(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0103269
(43) 공개일자 2014년08월26일

(51) 국제특허분류(Int. Cl.)
H04L 9/08 (2006.01) *H04L 9/30* (2006.01)
H04L 9/32 (2006.01)
(21) 출원번호 10-2014-7015912
(22) 출원일자(국제) 2012년12월11일
심사청구일자 없음
(85) 번역문제출일자 2014년06월11일
(86) 국제출원번호 PCT/EP2012/075091
(87) 국제공개번호 WO 2013/087629
국제공개일자 2013년06월20일
(30) 우선권주장
11306672.4 2011년12월15일
유럽특허청(EPO)(EP)

(71) 출원인
톰슨 라이센싱
프랑스, 이씨레뮬리노 92130 잔다르크 뤼 1-5
(72) 발명자
엘 에마니, 라일라
프랑스 35000 렌 뤼 데 샤티옹 23
조이, 마르크
프랑스 쎄에스 176 16 35 576 쎄쏭 쎄비네 자크 데 샹 블랑 975 아브뉘 데 샹 블랑 떼끄니꼴로르 에르 에 데 프랑스
(74) 대리인
백만기, 양영준, 전경석

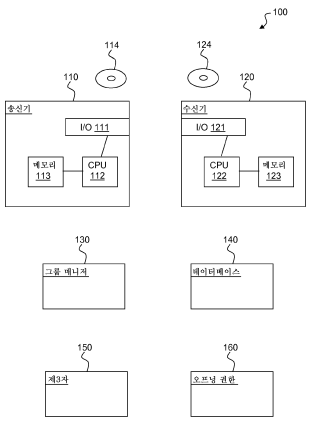
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **그룹 암호화 방법 및 디바이스**

(57) 요 약

본 발명은 수신자의 공개 키(public key) 자체 대신, 공개 키의 별명(alias)을 암호화함으로써 종래 기술에서의 그룹 암호화 스킴들을 개선한다. 그룹 매니저는 별명의 암호화, 상응하는 공개 키와 상응하는 인증서를 공용 데이터베이스(DB)에 게시한다. 별명은 공개 키에 있는, 적절하게 선택된 함수 f의 결과 값이고, 공개 키의 해시(hash)로 보일 수 있다. 별명이 공개 키보다 작게 만들어질 수 있기 때문에, 그 결과로 발생하는 구성의 사이즈와 비용의 상당한 감소를 허용한다. 특히, 수신자의 공개 키에 있는 그룹 요소의 개수만큼 두 번째 암호화를 적용할 필요가 없다.

대 표 도 - 도1



특허청구의 범위

청구항 1

공개 키 pk를 가진 수신자가 암호문 c를 획득하도록 디바이스(110)에서 태그 t에 관하여 평문 m을 그룹 암호화하는 방법으로서,

서명 키 OTS.sk 및 검증 키 OTS.vk를 획득하는 단계;

$c_1 = E_1.Encrypt_{\{pk\}}(m,OTS.vk)$ 및 $c_2 = E_2.Encrypt_{\{pkOA\}}(f(pk),OTS.vk)$를 계산함으로써 제1 암호화된 값 c_1과 제2 암호화된 값 c_2를 생성하는 단계 - E_1은 제1 암호화 알고리즘이고, E_2는 제2 암호화 알고리즘이고, f는 맵핑 함수임 -;

$s = OTS.Sign_{\{OTS.sk\}}(c_1,c_2,t)$를 계산함으로써, 상기 서명 키 OTS.sk를 사용하여 상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2, 및 상기 태그 t에 대해 서명 s를 생성하는 단계 - OTS.Sign은 서명 알고리즘임 -; 및

상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2, 상기 검증 키 OTS.vk 및 상기 서명 s를 포함하는 상기 암호문 c를 출력하는 단계

를 포함하는, 방법.

청구항 2

제1항에 있어서,

메시지 m은, 공적으로 검증가능한 관계(publicly verifiable relation) R을 만족시키는, 방법.

청구항 3

디바이스(120)에서 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk 및 서명 s를 포함하는 그룹 암호화 c를 복호화하는 방법으로서,

상기 서명 s는 상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2, 및 태그 t에 대한 것이고,

상기 그룹 암호화 c를 수신하는 단계;

검증 키 OTS.vk에 관하여 상기 서명 s를 검증하는 단계; 및

상기 서명 s가 성공적으로 검증된 경우, 복호화 알고리즘 E_1과 상기 검증 키 OTS.vk를 사용하여 상기 제1 암호화된 값 c_1을 복호화하는 단계

를 포함하는, 방법.

청구항 4

제3항에 있어서,

상기 서명을 검증하는 단계는 상기 제1 암호화된 값 c_1의 복호화가 공적 관계 R을 만족시키는 것을 검증하는 단계

를 더 포함하는, 방법.

청구항 5

공개 키 pk를 가진 수신자가 암호문 c를 획득하도록, 태그 t에 관하여 평문 m을 그룹 암호화하는 디바이스(110)로서,

서명 키 OTS.sk 및 검증 키 OTS.vk를 획득하고;

$c_1 = E_1.Encrypt_{\{pk\}}(m,OTS.vk)$ 및 $c_2 = E_2.Encrypt_{\{pkOA\}}(f(pk),OTS.vk)$를 계산함으로써 제1 암호화된 값 c_1과 제2 암호화된 값 c_2를 생성하고 - E_1은 제1 암호화 알고리즘이고, E_2는 제2 암호화 알고리즘이고 f는 맵핑 함수임 -;

$s = OTS.Sign_{\{OTS.sk\}}(c_1,c_2,t)$ - OTS.Sign은 서명 알고리즘임 -를 계산함으로써, 상기 서명 키 OTS.sk를 사용하여 상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2,및 상기 태그 t에 대한 서명 s를 생성하고;

상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2, 상기 검증 키 OTS.vk, 및 상기 서명 s를 포함하는 상기 암호문 c를 출력하도록

구성된 프로세서(112)를 포함하는, 디바이스.

청구항 6

제5항에 있어서,

메시지 m은 공적으로 검증가능한 관계 R을 만족시키는, 디바이스.

청구항 7

제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk 및 서명 s를 포함하는 그룹 암호화 c를 복호화하는 디바이스(120)로서,

상기 서명 s는 상기 제1 암호화된 값 c_1, 상기 제2 암호화된 값 c_2,및 태그 t에 대한 것이고,

상기 그룹 암호화 c를 수신하고;

검증 키 OTS.vk에 관하여 상기 서명 s를 검증하고;

상기 서명 s가 성공적으로 검증된 경우, 복호화 알고리즘 E_1과 상기 검증 키 OTS.vk를 사용하여 상기 제1 암호화된 값 c_1을 복호화하도록

구성된 프로세서(122)를 포함하는, 디바이스.

청구항 8

제7항에 있어서,

상기 프로세서는 상기 제1 암호화된 값 c_1의 복호화가 공적 관계 R을 만족시키는 것을 검증하도록 더 구성된 것인, 디바이스.

청구항 9

프로세서에 의해 실행될 때, 제1항 또는 제2항에 따른 방법을 수행하는 명령어들이 저장된, 컴퓨터 프로그램 제품(114).

청구항 10

프로세서에 의해 실행될 때, 제3항 또는 제4항에 따른 방법을 수행하는 명령어들이 저장된, 컴퓨터 프로그램 제품(124).

명 세 서

기 술 분 야

[0001] 본 발명은 일반적으로 암호 기법(cryptography)에 관한 것이며, 특히 그룹 암호화에 관한 것이다.

배 경 기 술

[0002] 본 배경기술 부분은, 이하 기술된 그리고/또는 청구된 본 발명의 다양한 측면들과 관련이 있을 수 있는 기술의 다양한 측면들을 독자에게 소개하고자 한다. 이러한 논의는 본 발명의 다양한 측면들에 대한 한층 더 깊은 이해를 용이하게 하기 위해서, 독자에게 배경 정보를 제공하는 데에 도움이 된다고 생각된다. 따라서, 본 서술들은 종래 기술의 인정이 아니라, 이러한 관점에서 해석되어야 한다는 점이 이해되어야 한다.

[0003] 본 부분에서, 그룹 암호화 프리미티브가 정의되고, 필요한 빌딩 블록들 - 공개 키 암호화, 태그-기반 암호화, 및 일회용 서명(one-time signature) - 이 제시되며, 그룹 암호화에서의 최신 기술이 기술된다.

[0004] 그룹 암호화는 Kiayias-Tsiounis-Yung에 의해 그룹 서명의 암호화 아날로그로 소개되었다; Aggelos Kiayias, Yiannis Tsiounis 및 Moti Yung에 의한, ASIACRYPT 2007의 181-199면의 "Group Encryption"을 참조한다. 그룹 암호화는 정당한 사용자의 그룹 내의 수신자(디크립터)를 숨기길 원하는 상황에서 유용하다.

[0005] 실례로서, 발송할 광고들에 맞는 프로필을 가진 가입 고객에게 특정한 광고를 발송하길 원하는 네트워크 서비스 제공자(NSP)가 있다. 동시에, NSP는 자신의 고객, 즉 광고들을 전송하는 대가를 NSP에게 지급하는 회사에, 수신자의 정확한 신원을 비밀로 유지하면서, 당해 광고들을 구독자들의 그룹 내로 전송했다는 것을 증명하길 원한다. 광고들의 수신자의 프라이버시는 또한 NSP의 구독자의 그룹 내에서도 보존되어야 한다.

[0006] 그룹 암호화는 전송자(본 예에서 NSP)가 메시지(광고)를 타겟 사용자에 대해 암호화하는 것을 허용하고, 부가적으로, 형성된 암호문이 유효한지(예를 들어 상응하는 평문이 소정의 관계를 만족하는지) 및 구독자들의 그룹 중의 소정의 익명의 구성원이 그것을 복호할 수 있는지 검증자(verifier)가 체크하는 것을 가능하게끔 하므로, 이러한 문제에 대한 타당한 해결책이 된다. 그룹 암호화는 또한 지정된 권한에 의한 암호문의 오픈과, 분쟁의 경우에 수신자의 신원을 되찾는 기능을 지원한다.

[0007] 더욱 형식적으로, 그룹 암호화(GE; group encryption) 스킴은, 그룹 구성원들을 등록하는 그룹 매니저(GM; group manager) 및 상응하는 암호문으로부터 수신자의 신원을 복원할 수 있는 오프닝 권한(OA; opening authority)을 수반한다.

[0008] GE 스킴의 기저를 이루는 주요한 절차는 다음과 같다.

[0009] - **Join**. GM과 잠재적 그룹 구성원 간의 상호적인 프로토콜. GM이 그룹 구성원의 공개 키 pk_i에 대한 인증서 $cert_i$를 발급한다. GM은 더 나아가 (pk_i, $cert_i$) 쌍을 공용 데이터베이스 DB에 저장한다.

[0010] - **Encrypt**. 암호화의 컨텍스트를 명시하는 이진 스트링(binary string)인, 입력 태그 t에 관하여, 타겟이 된 그룹 구성원의 공개 키 pk 하에서, 입력 메시지 m에 대한 암호문 c를 생성한다. 위조 메시지를 전송하는 것을 방지하기 위해서, 암호화되는 메시지 m이 소정의 연역적인 관계(a priori relation)를 만족할 것이 요구된다: m은 관계 R에 관하여 "인스턴스" x(공개 값)의 "위트니스(witness)"이다; 즉, $(m,x)\in R$. 이런 의미에서, 암호화는 또한 암호화된 위트니스에 상응하는 인스턴스 x를 출력한다.

[0011] - **Decrypt**. 입력 암호문 c에 있는 암호화된 메시지 m을 입력 태그 t에 관하여, 공개 키 pk - 그 공개 키 하에서 암호문이 만들어졌음 - 에 상응하는 개인 키(private key) sk를 사용하여 복원한다. 이 절차는 더 나아가 복원된 메시지가 입력 인스턴스 x의 위트니스인지 여부, 즉 $(m,x)\in R$ 이 성립하는지 체크한다. 만약 그렇다면, 알고리즘은 m을 출력하고, 그렇지 않다면, 실패(Fail)를 출력한다.

[0012] - **Open**. 암호문 c, 태그 t, 및 OA의 개인 키를 입력하고, 입력 태그 t에 관하여 공개 키 pk - 그 공개 키 하에서 암호문이 만들어짐 - 를 복원한다.

[0013] - **Prove**. 암호문을 만드는 엔티티로부터, 상호적이거나 상호적이지 않은 증명을 임의의 검증자에게 제공한다. 검증자는 당해 암호문이 유효하다는 것(예컨대, 그 안의 메시지가 관계 R을 만족시킨다는 것)과 그것이 소정의 익명의 등록된 그룹 구성원에 의해서 복호될 수 있다는 것을 확신해야한다.

[0014] PKE(public key encryption) 스킴은 (공개 키, 개인 키) 형식의 쌍들을 생성하는 키 생성 알고리즘, 수신자의 공개 키를 사용하여 입력 메시지의 암호화를 생성하는 암호화 알고리즘, 및 적절한 개인 키를 사용하여 입력 암호문으로 암호화된 메시지를 복원하는 복호화 알고리즘을 포함한다.

[0015] TBE(tag-based encryption scheme)는 암호화 및 복호화 둘 다에 대한 부가적인 인수(argument)인, 태그를 필요로 한다. 약식으로, 태그는 암호화에 관한 정보(날짜, 컨택스트, 등...)를 명시하는 적절한 길이의 이진 스트링이다.

[0016] 일회용 디지털 서명(one-time digital signature) 스킴은 최대 하나의 메시지를 서명하는 데 사용될 수 있다; 그렇지 않다면, 서명들이 위조될 수 있다. 새로운 공개 키가 서명된 각각의 메시지에 대해 필요하다. 그것들은 '보통의' 디지털 서명들처럼, 키 생성 알고리즘, 서명 알고리즘 및 검증 알고리즘에 의해 정의된다. 일회용 서명 스킴들의 보안은, 주어진 공개 키에 대하여 메시지의 새로운 유효 쌍과 상응하는 서명을 찾는 것의 어려운 정도에 의존한다.

[0017] 본 명세서에서 앞서 언급된 Kiayias-Tsiounis-Yung에 의한 논문은, 사용자 공개 키들의 인증을 위한 디지털 서명 스킴 S, 메시지를 암호화하기 위한 태그-기반의 암호화 스킴 E_1, 수신자 공개 키를 암호화하기 위한 또 다른 태그-기반의 암호화 스킴 E_2, 및 사용된 키와 그것의 인증서에 커미팅(commiting)하기 위한 커미트먼트(commitment) 스킴을 사용하는 안전한 그룹 암호화 스킴을 위한 포괄적인 구성을 제공한다. 좀 더 자세하게는, 해당 스킴은 다음과 같이 작동한다:

[0018] - **Join**. GM이 그것의 개인 서명 키 S.sk를 사용하여, 사용자의 공개 키 pk에 대한 서명 s(다시 말해서, 인증서)를 생성한다. GM은 더 나아가 (pk,s)를 공용 데이터베이스 DB에 저장한다.

[0019] - **Encrypt**$_{\{pk\}}$**(m,t)**. 태그 t에 관하여, 공개 키 pk를 가진 수신자를 위해, 메시지 m(x가 공개 값일 때, $(m,x)\in R$ 을 만족하는 m)을 암호화하기 위해서 앨리스는:

[0020] - pk에 대한 커미트먼트 c_3, 및 인증서에 대한 c_4를 생성한다.

[0021] - 태그(t,c_3,c_4)에 관하여, E_2.Encrypt를 사용하여 오프닝 권한의 공개 키 pk_{OA} 하에서 pk를 암호화하고, 그 결과 c_2가 주어진다.

[0022] - 태그(t,c_2,c_3,c_4)에 관하여, E_1.Encrypt를 사용하여 공개 키 pk 하에서 m을 암호화하고, 그 결과 c_1이 주어진다.

[0023] - 튜플(tuple) (c_1,c_2,c_3,c_4)를 t에 관하여 pk 하에서 m의 그룹 암호화로서 리턴한다.

[0024] - **Decrypt**$_{\{sk\}}$**(c,t,x)**. 처음에 c를 (c_1,c_2,c_3,c_4)로 분해(parse)하고, 그 다음 개인 키 sk를 사용하여 c_1과 (t,c_2,c_3,c_4)에 대하여 E_1.Decrypt를 호출하며, $(m,x)\in R$ 인 경우 결과, 즉 m을 리턴하고, 그렇지 않으면 "실패"를 리턴한다.

[0025] - **Open**$_{\{skOA\}}$**(c,t)**. 처음에 c를 (c_1,c_2,c_3,c_4)로 분해하고, 그 다음 OA의 개인 키 sk_{OA}를 사용하여 c_2와 (t,c_3,c_4)에 대하여 E_2.Decrypt를 호출하며, 결과를 리턴한다.

[0026] - **Prove(c,t,x)**. 태그 t에 관하여 암호문 $c = (c_1,c_2,c_3,c_4)$를 생성한 엔티티인 앨리스는 암호문이 유효하며, 공개 키에 상응하는, 개인 키에 의해 복호화되며, c_2에 암호화되고 c_3 - 이것의 인증서는 c_4로 커미트됨 - 로 커미트될 수 있다는 증명을 제공한다. 앨리스는 더 나아가 암호문에 내재된 메시지가 공적 관계 R에 관하여 x에 대한 위트니스라는 것을 증명한다. 앨리스는 위의 증명을 제공하기 위하여, c를 생성하는 데 사용된 개인 코인들(private coins)(즉, 커미트먼트 c_3과 c_4 및 암호화 c_1, c_2를 생성하기 위해 사용되는 랜덤한 값들)을 사용한다.

[0027] Cathalo-Libert-Yung이 그룹 암호화 스킴의 구체적인 실현을 제공하였고, Julien Cathalo, Benoit Libert, Moti Yung에 의한, ASIACRYPT 2009의 179-196면의 "Group Encryption: Non-interactive Realization in the Standard Model"을 참조하기 바란다. 그 스킴은 메시지의 암호화를 위해 Shacham의 암호화 스킴을 사용하고 [Cryptology ePrint Archive, Report 2007/074의 Hovav Shacham의 "A Cramer-Shoup Encryption Scheme from the Linear Assumption and from Progressively Weaker Linear Variants"를 참조한다], 수신자의 공개 키의 암호화를 위해 Kiltz' 암호화를 사용한다[TCC 2006의 581-600면에 있는 Eike Kiltz의 "Chosen-Ciphertext Security from Tag-Based Encryption"을 참조]. 해당 솔루션은, Prove의 기저에 있는 증명에 대한 커미트먼트 c_3과 c_4를 포기함으로써, Kiayias-Tsiounis-Yung에 의해 제공되는 구성에서 벗어난다.

[0028] 더 정확하게는, 만약 S가 해당 논문에서 주어진 디지털 서명 스킴을 지칭하면, OTS는 임의의 안전한 일회성-서명 스킴을, [Kiltz]는 Kiltz의 암호화 스킴을, 그리고 [Shacham]은 Shacham의 암호화 스킴을 지칭하고, 스킴은

다음과 같이 정의된다.

[0029] - **Join** 입력 공개 키 pk에 대하여, GM이 그것의 개인 서명 키 S.sk를 사용하여 서명(또는 인증서)을 생성하고, (pk, cert)를 공용 데이터베이스 DB에 저장한다.

[0030] - **Encrypt**$_{\{pk\}}$**(m,t)** 태그 t에 관하여 공개 키 pk를 갖는 수신자를 위해 메시지 m(e가 메시지 공간을 이루는 페어링(pairing)이고, 가령 G, 즉 g는 해당 그룹의 생성자일 때, m은 어떠한 (x,y)의 Diffie-Hellman 솔루션이다: e(m,g)=e(x,y))을 암호화하기 위해서, 앨리스는:

[0031] - 서명 키와 검증 키의 쌍 (OTS.sk,OTS.vk)를 생성하기 위해 OTS.keygen을 호출한다.

[0032] - c_1 = [Schacham].Encrypt$_{\{pk\}}$(m,(OTS.vk,t)) 및 c_2 = [Kiltz].Encrypt$_{\{pkOA\}}$(pk,OTS.vk)를 생성한다. c_1을 위해 사용된 태그는 (OTS.vk,t)이고 c_2를 위해 사용된 태그는 검증 키 OTS.sk이다.

[0033] - OTS.sk를 사용하여 (c_1,c_2,t)에 대한 일회용 서명 s를 생성한다; s = OTS.Sign$_{(OTS.sk)}$(c_1,c_2,t)

[0034] - t에 관하여 pk 하에서 m의 그룹 암호화로서 c = (c_1,c_2,OTS.vk,s)를 리턴한다.

[0035] - **Decrypt**$_{\{sk\}}$**(c,t,x,y)**

[0036] - c를 (c_1,c_2,OTS.vk,s)로 분해한다.

[0037] - (c_1,c_2,t)에 대한 서명 s를 OTS.vk에 관하여 체크한다; 만약 OTS.Verify$_{\{OTS.vk\}}$(s,(c_1,c_2,t)) = 0 이면 Fail을 리턴하고, 그렇지 않으면 [Shacham].Decrypt$_{\{sk\}}$(c_1,(OTS.vk,t))를 계산하여, 계산 값이 (x,y)에 대한 Diffie-Hellman의 솔루션이면, 계산 값을 리턴하고, 그렇지 않으면 Fail을 리턴한다.

[0038] - **Open**$_{\{skOA\}}$**(c,t)**

[0039] - c를 (c_1,c_2,OTS.vk,s)로 분해한다.

[0040] - (c_1,c_2,t)에 대한 서명 s를 OTS.vk에 관하여 체크한다; 만약 OTS.Verify$_{\{OTS.vk\}}$(s,(c_1,c_2,t)) = 0 이면 Fail을 리턴하고, 그렇지 않으면 [Kiltz].Decrypt$_{\{skOA\}}$(pk,OTS.vk)를 호출하고 그 결과를 리턴한다.

[0041] - **Prove(c,t,x,y)**. 태그 t에 관하여 암호문 c를 생성한 엔티티인, 앨리스는 c가 잘 형성되었고, 그것은 인증된 공개 키를 가지고 있는 소정의 익명의 구성원에 의해 복호화될 수 있다는 비상호적인 증명을 제공한다.

[0042] Kiayias-Tsiounis-Yung 및 Cathalo-Libert-Yung에 의해 제공된 스킴들은 강력한 보안 개념들을 만족하는 암호화 스킴들(즉, 강력한 상대에 대해 안전한 암호화 스킴들)을 이용한 구성을 인스턴스화(instantiating)함으로써 안전한 그룹 암호화를 달성한다. 사용된 빌딩 블록들에 따라, 결론적인 실현들이 다음과 같이 비교된다:

[0043] 1. Kiayias-Tsiounis-Yung: 이들의 포괄적인 구성의 인스턴스화는 1024-비트 모듈라이(moduli)를 사용한 2.5kB 크기의 암호문과 70kB 크기의 증명을 달성한다. 더욱이, 증명은 검증자와의 상호작용을 수반하고, 따라서 동일한 증명을 여러 차례 수행하고자 하는 경우, 증명자는 암호문을 생성하기 위해 사용된 모든 랜덤함(randomness)을 기억할 것이 요구된다.

[0044] 2. Cathalo-Libert-Yung은 상기 스킴을 더 개선한다; 이것은 256-비트 모듈라이를 사용한 더 작은 1.25kB 크기의 암호문과 16.125kB의 더 작은 증명을 달성한다. 더욱이, 본 증명은 비상호적이고 따라서 상태 기반 증명자(stateful prover)를 필요로 하지 않는다. 그러나 본 증명은 수백 또는 수천 개의 페어링 식 검증을 필요로 하는 고가의 Groth-Sahai 증명 시스템을 사용하므로 상당히 비현실적이다.

[0045] 당해 기술 분야에서 통상의 지식을 가진 자는 Kiayias-Tsiounis-Yung과 Cathalo-Libert-Yung 모두 암호문과 증명의 크기나 비용 때문에 여전히 꽤 고가라는 것을 인식할 것이다.

[0046] 예를 들어, Kiayias-Tsiounis-Yung과 Cathalo-Libert-Yung 모두 공개 키 - 공개 키는 항상 그룹 요소들의 벡터로 이루어짐 - 의 각 컴포넌트를 암호화하는 것에 의존하고, 그 결과 동일한 (리소스 사용의 측면에서) 고가의 암호화("E_2" 또는 [Kiltz])를 n회 적용하는데, 여기에서 n은 수신자의 공개 키에 있는 요소들의 개수를 나타낸다.

[0047] 당해 기술 분야에서 통상의 지식을 가진 자는 개선된 GE 스킴을 제공하는 솔루션이 필요하다는 것을 인지할 것이다. 본 발명은 그러한 솔루션을 제공한다.

발명의 내용

[0048] 제1 측면에서, 본 발명은 공개 키 pk를 가지고 있는 수신자가 암호문 c를 획득하도록 태그 t에 관하여 평문 m을 그룹 암호화하는 방법에 대한 것이다. 디바이스는 서명 키 OTS.sk와 검증 키 OTS.vk를 획득하고; E_1이 제1 암호화 알고리즘이고, E_2가 제2 암호화 알고리즘이고, f가 맵핑 함수일 때, $c_1 = E_1.Encrypt_{\{pk\}}(m,OTS.vk)$ 및 $c_2 = E_2.Encrypt_{\{pkOA\}}(f(pk),OTS.vk)$를 계산함으로써 제1 암호화된 값 c_1 및 제2 암호화된 값 c_2를 생성하고; OTS.sign이 서명 알고리즘일 때, $s = OTS.Sign_{\{OTS.sk\}}(c_1,c_2,t)$를 계산함으로써 서명 키 OTS.sk를 사용하여 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 및 태그 t에 대한 서명 s를 생성하고; 암호문 c를 출력하여 암호문 c는 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk 및 서명 s를 포함한다.

[0049] 바람직한 제1 실시예에서, 메시지 m은 공적으로 검증가능한 관계(publicly verifiable relation) R을 만족한다.

[0050] 제2 측면에서, 본 발명은, 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk와 서명 s를 포함하는 그룹 암호문 c를 복호화하는 방법에 대한 것이며, 서명 s는 제1 암호화된 값 c_1, 제2 암호화된 값 c_2 및 태그 t에 대한 것이다. 디바이스는 그룹 암호문 c를 수신하고; 검증 키 OTS.vk에 관하여 서명 s를 검증하고; 만약 s가 성공적으로 검증되면, 복호화 알고리즘 E_1과 검증 키 OTS.vk를 사용하여 제1 암호화된 값 c_1을 복호화한다.

[0051] 바람직한 제1 실시예에서, 서명을 검증하는 것은 제1 암호화된 값 c_1의 복호화가 공적 관계 R을 만족시키는지 검증하는 것을 더 포함한다.

[0052] 제3 측면에서, 본 발명은 공개 키 pk를 가지고 있는 수신자가 암호문 c를 획득하도록 태그 t에 관하여 평문 m을 그룹 암호화하는 디바이스에 대한 것이다. 디바이스는 서명 키 OTS.sk와 검증 키 OTS.vk를 획득하고; E_1이 제1 암호화 알고리즘이고, E_2가 제2 암호화 알고리즘이고, f가 맵핑 함수일 때, $c_1 = E_1.Encrypt_{\{pk\}}(m,OTS.vk)$ 및 $c_2 = E_2.Encrypt_{\{pkOA\}}(f(pk),OTS.vk)$를 계산함으로써 제1 암호화된 값 c_1 및 제2 암호화된 값 c_2를 생성하고; OTS.sign이 서명 알고리즘일 때, $s = OTS.Sign_{\{OTS.sk\}}(c_1,c_2,t)$를 계산함으로써 서명 키 OTS.sk를 사용하여 제1 커미트먼트 c_1, 제2 커미트먼트 c_2 및 태그 t에 대한 서명 s를 생성하고; 암호문 c를 출력 - 암호문 c는 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk와 서명 s를 포함함 - 하도록 구성된 프로세서를 포함한다.

[0053] 바람직한 제1 실시예에서, 메시지 m은 공적으로 검증가능한 관계 R을 만족한다.

[0054] 제4 측면에서, 본 발명은 제1 암호화된 값 c_1, 제2 암호화된 값 c_2, 검증 키 OTS.vk 및 서명 s를 포함하는 그룹 암호화 c를 복호화하기 위한 디바이스에 대한 것이며, 여기에서 서명 s는 제1 암호화된 값 c_1, 제2 암호화된 값 c_2와 태그 t에 대한 것이다. 디바이스는 그룹 암호화 c를 수신하고; 검증 키 OTS.vk에 관하여 서명 s를 검증하고; 만약 서명 s가 성공적으로 검증되면, 복호화 알고리즘 E_1과 검증 키 OTS.vk를 사용하여 제1 암호화된 값 c_1을 복호화하도록 구정된 프로세서를 포함한다.

[0055] 바람직한 제1 실시예에서, 프로세서는 제1 암호화된 값 c_1의 복호화가 공적 관계 R을 만족시키는지 더 검증한다.

[0056] 제5 측면에서, 본 발명은, 프로세서에 의해 실행될 때, 제1 측면의 방법을 수행하는 명령어들이 저장된 컴퓨터 프로그램 제품에 대한 것이다.

[0057] 제6 측면에서, 본 발명은, 프로세서에 의해 실행될 때, 제2 측면의 방법을 수행하는 명령어들이 저장된 컴퓨터 프로그램 제품에 대한 것이다.

도면의 간단한 설명

[0058] 첨부된 도면들을 참조하여, 본 발명의 바람직한 특징들이 비제한적인 예로서, 기술될 것이다.

도 1은 본 발명의 바람직한 실시예에 따른 그룹 암호화 시스템을 도시한다.

발명을 실시하기 위한 구체적인 내용

[0059] 본 발명의 주요한 독창적인 아이디어는 수신자의 공개 키 자체 대신 그 공개 키의 별명(alias)을 암호화하는 것이다. 그룹 매니저(GM; Group Manager)는 공개 키, 별명의 상응하는 암호화, 및 공용 데이터베이스(DB)에 있는 인증서를 공개한다(publish). 별명은 공개 키에 적용된, 적합하게 선택된 맵핑 함수 f의 결과 값이다.

[0060] 바람직하게도 입력의 크기를 감소시키는 함수 f를 사용하는 계산들은 수행하기에 쉽고, 두 개의 상이한 입력 값은 데이터베이스(DB)에 있는 동일한 엔트리로 이어지면 안된다. 맵핑 함수 f는 예를 들어 데이터베이스에 있는 엔트리가 고유할 때까지 새로운 메시지를 랜덤화하는 것에 의해 충돌 회피성(collision resistant) 특성을 보장하는 그룹 매니저를 가짐으로써, 충돌 회피성을 갖는 해시 함수(hash function)의 일종이라고 할 수 있다.

[0061] 별명이 공개 키보다 더 작게 만들어질 수 있기 때문에, 결과 구조의 크기와 비용의 상당한 감소가 허용될 수 있다. 특히, 제2 암호화 스킴을, 수신자의 공개 키에 있는 그룹 요소들의 개수만큼 적용할 필요가 없다.

[0062] 그러나, 오픈 절차에서 오프닝 권한(OA; opening authority)이 데이터베이스(DB)에서 별명의 프리이미지(preimage, 공개 키)를 찾을 것이 요구된다는 단점이 있다. 다행히, 오픈 절차의 사용은 매우 드물게, 예를 들면 논쟁의 경우에서만 발생한다.

[0063] 본 발명의 그룹 암호화 스킴은 수 개의 빌딩 블록들을 사용한다(예들은 본 기술의 후반부에서 주어질 것이다):

[0064] - 암호화 스킴들: E_1과 E_2의 두 개의 암호화 스킴이 사용된다. 본 발명의 목적을 위해서, 아래 정의된 바와 같이, E_1과 E_2가 약하게 보호되는 것으로 충분하다는 점이 주목할 만하다:

[0065] 약하게 보안된 암호화 스킴은 "가장 높은" 보안 개념("highest" security notion)에 이르지 못하는 것이다. E_1에 대한 올바른 보안 개념은 IND-st-wCCA 및 ANO-st-wCCA(indistinguishable and anonymous under selective tag weak chosen ciphertext attacks)이다. E_2에 대해서는 오직 IND-st-wCCA 보안만 필요하다.

[0066] 보안 개념들 둘 모두는 보안 목표(IND 또는 ANO)와 공격 모델(st-wCCA)을 조합한다.

[0067] 구분불가능성(IND; indistinguishability) 목표는 비공식적으로 암호문으로부터 메시지에 관한 정보를 얻는 것의 어려운 정도를 나타낸다. 익명성(ANO; anonymity)은 암호문으로부터 공개 키에 관한 정보를 추론하는 것의 어려운 정도를 나타낸다.

[0068] 공격 모델 st-wCCA에 관하여, 그것은 공격자가 미리(챌린지 공개 키를 수신하기 전에), 그가 도전하고자 원하는 태그에 대해 커미트하고, 이러한 챌린지 태그를 수반하는 복호화 쿼리들의 발행(issue)이 허용되지 않는 시나리오를 나타낸다.

[0069] - **서명 또는 인증 스킴들**: 그룹 요소들을 서명하는 서명 스킴이 사용된다. 적절한 후보는 구조-보존(structure-preserving) 서명 스킴 S, 즉, 검증 키, 메시지들 및 서명들이 그룹 요소들이고, 검증 알고리즘이 페어링 식 검증들의 서술부(predicate)로 이루어진 스킴 이다.

[0070] - **일회용 서명 스킴들**: 안전한 일회용 서명 OTS가 사용된다.

[0071] - **관계 R**: 공개적으로 검증가능한 관계가 사용된다.

[0072] **함수 f**: 효율적으로 계산가능한(입력의 크기 내의 다항 시간(polynomial time) 내에 예측될 수 있는) 함수 f가 사용된다.

[0073] 이러한 빌딩 블록들을 사용하여 스킴은 다음과 같이 구성된다.

[0074] -**Join.** 입력 공개 키 pk에 대하여, GM이 S.sk(S는 사용된 인증 스킴임)를 사용하여 pk에 대한 서명 (또는 인증서) cert와 더불어, f(pk)를 계산한다. GM은 더 나아가 (pk,f(pk),cert)를 공용 데이터베이스 DB에 저장한다. GM은 충돌들을 피하기 위하여, 즉 두 개의 상이한 공개 키 pk와 pk'이 f를 사용하여 동일한 값에 맵핑되는 것을 피하기 위해서, 단순한 측정들을 계속 진행한다는 것에 주목한다. 특정 함수 f가 사용되는 하나의 가능한 수단이 이하에서 상세히 설명될 것이다.

[0075] - **Encrypt**$_{\{pk\}}$**(m,t).** 태그 t에 관하여 공개 키 pk를 갖는 수신자를 위해, 메시지 m(m은 알려진 관계 R에 관한 소정의 x의 위트니스임)을 암호화하기위해, 엔티티는:

[0076] - 서명 키와 검증 키의 쌍 (OTS.sk,OTS.vk)를 생성하기 위해 OTS.keygen을 호출한다.

[0077] - (c_1,c_2) = (E_1.Encrypt$_{\{pk\}}$(m,OTS.vk), E_2.Encrypt$_{\{pkOA\}}$(f(pk),OTS.vk))를 생성한다. OTS.vk는 태그로 간주된다는 것이 주목할만 할 것이다.

[0078] - OTS.sk를 사용하여, (c_1,c_2,t)에 관한 서명 s를 발생시킨다; s = OTS.Sign$_{\{OTS.sk\}}$$(c_1,c_2,t)$.

[0079] - t에 관하여 pk 하에서 m의 그룹 암호화로서 c = $(c_1,c_2$,OTS.vk,s)를 리턴한다.

[0080] - **Decrypt$_{\{sk\}}$(c,t,x)**

[0081] - c를 $(c_1,c_2$,OTS.vk,s)로 분해한다.

[0082] - (c_1,c_2,t)에 대한 서명 s를 OTS.vk에 관하여 검증한다. 만약 OTS.Verify$_{\{OTS.vk\}}$$(s,(c_1,c_2,t))$ = 0 이면 Fail을 리턴하고, 그렇지 않으면 E_1.Decrypt$_{\{sk\}}$$(c_1$,OTS.vk)를 계산하여, 계산된 값이 R에 관하여 x에 대한 위트니스이면, 계산된 값을 리턴하고, 그렇지 않으면 Fail을 리턴한다.

[0083] - **Open$_{\{skOA\}}$(c,t)**.

[0084] - c를 $(c_1,c_2$,OTS.vk,s)로 분해한다.

[0085] - (c_1,c_2,t)에 대한 서명 s를, OTS.vk에 관하여 검증한다. 만약 OTS.Verify$_{\{OTS.vk\}}$$(s,(c_1,c_2,t))$ = 0 이면 Fail을 리턴하고, 그렇지 않으면 F 값을 리턴하는 E_2.Decrypt$_{\{skOA\}}$$(c_2$,OTS.vk)를 호출한다.

[0086] - DB에서 함수 f에 관하여 F 값의 프리이미지를 찾아보고, 검색 결과를 리턴한다.

[0087] - **Prove(c,t,x)**. 태그 t에 관하여 암호문 c를 생성한 엔티티는 c = $(c_1,c_2$,OTS.vk,s)를 생성하는 데 사용된 랜덤 코인들을 사용하여, 이하의 증명들을 제공한다.

[0088] - 소정의 공개 키 pk 하의 태그 OTS.vk에 관하여, c_1의 기저에 있는 메시지에 관한 지식의 증명(proof of knowledge) 및 이 메시지가 관계 R에 관하여 x에 대한 위트니스임을 증명.

[0089] - 키 pk_{OA} 하에서 태그 OTS.vk에 관하여 c_2의 복호화에 관한 지식의 증명 및 해당 복호화가 pk에 관한 함수 f의 값임을 증명.

[0090] - pk에 대한 인증서 cert의 지식의 증명.

[0091] 서명과 암호화 스킴들의 특정 클래스들은 이러한 증명의 효율적인 수행을 허용한다.

[0092] 비상호적인 증명들에 대하여, Groth-Sahai에 의한 [Jens Groth, Amit Sahai: Efficient Non-interactive Proof Systems for Bilinear Groups. EUROCRYP 2008: 415-432]와 호환가능한 암호체계와 같이, 당해 위트니스(즉, 서명/암호화 스킴들의 경우에 메시지나 키, 함수 f의 경우에 프리이미지, 또는 관계 R의 경우에 위트니스)에 대한 지식의 효율적인 비상호적 증명을 수용하는 컴포넌트들을 사용하는 것이 선호된다. 이런 의미에서, 소위 자기 동형(automorphic) 서명들(즉, 검증 키 메시지와 결과 서명이 그룹 요소들이고 검증 알고리즘이 페어링 곱 식들(pairing product equations)의 결합(conjuction)으로 이루어지는 서명 스킴) 및 암호화 스킴들 - 암호화 알고리즘이 입력에 관한 그룹 또는 페어링 동작들을 수행하는(이것은 메시지, 공개 키 및 암호문이 그룹 요소들인 것을 수반함) 암호화 스킴들을 사용할 수 있다. 함수 f 또한 입력에 관한 그룹 (또는, 이선형 식들(bilinear equations)의 경우에는 페어링) 동작들을 수행한다. 관계 R에 대해서도 동일하게 적용된다.

[0093] 유사하게, 위트니스의 효율적인 상호적 증명들을 수용하는 컴포넌트들을 사용하는 것이 선호된다. 이런 의미에서, 서명 σ가 메시지 M에 대한 것이라는 전제 하에, 준동형(homomorphic) 함수 φ를 정의하여 φ(S,M)이 g(R,vk)를 평가하는 것을 가능하도록 하는 서명 스킴들을 사용하는 것이 선호되며, vk는 검증 키이고, g는 공개 함수이고, (S,R)은 σ로부터 변환된 쌍이고, R은 σ 또는 M에 관한 정보를 누설하지 않으며 S는 서명의 "필수적(vital)"인 부분이고; 기저의 변환 알고리즘은 CONVERT 알고리즘으로 지칭된다. 주어진 키와 주어진 태그에 관하여 복호화의 정확함의 효율적인 증명들을 수용하는 암호화 스킴들을 사용하는 것 또한 선호된다. 더욱이, 공개 키를 암호화하는 데 사용된 스킴(E_2)은, 메시지에 관하여 준동형이어야 하고, 스킴 E_1은 공개 키와 메시지 둘

모두에 관하여 준동형이어야 한다. 더 나아가, 암호화 스킴(E_1)은 COMPUTE 알고리즘으로 지칭되는 알고리즘을 수반하고, 그 알고리즘은 입력에서, 주어진 태그 t에 관한 공개 키 pk 하에서의 메시지 m의 암호화 c_1이 동일한 태그 t에 관한 또 다른 공개 키 pk' 하에서의 또 다른 메시지 m'의 또 다른 암호화 c_1'을 생성하여, c_1과 c_1'의 합성(composition)은, 태그 t에 관한 pk 및 pk'의 합성 하에서의 m 및 m'의 합성의 암호화와 같게 하는 것이고; 여기에서 합성은 관여된 요소들이 속하는 집합을 갖춘 대수적 그룹 연산(algebraic group operation)을 적용하는 것으로 이해되어야 한다. 게다가, 함수 f는 바람직하게도 준동형 함수이다(두 개의 입력의 구성에 적용된 f는, 그 두 개의 입력에서의 f값들의 합성이다). 그리고 유사하게, 인스턴스 x가 주어졌다는 전제 하에, 관계 R은 F_R(w)=I를 만족하도록 준동형 함수 F_R과 이미지 I를 정의하도록 허용해야 하고, 여기에서 w는 인스턴스 x에 상응하는 위트니스이다.

[0094] 본 발명과 함께 사용하는 데에 선호되는 암호화 스킴은, CRYPTO 2010의 209-236면에 있는 "Structure-Preserving Signatures and Commitments to Group Elements"에서 Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev 및 Miyako Ohkubo에 의해 제안되었다.

[0095] 본 발명과 함께 사용하는 데에 선호되는 서명 스킴은, David Cash, Eike Kiltz 및 Victor Shoup에 의해 Journal of Cryptology 22(4):470-502 (2009)의 "The Twin Diffie-Hellman Problem and Applications"에 제공된 약하게 보안된 태그-기반 이형(weakly secure tag-based variant)이다.

[0096] 만약 공개 키들이 그룹 요소들의 n-벡터들이면, $(G,\cdot)$은 소정의 d자릿수의 그룹이고, n은 소정의 정수이고, $a_1,\ldots,a_n$은 $\mathbb{Z}_d$로부터의 공개 요소들, 즉 정수 모듈로 d의 집합(set of integers modulo d)이라는 전제 하에, 선호되는 함수 f는 다음과 같다:

[0097] f: $G^n \rightarrow G$

[0098] $(X_1,\ldots,X_n) \rightarrow X_1^{a_1}\ldots X_n^{a_n}$

[0099] 그러면, 함수 f는 그 후로 그룹 G 내의 n개의 요소들의 튜플(tuple)을 그룹 G 내의 요소에 맵핑한다.

[0100] 이러한 f의 선택으로, GM은 키 pk를 조직적으로(systematically) 랜덤화함으로써 충돌들을 피할 수 있다. 더 정확히 말하면, GM은 함수 f의 충돌들을 피하기 위해, 지수 그룹 $\mathbb{Z}_d$에 있는 랜덤 $r = (r_1,\ldots,r_n)$을 고려하여 pk를 랜덤화한다; $pk = (X_1,\ldots,X_n) \leftarrow pk^r = (X_1^{r1},\ldots,X_n^{rn})$. 또한 GM은 r을 공개하여, 수신자가 그에 따라 그의 개인 키를 업데이트하는 것을 허용한다 - 이것은 $X_i = g_i^{x_i}$ 가 성립할 경우에만 가능한데, 여기에서 g_i는 G의 알려진 생성자이고 x_i는 X_i에 상응하는 개인 키이다. 인증서는 새로이 계산된 공개 키에 대해 계산되고, 공개 키 및 그것의 별명과 함께 DB에 저장된다.

[0101] 최종적으로 선호되는 관계 R은 $(m,x,y) \in R \leftrightarrow e(m,P)=e(x,y)$이고, 여기에서 e는 도메인 $G \times H$(G와 H는 암호학적 이선형 그룹들(cryptographical bilinear groups)임)와의 효율적인 페어링이고, P는 H로부터의 고정된 요소이다.

[0102] 공개 키 pk를 가진 수신자를 위해 암호문 c를 생성한 증명자와 임의의 검증자 사이의 상호적인 Prove 프로토콜은 세 단계(pass), 즉 커미트먼트, 챌린지, 및 응답으로 진행된다. 커미트먼트 단계에서, 증명자는 쌍 (S,R)을 획득하기 위하여 그룹 매니저의 공개 키 S.pk, 공개 키 pk 및 상응하는 인증서인 입력 상에서 CONVERT 알고리즘을 실행한다. 증명자는 또한 입력 c_1 상에서 COMPUTE 알고리즘을 실행하고, 튜플(pk',m',c'_1)을 획득한다. 다음으로, 증명자는 F'=f(pk'), I'_R=F_R(m') 및 I'=ϕ(S',pk')를 계산한다. 최종적으로, 증명자는 공개 키 pk_{OA} 하에서의 F'의 암호화인 c_2'를 계산한다. 증명자는 검증자에게 튜플 (R,I',I'_R,c'_2)를 전송한다. 챌린지 단계에서, 해당 튜플을 수신하면, 검증자는 랜덤으로 정수 b를 선택하고, F_R(m)= I_R이 성립하도록 I = g(R,S.pk)와 I_R을 계산한다. 검증자는 챌린지 b를 증명자에게 전송한다. 해당 챌린지를 수신하면, 증명자는 z_S, z_{pk},

z_m 및 z_F 값들을 계산하고 전송하는데, 여기에서 z_{pk}는 pk'와 pk^b의 합성, z_s은 S'와 S^b의 합성, z_m은 m'과 m^b의 합성, 그리고 z_F는 F'와 F^b의 합성이다. 최종적으로 증명자는, c'$_1$과 c_1^b의 합성 (PoK1)이 태그 t에 관한 공개 키 z_{pk} 하에서의 z_m의 암호화라는 지식, 및 c'$_2$와 c_2^b의 합성 (PoK2)가 태그 t에 관한 공개 키 pk_{OA} 하에서의 z_F의 암호화라는 지식을 증명한다. 프로토콜의 말미에서, (1) $\phi(z_s, z_{pk})$가 I'와 I^b의 합성과 같고, (2)F_R(z_m)이 I'_R과 I^b_R의 합성이고, (3)f(z_{pk})가 z_F와 같으며, (4)PoK1과 PoK2가 유효하면, 검증자는 수용한다. 당해 기술 분야에서 통상의 지식을 가진 자는, 선호된 암호화 스킴들로 인스턴스화될 때, 지식의 증명들 PoK1과 PoK2는 이산 대수들(discrete logarithms)의 대등함을 보여주는 것이 된다는 것을 관찰할 것이고, 그 방법으로서 효율적인 것은 Claus P.Schnorr의 세미나 주제인 "Efficient signature generationby smart cards", Journal of Cryptology, 4(3):161-179, 1991로부터 도출될 수 있다. 또한 Jan Camenisch에 의한 "Group signature schemes and payment systems based on the discrete logarithm problem", PhD 논문, ETH Series in Information Security and Cryptography의 vol.2, Hartung-Gorre Verlag, 1998 (ISBN 3-89649-286-1)를 참조하기 바란다.

[0103] 도 1은 본 발명의 바람직한 실시예에 따른 그룹 암호화를 위한 시스템(100)을 도시한다. 용이한 도시와 이해를 위해, 시스템 내의 디바이스들 간의 연결들은 생략되었다.

[0104] 시스템(100)은 송신기(110)와 수신기(120)를 포함하며, 이들 각각은 다른 디바이스와 통신하기 위해 구성된 적어도 하나의 인터페이스 유닛(111, 121), 적어도 하나의 프로세서("프로세서")(112, 122), 및 누산기(accumulator)와 중간의 계산 결과들과 같은 데이터를 저장하도록 구성된 적어도 하나의 메모리(113, 123)를 포함한다. 시스템(100)은 그룹 매니저(130), 데이터베이스(140), 제 3자(a third party)(150)와 오프닝 권한(160)을 더 포함하고; 간결함을 위하여 도시되지 않았지만, 이들 디바이스들 각각은 프로세서 및 메모리와 같은 필요한 하드웨어를 포함한다.

[0105] 송신기(110)의 프로세서(112)는 본 그룹 암호화 스킴의 Encrypt와 Prove 부분을 수행하도록 구성되고, 수신기(120)의 프로세서(122)는 수신한 그룹 암호화를 복호화하도록, 즉 Decrypt를 수행하도록 적응된다. 그룹 매니저(130)는 Join 부분을 수행하도록 구성되고, 그에 의해 데이터베이스(140)에 데이터를 저장한다. 제 3자(150)는 송신기에 의해 제공된 증명들을 검증하도록 구성되고, 오프닝 권한(160)은 그룹 암호화 스킴의 Open 부분을 수행하도록 구성된다. CD-ROM이나 DVD와 같은 제1 컴퓨터 프로그램 제품(114)은 송신기(110)의 프로세서(112)에 의해 실행될 때 본 발명에 따른 Encryption과 Prove를 수행하는, 저장된 명령어들을 포함한다. 제2 컴퓨터 프로그램 제품(124)은 수신기(120)의 프로세서(122)에 의해 실행될 때 본 발명에 따른 Decrypt를 수행하는, 저장된 명령어들을 포함한다.

[0106] 당해 기술 분야에서 통상의 지식을 가진 자는, 본 발명의 그룹 암호화 스킴이 종래 스킴들과 비교하여 크기와 비용의 상당한 감소를 가능하게 한다는 것을 인지할 것이다. 예컨대, 본 발명의 GE 스킴은, 이하 기술되는 것들로 인스턴스화된다면, (종래 기술에서의 1.25kB 또는 2.5kB 대신) 0.4kB의 암호문을 야기한다:

[0107] - Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev 및 Miyako Ohkubo에 의해 Crypto 2010의 209-236면의 "Structure-Preserving Signatures and Commitments to Group Elements"에서 제안된 서명 스킴.

[0108] - Jens Groth, Rafail Ostrovsky 및 Amit Sahai에 의해 CRYPTO 2006의 97-111면의 "Non-interactive Zaps and New Techniques for NIZK"에서 제공된 일회용 서명.

[0109] - E_1 및 E_2를 인스턴스화하기 위해 David Cash, Eike Kiltz 및 Victor Shoup에 의해 Journal of Cryptology 22(4)(2009)의 470-504면의 "The Twin Diffie-Hellman Problem And Applications"에 제공된 약하게 보안된 태그-기반의 변종.

[0110] 부가적으로, 증명들은 더 짧고, 검증자와 상호작용하며 또는 상호작용 없이 수행될 수 있고(상호적 증명에 대해서는 1kB, 비상호적 증명에 대해서는 2kB), 검증자가 저렴한, 상호적 증명을 수행할지 아니면 고가의 비상호적 증명을 수행할지 선택하게 한다. 게다가, 증명의 검증은 (종래 기술에서의 3895회의 페어링 계산과 비교하여) 325회의 페어링 계산을 필요로 한다.

[0111] 상기 언급했듯이, 본 발명의 GE 스킴은 공개 키의 별명의 프리이미지를 찾기 위해 각 오픈 절차에서 데이터베이

스(DB)를 액세스해야한다는 단점이 있다. 다행스럽게도, 오픈에의 의존은 충돌의 경우에만 발생하고, 따라서 매우 드물다.

[0112] 본 발명이 GE의 맥락에서 기술되었으나, 본 발명의 범위는 이러한 종류의 암호 스킴들로 제한되지 않는다. (온라인) 공용 DB에 존재하는 (긴) 메시지들의 암호화를 수반하는 어떠한 암호 스킴도 똑같이 본 발명으로부터 이득을 볼 수 있다. 본 발명을 적용하면, (짧은) 별명들이 DB에 있는 각각의 메시지와 연관될 것이고 DB에 부가될 것이다. 그렇다면, 메시지의 암호화가 별명의 암호화로 교체될 것이고, 따라서 암호문의 크기를 줄일 것이다. 복호화에서, 별명이 복원될 것이고, 온라인 DB에의 요청을 이용하여 연관된 메시지 또한 복원될 것이다.

[0113] 발명의 기술, (적절한) 청구항들과 도면에 개시된 각 특징은 독립적으로 제공되거나 임의의 적절한 조합으로 제공될 수 있다. 하드웨어로 구현되는 것으로 기술된 특징들은 소프트웨어로도 구현될 수 있고, 그 역으로도 가능하다. 청구항들에 있는 참조 번호들은 예시를 위한 것이고, 청구항들의 범위를 한정하는 효과를 나타내지 않을 것이다.

도면

도면1