

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0132922

(43) 공개일자 2023년09월19일

(51) 국제특허분류(Int. Cl.)

H04L 9/40 (2022.01) H04L 9/08 (2006.01)

H04L 9/32 (2006.01)

(52) CPC특허분류

H04L 63/0421 (2013.01)

H04L 63/0823 (2013.01)

(21) 출원번호 10-2022-0029778

(22) 출원일자 2022년03월10일

심사청구일자 2022년03월10일

(71) 출원인

상명대학교산학협력단

서울특별시 종로구 홍지문2길 20 (홍지동, 상명대학교)

고려대학교 산학협력단

서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)

(72) 발명자

박종환

서울특별시 종로구 홍지문 2길 20 상명대학교 자하관 N505호

김효승

서울특별시 성북구 고려대로22길 6-7

(74) 대리인

김등용

전체 청구항 수 : 총 15 항

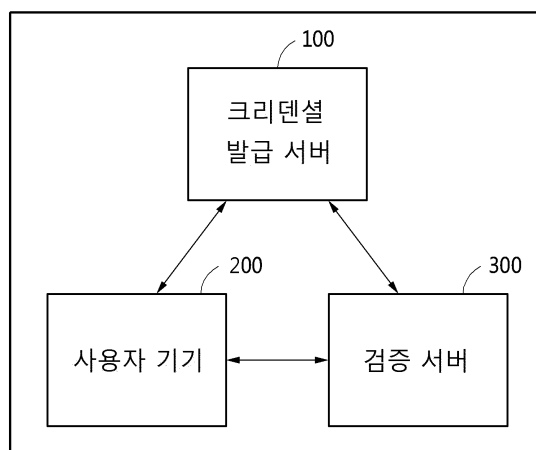
(54) 발명의 명칭 효율적으로 사용자 탈퇴 기능을 제공하는 익명 크리덴셜 인증 장치 및 방법

(57) 요약

효율적으로 사용자 탈퇴 기능을 제공하는 익명 크리덴셜 인증 시스템, 장치 및 방법이 개시된다. 상기 익명 크리덴셜 인증 시스템은 시스템 공개키와 마스터 비밀키를 생성하는 크리덴셜 발급 서버, 상기 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성하는 사용자 기기, 및 상기 사용자 기기로부터 수신된 익명 크리덴셜 서명을 검증하는 검증 서버를 포함하고, 상기 크리덴셜 발급 서버와 상기 사용자 기기는 크리덴셜 발급 알고리즘을 수행하여 크리덴셜을 발급하고, 상기 크리덴셜 발급 서버는 탈퇴한 사용자 목록에 기초한 정보를 생성하여 배포하고, 상기 검증 서버는 상기 탈퇴한 사용자 목록에 기초한 정보를 이용하여 상기 익명 크리덴셜 서명을 검증한다.

대표도 - 도4

10



(52) CPC특허분류

H04L 9/0825 (2013.01)

H04L 9/3236 (2013.01)

H04L 9/3247 (2013.01)

H04L 2209/42 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711134706
과제번호	2021-0-00518-001
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	데이터경제를위한블록체인기술개발(R&D)
연구과제명	블록체인 데이터 암호화 기반의 프라이버시 보호 기술개발
기 여 율	1/1
과제수행기관명	한양대학교산학협력단
연구기간	2021.04.01 ~ 2025.12.31

명세서

청구범위

청구항 1

시스템 공개키와 마스터 비밀키를 생성하는 크리덴셜 발급 서버;
 상기 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성하는 사용자 기기; 및
 상기 사용자 기기로부터 수신된 익명 크리덴셜 서명을 검증하는 검증 서버를 포함하고,
 상기 크리덴셜 발급 서버와 상기 사용자 기기는 크리덴셜 발급 알고리즘을 수행하여 크리덴셜을 발급하고,
 상기 크리덴셜 발급 서버는 탈퇴한 사용자 목록에 기초한 정보를 생성하여 배포하고,
 상기 검증 서버는 상기 탈퇴한 사용자 목록에 기초한 정보를 이용하여 상기 익명 크리덴셜 서명을 검증하는,
 익명 크리덴셜 인증 시스템.

청구항 2

제1항에 있어서,
 상기 크리덴셜 발급 서버는 보안 상수(λ)와 최대 속성 수(m , m 은 임의의 자연수)를 입력으로 하는 시스템키 생성 알고리즘을 수행하여 상기 시스템 공개키와 상기 마스터 비밀키를 생성하는,
 익명 크리덴셜 인증 시스템.

청구항 3

제2항에 있어서,
 상기 크리덴셜 발급 서버는,
 페어링 파라미터($BG = (p, g, \hat{g}, e, G_1, G_2, G_T)$)를 생성하고,
 암호학적 해쉬 함수($H: \{0,1\}^* \rightarrow Z_p^*$)를 선택하고,
 최대 사용자 수($N=2^n$, n 은 임의의 자연수)에 대하여 SD(Subset Difference) 기법의 셋업 알고리즘($SD.Setup(N)$)을 실행하여 바이너리 트리(BT)를 생성하고,
 임의의 $\gamma, \gamma_1, \gamma_2, \gamma_3 \in Z_p^*$ 를 생성하여 $\Gamma = e(g, \hat{g})^\gamma$ 와 $g_1 = g^{\gamma_1}$, $g_2 = g^{\gamma_2}$, $g_3 = g^{\gamma_3}$, $\hat{g}_1 = \hat{g}^{\gamma_1}$, $\hat{g}_2 = \hat{g}^{\gamma_2}$, $\hat{g}_3 = \hat{g}^{\gamma_3}$ 를 계산하고,
 임의의 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 를 생성하여 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 과 $\hat{h}_0 = \hat{g}^{\xi_0}, \dots, \hat{h}_m = \hat{g}^{\xi_m}$ 을 계산하여,
 상기 마스터 비밀키($isk = (\gamma, \{\gamma_u\}_{u=1}^3, \{\xi_v\}_{v=0}^m)$ 와 상기 시스템 공개키($ipk = (H, BG, BT, N, \{g_u, \hat{g}_u\}_{u=1}^3, \{h_v, \hat{h}_v\}_{v=0}^m, \Gamma)$)를 생성하는,
 익명 크리덴셜 인증 시스템.

청구항 4

제3항에 있어서,
 상기 사용자 기기는,

고유한 값($\alpha \in Z_p^*$)을 선택하고,

$f = h_0^\alpha$ 를 계산하여,

상기 사용자 비밀키($usk = \alpha$)와 상기 사용자 공개키($upk = f$)를 생성하는,

익명 크리덴셜 인증 시스템.

청구항 5

제4항에 있어서,

상기 사용자 기기는 상기 사용자 비밀키를 알고 있다는 사실을 증명하기 위해 비대화식 영지식 증명(Non-Interactive Zero-Knowledge, NIZK)의 증명 값을 생성하여 상기 크리덴셜 발급 서버로 송신하고,

상기 증명 값을 생성하기 위하여, 상기 사용자 기기는

상기 사용자 비밀키(usk)와 상기 사용자 공개키(ipk)를 이용하여 $f = h_0^\alpha$ 를 계산하고,

임의의 $\eta \in Z_p^*$ 를 선택하고,

상기 암호학적 해쉬 함수(H)를 이용하여 $c = H(h_0 \| f \| h_0^\eta)$ 을 계산하고,

$s = \eta - c\alpha$ 를 계산하여 상기 증명 값($\pi_0 = (s, c)$)을 생성하고,

상기 사용자 기기는 상기 증명 값(π_0), 상기 사용자 공개키(f), 속성 집합(A)을 상기 크리덴셜 발급 서버로 전송하는,

익명 크리덴셜 인증 시스템.

청구항 6

제5항에 있어서,

상기 크리덴셜 발급 서버는,

상기 증명 값(π_0)의 유효성을 검증하고,

SD 기법의 할당 알고리즘($SD.Assign(BT, i)$)을 실행하여 경로 집합($PV = \{(j, k)\}$)을 생성하고,

상기 경로 집합에 포함된 각각의 경로((j, k))에 대하여,

$a_{j, \ell}$ 를 Z_p^* 에서 임의로 선택하여 다항식($F_{j, k}(x) = a_{j, \ell}x + \gamma$)을 생성하고,

A 의 인덱스 집합을 $I_A \subset \{1, \dots, m\}$ 라고 하였을 때, 임의로 선택한 값($r \in Z_p^*$)과 상기 다항식을 이용하여

$K_1 = g^{F_{j, \ell}(k)} \left(g_1 f \prod_{u \in I_A} h_u^{t_u} \right)^r = g^{F_{j, \ell}(k)} \left(g_1 h_0^\alpha \prod_{u \in I_A} h_u^{t_u} \right)^r$ 와 $K_2 = g^r$ 을 계산하여,

결과 메시지($M = (\alpha, \{K_1, K_2\}, PV)$)를 생성하고,

상기 결과 메시지를 상기 사용자 기기로 전송하는,

익명 크리덴셜 인증 시스템.

청구항 7

제6항에 있어서,

상기 사용자 기기는 $\hat{f}=\hat{h}_0^\alpha$ 를 계산하여 상기 크리덴셜 ($cre=(\alpha, f, \hat{f}, \{K_1, K_2\}, PV)$)을 생성하는,
 익명 크리덴셜 인증 시스템.

청구항 8

제6항에 있어서,
 상기 사용자 기기는 상기 결과 메시지를 상기 크리덴셜로 설정하는,
 익명 크리덴셜 인증 시스템.

청구항 9

제7항 또는 제8항에 있어서,
 상기 크리덴셜 발급 서버는

SD 기법의 커버 알고리즘($SD.Cover(BT, RL_{\tau-1})$)을 실행하여 커버 집합 $CV=\{S_{j,k}\}$ 을 생성하고,

상기 커버 집합의 각 원소($S_{j,k}$)에 대하여

T_j 가 상기 바이너리 트리에 포함된 노드들 중 j번째 노드를 루트 노드로 하는 서브 트리이고, T_k 가 상기 바이너리 트리에 포함된 노드들 중 k번째 노드를 루트 노드로 하는 서브 트리이고, 1이 상기 T_j 의 깊이와 상기 T_k 의 깊이의 차이($\ell=d_j-d_k$)라고 정의하였을 때, 임의로 $s \in Z_p^*$ 를 선택하여 $U_1=g^{F_{j,\ell}(k)}(g_2g_3^\tau)^s$ 와 $U_2=g^s$ 를 계산하여,

상기 탈퇴한 사용자 목록에 기초한 정보($info_\tau=(\tau, \{U_1, U_2\}, CV)$)를 생성하고,

상기 탈퇴한 사용자 목록에 기초한 정보를 공개하는,

익명 크리덴셜 인증 시스템.

청구항 10

제9항에 있어서,

상기 사용자 기기는, 경로 (j,k) 에 대하여

$$\Delta_1 = \frac{-k'}{k-k'} \text{ 과 } \Delta_2 = \frac{-k}{k'-k} \text{ 를 계산하고,}$$

$I_B \subset I_A$ 가 공개하고 싶은 속성 집합(B)에 대한 인덱스 집합이고 $I_C \subset I_A \setminus I_B$ 가 A와 B의 차집합($C=A \setminus B$)

에 대한 인덱스 집합일 때 임의값 $\theta, \tilde{r}, \tilde{s} \in Z_{pq}^*$ 를 선택하여

$$\begin{aligned} \sigma_1 &= (K_1 K_2^\theta)^{\Delta_1} U_1^{\Delta_2} \left(g_1 f \prod_{u \in I_A} h_u^{t_u} g^\theta \right)^{\tilde{r}} (g_2 g_3^\tau)^{\tilde{s}} g^\gamma \left(g_1 h_0^\alpha \prod_{u \in I_A} h_u^{t_u} g^\theta \right)^R (g_2 g_3^\tau)^S, \\ \sigma_2 &= K_2^{\Delta_1} g^{\tilde{r}} = g^R, \quad \sigma_3 = U_2^{\Delta_2} g^{\tilde{s}} = g^S, \quad \sigma_4 = \hat{g}_1 \hat{f} \prod_{u \in I_A} \hat{h}_u^{t_u} g^{\hat{\theta}} = \hat{g}_1 \hat{h}_0^\alpha \prod_{u \in I_A} \hat{h}_u^{t_u} g^{\hat{\theta}}. \end{aligned}$$

를 계산하고($R = \Delta_1 r + \tilde{r}$, $S = \Delta_2 s + \tilde{s}$),

$$T_1 = \sigma_4 \hat{g}^{-1} \prod_{v \in I_B} \hat{h}_v^{t_v} \text{ 를 계산하고,}$$

임의의 $\eta_0, \{\eta_u\}_{u \in I_C}, \eta_0' \in Z_p^*$ 를 선택하고,

$c_1 = H\left(\hat{h}_0 \parallel \{\hat{h}_u\}_{u \in I_C} \parallel \hat{g} \parallel T_1 \parallel \hat{h}_0^{\eta_0} \prod_{u \in I_C} \hat{h}_u^{\eta_u} \hat{g}^{\eta_0'}\right)$ 를 계산하고,

$s_0 = \eta_0 - c_1 \alpha$, $\{s_u = \eta_u - c_1 t_u\}_{u \in I_C}$, $s_0' = \eta_0' - c_1 \theta$ 를 계산하고,

증명 값($\pi_1 = (s_0, \{s_u\}_{u \in I_C}, s_0')$)을 생성하고,

상기 익명 크리덴셜 서명($\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \pi_1)$)과 상기 공개하고 싶은 속성 집합(B)를 상기 검증 서버로 송신하는,

익명 크리덴셜 인증 시스템.

청구항 11

제10항에 있어서,

검증 서버는

$T_1 = \sigma_4 g_1^{\hat{-1}} \prod_{v \in I_B} \hat{h}_v^{t_v}$ 를 계산하고,

제1 검증식($c = H\left(\hat{h}_0 \parallel \{\hat{h}_u\}_{u \in I_C} \parallel \hat{g} \parallel T_1 \parallel \hat{h}_0^{s_0} \prod_{u \in I_C} \hat{h}_u^{s_u} \hat{g}^{s_0'} \sigma_4^{c_1} g_1^{\hat{-1}}\right)$)을 이용하여 상기 증명 값($\pi_1 = (s_0, \{s_u\}_{u \in I_C}, s_0')$)의 유효성을 검증하고,

$T_2 = g_2 g_3^{\hat{\tau}}$ 를 계산하고,

제2 검증식($e(\sigma_1, \hat{g}) = \Gamma \cdot e(\sigma_2, \sigma_4) \cdot e(\sigma_3, T_2)$)의 만족 여부에 따라 상기 익명 크리덴셜 서명을 검증하는,

익명 크리덴셜 인증 시스템.

청구항 12

익명 크리덴셜 인증 시스템에서 사용자 기기에게 크리덴셜을 발급하는 크리덴셜 발급 서버에 있어서,

보안 상수(λ)와 최대 속성 수(m , m 은 임의의 자연수)를 입력으로 하는 시스템키 생성 알고리즘을 수행하여 시스템 공개키와 마스터 비밀키를 생성하는 키 생성부; 및

상기 사용자 기기에게 상기 크리덴셜을 발급하는 크리덴셜 발급부를 포함하고,

상기 키 생성부는

페어링 파라미터($BG = (p, g, \hat{g}, e, G_1, G_2, G_T)$)를 생성하고,

암호학적 해쉬 함수($H: \{0, 1\}^* \rightarrow Z_p^*$)를 선택하고,

최대 사용자 수($N=2^n$, n 은 임의의 자연수)에 대응하는 바이너리 트리(BT)를 생성하고,

임의의 $\gamma, \gamma_1, \gamma_2, \gamma_3 \in Z_p^*$ 를 생성하여 $\Gamma = e(g, \hat{g})^\gamma$ 와 $g_1 = g^{\gamma_1}$, $g_2 = g^{\gamma_2}$, $g_3 = g^{\gamma_3}$, $\hat{g}_1 = \hat{g}^{\gamma_1}$, $\hat{g}_2 = \hat{g}^{\gamma_2}$, $\hat{g}_3 = \hat{g}^{\gamma_3}$ 를 계산하고,

임의의 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 를 생성하여 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 과 $\hat{h}_0 = \hat{g}^{\xi_0}, \dots, \hat{h}_m = \hat{g}^{\xi_m}$ 을 계산하여,

상기 마스터 비밀키($isk = (\gamma, \{\gamma_u\}_{u=1}^3, \{\xi_v\}_{v=0}^m)$ 와 상기 시스템 공개키

$(ipk = (H, BG, BT, N, \{g_u, \hat{g}_u\}_{u=1}^3, \{h_v, \hat{h}_v\}_{v=0}^m, \Gamma))$ 를 생성하는,

크리덴셜 발급 서버.

청구항 13

크리덴셜 발급 서버, 사용자 기기, 및 검증 서버를 포함하는 익명 크리덴셜 인증 시스템 상에서 수행되는 익명 크리덴셜 인증 방법에 있어서,

상기 크리덴셜 발급 서버가 시스템 공개키와 마스터 비밀키를 생성하는 단계;

상기 사용자 기기가 상기 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성하는 단계;

상기 크리덴셜 발급 서버가 상기 사용자 기기에 크리덴셜을 발급하는 단계;

상기 크리덴셜 발급 서버가 탈퇴한 사용자 목록에 기초한 정보를 생성하여 배포하는 단계;

상기 사용자 기기가 익명 크리덴셜 서명을 생성하고 상기 익명 크리덴셜 서명을 상기 검증 서버로 송신하는 단계; 및

상기 검증 서버가 상기 익명 크리덴셜 서명을 검증하는 단계를 포함하는 익명 크리덴셜 인증 방법.

청구항 14

제13항에 있어서,

상기 시스템 공개키와 상기 마스터 비밀키를 생성하는 단계는,

페어링 파라미터($BG = (p, g, \hat{g}, e, G_1, G_2, G_T)$)를 생성하는 단계;

암호학적 해쉬 함수($H: \{0,1\}^* \rightarrow Z_p^*$)를 선택하는 단계;

최대 사용자 수($N=2^n$, n은 임의의 자연수)에 대하여 SD(Subset Difference) 기법의 셋업 알고리즘(SD.Setup(N))을 실행하여 바이너리 트리(BT)를 생성하는 단계;

임의의 $\gamma, \gamma_1, \gamma_2, \gamma_3 \in Z_p^*$ 를 생성하여 $\Gamma = e(g, \hat{g})^\gamma$ 와 $g_1 = g^{\gamma_1}$, $g_2 = g^{\gamma_2}$, $g_3 = g^{\gamma_3}$, $\hat{g}_1 = \hat{g}^{\gamma_1}$, $\hat{g}_2 = \hat{g}^{\gamma_2}$, $\hat{g}_3 = \hat{g}^{\gamma_3}$ 를 계산하는 단계;

임의의 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 를 생성하여 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 과 $\hat{h}_0 = \hat{g}^{\xi_0}, \dots, \hat{h}_m = \hat{g}^{\xi_m}$ 을 계산하는 단계; 및

상기 마스터 비밀키($isk = (\gamma, \{\gamma_u\}_{u=1}^3, \{\xi_v\}_{v=0}^m)$ 와 상기 시스템 공개키

$(ipk = (H, BG, BT, N, \{g_u, \hat{g}_u\}_{u=1}^3, \{h_v, \hat{h}_v\}_{v=0}^m, \Gamma))$ 를 설정하는 단계를 포함하는,

익명 크리덴셜 인증 방법.

청구항 15

크리덴셜 발급 서버가 사용자 기기에 크리덴셜을 발급하는 방법에 있어서,

보안 상수(λ)와 최대 속성 수(m, m은 임의의 자연수)를 입력으로 하는 시스템키 생성 알고리즘을 수행하여 시스템 공개키와 마스터 비밀키를 생성하는 단계;

상기 사용자 기기에 상기 크리덴셜을 발급하는 단계를 포함하고,

상기 시스템 공개키와 마스터 비밀키를 생성하는 단계는

페어링 파라미터($BG = (p, g, \hat{g}, e, G_1, G_2, G_T)$)를 생성하는 단계;

암호학적 해쉬 함수($H: \{0,1\}^* \rightarrow Z_p^*$)를 선택하는 단계;

최대 사용자 수($N=2^n$, n은 임의의 자연수)에 대응하는 바이너리 트리(BT)를 생성하는 단계;

임의의 $\gamma, \gamma_1, \gamma_2, \gamma_3 \in Z_p^*$ 를 생성하여 $\Gamma = e(g, \hat{g})^\gamma$ 와 $g_1 = g^{\gamma_1}$, $g_2 = g^{\gamma_2}$, $g_3 = g^{\gamma_3}$, $\hat{g}_1 = \hat{g}^{\gamma_1}$, $\hat{g}_2 = \hat{g}^{\gamma_2}$, $\hat{g}_3 = \hat{g}^{\gamma_3}$ 를 계산하는 단계;

임의의 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 를 생성하여 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 과 $\hat{h}_0 = \hat{g}^{\xi_0}, \dots, \hat{h}_m = \hat{g}^{\xi_m}$ 을 계산하는 단계; 및

상기 마스터 비밀키($isk = (\gamma, \{\gamma_u\}_{u=1}^3, \{\xi_v\}_{v=0}^m)$ 와 상기 시스템 공개키($ipk = (H, BG, BT, N, \{g_u, \hat{g}_u\}_{u=1}^3, \{h_v, \hat{h}_v\}_{v=0}^m, \Gamma)$)를 설정하는 단계를 포함하는,

생성하는,

크리덴셜을 발급하는 방법.

발명의 설명

기술 분야

[0001] 본 발명은 익명 크리덴셜 인증 방법에 관한 것으로, 특히 사용자의 신원을 나타내는 다수의 속성을 크리덴셜로 인증받고, 이를 이용하여 원하는 속성만 노출하는 동시에 나머지 속성들에 대해서는 익명성을 제공하며 신원을 증명할 수 있는 익명 크리덴셜 방법에 관한 것이다.

배경 기술

[0002] 개인의 신원(identity)으로부터 다양한 서비스가 제공되고 서비스 제공자는 막대한 경제적인 이윤을 창출하고 있지만, 모순적으로 신원 데이터는 개인이 아니라 서비스 제공자가 소유하고 있다. 최근 자기결정권을 위한 데이터 주권이 필요하다는 인식이 강화되었으며, 개인이 스스로 자신의 신원 데이터에 대하여 주권을 갖고 인증을 수행하는 자기 주권 신원(self-sovereign identity) 개념이 도입되었다. 실제로 마이데이터 체계를 마련하고 블록체인을 활용한 분산 신원 인증(Decentralized identifier, DID) 서비스에 대하여 표준화가 진행 중이다.

[0003] 익명 인증 크리덴셜은 DID 실현화를 위한 코어 기술로써 사용자는 신원을 인증해주는 기관(기업)에서 크리덴셜을 발급받고, 검증자(서비스 제공자)가 요청할 때마다 익명 크리덴셜을 생성하여 신원 및 자격을 증명한다. 이때, 신원 정보는 다수의 속성으로 표현할 수 있는데 사용자는 자신이 원하는 속성만 노출하고 나머지는 안전하게 숨기면서 인증을 진행할 수 선택적 노출(selective disclosure) 기능을 기본적으로 제공한다.

[0004] 익명 인증 크리덴셜 기술과 비슷한 암호학적 기술로 DAA(Direct Anonymous Attestation)과 그룹 서명 기술이 있다. 해당 기술들은 익명 인증을 제공한다는 점에서 공통점이 있다. 하지만 DAA는 속성기반 기술이 아니므로 선택적 노출 기능을 제공하지 않는다. 대신 동일한 사용자가 값을 생성하였는지 확인할 수 있는 연결성을 제공하며 해당 성질은 사용자가 제어할 수 있다. 그룹 서명은 그룹 구성원 중 한 명이 서명하였다는 사실만 증명함으로써 익명성을 제공한다. 하지만 필요에 따라 서명자의 신원을 알아낼 수 있는 기관(opener)가 존재하기 때문에 완전한 익명성을 제공하지 않는다. 또한 그룹 서명도 선택적 노출 기능을 제공하지 않는다.

선행기술문헌

특허문헌

[0005] (특허문헌 0001) 대한민국 공개특허 제2021-0064076호 (2021.06.02. 공개)

발명의 내용

해결하려는 과제

[0006] 본 발명이 이루고자 하는 기술적인 과제는 효율적인 사용자 탈퇴 기능을 제공하는 익명 크리덴셜 인증 기법을 제공하는 것이다. 폐기 목록을 사용하는 기존 익명 크리덴셜 기술은 효율적으로 사용자 폐기 여부를 확인할 수 없다. 탈퇴한 사용자가 증명 값을 생성하였는지 확인하기 위해서 검증자는 먼저 검증 시간에 유효한 폐기 목록(RL)을 확인하고, 각 항목에 명시되어 있는 탈퇴한 사용자에게 대한 추적 값이 증명 값과 적합한지 정해진 검증식을 통하여 확인한다. 즉, 검증 계산량과 시간은 폐기 목록(RL)의 크기에 비례한다. 따라서 고정된 검증 계산량과 시간으로만 사용자 탈퇴를 효율적으로 지원하는 것은 중요한 해결 과제이다.

[0007] 본 발명이 이루고자 하는 다른 기술적인 과제는 이상적인 위조 불가능성을 만족하는 익명 크리덴셜 인증 기법을 제공하는 것이다. 기존 익명 크리덴셜 기술은 사용자가 탈퇴하여도 신원 증명 값을 생성하는 것을 방지할 수 없으며, 미리 생성된 서명의 유효성을 사후에 검증하는 방식이다. 탈퇴 목록을 이전 것으로 잘못 참조하거나, 확인하는 과정을 생략한다면 탈퇴 사용자가 생성한 증명 값이 여전히 유효한 것으로 인식되므로 이를 악용할 수 있다. 이상적인 해결책은 사용자가 탈퇴한 후 증명 값을 원천적으로 생성하지 못하는 위조 불가능성을 제공하는 것이다.

과제의 해결 수단

[0008] 본 발명의 일 실시예에 따른 익명 크리덴셜 인증 시스템은 시스템 공개키와 마스터 비밀키를 생성하는 크리덴셜 발급 서버, 상기 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성하는 사용자 기기, 및 상기 사용자 기기로부터 수신된 익명 크리덴셜 서명을 검증하는 검증 서버를 포함하고, 상기 크리덴셜 발급 서버와 상기 사용자 기기는 크리덴셜 발급 알고리즘을 수행하여 크리덴셜을 발급하고, 상기 크리덴셜 발급 서버는 탈퇴한 사용자 목록에 기초한 정보를 생성하여 배포하고, 상기 검증 서버는 상기 탈퇴한 사용자 목록에 기초한 정보를 이용하여 상기 익명 크리덴셜 서명을 검증한다.

[0009] 본 발명의 일 실시예에 따른 크리덴셜 발급 서버는 익명 크리덴셜 인증 시스템에서 사용자 기기에게 크리덴셜을 발급하고, 보안 상수와 최대 속성 수를 입력으로 하는 시스템키 생성 알고리즘을 수행하여 시스템 공개키와 마스터 비밀키를 생성하는 키 생성부 및 상기 사용자 기기에게 상기 크리덴셜을 발급하는 크리덴셜 발급부를 포함한다.

[0010] 본 발명의 일 실시예에 따른 익명 크리덴셜 인증 방법은 크리덴셜 발급 서버, 사용자 기기, 및 검증 서버를 포함하는 익명 크리덴셜 인증 시스템 상에서 수행되고, 상기 크리덴셜 발급 서버가 시스템 공개키와 마스터 비밀키를 생성하는 단계, 상기 사용자 기기가 상기 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성하는 단계, 상기 크리덴셜 발급 서버가 상기 사용자 기기에게 크리덴셜을 발급하는 단계, 상기 크리덴셜 발급 서버가 탈퇴한 사용자 목록에 기초한 정보를 생성하여 배포하는 단계, 상기 사용자 기기가 익명 크리덴셜 서명을 생성하고 상기 익명 크리덴셜 서명을 상기 검증 서버로 송신하는 단계, 및 상기 검증 서버가 상기 익명 크리덴셜 서명을 검증하는 단계를 포함한다.

[0011] 본 발명의 일 실시예에 따른, 크리덴셜 발급 서버가 사용자 기기에게 크리덴셜을 발급하는 방법은 보안 상수와 최대 속성 수를 입력으로 하는 시스템키 생성 알고리즘을 수행하여 시스템 공개키와 마스터 비밀키를 생성하는 단계, 상기 사용자 기기에게 상기 크리덴셜을 발급하는 단계를 포함한다.

발명의 효과

[0012] 본 발명의 실시예에 따른 익명 크리덴셜 인증 장치 및 방법에 의할 경우, 효율적인 사용자 탈퇴 기능을 제공할 수 있다.

[0013] 또한, 탈퇴한 사용자가 증명 값을 원천적으로 생성하지 못하게 함으로써 위조 불가능성을 제공할 수 있다.

도면의 간단한 설명

[0014] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 SD 기법에서 이용되는 바이너리 트리를 설명하기 위한 도면이다.

도 2는 SD 기법의 할당 알고리즘에서 도출되는 개인 집합을 설명하기 위한 도면이다.

도 3은 SD 기법의 커버 알고리즘에서 도출되는 커버 집합을 설명하기 위한 도면이다.

도 4는 본 발명의 일 실시예에 따른 익명 크리덴셜 인증 시스템을 도시한다.

도 5는 도 4에 도시된 크리덴셜 발급 서버의 기능 블록도이다.

도 6은 도 4에 도시된 익명 크리덴셜 인증 시스템 상에서 수행되는 익명 크리덴셜 인증 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0015] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.
- [0016] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.
- [0017] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.
- [0018] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0019] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0020] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0021] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시예들을 상세히 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.
- [0022] 우선, 본 발명에서 이용되는 배경 지식을 기술한다.
- [0023] [페어링 함수]
- [0024] 페어링 (접선형) 함수 $e: G_1 \times G_2 \rightarrow G_T$ 란 위수가 p 인 대수적 순환 그룹 G_1, G_2 의 임의의 원소 $g \in G_1, \hat{g} \in G_2$ 에 대해서, 접선형 성질 $e(g^a, \hat{g}^b) = e(g^b, \hat{g}^a) = e(g, \hat{g})^{ab}$ 을 제공하는 함수를 의미한다. 이러한 접선형 성질을 이용하여 $e(g^a, \hat{g}^b \hat{g}^c) = e(g^a, \hat{g}^b) e(g^a, \hat{g}^c)$ 를 만족한다.
- [0025] 페어링 함수는 사용하는 G_1, G_2 의 관계에 따라 Type 1, Type 2, Type 3로 구분할 수 있으며, 이 중 Type-3 비대칭 페어링 함수는 $G_1 \neq G_2$ 이면서 $\phi: G_2 \rightarrow G_1$ 인 동형함수가 존재하지 않을 때를 의미한다. 즉, 페어링 함수

값을 얻기 위해 반드시 $g \in G_1, \hat{g} \in G_2$ 는 정해진 위치에 입력해야 한다.

[0026] [SD(Subset Difference 기법]

[0027] SD 기법은 트리 기반 사용자 탈퇴 방법이며 브로드캐스트 암호 설계를 위해 제시된 방법이다(Dalit Naor, Moni Naor, Jeffery Lotspiech: Revocation and Tracing Schemes for Stateless Receivers. CRYPTO 2001).

[0028] SD 기법은 잎 노드(leaf node)의 수가 $N=2^n$ 인 바이너리 트리 BT 를 사용한다(도 1 참조). 바이너리 트리는 잎 노드를 제외한 루트 노드(root node)와 중간 노드들 각각은 2개의 자식 노드를 가진다. 바이너리 트리에 포함된 모든 노드, 즉 총 $2N-1$ 개의 노드를 식별자 i 를 이용하여 v_i 로 표기할 수 있다. 또한, i 는 보통 루트 노드를 1로 시작하여 왼쪽에서 오른쪽, 위에서 아래 방향으로 차례대로 할당한다. 이러한 규칙을 통해 잎 노드는 식별자 $i \in \{N, \dots, 2N-1\}$ 를 이용하여 나타낼 수 있다.

[0029] SD 기법은 4개의 알고리즘(SD.Setup, SD.Assign, SD.Cover, SD.Match)을 포함하며, 각 알고리즘에 대한 설명은 다음과 같다. 단, SD 기법에 대한 상세한 설명은 생략하기로 하고, 구체적인 내용에 관하여는 Kwangsu Lee 등의 논문(Kwangsu Lee, Woo Kwon Koo, Dong Hoon Lee, and Jong Hwan Park. Public-key revocation and tracing schemes with subset difference methods revisited. In Mirosław Kutylowski and Jaideep Vaidya, editors, ESORICS 2014, Part II) 등이 참조될 수 있다.

[0030] - $SD.Setup(N) \rightarrow BT$ (SD 기법의 셋업 알고리즘). 사용자 수 N 을 입력받아 잎 노드 수가 N 인 바이너리 트리 BT 를 생성한다.

[0031] - $SD.Assign(BT, i) \rightarrow PV$ (SD 기법의 할당 알고리즘). 사용자 i 를 잎 노드 v_i 로 할당한다. 사용자의 식별자 i 는 노드의 식별자와 같은 범위 $\{N, \dots, 2N-1\}$ 에 있음을 가정한다. 만약 사용자 N 명의 식별자를 일반집합 K 에서 선택한다면 일대일 함수 $Label: K \rightarrow \{N, \dots, 2N-1\}$ 를 통해 가정을 만족할 수 있다. 루트 노드부터 v_i 까지의 모든 경로(출발 노드의 식별자, 도착 노드의 식별자)들로 구성된 집합 $\{(j, k)\}$ 을 PV 로 출력한다. 예시적으로, $v_{10}(i=10)$ 에 대한 PV 는 도 2에 도시되어 있다. 여기서, PV 는 개인 집합(private set)이나 경로 집합으로 명명될 수 있다.

[0032] - $SD.Cover(BT, RL) \rightarrow CV$ (SD 기법의 커버 알고리즘). 탈퇴 사용자 목록 RL 에 대한 커버 집합 $CV = \{S_{j,k}\}$ 를 할당한다. 여기서 $S_{j,k}$ 에 대한 설명은 다음과 같다. v_i 가 루트 노드인 BT 의 서브 트리를 T_i 라 할 때 집합 S_i 는 T_i 의 모든 노드를 원소로 가지고 있으며, $S_{j,k}$ 는 S_j 에는 존재하지만 S_k 에는 존재하지 않는 노드로 구성되어 있는 차집합이다. CV 의 원소 $S_{j,k}$ 들은 서로 교집합이 없는 서로소 집합이며 탈퇴하지 않은 사용자의 노드들을 모두 포함(covering)한다. 도 3에는 탈퇴한 v_{12}, v_{13} 에 대한 CV 가 도시되어 있다.

[0033] - $SD.Match(PV, CV) \rightarrow ((j, k), S_{j,k})$ (SD 기법의 매치 알고리즘). 각각의 PV 의 원소 (j, k) 와 CV 의 원소 $S_{j,k}$ 에 대하여 다음 조건을 만족하는지 확인한다.

[0034] (1) $j = j'$

[0035] (2) $k \neq k'$

[0036] (3) $\ell = \ell'$ (여기서 ℓ 과 ℓ' 은 경로 (j, k) 와 (j', k') 의 길이를 의미한다.)

[0037] 모든 조건을 만족하면 해당 $((j, k), S_{j,k})$ 를 출력하고, 모든 원소에 대해서 조건을 만족하지 않으면 $\perp$ 을 출력한다. 즉, 매칭의 결과로 출력되는(모든 조건을 만족하는) $((j, k), S_{j,k})$ 가 존재하는 경우 해당 사용자는 탈

퇴하지 않은 것으로 볼 수 있고, $((j,k), S_{f,k})$ 가 존재하지 않는 경우 해당 사용자는 탈퇴한 사용자로 볼 수 있다.

[0038] 이하에서는 본 명세서에서 사용하는 표기와 알고리즘에 대해 설명한다.

[0039] - $x||y$: 문자열 x 와 y 의 연결(예, $x=010$ 이고 $y=110$ 이면 $x||y=010110$)

[0040] - Z_p^* : 집합 $\{1,2,...,p-1\}$

[0041] - $H: \{0,1\}^* \rightarrow Z_p^*$: 암호학적 해쉬 함수. 임의의 문자열을 입력받아 Z_p^* 의 한 원소를 출력

[0042] - λ : 보안 상수. 값이 클수록 높은 보안 등급을 제공

[0043] - t : 사용자를 나타내는 속성. 해쉬 함수를 이용하여 Z_p^* 값으로 할당하는 것을 가정 (예, 성별속성: "gender.att=men" $\rightarrow$ $H(\text{"gender.att=men"})=t$)

[0044] - A : 속성 집합 ($0 < |A| \leq m$)

[0045] - I_A : A 의 속성에 대한 메타 데이터의 인덱스 집합. 메타 데이터마다 고유의 인덱스가 할당 됨. 예를 들어, 사용자를 {홍길동, 남자, 21세}로 나타낸다면 각 속성을 나타내는 메타 데이터는 {Name, Gender, Age}임. 메타 데이터와 인덱스를 다음과 같이 $\text{Name} \rightarrow 1$, $\text{Gender} \rightarrow 2$, $\text{Age} \rightarrow 3$ 로 할당한다면, $A = \{t_1, t_2, t_3\}$ ($t_1 = H(\text{홍길동})$, $t_2 = H(\text{남자})$, $t_3 = H(\text{21세})$) 이고 $I_A = \{1,2,3\}$ 이 됨

[0046] - $X \setminus Y$: 집합 X 와 집합 Y 의 차집합 (예, $X=\{1,2,3\}$ 이고 $Y=\{2\}$ 이면 $X \setminus Y=\{1,3\}$)

[0047] - ipk/isk : 크리덴셜 발급 서버의 시스템 공개키/마스터 공개키

[0048] - upk/usk : 사용자의 공개키/비밀키

[0049] - cre : 크리덴셜

[0050] - σ : 익명 크리덴셜 서명

[0051] - τ : 시간(epoch). 특정 기간의 개념

[0052] - RL_τ : τ 시간에 탈퇴한 사용자 목록

[0053] - $info_\tau$: τ 시간에 익명 크리덴셜 서명을 생성하기 위한 탈퇴 목록 기반 정보

[0054] 익명 크리덴셜 시스템을 구성하는 객체는 I (크리덴셜 발급 서버), U (사용자 기기), 및 V (검증 서버)이다. 또한 I_{kg} (시스템키 발급) 알고리즘, $Issue$ (크리덴셜 발급 요청 및 발급), $Update$ (탈퇴 목록 기반 정보 생성), $Show$ (익명 크리덴셜 서명 생성) 알고리즘, $Verify$ (익명 크리덴셜 서명 검증) 알고리즘으로 구성된다. 이 중 $Issue$ 프로토콜은 U 가 수행하는 $Obtain$ (크리덴셜 발급 요청) 서브 알고리즘, I 가 수행하는 Iss (크리덴셜 발급) 서브 알고리즘으로 구성할 수 있다.

[0055] 각 알고리즘 및 프로토콜에 대한 상세한 설명은 다음과 같다.

[0056] - $\mathbf{Ikg}(1^\lambda, m) \rightarrow (ipk, isk)$. 크리덴셜 발급 서버(I)가 실행하는 알고리즘(시스템 키 발급 알고리즘, 시스템 키 생성 알고리즘 등으로 명명될 수 있음)으로, 보안 상수 λ 와 최대 속성 수 m (m 은 임의의 자연수)에 대해서 공개 키(시스템 공개키) ipk 와 마스터 비밀키 isk 를 다음과 같이 생성한다.

[0057] 1. Type-3 페어링 파라미터 $BG = (p, g, \hat{g}, e, G_1, G_2, G_T)$ 를 생성한다. 다만, 본 발명에서 다른 타입의 페어링 함수나 페어링 파라미터의 이용을 배제하는 것은 아니며, 실시예에 따라 다른 타입의 페어링 함수나 페어링 파라

미터가 이용될 수도 있다.

- [0058] 2. 암호학적 해쉬 함수(즉, 임의의 해쉬 함수) $H: \{0,1\}^* \rightarrow Z_p^*$ 를 선택한다.
- [0059] 3. 최대 사용자 수 $N=2^n$ (n 은 임의의 자연수)에 대하여 SD 기법의 셋업 알고리즘($SD.Setup(N)$)을 실행하고 그 결과로 바이너리 트리 BT 를 얻는다.
- [0060] 4. 임의의 $\gamma, \gamma_1, \gamma_2, \gamma_3 \in Z_p^*$ 를 생성하여 $\Gamma = e(g, \hat{g})^\gamma$ 와 $g_1 = g^{\gamma_1}$, $g_2 = g^{\gamma_2}$, $g_3 = g^{\gamma_3}$, $\hat{g}_1 = \hat{g}^{\gamma_1}$, $\hat{g}_2 = \hat{g}^{\gamma_2}$, $\hat{g}_3 = \hat{g}^{\gamma_3}$ 를 계산한다.
- [0061] 5. 임의의 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 를 생성하여 $\xi_0, \xi_1, \dots, \xi_m \in Z_p^*$ 과 $\hat{h}_0 = \hat{g}^{\xi_0}, \dots, \hat{h}_m = \hat{g}^{\xi_m}$ 을 계산한다.
- [0062] 6. $isk = (\gamma, \{\gamma_u\}_{u=1}^3, \{\xi_v\}_{v=0}^m)$ 와 $ipk = (H, BG, BT, N, \{g_u, \hat{g}_u\}_{u=1}^3, \{h_v, \hat{h}_v\}_{v=0}^m, \Gamma)$ 로 설정한다. 생성된 시스템 공개키 ipk 는 공개될 수 있다(또는 사용자 기기(U) 및/또는 검증 서버(V)로 전송될 수 있다).
- [0063] - $Ukg(ipk) \rightarrow (upk, usk)$. 사용자 기기(U)가 실행하는 알고리즘(사용자 키 발급 알고리즘, 사용자 키 생성 알고리즘 등으로 명명될 수 있음)으로, 사용자의 공개키 upk 와 비밀키 usk 를 생성하기 위해 아래 과정을 수행한다.
- [0064] 1. U 를 대표할 수 있는 고유한 값 $\alpha \in Z_p^*$ 를 선택한다.
- [0065] 2. $f = h_0^\alpha$ 를 계산한다.
- [0066] 3. $usk = \alpha$ 와 $upk = f$ 를 출력한다. 생성된 사용자 공개키 upk 는 공개될 수 있다(또는 크리덴셜 서버(I) 및/또는 검증 서버(V)로 전송될 수 있다).
- [0067] - $Issue(Obtain(i, usk, ipk, A) \leftarrow Iss(i, upk, isk, A)) \rightarrow (cre; M)$. 속성 집합 A 를 대표하는 (식별자가 i 인) 사용자 기기(U)가 크리덴셜을 발급받기 위하여 크리덴셜 발급 서버(I)와 함께 실행하는 프로토콜(알고리즘)이며 (크리덴셜 발급 알고리즘으로 명명될 수 있음), 서브 알고리즘으로 Obtain(제1 서브 알고리즘)과 Iss(제2 서브 알고리즘)는 최종 결과로 각각 사용자 기기(U)에 대한 크리덴셜 cre 와 결과 메시지 M 을 출력한다. 자세한 과정은 다음과 같이 순차적으로 진행된다.
- [0068] 1. Obtain : 사용자 기기(U)에 의해 수행되며, 아래 과정을 수행한다.
- [0069] (1-1). 사용자 기기(U)는 자신의 비밀키 α 를 알고 있다는 사실을 비대화식 영지식(non-interactive zero-knowledge, NIZK) 프로토콜을 이용하여 증명한다. 영지식 프로토콜의 증명 값은 $\pi_0 = NIZK\{(\alpha): f = h_0^\alpha\}$ 로 나타낼 수 있으며 아래 과정을 통해 생성한다.
- [0070] (1-1-1). usk 와 ipk 를 이용하여 $f = h_0^\alpha$ 를 계산한다.
- [0071] (1-1-2). 임의의 $\eta \in Z_p^*$ 를 선택한다.
- [0072] (1-1-3). 암호학적 해쉬 함수 H 를 이용하여 $c = H(h_0 \| f \| h_0^\eta)$ 을 계산한다.
- [0073] (1-1-4). $s = \eta - c\alpha$ 를 계산한다.
- [0074] (1-1-5). 증명 값을 $\pi_0 = (s, c)$ 로 설정한다.
- [0075] (1-2). 생성한 증명 값 π_0 과 함께 공개키 f , 속성 집합 A 를 함께 전송한다.
- [0076] 요컨대, Obtain 알고리즘은 비밀키 α 를 알고 있음을 증명하기 위한 영지식 프로토콜(또는 영지식 증명)의 증명

값을 생성하는 알고리즘을 의미할 수 있다.

2. Iss : 크리덴셜 발급 서버(I)에 의해 수행되며, 아래 과정을 수행한다.

(2-1). 전송받은 f 가 시스템 내 고유한 값인지 확인한다. 다른 크리덴셜 발급 세션에서 f 가 이미 사용되었다면 (또는 사용 중이라면) $M=\perp$ 을 출력하고 프로토콜을 중단한다.

(2-2). $c = H(h_0 || f || h_0^{sc})$ 가 만족하는지 확인하여 π_0 의 유효성을 검증한다. 등식이 성립하면 계속 진행하고, 성립하지 않으면 $M=\perp$ 을 출력하고 프로토콜을 중단한다.

(2-3). SD 기법의 할당 알고리즘($SD.Assign(BT, i)$)을 실행하여 경로 집합 $PV = \{(j, k)\}$ 을 얻는다.

(2-4). 각각의 경로 (j, k) 마다 경로의 길이를 ℓ 이라고 정의하였을 때 다음을 수행한다.

(2-4-1). 다항식 $F_{j,k}(x) = a_{j,\ell}x + \gamma$ 을 내부 저장소에 저장한다. 여기서 $a_{j,\ell}$ 은 Z_p^* 에서 임의로 선택한다. 즉, 경로 집합에 포함된 각 경로에 대응하는 다항식이 생성될 수 있다.

(2-4-2). A 의 인덱스 집합을 $I_A \subset \{1, \dots, m\}$ 라고 하였을 때, 임의로 선택한 값 $r \in Z_p^*$ 과 다항식을 이용하여 $K_1 = g^{F_{j,\ell}(k)} \left(g_1 f \prod_{u \in I_A} h_u^{t_u} \right)^r = g^{F_{j,\ell}(k)} \left(g_1 h_0^\alpha \prod_{u \in I_A} h_u^{t_u} \right)^r$ 와 $K_2 = g^r$ 을 계산한다.

(2-5). 결과 메시지 $M = (\alpha, \{K_1, K_2\}, PV)$ 를 전송한다. 결과 메시지 M 은 크리덴셜에 포함된 정보의 적어도 일부를 포함하는 메시지로, 실시예에 따라 크리덴셜 또는 제1 크리덴셜로 명명될 수도 있다.

3. Obtain : 사용자 기기(U)에 의해 수행되며, 아래 과정을 수행한다.

(3-1). $\hat{f} = \hat{h}_0^\alpha$ 를 계산한다.

(3-2). 크리덴셜 $cre = (\alpha, f, \hat{f}, \{K_1, K_2\}, PV)$ 을 저장한다.

(3-3). (optional) 만약 서명 생성 과정에서 연산량이 늘어나도 저장량을 줄이는 것이 더 중요하다면 (3-1)을 생략할 수 있다. 이때의 크리덴셜 $cre = (\alpha, \{K_1, K_2\}, PV)$ 이다. 즉, 크리덴셜 발급 서버(I)로부터 수신된 메시지 M 이 크리덴셜이 될 수 있다.

- **Update**($RL_{\tau-1}, isk \rightarrow info_\tau$). 크리덴셜 발급 서버(I)는 $\tau-1$ 시간에 탈퇴한 사용자 목록 $RL_{\tau-1}$ 을 이용하여 다음 시간 τ 에 대한 정보 $info_\tau$ 를 다음과 같이 생성한다. 이 알고리즘은 업데이트 알고리즘 등으로 명명될 수 있다.

1. SD 기법의 커버 알고리즘($SD.Cover(BT, RL_{\tau-1})$)을 실행하여 커버 집합 $CV = \{S_{j,k}\}$ 을 얻는다.

2. 커버 집합의 원소 $S_{j,k}$ 에 대하여 다음을 수행한다.

(2-1). T_j 와 T_k 의 깊이의 차이를 $\ell = d_j - d_k$ 라고 정의하였을 때, 이전에 정의한 $F_{j,\ell}(x) = a_{j,\ell}x + \gamma$ 를 불러온다.

(2-2). 임의로 $s \in Z_p^*$ 를 선택하여 $U_1 = g^{F_{j,\ell}(k)} (g_2 g_3^T)^s$ 와 $U_2 = g^s$ 를 계산한다.

(2-3). $info_\tau = (\tau, \{U_1, U_2\}, CV)$ 를 배포(또는 검증 서버(V)로 전송)한다.

- **Show**($ipk, info_\tau, A, B, cre \rightarrow (\sigma, B)$). 사용자 기기(U)에 의해 수행되는 익명 크리덴셜 서명 값 생성 알고리즘

으로, τ 시간에 사용자 기기(U)는 저장하고 있는 cre 를 이용하여 공개하고 싶은 속성들로 이루어진 부분 속성 집합 $B \subseteq A$ 에 대한 익명 크리덴셜 서명 값 σ 를 다음과 같이 생성한다.

[0096] 1. SD 기법의 매치 알고리즘($SD.Match(PV, CV)$)을 수행하여 $\perp$ 이 나오면 서명을 만들 수 없으므로 중단한다. 이는 이미 탈퇴한 사용자인기 때문이다. 정당한 $S_{j,k} \in CV$ 와 $(j,k) \in PV$ 를 얻을 수 있으면(즉, 탈퇴하지 않은 사용자라면) 다음 단계로 넘어간다.

[0097] 2. 경로 (j,k) 에 대한 $K_1 = g^{F_{j,\ell}(k)} \left(g_1 h_0^\alpha \prod_{u \in I_A} h_u^{t_u} \right)^r$ 과 $K_2 = g^r$ 를 cre 에서 찾는다.

[0098] 3. 경로 (j,k) 의 길이를 ℓ 이라고 하였을 때, $S_{j,k} \in CV$ 에 대한 $U_1 = g^{F_{j,\ell}(k)} (g_2 g_3^\tau)^s$ 와 $U_2 = g^s$ 를 $info$ 에서 찾는다.

[0099] 4. $\Delta_1 = \frac{-k'}{k-k'}$ 과 $\Delta_2 = \frac{-k}{k'-k}$ 를 계산한다. 이는 $F_{j,\ell}(x)$ 에서 $x=0$ 에 대한 함수값을 보간(interpolation)할 수 있는 라그랑주 계수(Lagrange's coefficient)이다. 즉, $\Delta_1 F_{j,\ell}(k) + \Delta_2 F_{j,\ell}(k') = F(0) = \gamma$ 를 만족하는 값이다.

[0100] 5. B (공개하고 싶은 속성 집합)에 대한 인덱스 집합을 $I_B \subseteq I_A$ 로 정의하고 A 와 B 의 차집합 $C = A \setminus B$ 숨기고 싶은 속성 집합)에 대한 인덱스 집합 $I_C \subseteq I_A \setminus I_B$ 을 정의하였을 때, 임의값 $\theta, \tilde{r}, \tilde{s} \in Z_p^*$ 를 선택하여 아래 값을 계산한다.

$$\begin{aligned} \sigma_1 &= (K_1 K_2^\theta)^{\Delta_1} U_1^{\Delta_2} \left(g_1 f \prod_{u \in I_A} h_u^{t_u} g^\theta \right)^{\tilde{r}} (g_2 g_3^\tau)^{\tilde{s}} g^\gamma \left(g_1 h_0^\alpha \prod_{u \in I_A} h_u^{t_u} g^\theta \right)^R (g_2 g_3^\tau)^S, \\ \sigma_2 &= K_2^{\Delta_1} g^{\tilde{r}} = g^R, \quad \sigma_3 = U_2^{\Delta_2} g^{\tilde{s}} = g^S, \quad \sigma_4 = \hat{g}_1 \hat{f} \prod_{u \in I_A} \hat{h}_u^{t_u} \hat{g}^\theta = \hat{g}_1 \hat{h}_0^\alpha \prod_{u \in I_A} \hat{h}_u^{t_u} \hat{g}^\theta. \end{aligned}$$

[0101] 위의 식에서 $R = \Delta_1 r + \tilde{r}$ 이고 $S = \Delta_2 s + \tilde{s}$ 이다.

[0102] 6. 고윳값 α , 사용한 난수 θ , 숨기고 싶은 속성들을 모두 알고 있다는 사실을 비대화식 영지식(NIZK) 프로토콜을 이용하여 증명한다. 영지식 프로토콜의 증명 값은 $\pi_1 = NIZK \left\{ \left(\alpha, \{t_u\}_{u \in I_C}, \theta \right) : \sigma_4 g_1^{-1} \prod_{v \in I_B} \hat{h}_v^{t_v} = \hat{h}_0^\alpha \prod_{u \in I_C} \hat{h}_u^{t_u} g^\theta \right\}$ 로 나타낼 수 있으며 아래 과정을 통해 생성한다.

[0103] (6-1) 앞선 정의에 의하여 $B = \{t_v\}_{v \in I_B}$ 이므로 이를 이용하여 $T_1 = \sigma_4 g_1^{-1} \prod_{v \in I_B} \hat{h}_v^{t_v}$ 를 계산한다.

[0104] (6-2) 임의의 $\eta_0, \{\eta_u\}_{u \in I_C}, \eta_0' \in Z_p^*$ 를 선택한다.

[0105] (6-3) 해쉬 함수를 이용하여 $c_1 = H \left(\hat{h}_0 \parallel \{ \hat{h}_u \}_{u \in I_C} \parallel \hat{g} \parallel T_1 \parallel \hat{h}_0^{\eta_0} \prod_{u \in I_C} \hat{h}_u^{\eta_u} \hat{g}^{\eta_0'} \right)$ 를 계산한다.

[0106] (6-4) $s_0 = \eta_0 - c_1 \alpha$, $\{s_u = \eta_u - c_1 t_u\}_{u \in I_C}$, $s_0' = \eta_0' - c_1 \theta$ 를 계산한다.

[0107] (6-5) 증명 값을 $\pi_1 = (s_0, \{s_u\}_{u \in I_C}, s_0')$ 로 설정한다.

[0108] 7. 익명 크리덴셜 서명 $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \pi_1)$ 와 함께 공개 속성 집합 B 를 출력한다. 여기서 B 에 할당된 본래 속성을 함께 공개하는 것을 가정한다.

- [0110] $\text{Verify}(ipk, info_\tau, \sigma, B) \rightarrow 1 \text{ or } 0$. 검증 서버(V)에 의해 수행되는 검증 알고리즘으로써, 검증 서버(V)는 익명 크리덴셜 서명 $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \pi_1)$ 을 검증하기 위해 아래 조건을 만족하는지 확인한다.
- [0111] 1. (6-1)에서처럼 B 를 이용하여 $T_1 = \sigma_4 g_1^{\hat{c}_1^{-1}} \prod_{v \in I_B} \hat{h}_v^{t_v}$ 를 계산한다.
- [0112] 2. $\pi_1 = (s_0, \{s_u\}_{u \in I_C}, s_0')$ 가 유효한지 $c = H(\hat{h}_0 \parallel \{\hat{h}_u\}_{u \in I_C} \parallel \hat{g} \parallel T_1 \parallel \hat{h}_0^{s_0} \prod_{u \in I_C} \hat{h}_u^{s_u} \hat{g}^{s_0'} \sigma_4^{c_1} g_1^{\hat{c}_1^{-1}})$ 로 확인한다. 성립하면 다음으로 넘어가고, 그렇지 않으면 중단하고 0을 출력한다.
- [0113] 3. $\hat{g}_2, \hat{g}_3 \in ipk$ 와 $\tau \in info_\tau$ 를 이용하여 $T_2 = \hat{g}_2 \hat{g}_3^\tau$ 를 계산한다.
- [0114] 4. $e(\sigma_1, \hat{g}) = \Gamma \cdot e(\sigma_2, \sigma_4) \cdot e(\sigma_3, T_2)$ 를 만족하는지 확인한다.
- [0115] 5. 위의 모든 과정을 통과하면 1 (검증 성공)을 출력하고, 그렇지 않으면 0 (검증 실패)를 출력한다.
- [0116] 도 4는 본 발명의 일 실시예에 따른 익명 크리덴셜 인증 시스템을 도시한다. 익명 크리덴셜 인증 시스템을 서명함에 있어 앞선 기재와 중복되는 내용에 관하여는 그 상세한 기재를 생략하기로 한다.
- [0117] 도 4를 참조하면, 익명 크리덴셜 시스템 등으로 명명될 수도 있는 익명 크리덴셜 인증 시스템(10)은 크리덴셜 발급 서버(100), 사용자 기기(200), 및 검증 서버(300)를 포함한다.
- [0118] 크리덴셜 발급 서버(100), 사용자 기기(200), 및 검증 서버(300)는 적어도 프로세서 및/또는 메모리를 포함하는 컴퓨팅 장치로 구현될 수 있다. 컴퓨팅 장치는 PC(Personal Computer), 태블릿 PC, 스마트폰, HMD, 노트북 등을 포함할 수 있다.
- [0119] 크리덴셜 발급 서버(100)는 시스템키 발급 알고리즘을 수행하여 공개키 ipk 와 마스터 비밀키 isk 를 생성하고, 생성된 공개키 ipk 를 배포(또는 사용자 기기(200) 및/또는 검증 서버(300)로 송신)할 수 있다. 또한, 크리덴셜 발급 서버(100)는 사용자 기기(200)와의 유무선 통신을 통해 크리덴셜 발급 알고리즘의 적어도 일부를 수행함으로써 결과 메시지(또는 크리덴셜)를 생성하고 생성된 결과 메시지를 사용자 기기(200)로 송신할 수 있다. 또한, 크리덴셜 발급 서버(100)는 업데이트 알고리즘을 수행함으로써 탈퇴 목록 기반 정보를 생성하고 생성된 탈퇴 목록 기반 정보를 배포(또는 사용자 기기(200) 및/또는 검증 서버(300)로 송신)할 수 있다.
- [0120] 사용자 기기(200)는 사용자키 발급 알고리즘을 수행함으로써 공개키 ipk 로부터 사용자 공개키 upk 와 사용자 비밀키 usk 를 생성할 수 있다. 또한, 사용자 기기(200)는 크리덴셜 발급 서버(100)와의 유무선 통신을 통해 크리덴셜 발급 알고리즘의 적어도 일부를 수행함으로써 크리덴셜 발급 서버(100)로부터 결과 메시지(또는 크리덴셜)를 수신할 수 있다. 실시예에 따라, 사용자 기기(200)는 수신된 결과 메시지를 포함하는 크리덴셜을 생성하는 것도 가능하다. 또한, 사용자 기기(200)는 익명 크리덴셜 서명 알고리즘을 수행함으로써 익명 크리덴셜 서명을 생성하고 생성된 익명 크리덴셜 서명을 검증 서버(300)로 송신할 수 있다. 이때, 사용자 기기(200)는 공개 속성 집합을 익명 크리덴셜 서명과 함께 전송할 수도 있다.
- [0121] 검증 서버(300)는 익명 크리덴셜 서명 검증 알고리즘을 수행함으로써 사용자 기기(200)로부터 수신된 익명 크리덴셜 서명을 검증할 수 있다.
- [0122] 도 5는 도 4에 도시된 크리덴셜 발급 서버의 기능 블록도이다.
- [0123] 도 4와 도 5를 참조하면, 크리덴셜 발급 서버(100)는 키 생성부(110), 크리덴셜 발급부(120), 및 업데이트부(130) 중 적어도 하나 이상을 포함한다. 실시예에 따라, 크리덴셜 발급 서버(100)는 저장부(140)를 더 포함할 수도 있다.
- [0124] 키 생성부(110)는 시스템키 발급 알고리즘을 수행하여 시스템 공개키 ipk 와 마스터 비밀키 isk 를 생성하고, 생성된 공개키 ipk 를 배포(또는 사용자 기기(200) 및/또는 검증 서버(300)로 송신)할 수 있다.
- [0125] 크리덴셜 발급부(120)는 사용자 기기(200)와의 유무선 통신을 통해 크리덴셜 발급 알고리즘의 적어도 일부를 수행함으로써 결과 메시지(또는 크리덴셜)를 생성하고 생성된 결과 메시지를 사용자 기기(200)로 송신할 수 있다.
- [0126] 업데이트부(130)는 업데이트 알고리즘을 수행함으로써 탈퇴 목록 기반 정보를 생성하고 생성된 탈퇴 목록 기반

정보를 배포(또는 사용자 기기(200) 및/또는 검증 서버(300)로 송신)할 수 있다.

- [0127] 저장부(140)에는 키 생성부(110)에 의한 키 생성 동작 과정에서 이용되는 데이터, 일시적으로 또는 비밀시적으로 생성되는 데이터, 크리덴셜 발급부(120)에 의한 크리덴셜 발급 동작 과정에서 이용되는 데이터, 일시적으로 또는 비밀시적으로 생성되는 데이터, 및 업데이트부(130)에 의한 업데이트 동작 과정에서 이용되는 데이터, 일시적으로 또는 비밀시적으로 생성되는 데이터가 저장될 수 있다.
- [0128] 도 6은 도 4에 기재된 익명 크리덴셜 인증 시스템 상에서 수행되는 익명 크리덴셜 인증 방법을 설명하기 위한 흐름도이다. 도 6에 도시된 방법을 구성하는 단계들 각각은 크리덴셜 발급 서버(100), 사용자 기기(200), 또는 검증 서버(300)에 의해 수행될 수 있다. 다시 말하면, 각 단계는 크리덴셜 발급 서버(100), 사용자 기기(200), 또는 검증 서버(300)에 포함된 프로세서의 동작으로 이해될 수도 있다.
- [0129] 우선, 크리덴셜 발급 서버(100)는 소정의 알고리즘을 수행함으로써 시스템 공개키와 마스터 비밀키를 생성할 수 있다(S110). 생성된 시스템 공개키는 공개(배포, 또는 전송)될 수 있다.
- [0130] 사용자 기기(200)는 수신된(또는 공개된) 시스템 공개키를 이용하여 사용자 공개키와 사용자 비밀키를 생성할 수 있다(S120). 생성된 사용자 공개키는 공개(배포, 또는 전송)될 수 있다.
- [0131] S130 단계에서, 크리덴셜 발급 서버(100)와 사용자 기기(200)는 소정의 데이터(또는 메시지)를 주고받으면서 크리덴셜 발급 알고리즘을 수행함으로써 크리덴셜 발급 동작을 수행할 수 있다.
- [0132] 또한, 크리덴셜 발급 서버(100)는 탈퇴한 사용자가 크리덴셜 서명을 생성할 수 없도록 탈퇴 목록 기반 정보를 생성하고 생성된 탈퇴 목록 기반 정보를 공개(배포, 또는 전송)할 수 있다(S140).
- [0133] 다음으로, 사용자 기기(200)는 익명 크리덴셜 서명을 생성하고, 생성된 익명 크리덴셜 서명과 공개 속성 집합을 검증 서버(300)로 송신할 수 있다(S150).
- [0134] 익명 크리덴셜 서명을 수신한 검증 서버(300)는 익명 크리덴셜 검증 알고리즘을 수행함으로써 사용자 기기(200)로부터 수신된 익명 크리덴셜에 대한 검증 동작을 수행할 수 있다(S160).
- [0135] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로컴퓨터, FPA(Field Programmable array), PLU(Programmable Logic Unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(Operation System, OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술 분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(Processing Element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(Parallel Processor)와 같은, 다른 처리 구성(Processing Configuration)도 가능하다.
- [0136] 소프트웨어는 컴퓨터 프로그램(Computer Program), 코드(Code), 명령(Instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(Collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성 요소(Component), 물리적 장치, 가상 장치(Virtual Equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(Signal Wave)에 영구적으로, 또는 일시적으로 구체화(Embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.
- [0137] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(Magnetic Media), CD-

ROM, DVD와 같은 광기록 매체(Optical Media), 플롭티컬 디스크(Floptical Disk)와 같은 자기-광 매체(Magneto-optical Media), 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

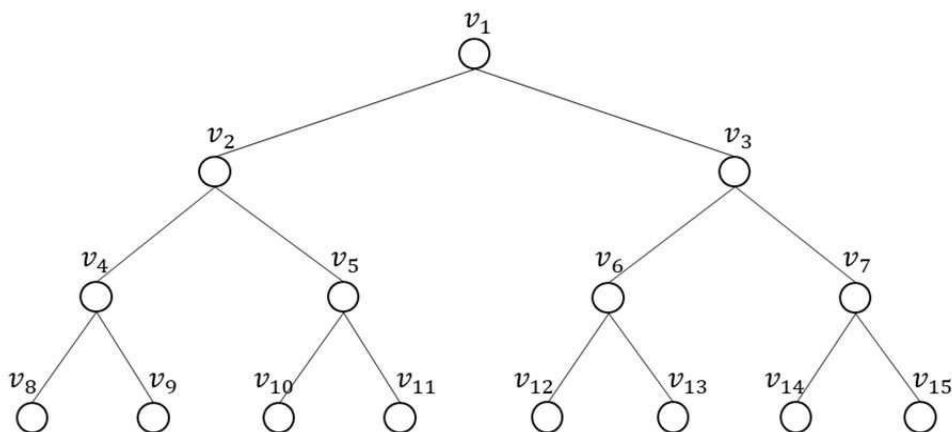
[0138] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성 요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

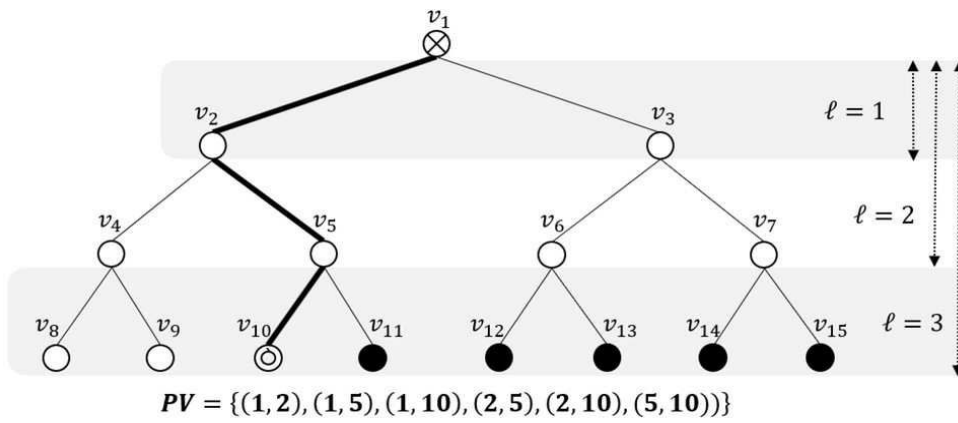
[0139] 10 : 익명 크리덴셜 인증 시스템
 100 : 크리덴셜 발급 서버
 110 : 키 생성부
 120 : 크리덴셜 발급부
 130 : 업데이트부
 140 : 저장부
 200 : 사용자 기기
 300 : 검증 서버

도면

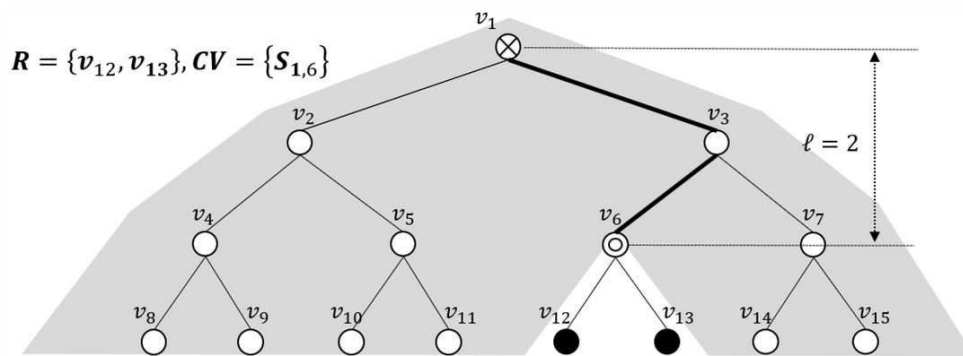
도면1



도면2

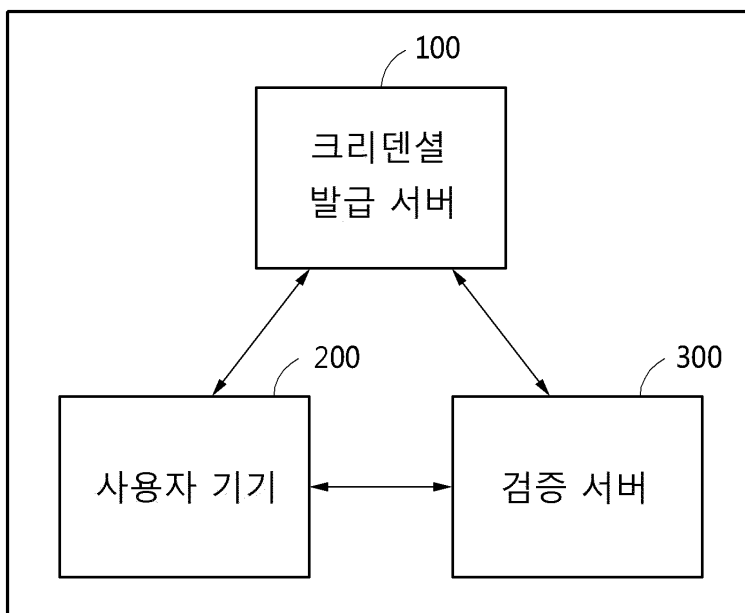


도면3



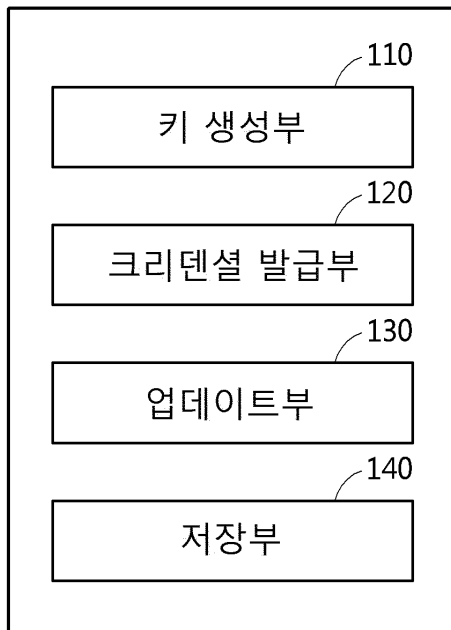
도면4

10



도면5

100



도면6

