(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0137789
(43) 공개일자 2024년09월20일

(51) 국제특허분류(Int. Cl.)
H04L 9/08 (2006.01)
(52) CPC특허분류
H04L 9/0891 (2013.01)
H04L 9/0822 (2013.01)
(21) 출원번호 10-2023-0030991
(22) 출원일자 2023년03월09일
심사청구일자 2023년03월09일

(71) 출원인
국방과학연구소
대전광역시 유성구 북유성대로488번길 160 (수남동)
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
이광수
서울특별시 광진구 능동로 209 (군자동)
(74) 대리인
제일특허법인(유)

전체 청구항 수 : 총 24 항

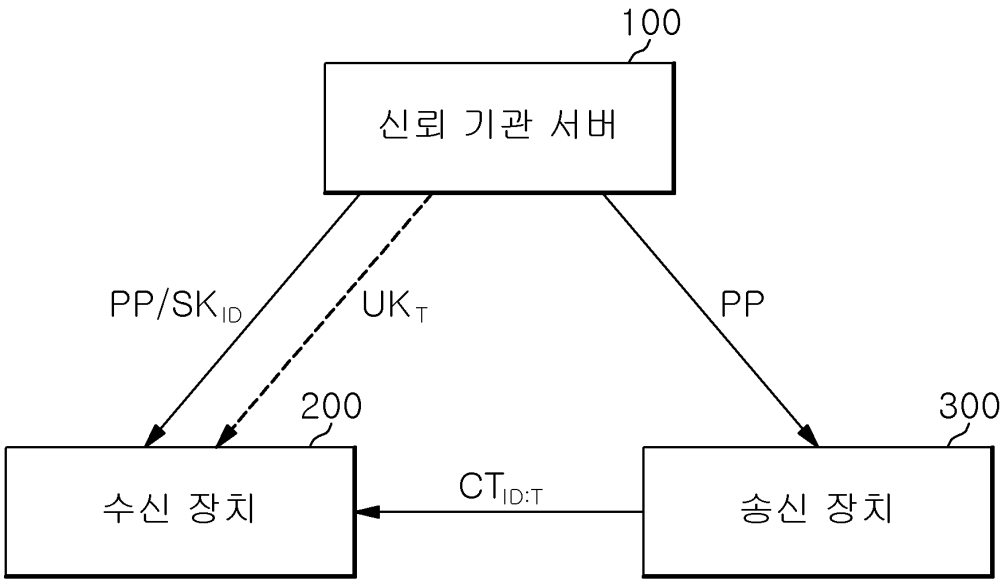
(54) 발명의 명칭 **폐기 가능한 아이디 기반의 암호화 방법, 이를 위한 암호화 시스템**

(57) 요 약

본 발명은 암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법에 관한 것으로, 상기 암호화 시스템의 신뢰기관 서버가 공개 정보를 생성하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 공개 파라미터를 수신 장치 및 송신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비

(뒷면에 계속)

대 표 도 - 도4



밀 키를 생성하는 단계; 상기 신뢰기관 서버가 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 상기 폐기 리스트가 갱신되면 상기 신뢰기관 서버가 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계; 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 갱신 키와 상기 사용자 비밀 키를 기초로 복호화 키를 생성하는 단계; 및 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 복호화 키를 이용하여 상기 송신 장치로부터 전송된 암호문을 복호화하는 단계;를 포함할 수 있다.

(52) CPC특허분류
H04L 9/0861 (2013.01)
H04L 2209/42 (2013.01)

명 세 서

청구범위

청구항 1

암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법에 있어서,

공개 정보를 생성하는 단계;

상기 공개 정보 중 공개 파라미터를 상기 암호화 시스템 내의 수신 장치 및 송신 장치로 전송하는 단계;

상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계;

상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계;

상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및

상기 폐기 리스트가 갱신되면 상기 수신 장치 중 비폐기 사용자 아이디(identification, ID)를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함하는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 2

제 1 항에 있어서,

상기 갱신 키는 상기 폐기 리스트, 갱신 시간, 상기 마스터 비밀 키, 상태 정보 및 상기 공개 파라미터를 기초로 생성되는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 3

제 1 항에 있어서,

상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는 단계;를 더 포함하는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 4

제 1 항에 있어서,

상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성되는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 5

제 1 항에 있어서,

상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신되는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 6

제 1 항에 있어서,

상기 판단하는 단계는 CS(complete subset) 기법이 적용되는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 7

제 6 항에 있어서,

상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서(cover) 전체 폐기 리스트를 갱신하는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 8

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법에 있어서,

상기 암호화 시스템의 신뢰기관 서버가 공개 정보를 생성하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 공개 파라미터를 수신 장치 및 송신 장치로 전송하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계;

상기 신뢰기관 서버가 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및

상기 폐기 리스트가 갱신되면 상기 신뢰기관 서버가 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함하는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 9

제 8 항에 있어서,

상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 갱신 키와 상기 사용자 비밀 키를 기초로 복호화 키를 생성하는 단계; 및

상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 복호화 키를 이용하여 상기 송신 장치로부터 전송된 암호문을 복호화하는 단계;를 더 포함하는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 10

제 8 항에 있어서,

상기 갱신 키는 상기 폐기 리스트, 갱신 시간, 상기 마스터 비밀 키, 상태 정보 및 상기 공개 파라미터를 기초로 생성되는

폐기 가능한 아이디 기반의 암호화 방법.

청구항 11

제 8 항에 있어서,

상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는 단계;를 더 포함하는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 12

제 8 항에 있어서,

상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성되는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 13

제 8 항에 있어서,

상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신되는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 14

제 8 항에 있어서,

상기 판단하는 단계는 CS 기법이 적용되는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 15

제 14 항에 있어서,

상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서 전체 폐기 리스트를 갱신하는

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법.

청구항 16

암호문을 송신하는 송신 장치;

상기 송신 장치로부터 암호문을 수신하고, 비폐기 사용자 아이디를 갖는 경우에 갱신 키와 사용자 비밀 키를 기초로 복호화 키를 생성하며, 상기 복호화 키를 기초로 상기 암호문을 복호화하는 수신 장치; 및

암호화 시스템의 공개 정보를 생성하고, 상기 공개 정보 중 공개 파라미터를 상기 수신 장치 및 상기 송신 장치로 전송하며, 상기 공개 정보 중 마스터 비밀 키를 이용하여 상기 사용자 비밀 키를 생성하고, 상기 사용자 비밀 키를 상기 수신 장치로 전송하며, 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하고, 상기 폐기 리스트가 갱신되면 상기 수신 장치로 상기 갱신 키를 전송하는 신뢰기관 서버;를 포함하는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 17

제 16 항에 있어서,

상기 신뢰기관 서버는 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 18

제 16 항에 있어서,

상기 신뢰기관 서버는 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 19

제 16 항에 있어서,

상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성되는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 20

제 16 항에 있어서,

상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신되는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 21

제 16 항에 있어서,

상기 신뢰기관 서버는 CS 기법이 적용되는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 22

제 21 항에 있어서,

상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서 전체 폐기 리스트를 갱신하는

폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템.

청구항 23

컴퓨터 프로그램을 저장하고 있는 컴퓨터 판독 가능 기록매체로서,

상기 컴퓨터 프로그램은,

암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 프로세서가 수행하도록 하기 위한 명령어를 포함하고,

상기 방법은,

암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법에 있어서,

공개 정보를 생성하는 단계;

상기 공개 정보 중 공개 파라미터를 상기 암호화 시스템 내의 수신 장치 및 송신 장치로 전송하는 단계;

상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계;

상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계;

상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및

상기 폐기 리스트가 갱신되면 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함하는

컴퓨터 판독 가능한 기록매체.

청구항 24

컴퓨터 판독 가능 기록매체에 저장된 컴퓨터 프로그램으로서,

상기 컴퓨터 프로그램은,

암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법을 프로세서가 수행하도록 하기 위한 명령어를 포함하고,

상기 방법은,

상기 암호화 시스템의 신뢰기관 서버가 공개 정보를 생성하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 공개 파라미터를 수신 장치 및 송신 장치로 전송하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계;

상기 신뢰기관 서버가 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계;

상기 신뢰기관 서버가 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및

상기 폐기 리스트가 갱신되면 상기 신뢰기관 서버가 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계;

상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 갱신 키와 상기 사용자 비밀 키를 기초로 복호화 키를 생성하는 단계; 및

상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 복호화 키를 이용하여 상기 송신 장치로부터 전송된 암호문을 복호화하는 단계;를 포함하는

기록매체에 저장된 컴퓨터 프로그램.

발명의 설명

기 술 분 야

[0001] 본 발명은 암호화 시스템과 관련이 있고, 특히 사용자 아이디(ID, identification) 기반의 암호화 기술과 관련

이 있다.

배 경 기 술

[0003] 클라우드 서버나 블록체인과 같이 공개된 플랫폼에 데이터를 업로드하는 현대의 데이터 환경에는 프라이버시 보존을 위해 익명(anonymous)의 기능이 중요하다. 하지만 기존의 사용자 아이디 기반의 암호화 기술에서는 아래 두 가지의 프라이버시 노출 요소가 존재한다.

[0004] 첫째는 암호문에서 사용자의 아이디가 노출될 위협이 있다는 점이고, 둘째는 갱신 키 발급시에 폐기된 사용자에 대한 정보가 노출된다는 점이다. 갱신 키를 통해 폐기된 사용자들이 노출된다면 그룹을 정상적으로 탈퇴한 사용자들의 프라이버시 노출 위협이 발생한다.

선행기술문헌

특허문헌

[0006] (특허문헌 0001) 등록특허공보 제10-2463640호 (2022년11월09일 등록공고)

발명의 내용

해결하려는 과제

[0007] 본 발명의 실시예에서는, 암호문에서 사용자의 아이디가 노출될 수 있는 위협으로부터 보호할 수 있는 폐기 가능한 아이디 기반의 암호화 기술을 제안하고자 한다.

[0008] 본 발명의 실시예에서는, 갱신 키 발급시에 폐기된 사용자에 대한 정보가 노출될 수 있는 프라이버스 문제를 해결할 수 있는 폐기 가능한 아이디 기반의 암호화 기술을 제안하고자 한다.

[0009] 본 발명이 해결하고자 하는 과제는 상기에서 언급한 것으로 제한되지 않으며, 언급되지 않은 또 다른 해결하고자 하는 과제는 아래의 기재들로부터 본 발명이 속하는 통상의 지식을 가진 자에 의해 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0011] 본 발명의 실시예에 따르면, 암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법에 있어서, 공개 정보를 생성하는 단계; 상기 공개 정보 중 공개 파라미터를 상기 암호화 시스템 내의 수신 장치 및 송신 장치로 전송하는 단계; 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계; 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계; 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및 상기 폐기 리스트가 갱신되면 상기 수신 장치 중 비폐기 사용자 아이디(identification, ID)를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함하는 폐기 가능한 아이디 기반의 암호화 방법을 제공할 수 있다.

[0012] 여기서, 상기 갱신 키는 상기 폐기 리스트, 갱신 시간, 상기 마스터 비밀 키, 상태 정보 및 상기 공개 파라미터를 기초로 생성될 수 있다.

[0013] 또한, 상기 방법은 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는 단계;를 더 포함할 수 있다.

[0014] 또한, 상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성될 수 있다.

[0015] 또한, 상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신될 수 있다.

[0016] 또한, 상기 판단하는 단계는 CS(complete subset) 기법이 적용될 수 있다.

[0017] 또한, 상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서(cover) 전체 폐기 리스트를 갱신할 수 있다.

[0018] 본 발명의 실시예에 따르면, 암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법에 있어서, 상기 암호화 시스템의 신뢰기관 서버가 공개 정보를 생성하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 공개 파라미터를 수신 장치 및 송신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계; 상기 신뢰기관 서버가 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및 상기 폐기 리스트가 갱신되면 상기 신뢰기관 서버가 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함하는 암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법을 제공할 수 있다.

[0019] 여기서, 상기 방법은 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 갱신 키와 상기 사용자 비밀 키를 기초로 복호화 키를 생성하는 단계; 및 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 복호화 키를 이용하여 상기 송신 장치로부터 전송된 암호문을 복호화하는 단계;를 더 포함할 수 있다.

[0020] 또한, 상기 갱신 키는 상기 폐기 리스트, 갱신 시간, 상기 마스터 비밀 키, 상태 정보 및 상기 공개 파라미터를 기초로 생성될 수 있다.

[0021] 또한, 상기 방법은 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하는 단계;를 더 포함할 수 있다.

[0022] 또한, 상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성될 수 있다.

[0023] 또한, 상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신될 수 있다.

[0024] 또한, 상기 판단하는 단계는 CS 기법이 적용될 수 있다.

[0025] 또한, 상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서 전체 폐기 리스트를 갱신할 수 있다.

[0026] 본 발명의 실시예에 따르면, 암호문을 송신하는 송신 장치; 상기 송신 장치로부터 암호문을 수신하고, 비폐기 사용자 아이디를 갖는 경우에 갱신 키와 사용자 비밀 키를 기초로 복호화 키를 생성하며, 상기 복호화 키를 기초로 상기 암호문을 복호화하는 수신 장치; 및 암호화 시스템의 공개 정보를 생성하고, 상기 공개 정보 중 공개 파라미터를 상기 수신 장치 및 상기 송신 장치로 전송하며, 상기 공개 정보 중 마스터 비밀 키를 이용하여 상기 사용자 비밀 키를 생성하고, 상기 사용자 비밀 키를 상기 수신 장치로 전송하며, 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하고, 상기 폐기 리스트가 갱신되면 상기 수신 장치로 상기 갱신 키를 전송하는 신뢰기관 서버;를 포함하는 폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템을 제공할 수 있다.

[0027] 여기서, 상기 신뢰기관 서버는 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인할 수 있다.

[0028] 또한, 상기 신뢰기관 서버는 상기 폐기 리스트가 갱신되면 갱신된 폐기 리스트를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인할 수 있다.

[0029] 또한, 상기 사용자 비밀 키는 사용자 아이디 및 상태 정보를 기초로 생성될 수 있다.

[0030] 또한, 상기 폐기 리스트는 사용자 아이디가 폐기될 때 갱신될 수 있다.

[0031] 또한, 상기 신뢰기관 서버는 CS 기법이 적용될 수 있다.

[0032] 또한, 상기 CS 기법은 폐기되는 사용자 아이디의 부분 트리로 전체 트리를 덮어서 전체 폐기 리스트를 갱신할 수 있다.

[0033] 본 발명의 실시예에 따르면, 컴퓨터 프로그램을 저장하고 있는 컴퓨터 판독 가능 기록매체로서, 상기 컴퓨터 프로그램은, 암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 프로세서가 수행하도록 하기 위한 명령어를 포함하고, 상기 방법은, 암호화 시스템의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법에 있어서, 공개 정보를 생성하는 단계; 상기 공개 정보 중 공개 파라미터를 상기 암호화 시스템 내의 수신 장치 및 송신 장치로 전송하는 단계; 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계; 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계; 상기 공개 정보

중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및 상기 폐기 리스트가 갱신되면 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계;를 포함할 수 있다.

[0034] 본 발명의 실시예에 따르면, 컴퓨터 판독 가능 기록매체에 저장된 컴퓨터 프로그램으로서, 상기 컴퓨터 프로그램은, 암호화 시스템의 폐기 가능한 아이디 기반의 암호화 방법을 프로세서가 수행하도록 하기 위한 명령어를 포함하고, 상기 방법은, 상기 암호화 시스템의 신뢰기관 서버가 공개 정보를 생성하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 공개 파라미터를 수신 장치 및 송신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 마스터 비밀 키를 이용하여 사용자 비밀 키를 생성하는 단계; 상기 신뢰기관 서버가 상기 사용자 비밀 키를 상기 수신 장치로 전송하는 단계; 상기 신뢰기관 서버가 상기 공개 정보 중 폐기 리스트가 갱신되는지 여부를 판단하는 단계; 및 상기 폐기 리스트가 갱신되면 상기 신뢰기관 서버가 상기 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치로 갱신 키를 전송하는 단계; 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 갱신 키와 상기 사용자 비밀 키를 기초로 복호화 키를 생성하는 단계; 및 상기 비폐기 사용자 아이디를 갖는 수신 장치가 상기 복호화 키를 이용하여 상기 송신 장치로부터 전송된 암호문을 복호화하는 단계;를 포함할 수 있다.

발명의 효과

[0036] 본 발명의 실시예에 의하면, 익명 기능을 지원하는 폐기 가능한 사용자 아이디 기반의 암호화 기법을 제안하였다. 익명 기능과 사용자 폐기 기능을 동시에 지원하기 때문에 암호문이 공개되어도 어떤 아이디의 사용자에게 보내는 암호문인지 확인할 수 없다. 또한, 갱신 키에서도 폐기된 사용자에 대한 정보가 숨겨지기 때문에 그룹 내 사용자가 탈퇴를 원할 때 안전성을 지키면서 보다 효율적으로 키를 관리할 수 있다. 따라서, 실제 환경에서 사용될 때 사용자의 프라이버시를 보존하는데 큰 도움이 될 것으로 보인다. 또한, 페어링과 래티스(lattice) 두 가지를 기반으로 두 방식을 설계했기 때문에 선택 암호문 공격에 안전할 뿐만 아니라 양자 내성을 가지는 익명성이 보장되는 폐기 가능한 사용자 아이디 기반의 암호화를 구현할 수 있다. 본 발명을 이용하면 필요한 목적에 따라 적합한 설계 방식을 선택해 사용할 수 있다.

도면의 간단한 설명

[0038] 도 1은 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS(complete subset) 기법의 경로를 예시적으로 설명하는 도면이다.

도 2는 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS 기법의 커버(cover)들의 집합을 출력하는 경우를 예시적으로 설명하는 도면이다.

도 3은 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS 기법의 매치(match) 알고리즘의 동작을 예시적으로 설명하는 도면이다.

도 4는 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템의 블록도이다.

도 5는 본 발명의 실시예에 따른 암호화 시스템 내의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 예시적으로 설명하는 흐름도이다.

도 6은 본 발명의 실시예에 따른 암호화 시스템에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 예시적으로 설명하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0039] 본 발명의 이점 및 특징, 그리고 그것들을 달성하는 방법은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 그러나 본 발명은 이하에서 개시되는 실시예들에 한정되는 것이 아니라 다양한 형태로 구현될 수 있으며, 단지 본 실시예들은 본 발명의 개시가 완전하도록 하고, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 발명의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 발명의 범주는 청구항에 의해 정의될 뿐이다.

[0040] 본 발명의 실시예들을 설명함에 있어서 공지 기능 또는 구성에 대한 구체적인 설명은 본 발명의 실시예들을 설명함에 있어 실제로 필요한 경우 외에는 생략될 것이다. 그리고 후술되는 용어들은 본 발명의 실시예에서의 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.

[0041] 기존의 사용자 아이디 기반의 암호화 기술에서는 프라이버시 노출 이슈가 존재한다.

[0042] 본 발명의 실시예에서는, 암호문에서 사용자의 아이디가 노출될 수 있는 위협으로부터 보호하고, 갱신 키 발급 시에 폐기된 사용자에 대한 정보가 노출될 수 있는 프라이버스 문제를 해결할 수 있는 폐기 가능한 아이디 기반의 암호화 기술을 제안하고자 한다.

[0043] 먼저, 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화 기술을 뒷받침하는 기술들을 개략적으로 설명하면 다음과 같다.

[0044] 의사 난수 함수(pseudo-random function, PRF)

[0045] 의사 난수 함수는 $\mathcal{K}$가 키 스페이스, $\mathcal{X}$가 정의역이고 $\mathcal{Y}$가 치역일 때 효율적으로 계산 가능한 함수 $F:\mathcal{K}\times\mathcal{X}\rightarrow\mathcal{Y}$를 의미한다. $F(k,\cdot)$가 임의로 고른 $k\in\mathcal{K}$의 오라클이고 $f(\cdot)$를 임의로 고른 함수 $f:\mathcal{X}\rightarrow\mathcal{Y}$의 오라클이라고 할 때, 만약 다항 시간 안에 $F(k,\cdot)$를 공격에 성공할 확률과 $f(\cdot)$를 공격에 성공할 확률의 차이가 무시할 수 있을 정도로 작으면 의사 난수 함수가 안전하다고 정의한다.

[0046] 대칭 키 암호(symmetric key encryption)

[0047] 대칭 키 암호는 다음과 같은 세 가지의 알고리즘으로 구성된다.

[0048] 1) 설정($GenKey(1^{\lambda})$)

[0049] 본 알고리즘은 보안상수 1^{λ}를 입력받아 대칭 키 K를 출력한다.

[0050] 2) 암호화($Encrypt(M,K)$)

[0051] 본 알고리즘은 메시지 M과 대칭 키 K를 입력받아 암호문 C를 출력한다.

[0052] 3) 복호화($Decrypt(C,K)$)

[0053] 본 알고리즘은 암호문 C와 대칭 키 K를 입력받아 메시지 M이나 $\perp$을 출력한다.

[0054] 대칭 키 암호의 correctness: $GenKey(1^{\lambda})$에 의해 생성된 모든 K와 임의의 메시지 M은 다음 [수학식 1]을 만족한다.

수학식 1

[0055] $$Decrypt(Encrypt(M,K),K)=M$$

[0056] 겹선형 그룹(bilinear groups)

[0057] 겹선형 그룹 생성자 는 보안 상수 λ를 입력받아 다음 [수학식 2]의 튜플을 출력한다.

수학식 2

[0058] $$(p,\mathbb{G},\hat{\mathbb{G}},\mathbb{G}_T,e)$$

[0059] [수학식 2]에서, p는 임의의 소수, $\mathbb{G},\hat{\mathbb{G}},\mathbb{G}_T$는 소수 위수 p의 순환군이다. g와 $\hat{g}$를 $\mathbb{G},\hat{\mathbb{G}}$의 생성자라고 할 때, 쌍선형 맵 $e: \mathbb{G}\times\hat{\mathbb{G}}\rightarrow\mathbb{G}_T$은 다음 두 조건을 따른다.

[0060] 1) bilinearity : $\forall u\in\mathbb{G}, \forall \hat{v}\in\hat{\mathbb{G}}, \forall a,b\in\mathbb{Z}_p$에서 $e(u^a,\hat{v}^b)=e(u,\hat{v})^{ab}$이다.

[0061] 2) non-degeneracy : $\forall g\in\mathbb{G}, \forall \hat{g}\in\hat{\mathbb{G}}, e(g,\hat{g})$가 $\mathbb{G}_T$에서 위수 p를 가진다.

[0062] 격자 구조(lattice)

[0063] 래티스란 n차원 공간 $\mathbb{R}$에 속하는 기저(기저) 벡터 값들과 모든 정수의 선형 조합 값들이 이루는 집합이다. 기하학적으로는 n차원 공간 $\mathbb{R}$에서 벡터가 이루는 점들이 격자무늬 배열로 무한하게 배치되어 있는 것을 의미한다.

[0064] $m \geq 2n\log q$, q는 소수 조건을 만족하는 양의 정수 $n,m,\widetilde{m},q$가 있을 때, 래티스의 짧은 기저를 생성하는 다항 시간 알고리즘은 다음과 같다.

[0065] 1) $GenTrap(1^n,1^m,q)$: A가 통계적으로 균일한(uniform) 분포에 가깝고 $\|T_A\|_{GS}=O(\sqrt{n\log q})$일 때, full rank 행렬 $A\in\mathbb{Z}_q^{n\times m}$와 $\Lambda_q^{\perp}(A)$의 트랩도어(trapdoor) 기저 $T_A\in\mathbb{Z}^{m\times m}$를 임의로 선택하여 출력한다.

[0066] 2) $ExtendRandLeft(A,F,T_A,q)$: 본 알고리즘은 행렬 $A\in\mathbb{Z}_q^{n\times m}$와 $F\in\mathbb{Z}_q^{n\times\bar{m}}$, $\Lambda_q^{\perp}(A)$의 기저 T_A, 가우시안 파라미터 $\sigma\geq\|T_A\|_{GS}\cdot w(\sqrt{\log n})$를 입력받아 통계적으로 L]에 가까운 행렬 $T_{[A|F]}\in\mathbb{Z}^{(m+\bar{m})\times(m+\bar{m})}$를 임의로 선택하여 출력한다.

[0067] 3) $SampleLeft(A,F,T_A,u,\sigma)$: 본 알고리즘은 full rank 행렬 $A\in\mathbb{Z}_q^{n\times m}$와 $F\in\mathbb{Z}_q^{n\times\bar{m}}$, $\Lambda_q^{\perp}(A)$의 기저 $T_A\in\mathbb{Z}^{m\times m}$, 벡터 $u\in\mathbb{Z}_q^n$ 와 가우시안 파라미터 $\sigma\geq\|T_A\|_{GS}\cdot w(\sqrt{\log n})$를 입력으로 받아 $\Lambda_q^{u}([A|F]),\sigma$와 통계적으로 유사한 분포에서 벡터 $u\in\mathbb{Z}^{m+\bar{m}}$ 를 샘플링하여 출력한다.

[0068] 폐기 가능한 ID 기반 암호(revocable identity-based encryption, RIBE)

[0069] ID 기반 암호에서는 계약 기간이 만료되거나 개인키가 노출된 사용자에 대한 효율적인 폐기 메커니즘을 제공해야 한다. 폐기 가능한 ID 기반 암호 기법은 사용자가 등록 과정에서 PKG로부터 개인키를 발급받고, 이후 주기적으로 PKG가 폐기되지 않은 사용자들에 대한 갱신 키를 발급하여 폐기되지 않은 사용자들만 복호화 키를 생성할 수 있는 기법이다. 폐기 가능한 ID 기반 암호는 설정(setup), 키 발급(genkey), 키 갱신(updatekey), 복호화 키 생성(derivekey), 암호화(encrypt), 복호화(decrypt) 및 폐기(revoke)의 7개의 알고리즘으로 구성되어 있다.

[0070] 1) 설정($Setup(1^\lambda)\rightarrow PP,RL,MK$)

[0071] 본 알고리즘은 보안 상수 1^λ를 입력받아 마스터키 MK와 공개 파라미터 PP, 그리고 폐기된 사용자 목록 RL을 출력한다.

[0072] 2) 키 발급($Genkey(ID,MK,PP)\rightarrow SK_{ID}$)

[0073] 본 알고리즘은 사용자의 ID와 마스터키 MK, 그리고 공개 파라미터 PP를 입력받아 비밀키 SK_{ID}를 출력한다.

[0074] 3) 키 갱신($Updatekey(T,RL,MK,PP)\rightarrow UK_T$)

[0075] 본 알고리즘은 갱신 시간 T, 폐기된 사용자 목록 RL, 마스터키 MK와 공개 파라미터 PP를 입력받아 갱신 키 UK_T를 출력한다.

[0076] 4) 복호화키 생성($Derivekey(SK_{ID}, UK_T, PP) \rightarrow DK_{ID,1}$)

[0077] 본 알고리즘은 비밀 키 SK_{ID}, 갱신 키 UK_1와 공개 파라미터 PF를 입력받아 복호화 키 $DK_{ID,1}$를 출력한다.

[0078] 5) 암호화($Encrypt(ID, T, M, PP) \rightarrow CT_{ID,1}$)

[0079] 본 알고리즘은 사용자의 ID와 시간 T, 메시지 M, 그리고 공개 파라미터 PF를 입력받아 암호문 $CT_{ID,1}$를 출력한다.

[0080] 6) 복호화($Decrypt(CT_{ID,T}, DK_{ID,T}, PP) \rightarrow M$)

[0081] 본 알고리즘은 암호문 $CT_{ID,1}$와 비밀 키 $SK_{ID,1}$, 그리고 공개 파라미터 PF를 입력받아 메시지 M을 출력한다.

[0082] 7) 폐기($Revoke(ID, T, RL) \rightarrow RL$)

[0083] 본 알고리즘은 폐기될 사용자의 ID와 폐기 시간 T, 그리고 폐기된 사용자 목록 RL을 입력받아 갱신 된 폐기된 사용자 목록 RL을 출력한다.

[0084] 본 발명에서는 기존에 제안된 폐기 가능한 ID 기반 암호 기법을 기반으로 익명 기능을 추가한 프라이버시가 보존되는 폐기 가능한 ID 기반 암호를 제안하였으며, 두 가지 설계 방식(겹선형 및 래티스)을 사용하여 보다 효율적인 기법 선택이 가능하다.

[0085] 이진 트리(binary tree)

[0086] 완전 이진 트리 BT는 모든 안쪽의 노드(node)는 각각 두 개의 자식 노드를 가지고, 제일 밑단의 잎(leaf) 노드는 모두 같은 깊이를 갖는다. $N=2^r$개의 잎 노드가 존재하는 트리 BT는 총 $2N-1$개의 노드를 갖고 $1 \leqq i \leqq 2N-1$에 대하여 노드 v_i로 나타낸다. 제일 상위에 위치하는 루트(root) 노드는 깊이가 0이다. BT의 깊이는 루트 노드부터 잎 노드까지의 경로의 길이를 나타낸다.

[0087] 각각의 노드 v_i는 유니크(unique)한 식별자(identifier) $L_i \in \{0,1\}^*$를 가지며 L_i는 루트 노드부터 노드 v_i까지의 경로를 나타낸다.(두개의 자식 노드에서 1이면 왼쪽, 0이면 오른쪽으로 경로를 가진다.) 루트 노드는 식별자를 가지지 않으며, $Label(v_i)$는 노드 v_i의 식별자를 나타내고, $Depth(v_i)$는 노드 v_i의 깊이를 나타낸다.

[0088] 트리 BT의 부분 트리 T_i는 노드 v_i를 루트 노드로 가지는 트리를 의미한다. 부분집합 S_i는 T_i에 포함된 모든 잎 노드를 원소로 가진다.

[0089] v_j가 v_i의 후손인 관계를 가진 두 개의 노드 v_i, v_j에 대해 $T_{i,j}$는 $T_i - T_j$인 트리를 의미하고, $S_{i,j}$는 $T_{i,j}$에 포함된 모든 잎 노드를 원소로 가진다.

[0091] 이하, 첨부된 도면을 참조하여 본 발명의 실시예에 대해 상세히 설명하기로 한다.

[0092] 도 1은 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS(complete subset) 기법의 경로를 예시적으로 설명하는 도면이고, 도 2는 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS 기법의 커버(cover)들의 집합을 출력하는 경우를 예시적으로 설명하는 도면이며, 도 3은 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화에 적용되는 CS 기법의 매치(match) 알고리즘의 동작을 예시적으로 설명하는 도면이다.

[0093] CS 방식(complete subset method)

[0094] 도 1에 도시한 바와 같이, CS 방식은 완전 이진트리를 사용하고, 각 사용자는 트리의 잎 노드에 위치한다.

[0095] 1) 설정($Setup(N) \rightarrow BT$)

[0096] 본 알고리즘은 깊이 n인 완전 이진 트리 BT를 출력한다. 편의를 위해 $N=2^n$로 정의한다. 각 사용자는 잎 노드에 등록된다. 집합 S는 서브루트(subroot) $v_i \in BT$을 가지는 모든 서브트리(subtree) T_i의 잎들의 집합 $\{S_i\}$를 의미한다.

[0097] 2) 배정($Assign(BT,v) \rightarrow PV$)

[0098] 본 알고리즘은 이진 트리 BT와 사용자의 ID v를 입력받아 루트 노드부터 사용자가 등록된 잎 노드까지의 경로의 부분 집합들인 PV를 출력한다. $(v_{k_0}, v_{k_1}, \ldots, v_{k_n})$를 루트 노드부터 잎 노드까지의 경로라 하면 v_{k_0}는 루트 노드가 되고, $v_{k_n} = v$가 된다. 초기의 경로 집합 PV는 비어있으며, $j \in \{k_0, \ldots, k_n\}$인 j에 대해 부분집합 S_j을 PV에 추가한다. 이후 경로의 집합 $PV = \{S_j\}$을 출력한다.

[0099] 3) 커버($Cover(BT,R) \rightarrow CV$)

[0100] 도 2에 도시한 바와 같이, 본 알고리즘은 이진트리 BT와 폐기될 사용자 집합 R을 입력받아, 폐기되지 않은 잎 노드들을 효율적으로 덮을 수(cover) 있는 커버들의 집합 CV을 출력한다.

[0101] 우선, Steiner tree ST_R을 계산하고, BT의 서브트리를 $T_{k_1}, \ldots, T_{k_m}$로 정의한다.

[0102] 이때, 모든 서브트리의 루트 $v_{k_1}, \ldots, v_{k_m}$는 노드에 인접한 출력 차수(outdegree) 1을 제외하고는 ST_R에 존재하지 않는다. 커버들의 집합 CV는 초기에 비어있으며, $i \in \{k_1, \ldots, k_m\}$인 i에 대해 S_i을 CV에 추가한다. 이후 커버들 집합 $CV = S_i$를 출력한다.

[0103] 4) 매치($Match(CV,PV) \rightarrow (S_i, S_j)$)

[0104] 도 3에 도시한 바와 같이, 본 알고리즘은 $S_k \in CV$와 $S_k \in PV$인 공통 부분집합 S_k를 찾는 알고리즘이다. 만약, 두 집합에 공통 부분집합이 존재하면 (S_k, S_k)를 출력하고, 그렇지 않을 경우 $\perp$을 출력한다.

[0105] 본 발명의 실시예에 따른 CS 방식은 사용자 ID의 폐기에 의해 폐기 리스트를 업데이트 한 이후, 업데이트키 생성 시 폐기 되지 않은 사용자에게만 업데이트키 UK_1를 생성하기 위해 폐기 리스트로부터 해당 사용자의 폐기 여부를 조회할 때 사용된다. CS 방식을 사용하지 않고 폐기된 사용자들을 관리하려면 폐기가 일어날 때마다 모든 폐기 리스트에 존재하는 사용자의 정보를 업데이트해야 한다. 하지만 CS 방식을 사용하면 모든 사용자의 정보를 업데이트하지 않아도 폐기하는 사용자의 부분 트리로 전체 트리를 덮어서(cover) 전체 폐기 리스트를 업데이트 할 수 있다. 따라서 CS 방식을 사용하면 보다 효율적으로 사용자 폐기 리스트를 관리할 수 있다.

[0106] 본 발명은 기존에 정의된 사용자 폐기를 지원하는 아이디 기반 암호(revocable identity-based encryption)에 익명(anonymous) 기능을 지원하도록 확장하였다. 아이디 기반 암호에서의 익명이란 암호문과 그 암호문을 만들 때 사용된 아이디의 관련성을 없애는 것을 의미한다. 따라서 암호문이 공개되어도 그 암호문을 수신하는 사용자에 대한 정보를 알 수 없어서 사용자의 프라이버시를 보존할 수 있다.

[0107] 본 발명은 겹선형 그룹 및 래티스를 기반으로 각각 설계한 두 가지 기법에 대한 서술이다. 두 개의 기법은 도 4와 같은 시스템 모델을 따른다.

[0108] 설계 기반이 되는 수학적 가정에 따라 겹선형 그룹, 래티스 기반 그룹으로 나뉜다. 전자는 선택 암호문 공격에 안전한 방식을 기반으로 하며 후자는 양자 컴퓨터에도 안전한 암호 알고리즘이다.

[0109] 두 기법은 전체적인 기법의 동작 과정과 정확성은 동일하지만, 기반이 되는 난제가 다르기 때문에 기법의 안전성과 효율성에서 차이가 있다. 따라서 사용 환경에 맞추어 적합한 방식을 선택해 효율적으로 사용할 수 있다.

[0110] 도 4는 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템의 블록도이다.

[0111] 도 4에 도시한 바와 같이, 폐기 가능한 아이디 기반의 암호화 방법을 수행하는 암호화 시스템은, 신뢰기관 서버

(100), 수신 장치(200) 및 송신 장치(300)를 포함할 수 있다.

[0112] 신뢰기관 서버(100)는 공개 정보를 생성하고, 공개 정보 중 공개 파라미터(PF)를 암호화 시스템 내의 수신 장치(200) 및 송신 장치(300)로 전송하며, 공개 정보 중 마스터 비밀 키(MK)를 이용하여 사용자 비밀 키(SK_{ID})를 생성하고, 사용자 비밀 키(SK_{ID})를 수신 장치(200)로 전송하며, 공개 정보 중 폐기 리스트(RL)가 갱신되는지 여부를 판단하고, 폐기 리스트(RL)가 갱신되면 수신 장치(200) 중 비폐기 사용자 아이디(identification, ID)를 갖는 수신 장치(200)로 갱신 키(UK_T)를 전송할 수 있다.

[0113] 여기서, 공개 정보는 공개 파라미터(PF), 마스터 비밀 키(MK), 폐기 리스트(RL) 및 상태 정보(ST) 중 적어도 하나를 포함할 수 있다.

[0114] 이때, 신뢰기관 서버(100)는 폐기 리스트(RL)가 갱신되면 갱신된 폐기 리스트(RL)를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인할 수 있다. 또한, 갱신 키(UK_T)는 폐기 리스트(RL), 갱신 시간(T), 마스터 비밀 키(MK), 상태 정보(ST) 및 공개 파라미터(PF)를 기초로 생성될 수 있으며, 신뢰기관 서버(100)는 이러한 갱신 키(UK_T)를 비폐기 사용자 아이디를 갖는 수신 장치(200)에게만 전송할 수 있다.

[0115] 또한, 사용자 비밀 키(SK_{ID})는 사용자 아이디 및 상태 정보(ST)를 기초로 생성될 수 있으며, 폐기 리스트(RL)는 사용자 아이디가 폐기될 때 갱신될 수 있다.

[0116] 비폐기 사용자 아이디를 갖는 수신 장치(200)는 갱신 키(UK_T)를 수신하고, 수신되는 갱신 키(UK_T)와 사용자 비밀 키(SK_{ID})를 기초로 복호화 키 ($DK_{ID,T}$)를 생성할 수 있다.

[0117] 또한, 비폐기 사용자 아이디를 갖는 수신 장치(200)는 송신 장치(300)로부터 암호문($CT_{ID,T}$)을 수신할 수 있으며, 비폐기 사용자 아이디를 갖는 수신 장치(200)는 복호화 키($DK_{ID,T}$)를 이용하여 암호문($CT_{ID,T}$)을 복호화할 수 있다.

[0118] 이러한 도 4를 기초로, 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화 시스템의 기능들을 보다 상세히 설명하면 다음과 같다.

[0119] 1) 설정($Setup(1^{\lambda}, N) \rightarrow PP, RL, MK, ST$)

[0120] 본 알고리즘은 신뢰기관 서버(100)에 의해 동작하며, 보안 상수 1^{λ}와 최대 사용자의 수 N을 입력받아 마스터키(MK)와 공개 파라미터(PF), 비어있는 폐기된 사용자 목록(RL), 상태 정보(ST)를 출력한다.

[0121] 2) 키 발급($Genkey(ID, MK, ST, PP) \rightarrow SK_{ID}$)

[0122] 본 알고리즘은 마스터 비밀 키(MK)를 보유한 신뢰기관 서버(100)에 의해 동작하며, 사용자의 아이디와 마스터 비밀 키(MK), 상태 정보(ST), 그리고 공개 파라미터(PF)를 입력받아 사용자 비밀 키(SK_{ID})를 출력한다.

[0123] 3) 키 갱신($Updatekey(T, RL, MK, ST, PP) \rightarrow UK_T$)

[0124] 본 알고리즘은 마스터 비밀 키(MK)를 보유한 신뢰기관 서버(100)에 의해 동작하며, 갱신 시간(T), 폐기된 사용자 목록(RL), 마스터 비밀 키(MK), 상태 정보(ST)와 공개 파라미터(PF)를 입력받아 갱신 키(UK_T)를 출력한다.

[0125] 4) 복호화 키 생성($Derivekey(ID, T, SK_{ID}, UK_T, PP) \rightarrow DK_{ID,T}$)

[0126] 본 알고리즘은 아이디 기반의 사용자 비밀 키 (SK_{ID})를 보유한 시스템 사용자들에 의해 동작하며, 사용자 비밀 키(SK_{ID}), 갱신 키(UK_T) 및 공개 파라미터(PF)를 입력받아 복호화 키($DK_{ID,T}$)를 출력한다.

[0127] 5) 암호화($Encrypt(ID, T, M, PP) \rightarrow CT_{ID,T}$)

[0128] 본 알고리즘은 누구나 진행할 수 있으며, 사용자의 아이디와 시간(T), 메시지(M), 그리고 공개 파라미터(PP)를 입력받아 암호문($CT_{ID,T}$)을 출력한다.

[0129] 6) 복호화($Decrypt(CT_{ID,T}, DK_{ID,T}, PP) \rightarrow M$)

[0130] 본 알고리즘은 복호화 키($DK_{ID,T}$)를 보유한 폐기되지 않은 사용자에 의해 동작하며, 암호문($CT_{ID,T}$)와 복호화 키($DK_{ID,T}$), 그리고 공개 파라미터(PP)를 입력받아 메시지 M을 출력한다.

[0131] 이하, 상술한 구성과 함께, 본 발명의 실시예에 따른 폐기 가능한 아이디 기반의 암호화 방법을, 첨부하는 도 5 및 도 6의 흐름도를 참조하여 상세히 설명하기로 한다.

[0132] 도 5는 본 발명의 실시예에 따른 암호화 시스템 내의 신뢰기관 서버에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 예시적으로 설명하는 흐름도이고, 도 6은 본 발명의 실시예에 따른 암호화 시스템에서 수행되는 폐기 가능한 아이디 기반의 암호화 방법을 예시적으로 설명하는 흐름도이다.

[0133] 먼저, 도 5에 도시한 바와 같이, 암호화 시스템의 신뢰기관 서버(100)가 공개 정보를 생성할 수 있다(S100). 여기서, 공개 정보는 공개 파라미터(PP), 마스터 비밀 키(MK), 폐기 리스트(RL) 및 상태 정보(ST) 중 적어도 하나를 포함할 수 있다.

[0134] 이후, 신뢰기관 서버(100)는 공개 정보 중 공개 파라미터(PP)를 수신 장치(200) 및 송신 장치(300)로 전송할 수 있다(S102).

[0135] 공개 파라미터(PP)를 수신 장치(200) 및 송신 장치(300)로 전송한 후 신뢰기관 서버(100)는 공개 정보 중 마스터 비밀 키(MK)를 이용하여 사용자 비밀 키(SK_{ID})를 생성할 수 있다(S104). 여기서, 사용자 비밀 키(SK_{ID})는 사용자 아이디 및 상태 정보(ST)를 기초로 생성될 수 있다.

[0136] 사용자 비밀 키(SK_{ID})가 생성되면, 신뢰기관 서버(100)는 사용자 비밀 키(SK_{ID})를 수신 장치(200)로 전송할 수 있다(S106).

[0137] 이후, 신뢰기관 서버(100)는 공개 정보 중 폐기 리스트(RL)가 갱신되는지 여부를 판단할 수 있다(S108).

[0138] 폐기 리스트(RL)가 갱신된 것으로 판단되면, 신뢰기관 서버(100)는 갱신된 폐기 리스트(RL)를 기초로 폐기 사용자 아이디와 비폐기 사용자 아이디를 확인하고, 폐기 리스트(RL), 갱신 시간(T), 마스터 비밀 키(MK), 상태 정보(ST) 및 공개 파라미터(PP)를 기초로 갱신 키(UK_T)를 생성할 수 있다(S110).

[0139] 신뢰기관 서버(100)는 수신 장치 중 비폐기 사용자 아이디를 갖는 수신 장치(200)로 갱신 키(UK_T)를 전송할 수 있다(S112).

[0140] 즉, 본 발명의 실시예서는 폐기 리스트(RL), 갱신 시간(T), 마스터 비밀 키(MK), 상태 정보(ST) 및 공개 파라미터(PP)를 기초로 갱신 키(UK_T)를 생성하되, 생성된 갱신 키(UK_T)를 비폐기 사용자 아이디를 갖는 수신 장치(200)에게만 전송하는 것을 특징으로 한다.

[0141] 이후, 도 6의 단계(S202)에서는 비폐기 사용자 아이디를 갖는 수신 장치(200)가 갱신 키(UK_T)를 수신하고, 수신 장치(200)는 수신된 갱신 키(UK_T)와 사용자 비밀 키(SK_{ID})를 기초로 복호화 키($DK_{ID,T}$)를 생성할 수 있다(S204).

[0142] 이후, 비폐기 사용자 아이디를 갖는 수신 장치(200)는 송신 장치(300)로부터 암호문($CT_{ID,T}$)이 수신되는지를 판단하고(S206), 암호문($CT_{ID,T}$)이 송신 장치(300)로부터 수신되면, 수신 장치(200)는 복호화 키($DK_{ID,T}$)를 이

용하여 송신 장치(300)로부터 전송된 암호문($CT_{ID,t}$)을 복호화할 수 있다(S208).

[0144] 한편, 겹선형 그룹을 기반으로 설계된 기법은 다음과 같다.

[0145] $Setup(1^{\lambda}, N)$ (λ 은 보안 파라미터이며 N은 사용자의 최대 수, $n=\lambda$라고 정의한다.)

[0146] 1) 위수 p의 겹선형 그룹 $\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T$와 $\mathbb{G}, \hat{\mathbb{G}}$의 생성자 $g, \hat{g}$를 생성한다.

[0147] 2) 랜덤하게 선택한 지수 $\alpha, u'_1, h'_1, u'_2, h'_2, w' \in$ 를 선택하고 $u_1 = g^{u'_1}, h_1 = g^{h'_1}, u_2 = g^{u'_2}, h_2 = g^{h'_2}, w = g^{w'}, \hat{u}_1 = \hat{g}^{u'_1}, \hat{h}_1 = \hat{g}^{h'_1}, \hat{u}_2 = \hat{g}^{u'_2}, \hat{h}_2 = \hat{g}^{h'_2}, \hat{w} = \hat{g}^{w'}$ 를 생성한다. $F_1(ID) = (u_1^{ID}, h_1), \hat{F}_1(ID) = (\hat{u}_1^{ID}, \hat{h}_1), F_2(T) = (u_2^T, h_2), \hat{F}_2(T) = (\hat{u}_2^T, \hat{h}_2)$로 정의한다.

[0148] 3) PRF 키 z_1, z_2를 생성하고 $CS.Setup(N)$을 실행하여 이진트리 BT를 얻는다.

[0149] 4) 마스터키를 $MK = (\alpha, z_1, z_2)$로 설정하고, 비어있는 폐기 리스트(revocation list) RL, 상태(state) $ST = (BT)$, 공개 파라미터를 다음과 같이 설정하여 출력한다.

[0150] $PP = ((p, \mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T, e), g, u_1, h_1, u_2, h_2, w, \hat{g}, \hat{u}_1, \hat{h}_1, \hat{u}_2, \hat{h}_2, \hat{w}, \Omega = e(g, \hat{g})^{\alpha})$

[0151] $GenKey(ID, MK, ST, PP)$

[0152] 1) $CS.Assign(BT, v)$를 실행시켜 경로 집합 $PV = \{S_0, \cdots, S_n\}$을 포함하여 ID를 랜덤 리프 노드 $v \in BT$에 할당한다.

[0153] 2) $L_i = Label(S_i)$를 라벨 스트링(Label string)이라고 두고, $\kappa_i \in \{0,1\}^{\lambda}$, $\gamma_i, w_i \in \mathbb{Z}_p$일 때 $(\gamma_i, w_i) = PRF_1(z_1, L_i)$, $\kappa_i = PRF_2(z_2, L_i)$를 계산한다. 랜덤 지수 $r_1, r_2 \in \mathbb{Z}_p$를 선택하고 노드 비밀 키 $NSK_j = (K_{j,0} = \hat{g}^{\gamma_i} \hat{F}_1(ID)^{r_1} \hat{w}^{r_2}, K_{j,1} = \hat{g}^{-r_1}, K_{j,2} = \hat{g}^{-r_2}, K_{j,3} = (w_j, \kappa_j))$를 생성한다. $j=0$부터 n까지 위 과정을 수행한다.

[0154] 3) 비밀 키 $SK_{ID} = (NSK_0, \ldots, NSK_n)$를 출력한다.

[0155] $UpdateKey(T, RL, MK, ST, PP)$

[0156] 1) RL로부터 시간 T에서 폐기된 노드의 집합 R을 구하고 $CS.Cover(BT, R)$을 실행시켜 커버들의 집합 $CV = \{S_1, \cdots, S_l\}$을 얻는다. $r = |R|, l = |CV|, l_m = [rlog(N/r)]$으로 정의한다.

[0157] 2) 랜덤 지수 $s \in \mathbb{Z}_p$를 선택하고 $V = \hat{g}^s$를 정의한다.

[0158] 3) $L_i = Label(S_i)$를 라벨 스트링이라고 두고, $\gamma_i, w_i \in \mathbb{Z}_p$일 때 $(\gamma_i, w_i) = PRF_1(z_1, L_i)$, $\kappa_i = PRF_2(z_2, L_i)$ 를 계산한다. 랜덤 지수 $r \in \mathbb{Z}_p$를 선택하고 원소 $U_{i,0} = \hat{g}^{\alpha - \gamma_i} \hat{F}_2(T)^r, U_{i,1} = \hat{g}^{-r}$를 정의한다. 암호문 $CU_i = SKE\text{-}Encrypt(\kappa_i, U_{i,0} \| U_{i,1})$를 생성하고 힌트 값을 $Y_i = V^{w_i}$로 설정한 후 노드 업데이트 키 $NUK_i = (CU_i, Y_i)$를 생성한다. $1 \le i \le l$까지 위 과정을 수행한다.

[0159] 4) $l+1 \le i \le l_m$일 때, 랜덤 원소 $U_{i,0}, U_{i,1} \in \hat{\mathbb{G}}$를 고르고 랜덤 키 κ_i를 사용하여 암호문 $CU_i = SKE\text{-}Encrypt(\kappa_i, U_{i,0} \| U_{i,1})$ 를 얻는다. 랜덤 $Y_i \in \hat{\mathbb{G}}$를 선택한 후 노드 업데이트 키 $NUK_i = (CU_i, Y_i)$를 생성한다.

[0160] 5) 랜덤 순열 $\pi : \{1, \cdots, l_m\} \rightarrow \{1, \cdots, l_m\}$을 선택하고 업데이트 키 $UK_T = (V, NUK_{\pi(1)}, \cdots, NUK_{\pi(l_m)})$를 출력한다.

[0161] $DeriveKey(ID, T, SK_{ID}, UK_T, PP)$

[0162] 1) 모든 $j \in \{0, \cdots, n\}$에 대해 NSK_j로부터 w_j를 검색하고 리스트를 $(Y'_0 = V^{w_0}, \cdots, Y'_n = V^{w_n})$로 설정한다. 또한 모든 $i \in \{1, \cdots, l_m\}$에 대해 NUK_i로부터 Y_i를 검색하고 리스트를 $(Y_1, \cdots, Y_{l_m})$로 설정한다.

[0163] 2) 리스트 안에서 $Y'_i = Y_i$인 두 인덱스 $i \in \{1, \cdots, l_m\}$, $j \in \{0, \cdots, n\}$를 찾고, 그에 대응되는 $NUK_i = (CU_i, Y_i)$와 $NSK_j = (K_{j,0}, K_{j,1}, K_{j,2}, K_{j,3} = (w_j, \kappa_j))$를 검색한다. 이후 $U_{i,0} \| U_{i,1} = SKE.Decrypt(\kappa_j, CU_i)$를 계산한다.

[0164] 3) 랜덤 지수 $r'_1, r'_2, r'_3 \in \mathbb{Z}_k$를 선택하고 복호화 키를 다음과 같이 계산한다.

[0165] $D_0 = K_{j,0} U_{i,0} \cdot F_1(ID)^{r'_1} F_2(T)^{r'_2} w^{r'_3}, D_1 = K_{j,1} g^{-r'_1}, D_2 = U_{i,1} g^{-r'_2}, D_3 = K_{j,2} g^{-r'_3}$

[0166] 4) 복호화 키 $DK_{ID,T} = (D_0, D_1, D_2, D_3)$를 출력한다.

[0167] $Encrypt(ID, T, M, PP)$

[0168] 1) 랜덤 지수 $t \in \mathbb{Z}_k$와 다음과 같은 암호문 $CT_{ID,T} = (C, C_0, C_1, C_2, C_3)$를 출력한다.

[0169] $CT_{ID,T} = (C = \Omega^t M, C_0 = g^t, C_1 = F_1(ID)^t, C_2 = F_2(T)^t, C_3 = w^t)$

[0170] $Decrypt(CT_{ID,T}, DK_{ID,T}, PP)$

[0171] 1) 암호문과 복호화 키로 $M = C \cdot \left(\prod_{i=0}^{3} e(C_i, D_i) \right)^{-1}$ 를 계산해 메시지를 복호화한다.

[0172] 래티스를 기반으로 하는 설계 방법은 다음과 같다.

[0173] $Setup(1^\lambda, N)$ (λ 은 보안 파라미터이며 N은 사용자의 최대 수, $n = \lambda$라고 정의한다.)

[0174] 1) $GenTrap(1^n, 1^m, q)$를 동작시켜 (A_1, T_{A_1})와 (A_2, T_{A_2})를 얻는다. 랜덤 행렬 $B \leftarrow \mathbb{Z}_q^{n \times m}$와 벡터 $\mathbf{u} \in \mathbb{Z}_q^n$를 무작위로 샘플링한다.

[0175] 2) PRF 키 z_1, z_2를 생성하고, $CS.Setup(N)$을 실행하여 이진트리 BT를 얻는다.

[0176] 3) 마스터키를 $MK = (T_{A_1}, T_{A_2}, z_1, z_2)$로 설정하고, 폐기 리스트 $RL = \varnothing$, $ST = (BT)$, 공개 파라미터를 $PP = (A_1, A_2, B, \mathbf{u})$로 설정하여 출력한다.

[0177] $GenKey(ID, MK, ST, PP)$

[0178] 1) $Assign(BT, v)$를 실행시켜 경로 집합 $PV = \{S_0, \cdots, S_n\}$을 포함하여 ID를 랜덤 리프 노드 $v \in BT$에 할당한다.

[0179] 2) $L_i = Label(S_i)$를 라벨 스트링이라고 두고, $\mathbf{u}_0, \mathbf{w} \in \mathbb{Z}_q^n$일 때 $(\mathbf{u}_0, \mathbf{w}) = PRF_1(z_1, L_i), \kappa_i = PRF_2(z_2, L_i)$ 를 계산한다. $SampleLeft(A_1, B(ID), T_{A_1}, \mathbf{u}_0, \sigma)$를 실행시켜 $[A_1 | B(ID)]\mathbf{e}_0 = \mathbf{u}_0$ 를 만족하는 벡터 $\mathbf{e}_0$를 샘플링하고, 노드 비밀 키 $NSK_j = (\mathbf{e}_0, \mathbf{w}, \kappa_i)$를 생성한다. $j = 0$부터 n까지 위 과정을 수행한다.

[0180] 3) $ExtendRandLeft(A_2, B(ID), T_{A_2}, \sigma)$ 를 실행시켜 확장된 기저인 $T_{[A_2 | B(ID)]}$를 구한다.

[0181] 4) 비밀 키 $SK_{ID}=(NSK_0,\ldots,NSK_n,T_{[A_2|E(ID)|}$를 출력한다.

[0182] $UpdateKey(T,RL,MK,ST,PP)$

[0183] 1) RL로부터 시간 T에서 폐기된 노드의 집합 R을 구하고 $Cover(BT,R)$을 실행시켜 커버들의 집합 $CV=\{S_1,\cdots,S_l\}$을 얻는다. $|r=|R|,l=|CV|,l_m=\lceil r\log(N/r)\rceil$으로 정의한다.

[0184] 2) uniform하게 랜덤한 행렬 $V\in\mathbb{Z}_q^{n\times r}$을 선택한다.

[0185] 3) $L_i=Label(S_i)$를 라벨 스트링이라고 두고, $\mathbf{u}_0,\mathbf{w}_0\in\mathbb{Z}_q^r$일 때 $(\mathbf{u}_1,\mathbf{w}_1)=PRF_1(z_1,L_i),\kappa_i=PRF_2(z_2,L_i)$를 계산한다. $SampleLeft(A_1,F(T),T_A,\mathbf{u}-\mathbf{u}_1,\sigma)$를 실행시켜 $[A_1|F(T)]\mathbf{f}_1=\mathbf{u}_1$를 만족하는 벡터 $\mathbf{f}_1$를 샘플링하고, 암호문 $CU_i=SKE.Encrypt(\kappa_i,\mathbf{f}_1)$를 얻는다. 힌트 벡터를 $\mathbf{y}_i=\lfloor\mathbf{V}^\top\mathbf{w}_i\rceil_p$로 설정하고 노드 업데이트 키 $NUK_i=(CU_i,\mathbf{y}_i)$를 생성한다. $1\le i\le l$까지 위 과정을 수행한다.

[0186] 4) $l+1\le i\le l_m$일 때, 랜덤 $\mathbf{f}_i$를 고르고 랜덤 키 κ_i를 사용하여 암호문 $CU_i=SKE.Encrypt(\kappa_i,\mathbf{f}_i)$를 얻고 노드 업데이트 키 $NUK_i=(CU_i,\mathbf{y}_i)$를 생성한다.

[0187] 5) 랜덤 순열 $\pi:\{1,\cdots,l_m\}\to\{1,\cdots,l_m\}$을 선택하고 업데이트 키 $UK_T=(V,NUK_{\pi(1)},\cdots,NUK_{\pi(l_m)})$를 출력한다.

[0188] $DeriveKey(ID,T,SK_{ID},UK_T,PP)$

[0189] 1) 모든 $j\in\{0,\ldots,n\}$에 대해 NSK_j로부터 $\mathbf{w}_j$를 검색하고 리스트를 $(\mathbf{y}'_0=\lfloor\mathbf{V}^\top\mathbf{w}_0\rceil_p,\ldots,\mathbf{y}'_n=\lfloor\mathbf{V}^\top\mathbf{w}_n\rceil_p)$로 설정한다. 또한 모든 $i\in\{1,\cdots,l_m\}$에 대해 NUK_i로부터 $\mathbf{y}_i$를 검색하고 리스트를 $(\mathbf{y}_1,\cdots,\mathbf{y}_{l_m})$로 설정한다.

[0190] 2) 리스트 안에서 $\mathbf{y}'_j=\mathbf{y}_i$인 두 인덱스 $i\in\{1,\cdots,l_m\}$, $j\in\{0,\ldots,n\}$를 찾고, 그에 대응되는 $NUK_i=(CU_i,\ldots)$와 $NSK_j=(\mathbf{e}_j,\mathbf{w}_j,\kappa_j)$를 검색한다. 이후 $\mathbf{f}_i=SKE.Encrypt(\kappa_i,CU_i)$를 계산한다.

[0191] 3) $\mathbf{e}_{j,L},\mathbf{e}_{j,R},\mathbf{f}_{j,L},\mathbf{f}_{j,R}\in\mathbb{Z}^m$일 때 $\mathbf{e}_j=[\mathbf{e}_{j,L}|\mathbf{e}_{j,R}]$와 $\mathbf{f}_j=[\mathbf{f}_{j,L}|\mathbf{f}_{j,R}]$로 설정하고 $[A_1|E(ID)|F(T)]\mathbf{d}_j=\mathbf{u}_1$를 만족하는 $\mathbf{d}_j=[\mathbf{e}_{j,L}+\mathbf{f}_{j,L}|\mathbf{e}_{j,R}|\mathbf{f}_{j,R}]$를 계산한다.

[0192] 4) $SampleLeft([A_2|E(ID)],F(T),T_{[A_2|E(ID)|},\mathbf{u},\sigma)$를 실행시켜 $[A_2|E(ID)|F(T)]\mathbf{d}_0=\mathbf{u}$를 만족하는 $\mathbf{d}_0$를 샘플링한다.

[0193] 5) 복호화 키 $DK_{ID,T}=(\mathbf{d}_0,\mathbf{d}_1)$를 출력한다.

[0194] $Encrypt(ID,T,M,PP)$

[0195] 1) 균일한 분포를 가지는 랜덤 벡터 $\mathbf{s}_1,\mathbf{s}_2\in\mathbb{Z}_q^n$와 $x\leftarrow D_{\mathbb{Z},\alpha q}$, $\mathbf{x}_1,\mathbf{x}_2\leftarrow D_{\mathbb{Z}^m,\alpha' q}$를 샘플링하고 다음과 같이 정의한다.

[0196] $c_0=\mathbf{u}_0^\top(\mathbf{s}_1+\mathbf{s}_2)+x+M\left\lfloor\frac{q}{2}\right\rfloor$,

[0197] $\mathbf{c}_1=[A_1|E(ID)|F(T)]^\top\mathbf{s}_1+\mathbf{x}_1$,

[0198] $c_i = [A_2 | B(ID) | F(T)]^\top s_i + x_i$

[0199] 2) 암호문 $CT_{ID,T} = (c_0, c_1, c_2)$를 출력한다.

[0200] $Decrypt(CT_{ID,T}, DK_{ID,T}, PP)$

[0201] 1) 암호문과 복호화 키로 $c' = c_0 - c_1^\top d_1 - c_2^\top d_2$ 를 계산한다.

[0202] 2) 만약 $|c' - \lfloor \frac{q}{2} \rfloor| < \lfloor \frac{q}{4} \rfloor$ 이면 1, 그렇지 않으면 0을 출력한다.

[0203] $Revoke(ID, T, RL)$

[0204] 1) ID가 BT에 존재하지 않으면 $\perp$을 출력한다. 아닌 경우에는 RL에 (ID, T)를 추가한 뒤 RL을 출력한다.

[0206] 이상 설명한 바와 같은 본 발명의 실시예에 의하면, 폐기되지 않은 사용자는 갱신 키(UK_T)와 사용자 비밀 키(SK_{ID})를 사용하여 복호화 키($DK_{ID,T}$)를 생성할 수 있고, 이를 통해 암호화 주체가 생성한 암호문($CT_{ID,T}$)을 복호화 할 수 있다. 하지만, 폐기된 사용자의 경우에는 갱신 키(UK_T)를 발급받지 못하기 때문에 복호화 키를 생성할 수 없고, 따라서 암호문($CT_{ID,T}$)의 복호화 또한 불가능하도록 구현한 것이다.

[0207] 한편, 첨부된 블록도의 각 블록과 흐름도의 각 단계의 조합들은 컴퓨터 프로그램 인스트럭션들에 의해 수행될 수도 있다. 이들 컴퓨터 프로그램 인스트럭션들은 범용 컴퓨터, 특수용 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비의 프로세서에 탑재될 수 있으므로, 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비의 프로세서를 통해 수행되는 그 인스트럭션들이 블록도의 각 블록에서 설명된 기능들을 수행하는 수단을 생성하게 된다.

[0208] 이들 컴퓨터 프로그램 인스트럭션들은 특정 방식으로 기능을 구현하기 위해 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비를 지향할 수 있는 컴퓨터 이용 가능 또는 컴퓨터 판독 가능 기록매체(또는 메모리) 등에 저장되는 것도 가능하므로, 그 컴퓨터 이용 가능 또는 컴퓨터 판독 가능 기록매체(또는 메모리)에 저장된 인스트럭션들은 블록도의 각 블록에서 설명된 기능을 수행하는 인스트럭션 수단을 내포하는 제조 품목을 생산하는 것도 가능하다.

[0209] 그리고, 컴퓨터 프로그램 인스트럭션들은 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비 상에 탑재되는 것도 가능하므로, 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비 상에서 일련의 동작 단계들이 수행되어 컴퓨터로 실행되는 프로세스를 생성해서 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비를 수행하는 인스트럭션들은 블록도의 각 블록에서 설명된 기능들을 실행하기 위한 단계들을 제공하는 것도 가능하다.

[0210] 또한, 각 블록은 특정된 논리적 기능(들)을 실행하기 위한 적어도 하나 이상의 실행 가능한 인스트럭션들을 포함하는 모듈, 세그먼트 또는 코드의 일부를 나타낼 수 있다. 또, 몇 가지 대체 실시 예들에서는 블록들에서 언급된 기능들이 순서를 벗어나서 발생하는 것도 가능함을 주목해야 한다. 예컨대, 잇달아 도시되어 있는 두 개의 블록들은 사실 실질적으로 동시에 수행되는 것도 가능하고 또는 그 블록들이 때때로 해당하는 기능에 따라 역순으로 수행되는 것도 가능하다.

부호의 설명

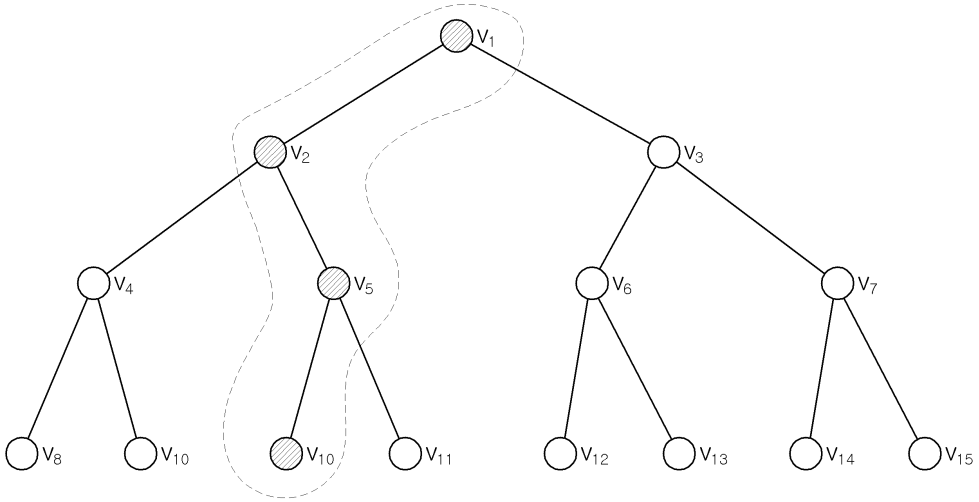
[0212] 100: 신뢰기관 서버

200: 수신 장치

300: 송신 장치

도면

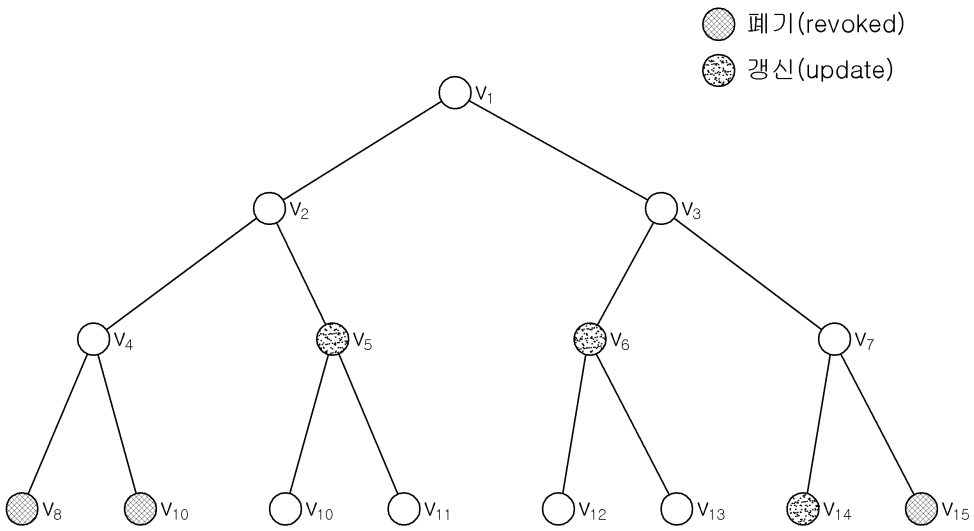
도면1



ID = 010

$PV_{ID} = \{V_1, V_2, V_5, V_{10}\}$

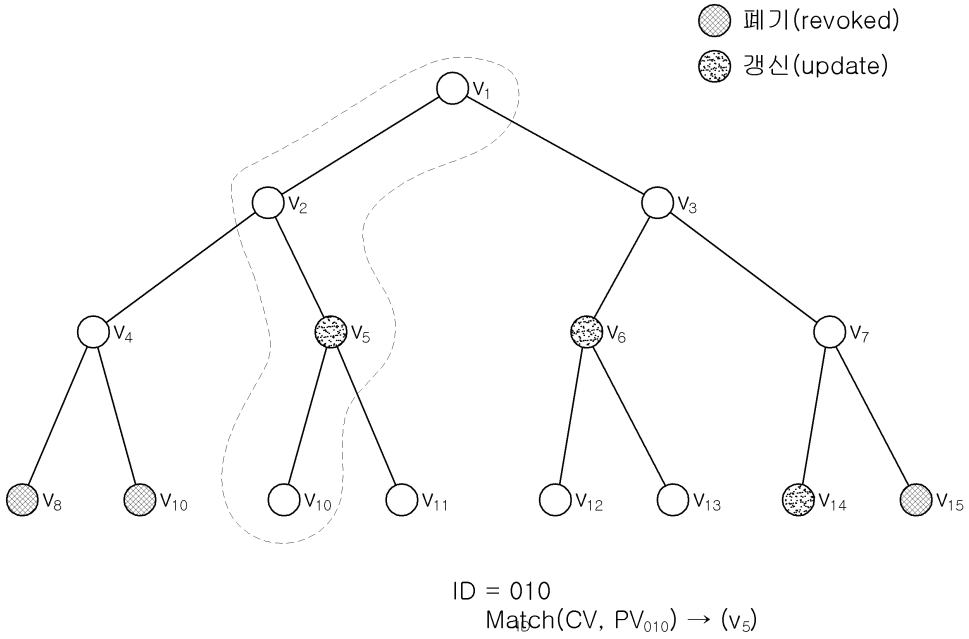
도면2



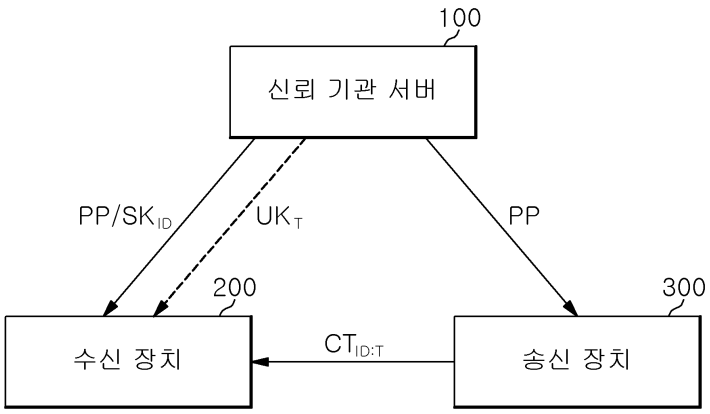
$R = \{V_8, V_9, V_{15}\}$

$CV = \{V_5, V_6, V_{14}\}$

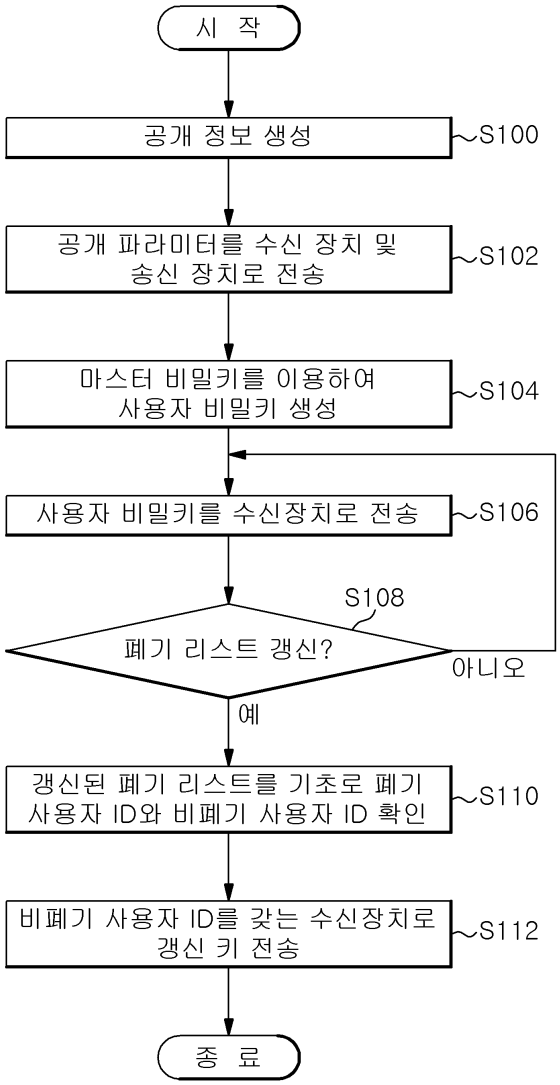
도면3


폐기(revoked)
갱신(update)
v1
v2
v3
v4
v5
v6
v7
v8
v10
v10
v11
v12
v13
v14
v15
ID = 010
Match(CV, PV010) → (v5)

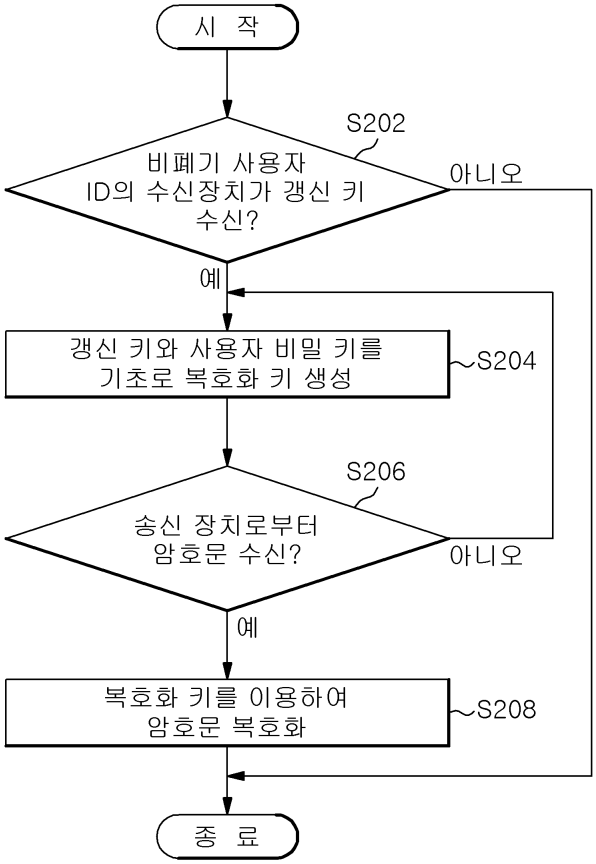
도면4


100
신뢰 기관 서버
PP/SKID
UKT
PP
200
300
수신 장치
CTID:T
송신 장치

도면5


시 작
공개 정보 생성
S100
공개 파라미터를 수신 장치 및 송신 장치로 전송
S102
마스터 비밀키를 이용하여 사용자 비밀키 생성
S104
사용자 비밀키를 수신장치로 전송
S106
S108
폐기 리스트 갱신?
아니오
예
갱신된 폐기 리스트를 기초로 폐기 사용자 ID와 비폐기 사용자 ID 확인
S110
비폐기 사용자 ID를 갖는 수신장치로 갱신 키 전송
S112
종 료

도면6


시 작
S202
비폐기 사용자 ID의 수신장치가 갱신 키 수신?
아니오
예
갱신 키와 사용자 비밀 키를 기초로 복호화 키 생성
S204
S206
송신 장치로부터 암호문 수신?
아니오
예
복호화 키를 이용하여 암호문 복호화
S208
종 료