, 본 명세서에서 모듈이라 함은, 본 발명의 기술적 사상을 수행하기 위한 하드웨어 및 상기 하드웨어를 구동하기 위한 소프트웨어의 기능적, 구조적 결합을 의미할 수 있다. 예컨대, 상기 모듈은 소정의 코드와 상기 소정의 코드가 수행되기 위한 하드웨어 리소스의 논리적인 단위를 의미할 수 있으며, 반드시 물리적으로 연결된

코드를 의미하거나, 한 종류의 하드웨어를 의미하는 것이 아니다.

[0094] 이하에서는, 본 발명에서 제안한 방식을 이용하여 Bellare와 Neven이 제안한 ID-based Multi-signature from RSA를 효율적으로 변경한 폐기 가능한 ID 기반 서명의 예제이다.

[0095] 1. 설정 단계

[0096] 관리자(PKG)는 $Setup(1^{\lambda},\mathrm{N})$ 알고리즘을 통해 (N,e,d) 와 ID_{KGC} , $H_1,H_2,H_3,$ RL 을 생성한다. (N,e,d) 는 RSA 암호 시스템에서의 키 생성 방식을 이용해서 생성되고, ID_{KGC}는 N의 원소가 아닌 임의의 문자열이며, H_1,H_2,H_3 는 암호학적 해시함수로 임의의 $l_1,l_2\in\mathrm{N}$에 대해 $H_1:\{0,1\}^*\rightarrow\{0,1\}^{l_1}$, $H_2:\{0,1\}^*\rightarrow Z_N^*$, 및 $H_3:\{0,1\}^*\rightarrow\{0,1\}^{l_2}$ 을 만족한다. RL은 폐기된 ID를 저장하는 집합으로 이 알고리즘에서는 공집합으로 정의되며, 이후 시간(T)에 따라 폐기된 ID를 포함한다. 알고리즘을 통해 생성된 변수를 이용하여 $PP=<N,e,ID_{KGC}>$ 를 공개 상수로 공개하고, $msk=<d,H_2(ID_{KGC})^d \bmod N>$ 는 마스터키로 관리자가 관리하며, 사용자의 개인키를 생성할 때 사용된다.

[0097] 2. 사용자 등록 단계

[0098] 관리자는 등록을 원하는 ID에 대해 $PriKeyGen(MSK,\ ID)$ 알고리즘을 수행한다. 사용자의 ID를 해시함수(H_2), 공개 상수(N), 및 마스터키(d)를 이용하여 개인키($SK_{ID}=H_2(ID)^d \bmod N$)를 생성하고, 생성된 개인키를 사용자에게 안전하게 전달한다.

[0099] 3. 업데이트 단계

[0100] 관리자는 폐기되지 않은 사용자에 대해 업데이트키를 생성하기 위해 $UpKeyGen(MSK,T,RL)$ 알고리즘을 수행한다. 사용자의 집합(N), 폐기 목록(RL), 및 ID_{KGC} 를 이용하여, 폐기되지 않은 사용자에 대한 집합($\mathrm{S=N}\setminus RL\cup\{ID_{KGC}\}$)로 정의한다. 이때, 집합(S)에 포함된 각각의 원소(ID_i)에 대해 임의의 r_i 를 선택한다. 선택된 r_i 들과 공개 상수($e,\ N$)을 이용하여 $R=\prod_{ID_i\in S} r_i^e \bmod N$ 을 계산하고, 계산된 R , 시간(T), 해시함수(H_1)를 이용하여 $c=H_1(R\|T)$ 를 계산하고, 집합(S)에 포함된 사용자 ID_i 의 개인키(SK_i)와 선택된 r_i 를 이용하여 $s=\prod_{ID_i\in S} r_i SK_i^c$ 를 계산한다. 계산된 $UK_{T,RL}=<c,s>$ 가 업데이트키가 되고, 이를 이용하여 사용자의 개인키를 업데이트할 수 있다.

[0101] 4. 서명키 생성 단계

[0102] 개인의 서명키는 $DeriveKeyGen(UK_{T,RL},SK_{ID})$ 알고리즘에 사용자 등록 단계에서 받은 SK_{ID}

와 업데이트 단계에서 생성된 $UK_{T,RL}$ 를 이용하여 생성된다. 사용자의 ID가 집합(S)에 포함되어 있다면 $ssk_1 = SK_{ID}$ 로 하고, 공개 상수(e)를 이용하여 $ssk_2 = s^e$ 를 계산한다. 집합(S)에서 서명키를 생성할 ID와 ID_{KGC} 를 제외한 집합($S' = S \setminus \{ID, ID_{KGC}\}$)를 생성하여 $ssk_3 = \prod_{ID_i \in S'} H_2(ID_i)$ 를 계산하고, 업데이트키에 포함된 c 를 ssk_4 라 한다. $ssk_{ID|T} = < ssk_1, ssk_2, ssk_3, ssk_4 >$ 를 서명키로 사용한다.

[0103] 사용자의 ID가 RL에 포함되어 있다면, 집합(S)에 포함되어 있지 않으므로 업데이트키와 조합하여 서명키를 만들 수 없다.

[0104] 5. 서명 단계

[0105] 사용자는 보내고자 하는 메시지(M)에 대해 자신의 서명키($ssk_{ID|T}$)를 이용하여 $Sign(ID, ssk_{ID,T}, M)$ 알고리즘을 수행한다. 임의의 원소(A)를 선택하여 $\sigma_1 = A^e$ 로 둔다. 주어진 해시함수(H_3)를 이용하여 $h = H_3(ID, M, \sigma_1)$ 을 계산하고, 서명키($ssk_{ID|T}$)에 포함된 ssk_1 을 이용하여 $\sigma_2 = A \cdot ssk_1^h$ 를 생성한다. 나머지 서명키(ssk_2, ssk_3, ssk_4)를 각각 $\sigma_3 = ssk_2,\ \sigma_4 = ssk_3,\ \sigma_5 = ssk_4$ 로 하여 서명($\sigma = < \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5 >$)을 생성한다.

[0106] 6. 검증 단계

[0107] 메시지(M)에 대해 ID 와 T 에 대한 서명($\sigma = < \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5 >$)을 받으면 $Verify(\sigma, M, ID, T)$ 알고리즘을 통해 검증한다. σ_3, σ_4, σ_5 , 해시함수(H_2), ID , 및 ID_{KGC} 를 이용하여 $R' = \sigma_3 \cdot (\sigma_4 \cdot H_2(ID) \cdot H_2(ID_{KGC}))^{-\sigma_5} \bmod N$ 을 계산하고, σ_1 , 해시함수(H_3), ID , 및 메시지(M)을 이용하여 $h' = H_3(ID, M, \sigma_1)$ 을 계산한다. 계산된 R' 과 h' 을 이용하여 $H_1(R' \| T)$ 를 계산했을 때, σ_5 와 동일한 결과가 나오는지 여부를 이용하여 시간(T)에서 ID 가 폐기되지 않았음을 확인할 수 있고, $\sigma_1 \cdot H_2(ID)^{h'}$ 을 계산했을 때, σ_2^e 의 값과 동일한 결과가 나오는지 여부를 통해 ID 와 메시지(M)에 대해 검증할 수 있다.

[0108] 7. 폐기 단계

[0109] 시간(T)에 폐기될 ID 를 기존의 RL_T 에 포함하여 새로운 RL_{T+1} 을 생성한다.

[0110] 이상 설명한 기법에 대한 정확성(Correctness)은 아래와 같이 확인될 수 있다.

[0111] 검증 단계에서 계산하는 R' 은 $R' = \sigma_3 \cdot (\sigma_4 \cdot H_2(ID) \cdot H_2(ID_{KGC}))^{-\sigma_5} \bmod N$ 으로 계산되고, $\sigma_3 = ssk_2$, $\sigma_4 = ssk_3$, $\sigma_5 = ssk_4$ 이고, $ssk_2 = s^e$, $ssk_3 = \prod_{ID_i \in S'} H_2(ID_i)$, 및 $ssk_4 = c$ 이다. 또한, $S' = S \setminus \{ID, ID_{KGC}\}$ 이다. 이를 이용하여 다음과 같이 R' 을 계산할 수 있다.

$$
\begin{aligned}
R' &= \sigma_3 \cdot (\sigma_4 \cdot H_2(ID) \cdot H_2(ID_{KGC}))^{-\sigma_5} \bmod N \\
&= s^e \cdot (\prod_{ID_i \in S'} H_2(ID_i) \cdot H_2(ID) \cdot H_2(ID_{KGC}))^{-c} \bmod N \\
&= s^e \cdot (\prod_{ID_i \in S} H_2(ID_i))^{-c} \bmod N \\
&= (\prod_{ID_i \in S} r_i SK_i^c)^e \cdot (\prod_{ID_i \in S} H_2(ID_i))^{-c} \bmod N \\
&= (\prod_{ID_i \in S} r_i^e H_2(ID_i)^{edc} H_2(ID_i)^{-c}) \bmod N \\
&= \prod_{ID_i \in S} r_i^e \bmod N = R
\end{aligned}
$$

[0112]

[0113] 따라서, 올바른 서명키로 생성된 R' 에 대해서는 $H_1(R' \| T) = c = ssk_4 = \sigma_5$ 가 된다.

[0114] 검증 단계에서 계산하는 h' 은 $h' = H_3(ID, M, \sigma_1)$ 으로 계산되고, 올바른 서명키에 대해서는 $h' = H_3(ID, M, \sigma_1) = h$ 가 되고, $\sigma_2^e = (A \cdot ssk_1^h)^e = (A^e \cdot H_2(ID)^{edh}) = \sigma_1 \cdot H_2(ID)^h$ 이 되므로, 올바른 서명키로 생성된 h' 에 대해서는 $\sigma_2^e = \sigma_1 \cdot H_2(ID)^{h'}$ 이 된다.

[0115] 위의 방법을 통하여 서명($\sigma = < \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5 >$)과 시간(T)에서 폐기되지 않은 ID 에 대한 메시지(M)의 검증이 가능하다.

[0116] 본 발명은 도면에 도시된 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

[0117] 10 : 시스템

100 : 서버

110 : 셋업 모듈

130 : 개인키 생성 모듈

150 : 업데이트키 생성 모듈

170 : 폐기 모듈

300 : 제1 단말기

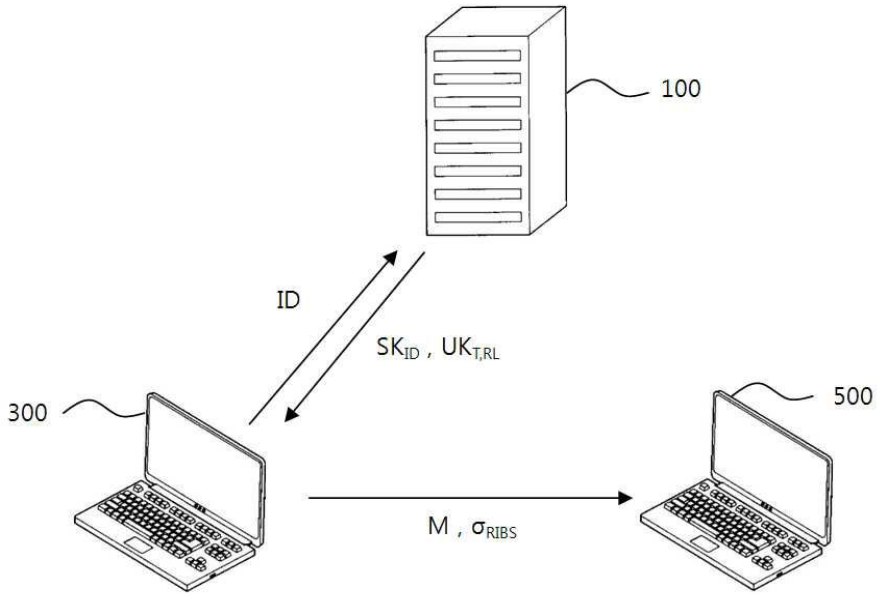
310 : 서명키 생성 모듈

330 : 서명 모듈

500 : 제2 단말기

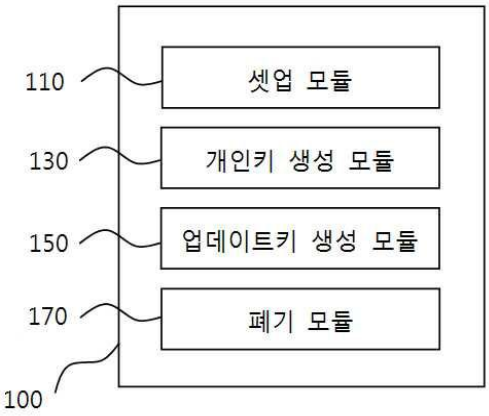
도면

도면1

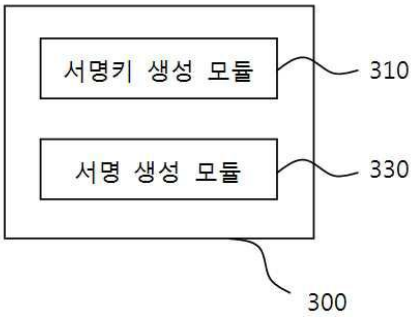


10

도면2


110
셋업 모듈
130
개인키 생성 모듈
150
업데이트키 생성 모듈
170
폐기 모듈
100

도면3


서명키 생성 모듈
310
서명 생성 모듈
330
300