(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년08월05일
(11) 등록번호 10-1425552
(24) 등록일자 2014년07월25일

(51) 국제특허분류(Int. Cl.)
H04L 9/30 (2006.01)
(21) 출원번호 10-2010-0096561
(22) 출원일자 2010년10월04일
심사청구일자 2010년10월04일
(65) 공개번호 10-2012-0035069
(43) 공개일자 2012년04월13일
(56) 선행기술조사문헌
US20100169656 A1*
JP2010103623 A
KR1020080021801 A
KR1020100018043 A
*는 심사관에 의하여 인용된 문헌
기술이전 희망 : **기술양도**, **실시권허여**, **기술지도**

(73) 특허권자
한국전자통신연구원
대전광역시 유성구 가정로 218 (가정동)
(72) 발명자
황정연
경기도 수원시 권선구 세지로12번길 25-10, 대우연립 10동 201호 (세류동)
이석준
대전광역시 유성구 배울1로 119, 대덕테크노밸리 우림필유 1207동 1102호 (용산동)
(뒷면에 계속)
(74) 대리인
제일특허법인, 김원준

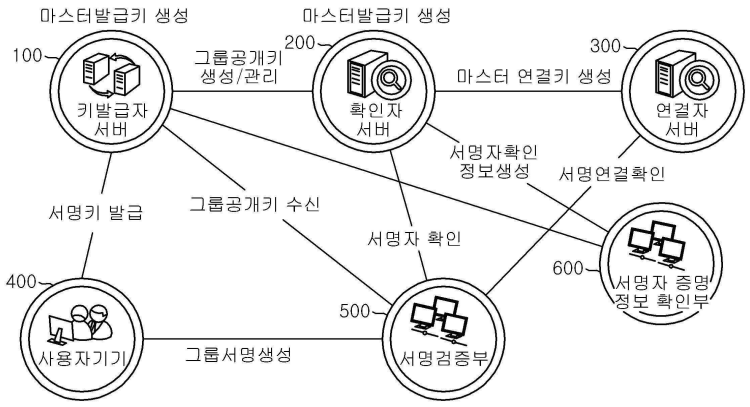
전체 청구항 수 : 총 17 항 심사관 : 이병수

(54) 발명의 명칭 **제어가능 연결성을 제공하는 그룹서명 시스템 및 방법**

(57) 요 약

본 발명은 제어가능 연결성을 제공하는 그룹서명 시스템 및 방법을 제공한다. 여기서 연결성은 주어진 "서명 값들이 한 서명키로부터 생성 되었음"를 의미한다. 본 발명에서는 사용자 가입/탈퇴에 대한 동적 맴버쉽(dynamic membership)을 지원하고 기본적으로 기존에 알려진 그룹서명의 모든 기능들을 제공하며 이에 더하여 제어가능 연결성을 제공한다. 그룹서명의 참가자들은 크게 키 발급자(issuer), 확인자(opener), 연결자(linker), 서명자(signer)(또는 사용자), 그리고 검증자(verifier)로 구성된다. 서명자는 가입을 한 후 키 발급자로부터 익명인증에 사용되는 키를 발급받고 이 후 익명성 기반의 서명을 생성하는데 사용한다. 검증자는 공개 그룹키를 이용하여 서명자가 누구인지는 모른체 서명이 정당한 사용자가 생성했음을 확인할 수 있다. 향후 키 분실 또는 사용자 탈퇴의 경우, 확인자는 자신의 확인키를 이용하여 사용자가 누구인지 증명할 수 있는 증명값들을 생성할 수 있다. 연결자는 자신의 연결키를 이용하여 주어진 두 개의 서명 값들이 서명자가 누구인지는 모른체 같은 사용자로부터 생성된 것인지 확인할 수 있다.

대 표 도 - 도1



(72) 발명자

배근태

대전광역시 서구 둔산대로117번길 66, 908호 (만년동, 골드타워)

이윤경

대전광역시 유성구 배울2로 19, 대덕테크노밸리 꿈에그린아파트 910동 1202호 (관평동)

문혜란

경기도 광주시 경충대로 1514-11, 이스트빌 3동 402호 (쌍령동)

이상우

대전광역시 유성구 배울2로 78, 대덕테크노밸리아파트 603동 804호 (관평동)

김신효

대전광역시 유성구 엑스포로 448, 404동 1106호 (전민동, 엑스포아파트)

정병호

대전광역시 유성구 가정로 306-6, 타운하우스 7동 203호 (도룡동)

조현숙

대전광역시 유성구 관평1로 12, 대덕테크노밸리7단지 금성백조아파트 701동 501호 (관평동)

이 발명을 지원한 국가연구개발사업

과제고유번호	KI001917
부처명	지식경제부
연구사업명	정보통신산업원천기술개발사업
연구과제명	익명성 기반의 u지식정보보호 기술 개발
기 여 율	1/1
주관기관	한국전자통신연구원
연구기간	2010.03.01 ~ 2011.02.28

특허청구의 범위

청구항 1

그룹서명 시스템으로서,

그룹 공개키의 제1파라미터를 생성하고 이에 대응하는 마스터 발급키를 생성하며, 사용자 기기의 가입이 있는 경우 상기 사용자에게 서명키를 발급하는 키발급자 서버와,

상기 그룹 공개키의 제2파라미터와 이에 대응하는 마스터 확인키와 마스터 연결키를 생성하는 확인자 서버와,

상기 그룹 공개키에 대응하는 유효한 두 개의 서명이 주어지는 경우 상기 마스터 연결키를 이용하여 상기 두 개의 서명이 서로 연결되었는지 확인하는 연결자 서버를 포함하며,

상기 키발급자 서버는, 그룹서명 시스템에 등록하고자 하는 사용자 기기로부터 가입 요청 메시지를 수신하는 경우 상기 가입 요청 메시지의 유효성을 검증한 후, 상기 사용자 기기로 서명키를 발급하는 것을 특징으로 하는 그룹서명 시스템.

청구항 2

제 1 항에 있어서,

상기 그룹서명 시스템은,

상기 주어진 서명에 대한 서명의 유효성을 확인하여 주는 서명검증부를 더 포함하는 그룹서명 시스템.

청구항 3

제 1 항에 있어서,

상기 그룹서명 시스템은,

상기 확인자 서버에서 생성된 서명자 확인정보의 유효성을 확인해주는 서명자증명정보 확인부를 더 포함하는 그룹서명 시스템.

청구항 4

제 1 항에 있어서,

상기 키발급자 서버는,

상기 제1파라미터와 제2파라미터를 이용하여 그룹 공개키를 생성하고, 이를 사용자 기기를 포함하는 모든 참가자에게 제공하는 그룹서명 시스템.

청구항 5

삭제

청구항 6

제 1 항에 있어서,

상기 가입 요청 메시지에는,

개인키 소유 증명 정보와 키발급과 관련된 증명정보가 포함되는 그룹서명 시스템.

청구항 7

제 1 항에 있어서,

상기 키발급자 서버는,

세션의 변경에 따라 키폐기가 발생하는 경우, 상기 그룹 공개키를 갱신시키는 그룹서명 시스템.

청구항 8

제 1 항에 있어서,

상기 키발급자 서버는,

이선형군 쌍과 상기 이선형군 쌍과 결합되는 이선형 함수를 이용하여 상기 마스터 발급키를 정의하는 그룹서명 시스템.

청구항 9

제 1 항에 있어서,

상기 확인자 서버는,

상기 유효한 서명이 주어지는 경우 상기 마스터 확인키를 이용하여 서명자가 누구인지 확인해주는 증명정보를 출력하는 그룹서명 시스템.

청구항 10

제 1 항에 있어서,

상기 사용자 기기와 키발급자 서버간에는,

인증을 위한 보안 채널이 형성되는 그룹서명 시스템.

청구항 11

키발급자 서버에서 그룹 공개키의 제1파라미터를 생성하고 이에 대응하는 마스터 발급키를 생성하는 단계와,

상기 키발급자 서버에서, 그룹서명 시스템에 등록하고자 하는 사용자 기기로부터 가입 요청 메시지를 수신하는 경우 상기 가입 요청 메시지의 유효성을 검증한 후, 상기 사용자 기기로 서명키를 발급하는 단계와,

확인자 서버에서 상기 그룹 공개키의 제2파라미터와 이에 대응하는 마스터 확인키와 마스터 연결키를 생성하는 단계와,

연결자 서버에서 유효한 두 개의 서명이 주어지는 경우 상기 마스터 연결키를 이용하여 상기 두 개의 서명이 서로 연결되었는지 확인하는 단계

를 포함하는 그룹서명 방법.

청구항 12

제 11 항에 있어서,

상기 확인단계이 후, 상기 유효한 서명이 주어지는 경우, 상기 확인자 서버에서 상기 마스터 확인키를 이용하여 서명자가 누구인지 확인해주는 증명정보를 출력하는 단계를 더 포함하는 그룹서명 방법.

청구항 13

키발급자 서버에서 그룹 공개키의 제1파라미터를 생성하고 마스터 발급키를 정의하는 단계와,

확인자 서버에서 마스터 확인키와 마스터 연결키를 각각 정의하고, 상기 그룹 공개키의 제2파라미터를 생성하여 상기 키발급자 서버로 제공하는 단계와,

상기 키발급자 서버에서 상기 제1파라미터와 제2파라미터를 조합하여 상기 그룹 공개키를 생성하는 단계를 포함하며,

상기 마스터 발급키는, 이선형군 쌍과 상기 이선형군 쌍과 결합되는 이선형 함수를 이용하여 정의되는 것을 특징으로 하는 그룹 공개키 생성 방법.

청구항 14

제 13 항에 있어서,

상기 그룹 공개키의 생성단계 이후, 상기 키발급자 서버에서 상기 그룹 공개키를 그룹서명 시스템에 등록된 사용자 기기로 제공하는 단계를 더 포함하는 그룹 공개키 생성 방법.

청구항 15

삭제

청구항 16

키발급자 서버는 세션이 변경되는 경우 키 갱신을 위한 폐기목록을 공개하는 단계와,

상기 키발급자 서버에서 새로운 그룹 공개키 생성하여 사용자 기기로 제공하는 단계와,

상기 사용자 기기에서 상기 새로운 그룹 공개키로 그룹 공개키를 갱신하는 단계와,

상기 사용자 기기에서 상기 새로운 그룹 공개키에 대응하는 서명키를 갱신하는 단계

를 포함하는 그룹 공개키 갱신 방법.

청구항 17

키발급자 서버에서 사용자 기기로부터 가입 요청 메시지를 수신하는 단계와,

상기 키발급자 서버에서 상기 가입 요청 메시지의 유효성을 검증하는 단계와,

상기 사용자 기기로부터 상기 유효성이 검증된 상기 가입 요청 메시지에 대한 서명을 전송받는 단계와,

상기 키발급자 서버에서 상기 서명의 유효성을 검증하여 상기 사용자 기기를 등록하는 단계와,

상기 사용자 기기에서 그룹 공개키에 대응하는 비밀 서명키를 생성하는 단계

를 포함하는 서명키 생성 방법.

청구항 18

제 17 항에 있어서,

상기 사용자 기기와 키발급자 서버간에는,

인증을 위한 보안 채널이 형성되는 서명키 생성 방법.

청구항 19

제 17 항에 있어서,

상기 가입 요청 메시지에는,

개인키 소유 증명 정보와 키발급과 관련된 증명정보가 포함되는 서명키 생성 방법.

청구항 20

삭제

명 세 서

기 술 분 야

[0001] 본 발명은 암호학적 그룹서명 기법에 관한 것으로, 특히 그룹의 정당한 사용자가 생성한 서명은 겉으로는 단지 그룹 맴버 중 한 사용자가 메시지에 대한 서명을 생성했음을 증명하지만 특별한 확인키(opening key)가 주어질 경우 서명자를 확인할 수 있으며 또한 특별한 연결키(linking key)가 주어질 경우에는 서명 값들이 서로 연결됨(즉, 한 서명자 또는 서명자키에 생성되었음)을 확인할 수 있도록 하여 다양한 수준으로 제어 가능한 익명성 및 연결성을 제공하는 그룹서명 시스템 및 방법에 관한 것이다.

배 경 기 술

[0002] 통상적으로, 그룹서명 기법은 사용자의 프라이버시(privacy)를 보호하기 위한 매우 중요한 암호학적 인증 기법 중 하나로 폭넓게 연구되고 있다. 이러한 그룹서명 기법은 1991년 Chaum과 Heyst에 의해 최초로 개념이 제시된 이후로 많은 발전을 이루어 왔으며 안전성 요구사항에 대한 형식적 모델은 물론 구체적인 기법들도 많이 제안되어 오고 있다.

[0003] 또한, 개인정보의 등록 및 확인 과정을 통해 개인정보 노출, 서비스 제공자의 과도한 개인정보 수집 및 관리 부주의로 인한 유출 등과 같은 많은 문제점을 안고 있는 아이디/패스워드(id/password) 인증 기법, 실명 기반 PKI 인증 기법과, 광범위한 행위 추적의 문제를 가지고 있는 i-Pin 기법 등을 대체할 수 있는 효과적인 익명 인증 기법으로 최근 활발히 연구되고 있다.

[0004] 하지만, 전통적인 그룹서명 기법은 서명자의 아이디를 은닉/복구하는 이분법적인 구조로 단순하게 익명성을 다루고 있어 실제적인 응용 환경에서 채택되기에는 충분하지 않다. 이는 서비스를 이용하는 쪽에서는 더욱 완전한 익명성의 장점을 선호하지만 서비스를 제공하는 측면에서는 익명성만으로 서비스 제공으로부터 얻은 본래의 목적을 달성하기 쉽지 않기 때문이다.

[0005] 예를 들어, 웹 기반의 익명 인증 서비스를 생각해 보면 양질의 서비스는 물론 다양한 개인화된 서비스를 구성하기 어렵다. 또한 데이터 마이닝의 경우 익명 인증 데이터로부터 유용한 정보를 얻어내기 힘들다.

[0006] 따라서 이러한 문제점을 해결하기 위해서 실용적인 관점에서 다양한 익명성 수준을 제어할 수 있으며 성능 관점에서도 우수한 그룹서명 기법 등의 개발이 절실히 필요하다.

[0007] 또한, 위에서 언급한 익명성 특성을 제공하는 효율적인 그룹서명 기법들을 설계 및 개발하기 위해서는, 기존에 알려진 이선형 군 (Bilinear Group) 상의 선형 암호 (Linear Encryption, LE) 기법만으로는 충분하지 않으며, 효율적으로 다중의 메시지 쌍을 암호화 시킬 수 있는 구조적으로 유연한 새로운 암호 기법 등이 동시에 개발되어야 한다.

발명의 내용

해결하려는 과제

[0008] 한편, 현재까지 익명 인증을 제공하기 위해서 다양한 그룹 서명 기법들이 제안되고 있으나 익명성을 처리하는 방식이 생성된 서명 속에 서명자의 아이디를 감추고 확인키 (master opening key)가 주어지면 다시 서명자 아이디를 복구할 수 있는 단순한 구조를 채택하고 있다. 이러한 단편적인 방식은 실제 응용 환경에서 활용되기에는 미흡하다. 서비스를 이용하는 쪽에서는 익명성의 장점을 선호하지만 서비스를 제공하는 측면에서는 단순한 익명성만으로 서비스 제공을 위한 유용한 목적을 달성하기 쉽지 않은 문제점이 있다. 예를 들어, 웹 기반의 익명 인증 서비스를 고려할 경우, 서비스 제공자는 익명화된 형태의 사용자 정보(예를 들어, 사용자의 소비 패턴 등)가 필요하며 이것이 뒷받침되지 않을 경우 다양한 개인화된 서비스 및 이와 결합된 양질의 서비스를 구성하기 어렵다. 또한 데이터 마이닝(Data Mining)의 경우 익명 인증 데이터로부터 개발자가 원하는 방식으로 유용한 정보를 얻어내기 힘들다.

[0009] 따라서, 본 발명의 목적은, 기존의 그룹 서명 기법이 갖는 익명성의 제한된 제어 모습을 극복하고 익명성의 수준을 다각화하는데 필요한 새로운 형태의 그룹 기반 익명 서명 기법을 제공하는 것이다. 구체적으로, 제어가능 연결성을 도입하여 익명성의 개념을 다양한 수준으로 세분화하고 이에 대응하는 제어 방법을 제공하는 그룹서명 시스템 및 방법을 제공하고자 한다. 즉, 특별한 키가 주어졌을 경우에만 서명자 아이디 또는 서명 값들 사이의 연결 정보가 확인되며 따라서 이런 측면에서 익명성은 제어 가능하다.

[0010] 본 발명의 다른 목적은, 선형결합암호 (linear combination encryption, LCE 기법과 이를 확장한 하이브리드 선형결합암호 (hybrid linear combination encryption, HLCE) 기법을 제안한다. 이 기법들은 그룹서명 기법의 설계를 위해 매우 필수적으로 이용되며 또한 독립적으로 다른 암호학적 기법의 설계를 위해 중요하게 이용될 수 있다. 이러한 암호 기법들은 결정적 DH (Decisional Diffie-Hellman, DDH) 문제가 쉬운 대수적 군(algebraic group), 예를 들어, 이선형함수(bilinear pairings)를 위해 정의되는 이선형 군(bilinear group)에서 안전하게 다중의 메시지를 효율적으로 암호화할 수 있다.

과제의 해결 수단

[0011] 상술한 본 발명은 그룹서명 시스템으로서, 그룹 공개키의 제1파라미터를 생성하고 이에 대응하는 마스터 발급키를 생성하며, 사용자 기기의 가입이 있는 경우 상기 사용자에게 서명키를 발급하고 사용자 키 폐기가 발생할 경우 키 갱신을 위해 폐기 목록 등을 관리하는 키발급자 서버와, 상기 그룹 공개키의 제2파라미터와 이에 대응하는 마스터 확인키와 마스터 연결키를 생성하는 확인자 서버와, 상기 그룹 공개키에 대응하는 유효한 두 개의 서명이 주어지는 경우 상기 마스터 연결키를 이용하여 상기 두 개의 서명들이 서로 연결되었는지 확인하는 연결자 서버를 포함한다.

[0012] 또한, 상기 그룹서명 시스템은, 발급된 서명키를 이용하여 서명을 생성하는 사용자 기기와, 상기 주어진 서명에 대한 서명의 유효성을 확인하여 주는 서명검증부와, 상기 확인자 서버에서 생성된 서명자 확인정보의 유효성을 확인해주는 서명자증명정보 확인부를 더 포함하는 것을 특징으로 한다.

[0013] 또한, 상기 키발급자 서버는, 상기 제1파라미터와 제2파라미터를 이용하여 그룹 공개키를 생성하고, 이를 사용자 기기를 포함한 모든 참가자에게 제공하는 것을 특징으로 한다.

[0014] 또한, 상기 키발급자 서버는, 이선형군 쌍과 상기 이선형군 쌍과 결합되는 이선형 함수를 이용하여 상기 마스터 발급키를 정의하는 것을 특징으로 한다.

[0015] 또한, 상기 키발급자 서버는, 그룹서명 시스템에 등록하고자 하는 사용자 기기로부터 가입 요청 메시지를 수신하는 경우 상기 가입 요청 메시지의 유효성을 검증한 후, 상기 사용자 기기로 서명키를 발급하는 것을 특징으로 한다.

[0016] 또한, 상기 가입 요청 메시지에는, 개인키 소유 증명 정보와 키 발급과 관련된 증명정보 등을 포함되는 것을 특징으로 한다.

[0017] 또한, 상기 사용자 기기와 키발급자 서버간에는, 인증을 위한 보안 채널이 형성되는 것을 특징으로 한다.

[0018] 또한, 상기 키발급자 서버는, 키폐기가 발생하는 경우 세션을 변경하고, 폐기 목록을 공개하여 상기 그룹 공개

키와 사용자 비밀키를 갱신하도록 하는 것을 특징으로 한다.

[0019] 또한, 상기 확인자 서버는, 상기 유효한 서명이 주어지는 경우 상기 마스터 확인키를 이용하여 서명자가 누구인지 확인해주는 증명정보를 출력하는 것을 특징으로 한다.

[0020] 또한 본 발명은 그룹서명 시스템에서 그룹서명 방법으로서, 키발급자 서버에서 그룹 공개키의 제1파라미터를 생성하고 이에 대응하는 마스터 발급키를 생성하는 단계와, 사용자 기기의 가입이 있는 경우 상기 사용자 기기로 서명키를 발급하는 단계와, 확인자 서버에서 상기 그룹 공개키의 제2파라미터와 이에 대응하는 마스터 확인키와 마스터 연결키를 생성하는 단계를 포함한다.

[0021] 또한, 상기 연결자 서버에서 유효한 두 개의 서명이 주어지는 경우 상기 마스터 연결키를 이용하여 상기 두 개의 서명이 서로 연결되었는지 확인하는 단계를 포함한다.

[0022] 또한, 상기 확인자 서버에서, 유효한 서명이 주어지는 경우 상기 마스터 확인키를 이용하여 서명자가 누구인지 확인해주는 증명정보를 출력하는 단계를 더 포함하는 것을 특징으로 한다.

[0023] 또한, 본 발명은 그룹서명 시스템에서 그룹 공개키 생성 방법으로서, 키발급자 서버에서 그룹 공개키의 제1파라미터를 생성하고 마스터 발급키를 정의하는 단계와, 확인자 서버에서 마스터 확인키와 마스터 연결키를 각각 정의하고, 상기 그룹 공개키의 제2파라미터를 생성하여 상기 키발급자 서버로 제공하는 단계와, 상기 키발급자 서버에서 상기 제1파라미터와 제2파라미터를 조합하여 상기 그룹 공개키를 생성하는 단계를 포함한다.

[0024] 또한, 상기 그룹 공개키의 생성단계 이후, 상기 키발급자 서버에서 상기 그룹 공개키를 상기 그룹서명 시스템에 등록된 사용자 기기로 제공하는 단계를 더 포함하는 것을 특징으로 한다.

[0025] 또한, 상기 마스터 발급키는, 이선형군 쌍과 상기 이선형군 쌍과 결합되는 이선형 함수 및 해쉬 함수를 이용하여 정의되는 것을 특징으로 한다.

[0026] 또한, 본 발명은 그룹서명 시스템에서 그룹 공개키 갱신 방법으로서, 키발급자 서버는 세션이 변경되는 경우 키 갱신을 위한 폐기 목록을 공개하는 단계와, 상기 키발급자 서버에서 새로운 그룹 공개키를 생성하여 사용자 기기로 제공하는 단계와, 상기 사용자 기기에서 상기 새로운 그룹 공개키로 그룹 공개키를 갱신하는 단계와, 상기 사용자 기기에서 상기 새로운 그룹 공개키에 대응하는 서명키를 갱신하는 단계를 포함한다.

[0027] 또한, 본 발명은 그룹서명 시스템에서 서명키 생성 방법으로서, 키발급자 서버에서 사용자 기기로부터 가입 요청 메시지를 수신하는 단계와, 상기 키발급자 서버에서 상기 가입 요청 메시지의 유효성을 검증하는 단계와, 상기 사용자 기기로부터 상기 유효성이 검증된 상기 가입 요청 메시지에 대한 서명을 전송받는 단계와, 상기 키발급자 서버에서 상기 서명의 유효성을 검증하여 상기 사용자 기기를 등록하는 단계와, 상기 사용자 기기에서 그룹 공개키에 대응하는 비밀 서명키를 생성하는 단계를 포함한다.

[0028] 또한, 상기 사용자 기기와 키발급자 서버간에는, 인증을 위한 보안 채널이 형성되는 것을 특징으로 한다.

[0029] 또한, 상기 가입 요청 메시지에는, 개인키 소유 증명 정보가 포함되는 것을 특징으로 한다.

[0030] 또한, 본 발명은 그룹서명 시스템에서 메시지 암호 방법으로서, 확인자 서버는 그룹 공개키의 제2파라미터에 포함되는 확인자 서버의 공개키를 정의하고 이에 대응되는 비밀키를 저장하는 단계와, 사용자기기가 상기 공개키를 이용하여 (그룹서명에 포함될) 자신의 서명키의 암호문을 출력하는 단계와, 확인자 서버가 상기 비밀키를 이용하여 상기 암호문으로부터 상기 메시지, 즉 서명키를 복구하는 단계를 포함한다. 상기 메시지 암호 방법은 상기 그룹서명 시스템과 독립적으로 정의되어 이용되거나 다른 시스템과 결합되어 이용될 수 있다.

발명의 효과

[0031] 본 발명에서는 제어가능 연결성을 포함하여 익명성을 다양한 수준으로 제어할 수 있는 그룹서명 기법을 통해 사용자 프라이버시를 보호할 수 있는 방법을 제공하며 기존의 다양한 응용서비스들과 결합되어 또는 독립적으로 쉽게 채택될 수 있는 이점이 있다.

[0032] 본 발명에 따르면, 제공되는 제어가능 연결성을 통해 익명성은 다양한 조건, 정책과 결합되어 세분화 될 수 있다. 기본적으로, 구성 기법은 기존에 알려진 그룹서명 기법의 모든 기능 및 보안특성들을 제공한다. 즉, 주어진 서명 값으로부터 단순히 서명자의 확인이나 연결성 정보의 확인은 불가능하다. 하지만 특별한 키들이 주어졌을 경우, 즉, 특별한 확인키 (opening key)가 주어질 경우 서명자를 확인할 수 있으며 또한 특별한 연결키

(linking key)가 주어질 경우에는 서명자는 서명 값들이 서로 연결됨(즉, 한 서명자 또는 서명자키로 생성되었음)을 확인할 수 있다.

[0033] 또한, 본 발명을 통해 결정적 DH (Decisional Diffie-Hellman) 문제가 쉬운 대수적 군(algebraic group)에서 안전하게 메시지를 암호/복호화 할 수 있는 방법을 제공하며, 또한 교통망을 위한 익명 인증(VSC), 미래인터넷 익명 패킷 인증 등 기존의 그룹 서명 기법이 적용 가능했던 용용 분야들뿐만 아니라 익명성 기반 웹 서비스, 의료정보보호, 클라우드 컴퓨팅 인증 등 다양한 차세대 IT 응용분야에서 활용될 수 있는 이점이 있다.

도면의 간단한 설명

[0034] 도 1은 본 발명의 실시 예에 따른 제어가능 연결성을 제공하는 그룹서명 시스템의 구성도,

도 2는 본 발명의 실시 예에 따라 그룹서명 시스템의 그룹 공개키와 마스터 발급키, 마스터 확인키, 마스터 연결키 들을 생성하는 동작 제어 흐름도,

도 3은 본 발명의 실시 예에 따라 키발급자 서버와 사용자 기기간 상호적으로 서명키를 생성하는 동작 제어 흐름도,

도 4는 본 발명의 실시 예에 따라 정당한 사용자 기기가 주어진 메시지에 대해 그룹서명을 생성하는 동작 제어 흐름도,

도 5는 본 발명의 실시 예에 따라 정당한 사용자 기기가 생성한 한 메시지에 대한 그룹서명에 대해 서명검증부를 통해 검증하는 동작 제어 흐름도,

도 6은 본 발명의 실시 예에 따라 정당한 사용자 기기가 생성한 한 메시지에 대한 그룹서명에 대해 증명정보를 생성하고, 증명정보의 유효성을 검증하는 동작 제어 흐름도,

도 7은 본 발명의 실시 예에 따라 정당한 사용자 기기가 생성한 두 그룹서명들에 대해 마스터 연결키를 이용하여 서명값들이 서로 연결되었는지를 확인해 주는 동작 제어 흐름도,

도 8은 본 발명의 실시 예에 따라 키 폐기가 발생한 경우 키발급자 서버가 관련 정보를 공개하고 그룹 공개키를 갱신하고 정당한 사용자 기기는 자신의 서명키를 갱신하는 동작 제어 흐름도,

도 9는 본 발명의 실시 예에 따른 선형결합암호기법(Linear Combination Encryption(LCE) Scheme) 을 설명하는 동작 제어 흐름도,

도 10은 본 발명의 실시 예에 따른 하이브리드선형결합암호기법(Hybrid Linear Combination Encryption(HLCE) Scheme)을 설명하는 동작 제어 흐름도.

발명을 실시하기 위한 구체적인 내용

[0035] 이하, 첨부된 도면을 참조하여 본 발명의 동작 원리를 상세히 설명한다. 하기에서 본 발명을 설명함에 있어서 공지 기능 또는 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략할 것이다. 그리고 후술되는 용어들은 본 발명에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다.

[0036] 도 1은 본 발명의 실시 예에 따른 제어가능 연결성을 제공하는 그룹서명 시스템의 구성을 도시한 것이다.

[0037] 도 1을 참조하면, 본 발명에 따른 기법의 참가 구성요소들은 크게 키발급자 서버(issuer sever)(100), 확인자 서버(opener server)(200), 연결자 서버(linker server)(300), 사용자 기기(400), 서명검증부(500), 그리고 서명자증명정보 확인부(600)로 구성된다. 여기서 서버는 주어진 입력값에 대해 특별한 값을 출력하는 알고리즘의 개념으로 사용 가능하다. 다만 참가자들의 구성은 이에 한정되는 것은 아니며 설계되는 방식에 따라 유연하게 참가자의 역할이 새로운 주체의 정의로 분리되거나 또는 통합될 수 있으며(예, 키발급자 서버와 확인자 서버의 통합 또는 확인자 서버와 연결자 서버의 통합), 필요 시 알려진 실명 인증 기법과 연동을 위해 새로운 참가자를 정의할 수 있다.

[0038] 키발급자 서버(100)는 신뢰된 주체로써 초기에 그룹 공개키의 제1파라미터 (group public parameters, gpp1)를

생성하고, 이에 대응하는 마스터 발급키(master issuing key, mik)를 생성한다. 키발급자 서버(100)는 새로운 사용자 기기(400)의 가입이 있을 경우 상호적인 프로토콜을 수행한 후, 사용자에게 서명키를 발급한다.

[0039] 또한, 키폐기(revocation)가 발생할 경우, 키발급자 서버(100)는 폐기목록을 포함한 관련 정보를 공개하고 참가자들이 필요한 경우 키 값들이 갱신되도록 한다.

[0040] 확인자 서버(200)는 초기에 그룹공개키의 제2파라미터(group public parameters, gpp2)와 이에 대응하는 마스터 확인키(master opening key, mok)와 마스터 연결키(master linking key, mlk)를 생성한다. 마스터 연결키는 연결자서버(300)에게 제공한다. 유효한 서명이 주어진 경우, 마스터 확인키를 이용하여 서명자가 누구인지 확인해 주는 증명정보를 출력한다. 출력된 증명정보는 누구든지 공개적으로 확인 가능하다.

[0041] 연결자(Linker)서버(300)는 초기에 마스터 연결키(master linking key, mlk)를 확인자 서버(400)로부터 받는다. 유효한 두 개의 서명이 주어진 경우, 마스터 연결키를 이용하여 서로 연결되었는지(즉, 두 명 서명을 한 서명자가 생성했는지) 확인할 수 있다.

[0042] 사용자 기기(400)는 정당한 그룹의 맴버로 가입하여 키발급자 서버(100)로부터 서명키를 발급받을 수 있다.

[0043] 이 때, 사용자 기기(400)와 키발급자 서버(100)는 상호적인 프로토콜을 수행한다. 이 후, 사용자 기기(400)는 발급받은 서명키를 이용하여 주어진 메시지에 대한 서명 값(group signature)을 생성한다. 키폐기(revocation)가 발생할 경우, 사용자 기기(400)는 키발급자 서버(100)로부터 공개된 폐기정보를 이용하여 키값들을 갱신한다. 서명검증부(500)는 주어진 서명에 대한 서명의 유효성을 확인해 주는 알고리즘이다. 서명자증명정보 확인부(600)는 확인자 서버(200)으로부터 생성된 서명자 확인정보의 유효성을 검증해 주는 알고리즘이다.

[0044] 키발급자 서버(100)는 위에서 생성된 제1파라미터(gpp1)과 제2파라미터(gpp2)를 조합하여 그룹 공개키(group public key, gpk)로 정의하며, 이와 같이 정의된 그룹 공개키를 그룹서명 시스템내 모든 참가 구성요소들에게 공개한다. 즉, gpk={gpp1,gpp2}이다. 향후, 키폐기가 발생할 때 마다, gpk는 갱신된다.

[0045] 도 2는 본 발명의 실시 예에 따라 키 발급자 서버(100)와 확인자 서버(200)가 그룹 공개키의 파라미터들 gpp1, gpp2으로 구성된 초기그룹 공개키 gpk와 이에 대응하는 마스터 발급키(mik), 마스터 확인키(mok), 마스터 연결키(mlk)를 생성하는 동작 제어 흐름을 도시한 것이다.

[0046] 이하, 도 2를 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0047] 초기에 보안 파라미터 K를 입력으로 받아서 키 발급자 서버(100)는 다음을 수행한다. 먼저 이선형군(bilinear group) 쌍 $(\overrightarrow{G_1},\overrightarrow{G_2})$와 이와 결합된 이선형 함수(bilinear map) $e:\overrightarrow{G_1}\times\overrightarrow{G_2}\to\overrightarrow{G_T}$와 해쉬 함수 $H:\{0,1\}^*\to\overrightarrow{Z_p^*}$를 생성한다. 임의의 원소 $h_1\in\overrightarrow{G_2}$과 임의의 $g_1,g_2,g_3,g\in\overrightarrow{G_1}$을 선택한다. 또한 임의의 $\theta\in\overrightarrow{Z_p^*}$를 선택하고 $h_0=h_1^{\theta}$을 계산한 후 $mik=\theta$를 마스터 발급키로 정의한다(S200).

[0048] 또한, 확인자 서버(200)는 임의의 $\eta_1,\eta_2,\xi_1,\xi_2\in\overrightarrow{Z_p^*}$을 선택하고 $U=h_1^{\xi_1}$, $V=h_1^{\xi_2}$을 계산한다. 또한 임의의 $u,v\in\overrightarrow{G}$을 선택하고 $w_1=u^{\eta_1}$, $w_2=v^{\eta_2}$, $d_1=u^{\xi_1}$, $d_2=v^{\xi_2}$ 을 계산한다. 여기서, $mok=(\eta_1,\eta_2,\xi_1,\xi_2)$는 마스터 확인키로, $mlk=(U,V)$은 마스터 연결키로 각각 정의된다. 그리고, 제2파라미터 gpp2=(u,v,w_1,w_2,d_1,d_2)를 키 발급자 서버(100)에게 송신한다(S202).

[0049] 키발급자 서버(100)는 확인자 서버(200)로부터 수신된 gpp2=(u,v,w_1,w_2,d_1,d_2)를 자신의 gpp1=$(e,\overrightarrow{G_1},\overrightarrow{G_2},g_1,g_2,g_3,g,h_1,h_0,H)$ 와 조합하여 초기 그룹 공개키 $gpk=(e,\overrightarrow{G_1},\overrightarrow{G_2},g_1,g_2,g_3,g,h_1,h_0,H,u,v,w_1,w_2,d_1,d_2)$를 만들고 공개적으로 이용할 수 있게 한다(S204). 이와 같은 초기 공개키는 키 폐기 사건이 발생할 때 마다 갱신된다. 편의상 초기 그룹 공개키를 gpk_0으로 나타내자. 그룹 공개키에서 키 발급자 서버(100)와 확인자 서버(200)가 관리하는 파라미터들은 공개적으로 인증된 방식으로 "검증가능하다"고 가정한다.

[0050] 도 3은 본 발명의 실시 예에 따라 키발급자 서버(100)가 사용자 기기(400)와 상호적으로 서명키를 생성하는 동작 제어 흐름을 도시한 것이다.

[0051] 이하, 도 3을 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0052] 먼저, 새롭게 그룹에 가입하려는 사용자 기기(400)와 키발급자 서버(100)는 상호적으로 다음과 같은 작업을 수

행한다. 두 참가자 사이에는 인증 및 보안채널이 형성되어 있다고 가정한다. 아래의 설명에서 Ext-Commit는 완전한 바인딩(perfect binding), 계산적 하이딩(computationally hiding)을 제공하는 추출 가능한 위탁 기법(extractable commitment scheme)을 나타낸다. 트랩도어(trapdoor) 정보가 주어지면 위탁된 값을 복구할 수 있다. NIZKEqDL(a,b,c)는 a에 위탁된 값과 $\log_c b$이 동일함을 증명하는 비상호적인 영지식증명기법(non-interactive zero-knowledge proof)을 나타낸다.

[0053] 또한, NIZKEqDL(B,D)는 $\log_D B$에 대한 지식을 증명하는 비상호적인 영지식증명기법을 나타낸다. 초기 그룹 공개키 $gpk_0=(e,g_1,g_2,g_3,T)$와 현재의 그룹 공개키 $gpk_k=(e,\widetilde{g_1},\widetilde{g_2},\widetilde{g_3},T)$가 주어졌다고 하자. (여기서 $T=(g,h_1,h_0,u,v,w_1,w_2,d_1,d_2,H)$이다.) 다음에서 사용자 기기(400)는 일반적인 서명 기법(PKI기반의 형태 가능) $\sum$=(KGen, Sign, Vrfy)을 이용한다. 다음 설명에서는 사전에 각 사용자 기기(400)는 서명 기법 $\sum$=(KGen, Sign, Vrfy)을 사용하기 위한 공개키와 비밀키 쌍을 생성했다고 가정한다.

[0054] (1) 사용자 기기(400)는 임의의 난수 $z_i \in \overrightarrow{Z_p^*}$를 선택하고 $upk[i]=Z_i=g_3^{z_i}$을 계산한다. 그리고, T_UO(c_U=Ext-Commit(z_i), NIZKEqDL(c_U,Z_i,g_3)) 를 생성한 후, 가입 요청 메시지 (Join, ID_i, ($upk[i]=Z_i$, T_U))를 키 발급자 서버(100)에게 보낸다(S300). 여기서, ($upk[i]=Z_i$, T_U)는 개인키 소유 증명(proof of possession, POP) 역할을 한다.

[0055] (2) 키발급자 서버(100)는 가입 요청 메시지 (Join, ID_i, ($upk[i]=Z_i$, T_U))를 수신한 후, 정해진 방식을 따라서 ($upk[i]=Z_i$, T_U)의 유효성을 검증한다. 유효하면, 사용자 등록 목록 REG에서 (ID_i, $H(g^{y_i})$, y_i, ... , $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$, ..)이 있는지 확인한다. 만일 존재하면, $x_i \in \overrightarrow{Z_p^*}$을 선택하고 $A_i=(g_1 g_2^{-y_i} Z_i^{-1})^{1/(\theta+x_i)} \in \overrightarrow{G_1}$을 계산한다. 만일 존재하지 않으면 $x_i, y_i \in \overrightarrow{Z_p^*}$을 선택하고 $A_i=(g_1 g_2^{-y_i} Z_i^{-1})^{1/(\theta+x_i)} \in \overrightarrow{G_1}$을 계산한다. 또한, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{x_i}$, $B=e(g_1 g_2^{-y_i} Z_i^{-1}, h_1)/e(A_i, h_\theta)$, $D=e(A_i, h_1)$을 계산하고 T_J=NIZKPoKDL($Y_{1,i}$,g_2)와 V_J=NIZKPoKDL(B,D)을 생성한다. 키발급자 서버(100)는 (A_i, T_J, V_J, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{x_i}$)을 사용자 기기(400)에게 송신한다(S302).

[0056] (3) 사용자 기기(400)는 (A_i, T_J, V_J, $Y_{1,i}=g_2^{y_i}$, $X_{2,i}=h_1^{x_i}$)을 수신한 후, $B=e(g_1 g_2^{-y_i} Z_i^{-1}, h_1)/e(A_i, h_\theta)$와 $D=e(A_i, h_1)$을 계산하고, T_J, V_J가 유효한지 확인한다. 또한 등식 $e(A_i, X_{2,i} h_\theta)=e(g_1 Y_{1,i}, g_3^{-z_i}, h_1)$이 성립하는지 확인한다. 만일 모든 검증이 성공 적이면 서명 $\sigma_{2,i}$O Signski(A_i, $upk[i]=Z_i$, $Y_{1,i}=g_2^{y_i}$, $X_{2,i}=h_1^{y_i}$)을 생성하고, $\sigma_{2,i}$를 키발급자 서버(100)에게 보낸다(S304).

[0057] (4) 키발급자 서버(100)는 서명 $\sigma_{2,i}$를 수신한 후, 서명의 유효성을 검증한다. 만일 서명이 유효하면, 키 발급자 서버(100)는 사용자 기기(400)에 (x_i, y_i)을 송신한다(S306).

[0058] (5) 사용자 기기(400)는 (x_i, y_i)을 수신한 후, 사용자 키 갱신 알고리즘을 이용하여 현재의 그룹 공개키에 대응하는 $\widetilde{gpk}=(g''_1 g''^{-y}_2 g''^{-z}_3)^{1/(\theta+x)}$을 계산한다. 다음 두 등식이 성립하는지 확인한다: $e(A_i, h_1^{x_i} h_\theta)=e(g_1, g_2^{-y_i} g_3^{-z_i}, h_1)$, $e(A_i, \widetilde{w_1})=e(\widetilde{g_1}, h_1)$.

[0059] 만일 두 등식이 성립하면 $usk[i]=(\widetilde{gpk}, x_i, y_i, z_i, A_i)$을 현재의 그룹 공개키에 대응하는 비밀 서명키로 안전하게 저장한다(S308). 마지막으로 사용자 기기(400)는 $e(X_{1,i}, h_1)=e(g, X_{2,i})$, $e(Y_{1,i}, h_1)=e(g_2, Y_{2,i})$이 성립하면 메시지 ($A_i$, $upk[i]=Z_i$, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$) 에 대한 서명 $\sigma_{judge,i}$O Signski(A_i, $upk[i]=Z_i$, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$)을 생성하고, $\sigma_{judge,i}$를 키발급자 서버(100)에게 보낸다(S310).

[0060] (6) 키발급자 서버(100)는 서명 $\sigma_{judge,i}$를 수신한 후, 서명의 유효성을 검증한다. 만일 서명이 유효하면, (ID_i, $H(g^{y_i})$, y_i, A_i, $upk[i]=Z_i$, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$, $\sigma_{judge,i}$)을 사용자 등록 목록 REG에 추가한다

(312).

[0061] 위의 설명에서는 마스터 연결키 mlk가 주어지면 사용자 기기(400)의 맴버쉽에 관계없이 연결성을 확인할 수 있는 구조를 제공하고 있다. 이를 수정하여, 만일 사용자 기기(400)가 정당한 맴버로 가입되어 있는 동안에만 연결성을 제공하는 경우를 고려한다면, (2)번 단계에서 키발급자 서버(100)는 사용자 가입시 마다 $y_i, H(\overline{Z_i})$을 매번 새로운 값으로 선택할 수 있다.

[0062] 도 4는 본 발명의 실시 예에 따라 정당한 사용자 기기(400)가 주어진 메시지에 대해 그룹서명을 생성하는 동작 제어 흐름을 도시한 것이다.

[0063] 이하, 도 4를 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0064] 먼저 메시지 M이 수신되는 경우 사용자 기기(400)는 주어진 현재의 그룹 공개키 gpk, 이에 대응하는 사용자 비밀 서명키 $usk[i]=(\mathcal{A}, x, y, z, A)$, 그리고 메시지 M을 입력으로 받는다(S400). 이어 사용자 기기(400)는 위와 같은 입력에 대해 다음과 같이 서명 σ을 생성한다(S402). 즉, 사용자 기기(400)는 먼저, 임의의 난수 $\alpha, \beta, \overline{Z_p}$를 선택하고 $D_1 = u^{\alpha}$, $D_2 = v^{\beta}$, $D_3 = \mathcal{A} w_1^{\alpha} w_2^{\beta}$, $D_4 = g^{y} d_1^{\alpha} d_2^{\beta}$과 $\gamma = x + \alpha$ mod p, $\delta = x + \beta$ mod p을 계산한다. 또한, 임의의 난수들 $r_{\alpha}, r_{\beta}, r_{\gamma}, r_{\delta}, r_x, r_y, r_z, \overline{Z_p}$를 선택하고 $R_1 = u^{r_{\alpha}}$, $R_2 = v^{r_{\beta}}$, $R_3 = e(D_3, h_1)^{r_x} e(w_1, h_0)^{-r_{\alpha}} e(w_2, h_0)^{-r_{\beta}} e(w_2, h_1)^{-r_{\gamma}} e(w_2, h_1)^{-r_{\delta}} e(g_2, h_1)^{r_x} e(g_3, h_1)^{r_z}$, $R_4 = g^{r_y} d_1^{r_{\alpha}} d_2^{r_{\beta}}$, $R_5 = D_1^{r_x} u^{-r_{\gamma}}$, $R_6 = D_2^{r_x} u^{-r_{\delta}}$을 계산한다. 또한, 해쉬함수를 이용하여 $c = H(M, D_1, D_2, D_3, D_4, R_1, R_2, R_3, R_4, R_5, R_6)$을 계산하고 $s_{\alpha} = r_{\alpha} + c\alpha$, $s_{\beta} = r_{\beta} + c\beta$, $s_{\gamma} = r_{\gamma} + c\gamma$, $s_{\delta} = r_{\delta} + c\delta$, $s_x = r_x + cx$ $s_y = r_y + cy$, $s_z = r_z + cz$을 계산한다. 마지막으로, $\sigma = (D_1, D_2, D_3, D_4, c, s_{\alpha}, s_{\beta}, s_{\gamma}, s_{\delta}, s_x, s_y, s_z)$을 서명으로 출력한다(S404).

[0065] 위의 설명에서, $D_3 = \mathcal{A} w_1^{\alpha} w_2^{\beta}$ 또는 $D_4 = g^{y} d_1^{\alpha} d_2^{\beta}$는 선택적으로 선형결합암호 기법 대신 선형암호(Linear Encryption)기법을 이용할 수 있다. 예를 들어, $D_4 = g^{y} d_1^{\alpha} d_2^{\beta}$ 대신 $D_4 = g^{y} d^{\alpha+\beta}$을 계산한다. 이 경우, 관련된 그룹 공개키 생성과 서명자 확인을 위한 증명정보 생성, 서명자증명정보확인을 위한 알고리즘 및, 연결성 정보의 확인 방법은 일관성을 위해 필요 시 적절히 수정된다. 수정은 자명하게 이루어지므로 생략한다.

[0066] 도 5는 본 발명의 실시 예에 따라 정당한 사용자 기기(400)가 생성한 한 메시지에 대한 그룹서명에 대해 서명검증알고리즘(500)을 통해 검증하는 동작 제어 흐름을 도시한 것이다.

[0067] 이하, 도 5를 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0068] 메시지 M에 대한 서명 $\sigma = (D_1, D_2, D_3, D_4, c, s_{\alpha}, s_{\beta}, s_{\gamma}, s_{\delta}, s_x, s_y, s_z)$가 주어졌다고 가정하자(S500). 그러면, 서명검증부(500)부는 $R_1 = u^{s_{\alpha}} D_1^{-c}$, $R_2 = v^{s_{\beta}} D_2^{-c}$, $R_3 = e(D_3, h_1)^{s_x} e(w_1^{-s_{\alpha}} w_2^{-s_{\beta}}, h_0) e(w_1^{-s_{\gamma}} w_2^{-s_{\delta}}, h_1) e(g_2, h_1)^{s_y} e(g_3, h_1)^{s_z} (e(D_3, h_0)/e(g_1, h_1))^{c}$, $R_4 = g^{s_y} d_1^{s_{\alpha}} d_2^{s_{\beta}} D_4^{-c}$, $R_5 = D_1^{s_x} u^{-s_{\gamma}}$, 그리고 $R_6 = D_2^{s_x} v^{-s_{\delta}}$을 계산한다(S502). 그리고 해쉬함수 값 $c' = H(M, D_1, D_2, D_3, D_4, R_1, R_2, R_3, R_4, R_5, R_6)$을 계산한 후 c와 c'이 같은지 확인한다. 만일 같다면 주어진 서명에 유효함을 나타내는 1을 출력한다. 만일 아니라면 0을 출력한다(S504).

[0069] 도 6은 본 발명의 실시 예에 따라 정당한 사용자 기기(400)가 생성한 한 메시지에 대한 그룹서명에 대해 마스터 확인키를 이용하여 서명자가 누구인지 확인해 주는 증명정보를 생성하고 서명자증명정보 확인부(600)를 이용하여 증명정보의 유효성을 검증하는 동작 제어 흐름을 도시한 것이다.

[0070] 이하, 도 6을 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0071] 메시지 M에 대한 서명 $\sigma = (D_1, D_2, D_3, D_4, c, s_{\alpha}, s_{\beta}, s_{\gamma}, s_{\delta}, s_x, s_y, s_z)$가 주어졌다고 가정하자(S600). 확인자 서버(200)는 마스터 확인키(mok) $mok = (\eta_1, \eta_2, \xi_1, \xi_2)$를 이용하여 다음과 같이 증명정보 i를생성한다. $g^{y} D_4 (D_1^{\xi_1} D_2^{\xi_2})^{-1}$와 $\mathcal{A} D_3 (D_1^{\eta_1} D_2^{\eta_2})^{-1}$을 통해서 $H(g^{y})$와 $\mathcal{A}$를 계산한다(S602).

[0072] 이어서, $H(g^{y})$을 이용하여 사용자 등록 목록 REG에서 이진 검색을 통해 효율적으로 $H(g^{y}) = H(g^{y_i})$을 만족하는 사용자 인덱스 i와 이에 대응하는 정보 $upk[i] = Z_i = g_3^{z_i}$, $Y_{1,i} = g_2^{y_i}$, $Y_{2,i} = h_1^{y_i}$, $X_{1,i} = g^{x_i}$, $X_{2,i} = h_1^{x_i}$를 찾는다. 여기서,

$upk[i]=Z_i=g_3^{z_i}$은 서명자가 가입 시 자신의 공개키로 등록한 정보이다(S604). 임의의 난수 $r_1, r_2 \in \overrightarrow{Z_p}$를 선택하고 $K_{12}=D_1^{\eta_1}D_2^{\eta_2}$, $W_1=u^{r_1}$, $W_2=v^{r_2}$, $W_{12}=D_1^{r_1}D_2^{r_2}$, $c_{12}=H(\sigma, u, v, K_{12}, W_1, W_2, W_{12})$, 그리고 $s_1=r_1+c_{12}\eta_1$, $s_2=r_2+c_{12}\eta_2$ (mod p)을 계산한다. 그리고, 주어진 서명과 메시지에 대한 서명자 증명정보로 ($\tau=(K_{12}, c_{12}, s_1, s_2)$, $upk[i]=Z_i=g_3^{z_i}$, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$, $\sigma_{judge,i}$)을 출력한다(S606). 여기서, $\sigma_{judge,i}$는 서명자 i가 지수값들 x_i, y_i, z_i을 알고 있음을 증명하는 (서명자 i에 의해서 서명 기법 $\sum$를 이용하여 생성된) 일반적인 서명이다.

[0073] 주어진 메시지 M에 대한 서명 $\sigma=(D_1, D_2, D_3, D_4, c, s_\alpha, s_\beta, s_\gamma, s_\delta, s_{x_1}, s_{y_1}, s_{z_1})$, 그리고 서명자 증명정보 ($\tau=(K_{12}, c_{12}, s_1, s_2)$, $upk[i]=Z_i=g_3^{z_i}$, $Y_{1,i}=g_2^{y_i}$, $Y_{2,i}=h_1^{y_i}$, $X_{1,i}=g^{x_i}$, $X_{2,i}=h_1^{y_i}$, $\sigma_{judge,i}$)에 대해 서명자증명정보 확인부(600)는 다음이 성립하는지 확인한다. (1) 먼저 $c'_{12}=H(\sigma, u, v, K_{12}, u^{s_1}w_1^{-c_{12}}, v^{s_2}w_2^{-c_{12}}, D_1^{s_1}D_2^{s_2}K_{12}^{-c_{12}})$를 계산하고 $c'_{12}=c_{12}$이 성립하는지 확인한다. (2) $e(D_3K_{12}^{-1}, X_{2,i}h_0)=e(g_1Y_{1,i}^{-1}Z_{1,i}^{-1}, \tilde{h}_1)$이 성립하는지 확인한다. 여기서 $\tilde{h}_1$는 현재의 그룹 공개키에 포함된 값이고 g, g_2, h_1, h_0는 초기 그룹공개키 gpk_0에 포함된 값이다. 만일 위의 등식들이 모두 성립하면 유효함을 나타내는 1을 출력한다. 만일 아니라면 0을 출력한다(S608).

[0074] 도 7은 본 발명의 실시 예에 따라 정당한 사용자 기기(400)가 생성한 두 그룹서명들에 대해 마스터 연결키를 이용하여 서명값들이 서로 연결되었는지를 확인해 주는 동작 제어 흐름을 도시한 것이다.

[0075] 이하, 도 7을 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0076] 주어진 메시지와 서명들의 쌍, (σ, M)와 (σ', M')을 입력받는 경우(S700), 연결자 서버(300)는 마스터 연결키 $mlk=(U, V)$를 이용하여 $B_1=e(D_4, h_1)[e(D_1, U)e(D_2, V)]^{-1}$과 $B_2=e(D'_4, h_1)[e(D'_1, U)e(D'_2, V)]^{-1}$을 계산한 후 등식 $B_1=B_2$이 성립하는지 확인한다(S702). 만일 등식이 성립하면 "연결됨"을 나타내는 1을 출력한다. 만일 아니라면 0을 출력한다(S704).

[0077] 선택적으로, 계산의 효율성을 증진시키기 위해서 등식 $e(D_4/D'_4, h_1)=e(D_1/D'_1, U)e(D_2/D'_2, V)$이 성립함을 확인할 수 있다.

[0078] 도 8은 본 발명의 실시 예에 따라 키 폐기가 발생한 경우 키발급자 서버(100)가 관련 정보를 공개하고, 그룹 공개키를 갱신하고 정당한 사용자 기기(400)는 자신의 서명키를 갱신하는 동작 제어 흐름을 도시한 것이다.

[0079] 이하, 도 8을 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0080] 각 세션마다 키들의 집합이 폐기되며, 각 세션을 구분하기 위해 인덱스 변수 κ를 이용하여 세션을 나타내기로 하자. 세션이 변하면 κ는 1씩 증가한다고 가정한다. 초기 그룹 공개키 $gpk_0=(T, g_1, g_2, g_3)$와 현재의 그룹 공개키 $gpk_{\kappa-1}=(T, g'_1, g'_2, g'_3)$가 각각 주어졌다고 가정하자. (여기서 $T=(e, \overrightarrow{G_1}, \overrightarrow{G_2}, g, h_1, h_0, H, u, v, w_1, w_2, d_1, d_2)$이다.)

[0081] 주어진 키 값들을 폐기 시키고 키들을 갱신시키기 위해서, 키발급자 서버(100)는 먼저 폐기목록 $RL=\{(T_{1,i}=g_1^{1/(\theta+x_{\kappa,i})}, T_{2,i}=g_2^{1/(\theta+x_{\kappa,i})}, T_{3,i}=g_3^{1/(\theta+x_{\kappa,i})}, x_{\kappa,i}) | i=1, \ldots, r_\kappa\}$ 을 공개한다(S800).

[0082] (1)그룹 공개키를 $gpk_{\kappa-1}$에서 gpk_κ으로 갱신하기 위해서 $g''_1=g'_1\bigcirc_{i=1}^{r_\kappa}T_{1,i}$, $g''_2=g'_2\bigcirc_{i=1}^{r_\kappa}T_{2,i}$, $g''_3=g'_3\bigcirc_{i=1}^{r_\kappa}T_{3,i}$,을 계산한다. 갱신된 그룹 공개키는 $gpk_\kappa=(T, g''_1, g''_2, g''_3)$ 이다(S802).

[0083] (2)사용자 기기(400)는 자신의 서명키를 $usk_{\kappa-1}[i]=(\mathcal{A}', x, y, z, A)$에서 $usk_\kappa[i]$으로 갱신하기 위해서 $\mathcal{A}''=\mathcal{A}'\bigcirc_{i=1}^{r_\kappa}[(T_{1,i}T_{2,i}^{-y}T_{3,i}^{-z}A^{-1})]^{1/(x-x_{\kappa,i})}=(g''_1g''^{-y}_2g''^{-z}_3)^{1/(\theta+x)}$ 을 계산한다. 현재의 그룹공개키 $gpk_\kappa=(T, g''_1, g''_2, g''_3)$에 대응하는 갱신된 서명키는 $usk_\kappa[i]=(\mathcal{A}'', x, y, z, A)$로 설정된다(S804).

[0084] 도 9는 본 발명의 실시 예에 따른 선형결합암호기법(Linear Combination Encryption(LCE) Scheme) 을 도시한

것이다.

[0085] 이하, 도 9를 참조하여 본 발명의 실시 예를 상세히 설명하기로 한다.

[0086] 다음과 같이 키 생성, 암호화, 복호화 알고리즘으로 정의한다. (1) 키 생성 알고리즘: 최초 사용자는 소수 위수 p인 대수적 군 $\overrightarrow{G_1}$와 $\overrightarrow{G_1}$의 임의의 생성원 u,v를 생성한다. 그리고 집합 $\overrightarrow{Z_p^*}$에서 임의의 x, y를 선택한 후 $w_1=u^x$, $w_2=v^y$을 계산한다. 여기서는 $\overrightarrow{G_1}$을 곱셈군(Multiplicative Group)으로 표현한다. 공개키를 $(\overrightarrow{G_1},u,v,w_1,w_2)$로 정의하고 공개하고 이에 대응하는 비밀키 (x,y)를 안전하게 저장한다(S900). (2) 암호화 알고리즘: 메시지 $M\in\overrightarrow{G_1}$을 암호화 하기 위해서, 암호화 알고리즘은 집합 $\overrightarrow{Z_p^*}$에서 임의의 a, b를 선택하고, $C_1=u^a$, $C_2=v^b$, $D_1=Mw_1^aw_2^b$을 계산한 후 암호문 (C_1, C_2, D_1)를 출력한다(S902). (3) 복호화 알고리즘: 주어진 암호문 (C_1, C_2, D_1)을 복호화하기 위해서, 비밀키 (x,y)를 이용하여 $M=D_1C_1^{-x}C_2^{-y}$을 계산하여 메시지를 복구한다(S904).

[0087] 도 10은 본 발명의 실시 예에 따라 n개의 메시지 쌍 $(M_1,\ldots,M_n)\in\overrightarrow{G_1}\times\cdots\times\overrightarrow{G_1}$을 암호화 하기 위해서 선형결합암호기법(Linear Combination Encryption(LCE) Scheme)을 효율적으로 확장시키는 하이브리드선형결합암호기법(Hybrid Linear Combination Encryption(HLCE) Scheme)을 도시한 것이다.

[0088] 이하, 다음은 이에 대한 자세한 설명이다. 다음과 같이 키 생성, 암호화, 복호화 알고리즘으로 정의한다. (1) 키 생성 알고리즘: 최초 사용자는 소수 위수 p인 대수적 군 $\overrightarrow{G_1}$와 $\overrightarrow{G_1}$의 임의의 생성원 u,v를 생성한다. 그리고 각 $i=1,\ldots,n$에 대하여 집합 $\overrightarrow{Z_p^*}$에서 임의의 x_i, y_i를 선택한 후 $w_{2i-1}=u^{x_i}$, $w_{2i}=v^{y_i}$을 계산한다. 여기서는 곱셈군(Multiplicative Group)으로 표현한다. 공개키를 $(\overrightarrow{G_1},u,v,w_1,w_2,\ldots,w_{2n-1},w_{2n})$으로 정의하고 공개하고 이에 대응하는 비밀키 $(x_1,y_1,\ldots,x_n,y_n)$를 안전하게 저장한다(S950). (2) 암호화 알고리즘: 메시지 $(M_1,\ldots,M_n)\in\overrightarrow{G_1}\times\cdots\times\overrightarrow{G_1}$를 암호화 하기 위해서, 암호화 알고리즘은 집합 $\overrightarrow{Z_p^*}$에서 임의의 a, b를 선택하고, $C_1=u^a$, $C_2=v^b$, $D_1=M_1w_1^aw_2^b$, $\ldots$, $D_j=M_jw_{2j-1}^aw_{2j}^b$, $\ldots$, $D_n=M_nw_{2n-1}^aw_{2n}^b$ 을 계산한 후 암호문 (C_1, C_2, D_1, $\ldots$, D_n)를 출력한다(S952). (3) 복호화 알고리즘: 주어진 암호문 (C_1, C_2, D_1, $\ldots$, D_n)을 복호화하기 위해서, 비밀키 $(x_1,y_1,\ldots,x_n,y_n)$를 이용하여 $M_j=D_jC_1^{-x_j}C_2^{-y_j}$을 계산하여 메시지를 복구한다(S954).

[0089] 한편 상술한 본 발명의 설명에서는 구체적인 실시 예에 관해 설명하였으나, 여러 가지 변형이 본 발명의 범위에서 벗어나지 않고 실시될 수 있다. 즉 본 발명에서는 제어가능 연결성을 제공하는 그룹서명 기법과 선형결합암호기법, 그리고 하이브리드선형결합암호기법에 대해 예시된 도면을 참조로 설명하였으나, 본 명세서에 개시된 실시예와 도면에 의해 본 발명은 한정되지 않고, 기술사상이 보호되는 범위 이내에서 응용될 수 있다. 따라서 발명의 범위는 설명된 실시 예에 의하여 정할 것이 아니고 특허청구범위에 의해 정하여져야 한다.

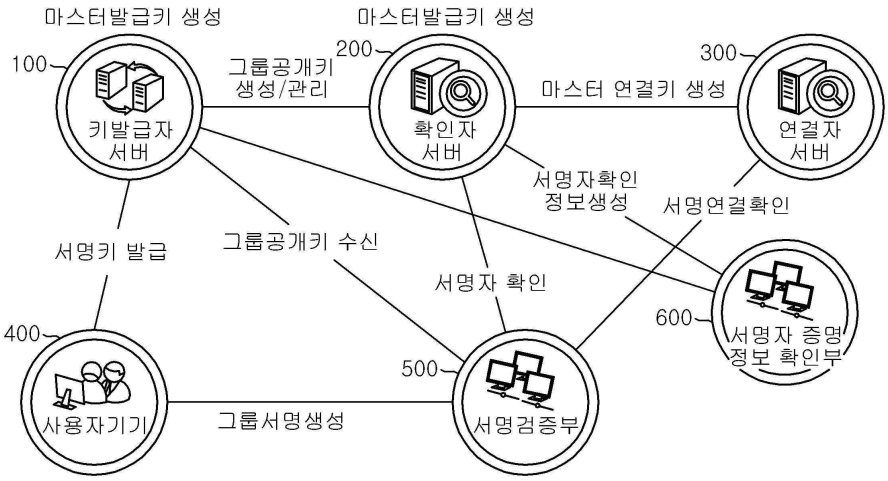
부호의 설명

[0090]

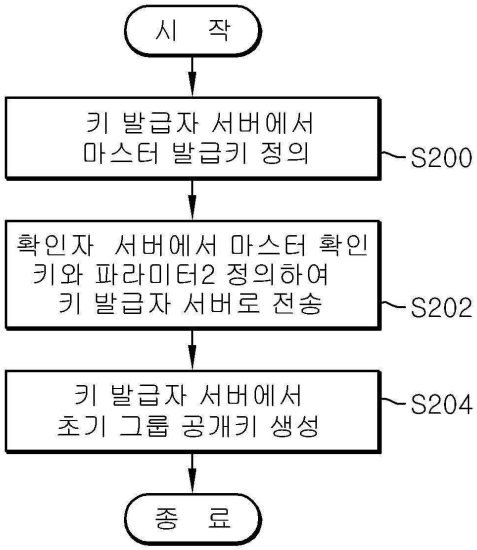
100 : 키발급자 서버	200 : 확인자 서버
300 : 연결자 서버	400 : 사용자 기기
500 : 서명검증부	600 : 서명자증명정보 확인부

도면

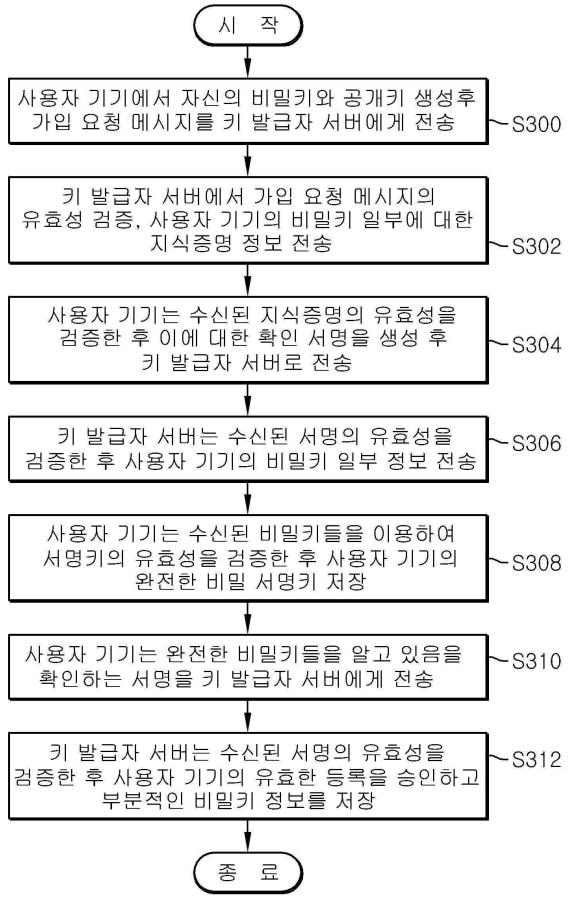
도면1


마스터발급키 생성
마스터발급키 생성
100
200
300
그룹공개키
생성/관리
마스터 연결키 생성
키발급자
서버
확인자
서버
연결자
서버
서명자확인
정보생성
서명연결확인
서명키 발급
그룹공개키 수신
서명자 확인
600
서명자 증명
정보 확인부
400
500
사용자기기
그룹서명생성
서명검증부

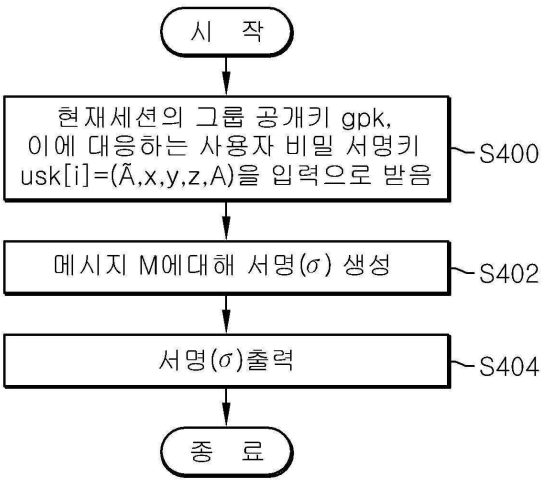
도면2


시 작
키 발급자 서버에서
마스터 발급키 정의
S200
확인자 서버에서 마스터 확인
키와 파라미터2 정의하여
키 발급자 서버로 전송
S202
키 발급자 서버에서
초기 그룹 공개키 생성
S204
종 료

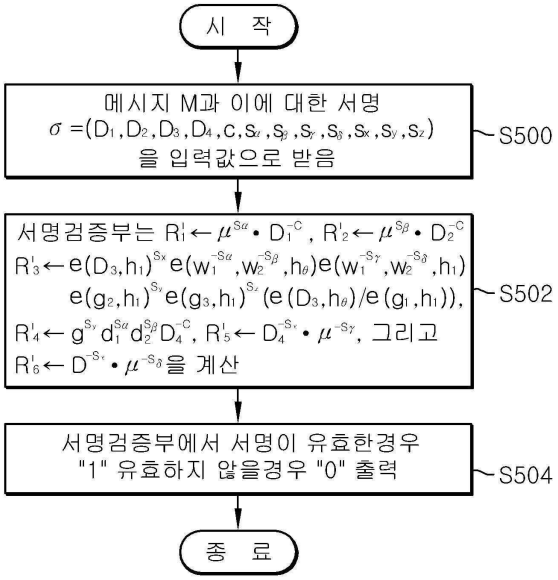
도면3


시 작
사용자 기기에서 자신의 비밀키와 공개키 생성후 가입 요청 메시지를 키 발급자 서버에게 전송
S300
키 발급자 서버에서 가입 요청 메시지의 유효성 검증, 사용자 기기의 비밀키 일부에 대한 지식증명 정보 전송
S302
사용자 기기는 수신된 지식증명의 유효성을 검증한 후 이에 대한 확인 서명을 생성 후 키 발급자 서버로 전송
S304
키 발급자 서버는 수신된 서명의 유효성을 검증한 후 사용자 기기의 비밀키 일부 정보 전송
S306
사용자 기기는 수신된 비밀키들을 이용하여 서명키의 유효성을 검증한 후 사용자 기기의 완전한 비밀 서명키 저장
S308
사용자 기기는 완전한 비밀키들을 알고 있음을 확인하는 서명을 키 발급자 서버에게 전송
S310
키 발급자 서버는 수신된 서명의 유효성을 검증한 후 사용자 기기의 유효한 등록을 승인하고 부분적인 비밀키 정보를 저장
S312
종 료

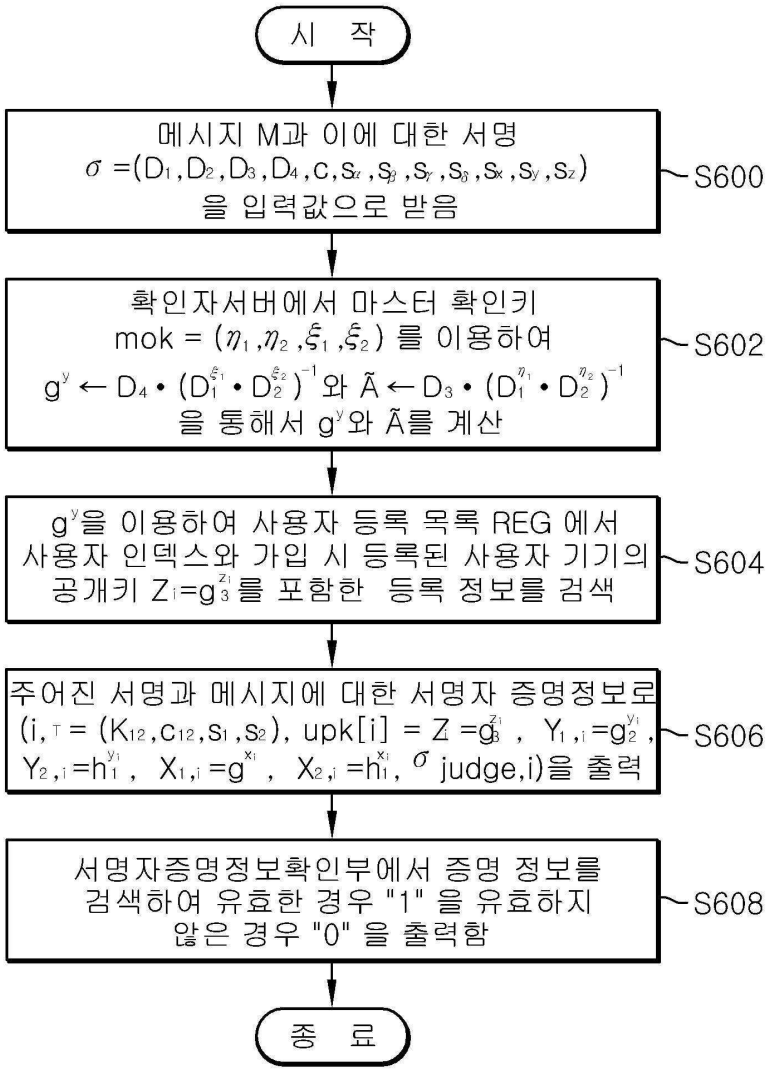
도면4


시 작
현재세션의 그룹 공개키 gpk, 이에 대응하는 사용자 비밀 서명키 usk[i]=(Ã,x,y,z,A)을 입력으로 받음
S400
메시지 M에대해 서명(σ) 생성
S402
서명(σ)출력
S404
종 료

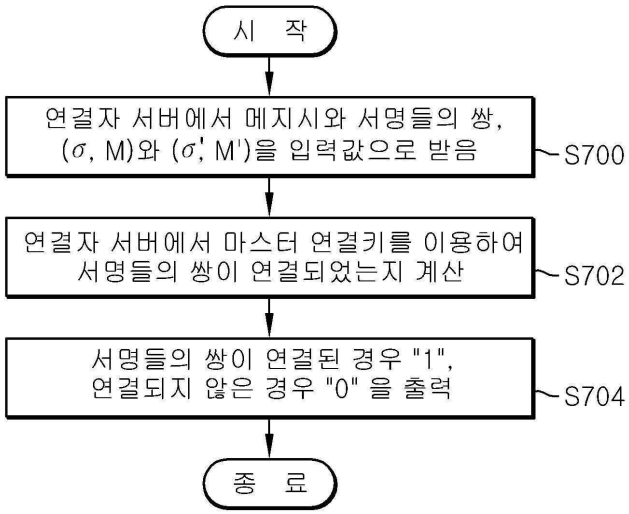
도면5


시 작
메시지 M과 이에 대한 서명
$\sigma=(D_1,D_2,D_3,D_4,c,s_\alpha,s_\beta,s_\gamma,s_\delta,s_x,s_y,s_z)$
을 입력값으로 받음
S500
서명검증부는 $R_1' \leftarrow \mu^{s_\alpha} \cdot D_1^{-c}$, $R_2' \leftarrow \mu^{s_\beta} \cdot D_2^{-c}$
$R_3' \leftarrow e(D_3,h_1)^{s_x} e(w_1^{-s_\alpha},w_2^{-s_\beta},h_\theta) e(w_1^{-s_\gamma},w_2^{-s_\delta},h_1)$
$e(g_2,h_1)^{s_y} e(g_3,h_1)^{s_z} (e(D_3,h_\theta)/e(g_1,h_1))$,
$R_4' \leftarrow g^{s_y} d_1^{s_\alpha} d_2^{s_\beta} D_4^{-c}$, $R_5' \leftarrow D_4^{-s_x} \cdot \mu^{-s_\gamma}$, 그리고
$R_6' \leftarrow D^{-s_\gamma} \cdot \mu^{-s_\delta}$을 계산
S502
서명검증부에서 서명이 유효한경우
"1" 유효하지 않을경우 "0" 출력
S504
종 료

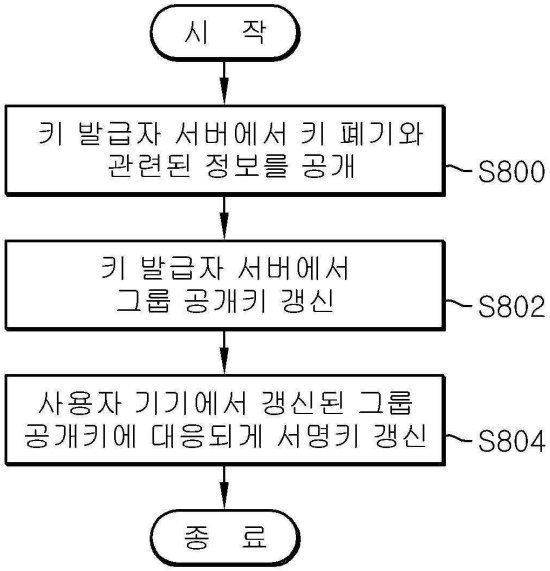
도면6


시 작
메시지 M과 이에 대한 서명
σ =(D1,D2,D3,D4,c,sα,sβ,sγ,sδ,sx,sy,sz)
을 입력값으로 받음
S600
확인자서버에서 마스터 확인키
mok = (η1,η2,ξ1,ξ2) 를 이용하여
g^y ← D4 • (D1^ξ1 • D2^ξ2)^-1 와 Ã ← D3 • (D1^η1 • D2^η2)^-1
을 통해서 g^y와 Ã를 계산
S602
g^y을 이용하여 사용자 등록 목록 REG 에서
사용자 인덱스와 가입 시 등록된 사용자 기기의
공개키 Zi=g3^zi를 포함한 등록 정보를 검색
S604
주어진 서명과 메시지에 대한 서명자 증명정보로
(i,T = (K12,c12,s1,s2), upk[i] = Zi =g3^zi , Y1,i =g2^yi ,
Y2,i =h1^yi , X1,i =g^xi , X2,i =h1^xi , σ judge,i)을 출력
S606
서명자증명정보확인부에서 증명 정보를
검색하여 유효한 경우 "1" 을 유효하지
않은 경우 "0" 을 출력함
S608
종 료

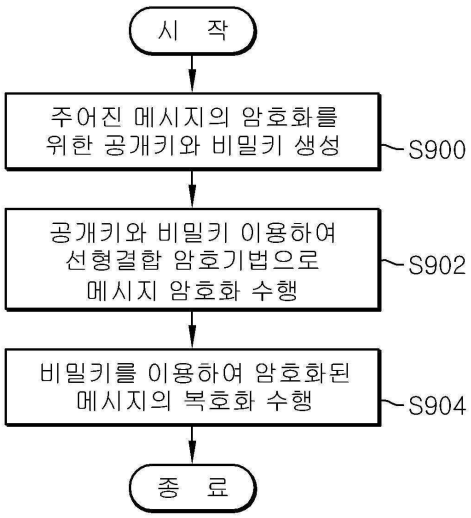
도면7


시 작
연결자 서버에서 메지시와 서명들의 쌍,
(σ, M)와 (σ', M')을 입력값으로 받음
S700
연결자 서버에서 마스터 연결키를 이용하여
서명들의 쌍이 연결되었는지 계산
S702
서명들의 쌍이 연결된 경우 "1",
연결되지 않은 경우 "0" 을 출력
S704
종 료

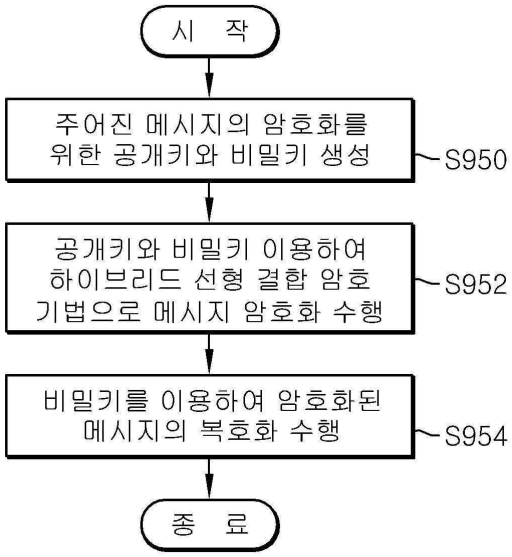
도면8


시 작
키 발급자 서버에서 키 폐기와 관련된 정보를 공개
S800
키 발급자 서버에서 그룹 공개키 갱신
S802
사용자 기기에서 갱신된 그룹 공개키에 대응되게 서명키 갱신
S804
종 료

도면9


시 작
주어진 메시지의 암호화를 위한 공개키와 비밀키 생성
S900
공개키와 비밀키 이용하여 선형결합 암호기법으로 메시지 암호화 수행
S902
비밀키를 이용하여 암호화된 메시지의 복호화 수행
S904
종 료

도면10


시 작
주어진 메시지의 암호화를 위한 공개키와 비밀키 생성
S950
공개키와 비밀키 이용하여 하이브리드 선형 결합 암호 기법으로 메시지 암호화 수행
S952
비밀키를 이용하여 암호화된 메시지의 복호화 수행
S954
종 료