(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2021년02월16일
(11) 등록번호 10-2215245
(24) 등록일자 2021년02월05일

(51) 국제특허분류(Int. Cl.)
H04L 9/00 (2006.01) *H04L 9/06* (2006.01)
H04L 9/08 (2006.01) *H04L 9/30* (2006.01)
H04L 9/32 (2006.01)
(52) CPC특허분류
H04L 9/008 (2013.01)
H04L 9/0643 (2013.01)
(21) 출원번호 10-2019-7011576
(22) 출원일자(국제) 2018년11월07일
심사청구일자 2019년04월22일
(85) 번역문제출일자 2019년04월22일
(65) 공개번호 10-2020-0054128
(43) 공개일자 2020년05월19일
(86) 국제출원번호 PCT/CN2018/114344
(87) 국제공개번호 WO 2019/072264
국제공개일자 2019년04월18일
(56) 선행기술조사문헌
Fanca, B. F. "Homomorphic mini-blockchain scheme." (2015).
(뒷면에 계속)

(73) 특허권자
어드밴스드 뉴 테크놀로지스 씨오., 엘티디.
케이만 군도, 그랜드 케이만 케이와이1-9008, 조지 타운, 27 하스피탈 로드, 케이만 코포레이트 센터
(72) 발명자
마 바올리
중국 저지앙 311121 항저우 유항 디스트릭트 웨스트 웬 이 로드 넘버969 빌딩 3 알리바바 그룹 리갈 디파트먼트 5층
장 웬빈
중국 저지앙 311121 항저우 유항 디스트릭트 웨스트 웬 이 로드 넘버969 빌딩 3 알리바바 그룹 리갈 디파트먼트 5층
(74) 대리인
김태홍, 김진희

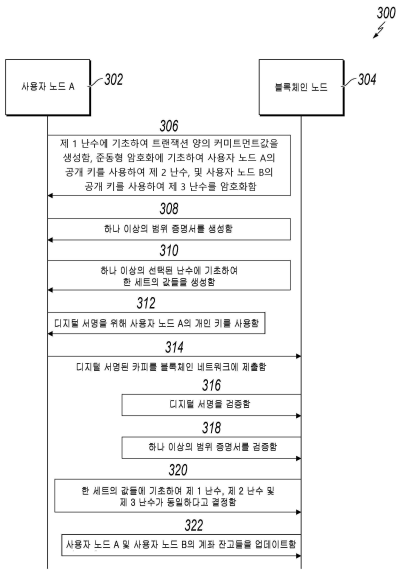
전체 청구항 수 : 총 16 항 심사관 : 양종필

(54) 발명의 명칭 **준동형 암호화를 사용하는 블록체인 데이터 보호**

(57) 요 약

본 개시의 구현예들은 제 1 난수에 기초하여 생성된 트랜잭션 양 중 제 1 양의 커미트먼트값의 디지털 서명된 카피, 제 1 계좌의 공개 키를 사용하여 암호화된 제 1 난수 및 제 1 양의 잔고 전송, 제 2 계좌의 공개 키를 사용하여 암호화된 제 2 난수 및 제 2 양의 잔고 전송, 및 하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들을, 제 1 계좌로부터 수신하는 것을 포함한다. 제 1 계좌는 한 세트의 값들에 기초하여 제 1 난수 및 제 2 난수가 동일한지 그리고 제 1 양 및 제 2 양이 동일한지를 결정하고, 제 1 양의 잔고 전송에 기초하여 제 2 계좌의 잔고 및 제 1 계좌의 잔고를 업데이트한다.

대 표 도 - 도3



(52) CPC특허분류
H04L 9/0869 (2013.01)
H04L 9/3066 (2013.01)
H04L 9/3247 (2013.01)
H04L 9/3263 (2013.01)
H04L 2209/38 (2013.01)

(56) 선행기술조사문헌
Wang, Qin, et al. "Preserving transaction privacy in bitcoin." Future Generation Computer Systems (2017).).
Acar, Abbas, et al. "A survey on homomorphic encryption schemes: Theory and implementation." ACM Computing Surveys (CSUR) 51.4 (2018): 1-35.
JP2008176192 A
US8861716 B2

명 세 서

청구범위

청구항 1

블록체인 네트워크의 합의 노드(consensus node)에 의해 수행되는 컴퓨터로 구현되는(computer-implemented) 방법에 있어서,

제 1 계좌(account)로부터,

제 1 난수(random number)에 기초하여 생성되는, 상기 제 1 계좌로부터 제 2 계좌로의 잔고 전송(balance transfer)의 제 1 양(amount)의 커미트먼트값(commitment value)의 디지털 서명된 카피(digitally signed copy) - 상기 잔고 전송의 제 1 양 및 상기 제 1 난수는 확률론적(probabilistic) 준동형 암호화(HE: homomorphic encryption) 알고리즘에 기초하여 상기 제 1 계좌의 제 1 공개 키(public key)를 사용하여 암호화됨 - ;

상기 잔고 전송의 제 2 양 및 제 2 난수 - 상기 잔고 전송의 제 2 양 및 상기 제 2 난수는 상기 확률론적 HE 알고리즘에 기초하여 상기 제 2 계좌의 공개 키를 사용하여 암호화됨 - ;

하나 이상의 범위 증명서(range proof); 및

하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들

을 수신하는 단계;

상기 디지털 서명된 카피에 대응하는 디지털 서명을, 상기 디지털 서명을 생성하는데 사용된 개인 키(private key)에 대응하는 상기 제 1 계좌의 제 2 공개 키를 사용하여 검증하는 단계;

상기 하나 이상의 범위 증명서가, 상기 잔고 전송의 제 1 양이 0보다 크고 상기 제 1 계좌의 잔고 이하임을 증명한다고 결정하는 단계;

상기 암호화된 제 1 양과 제 2 양, 상기 암호화된 제 1 난수와 제 2 난수, 및 상기 한 세트의 값들에 기초하여 상기 제 1 양 및 상기 제 2 양이 동일한지 그리고 상기 제 1 난수 및 상기 제 2 난수가 동일한지를 결정하는 단계; 및

상기 제 1 양 및 상기 제 2 양이 동일하고 상기 제 1 난수 및 상기 제 2 난수가 동일하다고 결정한 것에 응답하여, 상기 잔고 전송의 제 1 양에 기초하여 상기 제 1 계좌의 잔고 및 상기 제 2 계좌의 잔고를 업데이트하는 단계 - 상기 선택된 난수는 $r*$, $t*$, $z1*$, 및 $z2*$로 표시되고, 상기 선택된 난수는 a, b, c, d, C, D, 및 E를 생성하는데 사용되며, 여기서, $a = r* + xr$, $b = t* + xt$, $c = z1* + xz1$, $d = z2* + xz2$, $C = g^{r*}h^{t*}$, $D = u2^{r*}v2^{z1*}$, 및 $E = u2^{t*}v2^{z2*}$이고, r은 상기 제 1 난수이고, t는 상기 잔고 전송의 제 1 양이고, $z1$ 및 $z2$는 상기 잔고 전송의 제 2 양 및 상기 제 2 난수를 암호화하는데 사용되고, x는 C, D, E, 및 g를 해싱(hashing)하는 것에 기초하여 생성되는 해시값(hash value)이며, h, $u2$, 및 $v2$는 타원형 곡선의 생성자(generator)들임 -

를 포함하는, 블록체인 네트워크의 합의 노드에 의해 수행되는 컴퓨터로 구현되는 방법.

청구항 2

제 1 항에 있어서, 상기 커미트먼트값은, 준동형(homomorphic)인 커미트먼트 스킴(commitment scheme)을 사용하여 생성되는 것인, 컴퓨터로 구현되는 방법.

청구항 3

제 2 항에 있어서, 상기 커미트먼트 스킴은 페더슨(Pedersen) 커미트먼트 스킴인 것인, 컴퓨터로 구현되는 방법.

청구항 4

제 1 항에 있어서, 상기 확률론적 HE 알고리즘은 오카모토 우치야마(Okamoto-Uchiyama) HE 알고리즘인 것인, 컴퓨터로 구현되는 방법.

청구항 5

제 1 항에 있어서, 상기 한 세트의 값들은 또한 C, D, 및 E에 기초하여 생성되는 것인, 컴퓨터로 구현되는 방법.

청구항 6

제 5 항에 있어서, 확률론적 HE의 속성들에 기초하여 상기 제 1 양 및 상기 제 2 양이 동일한 것으로 결정되고 상기 제 1 난수 및 상기 제 2 난수가 동일한 것으로 결정되는 것인, 컴퓨터로 구현되는 방법.

청구항 7

제 6 항에 있어서, $g^a h^b$ = CT^x, $u2^a v2^c$ = DZ_B1^x, 및 $u2^b v2^d$ = EZ_B2^{x}이면 상기 제 1 양 및 상기 제 2 양이 동일한 것으로 결정되고 상기 제 1 난수 및 상기 제 2 난수가 동일한 것으로 결정되며, 여기서, T = $g^r h^t$는 상기 잔고 전송의 제 1 양의 커미트먼트값이며, Z_B1 = $u2^r v2^{z1}$, Z_B2 = $u2^t v2^{z2}$인 것인, 컴퓨터로 구현되는 방법.

청구항 8

제 1 항에 있어서, 상기 제 1 계좌의 잔고 및 상기 제 2 계좌의 잔고를 업데이트하는 단계는 HE에 기초하여 수행되는 것인, 컴퓨터로 구현되는 방법.

청구항 9

동작들을 수행하도록 컴퓨터 시스템에 의해 실행가능한 하나 이상의 명령어를 저장하는 비일시적 컴퓨터 판독가능 매체에 있어서, 상기 동작들은,

제 1 계좌로부터,

제 1 난수에 기초하여 생성되는, 상기 제 1 계좌로부터 제 2 계좌로의 잔고 전송의 제 1 양의 커미트먼트값의 디지털 서명된 카피 - 상기 잔고 전송의 제 1 양 및 상기 제 1 난수는 확률론적 준동형 암호화(HE) 알고리즘에 기초하여 상기 제 1 계좌의 제 1 공개 키를 사용하여 암호화됨 - ;

상기 잔고 전송의 제 2 양 및 제 2 난수 - 상기 잔고 전송의 제 2 양 및 상기 제 2 난수는 상기 확률론적 HE 알고리즘에 기초하여 상기 제 2 계좌의 공개 키를 사용하여 암호화됨 - ;

하나 이상의 범위 증명서; 및

하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들

을 수신하는 동작;

상기 디지털 서명된 카피에 대응하는 디지털 서명을, 상기 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 상기 제 1 계좌의 제 2 공개 키를 사용하여 검증하는 동작;

상기 하나 이상의 범위 증명서가, 상기 잔고 전송의 제 1 양이 0보다 크고 상기 제 1 계좌의 잔고 이하임을 증명한다고 결정하는 동작;

상기 암호화된 제 1 양과 제 2 양, 상기 암호화된 제 1 난수와 제 2 난수, 및 상기 한 세트의 값들에 기초하여 상기 제 1 양 및 상기 제 2 양이 동일한지 그리고 상기 제 1 난수 및 상기 제 2 난수가 동일한지를 결정하는 동작; 및

상기 제 1 양 및 상기 제 2 양이 동일하고 상기 제 1 난수 및 상기 제 2 난수가 동일하다고 결정한 것에 응답하여, 상기 잔고 전송의 제 1 양에 기초하여 상기 제 1 계좌의 잔고 및 상기 제 2 계좌의 잔고를 업데이트하는 동작 - 상기 선택된 난수는 $r*$, $t*$, $z1*$, 및 $z2*$로 표시되고, 상기 선택된 난수는 a, b, c, d, C, D, 및 E를 생성하는데 사용되며, 여기서, $a = r* + xr$, $b = t* + xt$, $c = z1* + xz1$, $d = z2* + xz2$, C = $g^{r*} h^{t*}$, D = $u2^{r*} v2^{z1*}$, 및 E = $u2^{t*} v2^{z2*}$이고, r은 상기 제 1 난수이고, t는 상기 잔고 전송의 제 1 양이고, $z1$ 및 $z2$는 상기

잔고 전송의 제 2 양 및 상기 제 2 난수를 암호화하는데 사용되고, *x*는 C, D, E, 및 *g*를 해싱하는 것에 기초하여 생성되는 해시값이며, *h*, *u2*, 및 *v2*는 타원형 곡선의 생성자들임 -

을 포함하는 것인, 동작들을 수행하도록 컴퓨터 시스템에 의해 실행가능한 하나 이상의 명령어를 저장하는 비일시적 컴퓨터 판독가능 매체.

청구항 10

제 9 항에 있어서, 상기 커미트먼트값은, 준동형인 커미트먼트 스킴을 사용하여 생성되는 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 11

제 10 항에 있어서, 상기 커미트먼트 스킴은 페더슨 커미트먼트 스킴인 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 12

제 9 항에 있어서, 상기 확률론적 HE 알고리즘은 오카모토 우치야마 HE 알고리즘인 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 13

제 9 항에 있어서, 상기 한 세트의 값들은 또한 C, D, 및 E에 기초하여 생성되는 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 14

제 13 항에 있어서, 확률론적 HE의 속성들에 기초하여 상기 제 1 양 및 상기 제 2 양이 동일한 것으로 결정되고 상기 제 1 난수 및 상기 제 2 난수가 동일한 것으로 결정되는 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 15

제 14 항에 있어서, $g^a h^b = CT^x$, $u2^a v2^c = DZ\_B1^x$, 및 $u2^b v2^d = EZ_B2^x$이면 상기 제 1 양 및 상기 제 2 양이 동일한 것으로 결정되고 상기 제 1 난수 및 상기 제 2 난수가 동일한 것으로 결정되며, 여기서, $T = g^r h^t$는 상기 잔고 전송의 제 1 양의 커미트먼트값이며, $Z\_B1 = u2^r v2^{z1}$, $Z_B2 = u2^t v2^{z2}$인 것인, 비일시적 컴퓨터 판독가능 매체.

청구항 16

시스템에 있어서,

하나 이상의 컴퓨터; 및

상기 하나 이상의 컴퓨터에 커플링되고, 동작들을 수행하도록 상기 하나 이상의 컴퓨터에 의해 실행가능한 명령어들이 저장되어 있는 하나 이상의 컴퓨터 판독가능 메모리

를 포함하고, 상기 동작들은,

제 1 계좌로부터,

제 1 난수에 기초하여 생성되는, 상기 제 1 계좌로부터 제 2 계좌로의 잔고 전송의 제 1 양의 커미트먼트값의 디지털 서명된 카피 - 상기 잔고 전송의 제 1 양 및 상기 제 1 난수는 확률론적 준동형 암호화(HE) 알고리즘에 기초하여 상기 제 1 계좌의 제 1 공개 키를 사용하여 암호화됨 - ;

상기 잔고 전송의 제 2 양 및 제 2 난수 - 상기 잔고 전송의 제 2 양 및 상기 제 2 난수는 상기 확률론적 HE 알고리즘에 기초하여 상기 제 2 계좌의 공개 키를 사용하여 암호화됨 - ;

하나 이상의 범위 증명서; 및

하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들

을 수신하는 동작;

상기 디지털 서명된 카피에 대응하는 디지털 서명을, 상기 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 상기 제 1 계좌의 제 2 공개 키를 사용하여 검증하는 동작;

상기 하나 이상의 범위 증명서가, 상기 잔고 전송의 제 1 양이 0보다 크고 상기 제 1 계좌의 잔고 이하임을 증명한다고 결정하는 동작;

상기 암호화된 제 1 양과 제 2 양, 상기 암호화된 제 1 난수와 제 2 난수, 및 상기 한 세트의 값들에 기초하여 상기 제 1 양 및 상기 제 2 양이 동일한지 그리고 상기 제 1 난수 및 상기 제 2 난수가 동일한지를 결정하는 동작; 및

상기 제 1 양 및 상기 제 2 양이 동일하고 상기 제 1 난수 및 상기 제 2 난수가 동일하다고 결정한 것에 응답하여, 상기 잔고 전송의 제 1 양에 기초하여 상기 제 1 계좌의 잔고 및 상기 제 2 계좌의 잔고를 업데이트하는 동작 - 상기 선택된 난수는 $r*$, $t*$, $z1*$, 및 $z2*$로 표시되고, 상기 선택된 난수는 a, b, c, d, C, D, 및 E를 생성하는데 사용되며, 여기서, $a = r* + xr$, $b = t* + xt$, $c = z1* + xz1$, $d = z2* + xz2$, $C = g^{r*}h^{t*}$, $D = u2^{r*}v2^{z1*}$, 및 $E = u2^{t*}v2^{z2*}$이고, r은 상기 제 1 난수이고, t는 상기 잔고 전송의 제 1 양이고, $z1$ 및 $z2$는 상기 잔고 전송의 제 2 양 및 상기 제 2 난수를 암호화하는데 사용되고, x는 C, D, E, 및 g를 해싱하는 것에 기초하여 생성되는 해시값이며, h, $u2$, 및 $v2$는 타원형 곡선의 생성자들임 -

을 포함하는 것인, 시스템.

발명의 설명

기 술 분 야

배 경 기 술

[0001] 블록체인(blockchain) 시스템, 합의(consensus) 네트워크, 분산형 원장(ledger) 시스템 네트워크 또는 블록체인으로 지칭될 수 있는 블록체인 네트워크는, 참여 엔티티(entity)들이 데이터를 안전하고 변경 불가능하게(immutably) 저장할 수 있게 한다. 블록체인은 트랜잭션(transaction)들의 원장 시스템으로 설명될 수 있으며, 원장의 여러 카피들이 블록체인 네트워크에 걸쳐 저장된다. 블록체인의 예시적인 유형들은 공개(public) 블록체인, 허가형 블록체인 및 개인(private) 블록체인을 포함할 수 있다. 공개 블록체인은 모든 엔티티들이 블록체인을 사용하고 합의 프로세스에 참여하도록 개방되어 있다. 허가형 블록체인은 공개 블록체인과 유사하지만 가입을 허가받은 엔티티들에 대해서만 개방되어 있다. 개인 블록체인은 특정 엔터티에 대해 제공되며, 판독 및 기록 허가들을 중앙에서 제어한다.

[0002] 블록체인들은 참여자들이 암호 화폐(crypto-currency)를 사용하여 상품 및/또는 서비스를 구매/판매하기 위한 트랜잭션들을 수행할 수 있게 하는 암호 화폐 네트워크들에서 사용된다. 통상적인 암호 화폐는 비트코인(Bitcoin)을 포함한다. 암호 화폐 네트워크들에서 기록 보존(record-keeping) 모델들은 사용자들 간의 트랜잭션들을 기록하는데 사용된다. 예시적인 기록 보존 모델들은 미지출(unspent) 트랜잭션 출력(UTXO: unspent transaction output) 모델 및 계좌 잔고(account balance) 모델을 포함한다. UTXO 모델에서, 각 트랜잭션은 이전 트랜잭션들의 출력을 지출하고 후속 트랜잭션들에서 지출될 수 있는 새로운 출력들을 생성한다. 사용자의 미지출 트랜잭션들이 추적되고 사용자가 소유한 잔고는 모든 사용자의 미지출 트랜잭션들의 합계로 계산된다. 계좌 잔고 모델에서, 각 사용자의 계좌 잔고가 전역 상태(global state)로서 추적된다. 각 트랜잭션마다 지출 계좌의 잔고가 트랜잭션 양(amount)보다 크거나 같은지를 확보하기 위해 검사된다. 이것은 전통적인 은행 업무와 비슷하다.

[0003] 블록체인 원장은 일련의 블록들을 포함하며, 블록들의 각각은 네트워크에서 실행되는 하나 이상의 트랜잭션을 포함한다. 각 블록은 원장의 페이지로 유추될 수 있지만 블록체인 자체는 원장의 전체 카피이다. 개별 트랜잭션들이 확인되고, 블록체인에 부가되는 블록에 부가된다. 블록체인 원장의 카피들은 네트워크의 노드들에 걸쳐 복제된다. 이러한 방식으로, 블록체인의 상태에 대한 글로벌 합의가 존재하게 된다. 게다가, 적어도 공개 네트워크들의 경우 블록체인은 모든 노드들이 볼 수 있도록 개방된다. 블록체인 사용자들의 프라이버시를 보호하기 위해 암호화 기술들이 구현될 수 있다.

[0004] 계좌 모델 하에서는, 트랜잭션의 양측 당사자들이 확정(commit)하는 값들을 숨기기 위해 커미트먼트 스킴(commitment scheme)들이 사용될 수 있다. 커미트먼트 스킴들은 당사자들이 선택 또는 값을 확정할 필요가 생길 때 발생할 수 있으며 나중에 그 가치를 연루된 다른 당사자들에게 전달할 수 있다. 예를 들어, 상호 작용하는 페더슨 커미트먼트(Pedersen Commitment)에서, 당사자 A는 랜덤 값 r에 기초하여 생성된 커미트먼트값 $PC(r, t)$를 송신함으로써 트랜잭션 양 t를 확정할 수 있다. 커미트먼트값이 생성되고, 당사자 B는 난수(random number) r을 획득함으로써 트랜잭션 양 t 만을 드러낼 수 있다.

발명의 내용

[0005] 본 개시의 구현예들은 사용자 확인, 상호 작용 없이 그리고 트랜잭션 양들 또는 계좌 잔고들을 드러내지 않은, 블록체인 트랜잭션들의 프라이버시 보호형 검증을 위한 컴퓨터로 구현되는 방법들을 포함한다. 보다 구체적으로, 본 개시의 구현예들은 커미트먼트 스킴들에 기초한 블록체인 사용자들 간의 트랜잭션들, 그리고 트랜잭션 양, 계좌 잔고들 또는 다른 블록체인 노드들에 대한 커미트먼트들을 생성하기 위한 난수들을 드러내지 않은 준동형 암호화(homomorphic encryption)를 유효성검사(validate)하는 것에 관한 것이다.

[0006] 일부 구현예들에서, 액션들은, 제 1 난수에 기초하여 생성되는, 제 1 계좌로부터 제 2 계좌로 전송될 트랜잭션 양 중 제 1 양의 커미트먼트값의 디지털 서명된 카피(digitally signed copy), 제 1 계좌의 공개 키(public key)를 사용하여 암호화된 제 1 양의 잔고 전송(balance transfer) 및 제 1 난수, 제 2 계좌의 공개 키를 사용하여 암호화된 제 2 양의 잔고 전송 및 제 2 난수, 하나 이상의 범위 증명서(range proof), 및 하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들을, 제 1 계좌로부터 수신하는 액션; 디지털 서명된 카피에 대응하는 디지털 서명을, 디지털 서명을 생성하는데 사용된 개인 키(private key)에 대응하는 제 1 계좌의 공개 키를 사용하여 검증하는 액션; 하나 이상의 범위 증명서가, 잔고 전송의 양이 0보다 크고 제 1 계좌의 잔고 이하임을 증명한다고 결정하는 액션; 한 세트의 값들에 기초하여 제 1 난수 및 제 2 난수가 동일한지 그리고 제 1 양 및 제 2 양이 동일한지를 결정하는 액션; 및 제 1 양 및 제 2 양이 동일하고 제 1 난수 및 제 2 난수가 동일하면, 제 1 양의 잔고 전송에 기초하여 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트하는 액션을 포함한다. 다른 구현예들은 컴퓨터 저장 디바이스들 상에 인코딩된, 방법들의 액션들을 수행하도록 구성된 대응 시스템들, 장치 및 컴퓨터 프로그램들을 포함한다.

[0007] 이들 및 다른 구현형태들은 이하의 특징들 중 하나 이상을 각각 선택적으로 포함할 수 있다: 커미트먼트값은, 준동형인 커미트먼트 스킴을 사용하여 생성된다; 커미트먼트 스킴은 페더슨(Pedersen) 커미트먼트 스킴이다; 제 1 양의 잔고 전송 및 제 1 난수는 확률론적(probabilistic) 준동형 암호화(HE: homomorphic encryption) 알고리즘에 기초하여 제 1 계좌의 공개 키를 사용하여 암호화되고, 제 2 양의 잔고 전송 및 제 2 난수는 확률론적 HE 알고리즘에 기초하여 제 2 계좌의 공개 키를 사용하여 암호화된다; 확률론적 HE 알고리즘은 오카모토 우치야마(Okamoto-Uchiyama) HE 알고리즘이다; 선택된 난수는 $r*$, $t*$, $z1*$ 및 $z2*$로 표시되고, 선택된 난수는 a, b, c 및 d를 생성하는데 사용되며, 여기서, $a = r* + xr$, $b = t* + xt$, $c = z1* + xz1$ 및 $d = z2* + xz2$이고, r은 제 1 난수이며, t는 제 1 양의 잔고 전송이고, x는 해시값(hash value)이다; 한 세트의 값들은 또한 C, D 및 E에 기초하여 생성되고, 여기서 $C = g^{r*}h^{t*}$, $D = u2^{r*}v2^{z1*}$, $E = u2^{t*}v2^{z2*}$, g, h, $u2$ 및 $v2$는 타원형 곡선의 생성자(generator)들이고, x는 C, D 및 E를 해싱(hashing)하는 것에 기초하여 생성된다; 확률론적 HE의 속성들에 기초하여 제 1 난수 및 제 2 난수가 동일한 것으로 결정되고 제 1 양 및 제 2 양이 동일한 것으로 결정된다; $g^ah^b = CT^x$, $u2^av2^c = DZ\_B1^x$ 및 $u2^bv2^d = EZ_B2^x$이면, 제 1 난수 및 제 2 난수는 동일한 것으로 결정되고 제 1 양과 제 2 양이 동일한 것으로 결정되며, 여기서, $T = g^rh^t$는 잔고 전송의 양의 커미트먼트값이고, $Z\_B1 = u2^rv2^{z1}$, $Z_B2 = u2^tv2^{z2}$이며, $z1$ 및 $z2$는 확률론적 HE 스킴에 기초하여 제 2 양의 잔고 전송 및 제 2 난수를 암호화하는데 사용되는 난수들이다; 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트하는 액션은 HE에 기초하여 수행된다.

[0008] 본 개시는 또한 하나 이상의 프로세서에 커플링되고, 하나 이상의 프로세서에 의해 실행될 때 하나 이상의 프로세서가 본 명세서에 제공된 방법들의 구현예들에 따른 동작들을 수행하도록 하는 명령어들이 저장되어 있는 하나 이상의 비일시적(non-transitory) 컴퓨터 판독가능 저장 매체를 제공한다.

[0009] 본 개시는 본 명세서에 제공된 방법들을 구현하기 위한 시스템을 더 제공한다. 시스템은 하나 이상의 프로세서, 및 하나 이상의 프로세서에 의해 실행될 때 하나 이상의 프로세서가 본 명세서에 제공된 방법들의 구현예들에 따른 동작들을 수행하도록 하는 명령어들이 저장되어 있는 하나 이상의 프로세서에 커플링된 컴퓨터 판독가능 저장 매체를 포함한다.

[0010] 본 개시에 따른 방법들이 본 명세서에 설명된 양태들 및 특징들의 임의의 조합을 포함할 수 있음을 이해될 것이다. 즉, 본 개시에 따른 방법들은 본 명세서에 구체적으로 설명된 양태들 및 특징들의 조합들로 제한되지 않으며, 제공되는 양태들 및 특징들의 임의의 조합을 포함한다.

[0011] 본 개시의 하나 이상의 구현예의 세부 내용은 첨부 도면들 및 아래의 설명에 제시되어 있다. 본 개시의 다른 특징들 및 이점들은 상세한 설명 및 도면, 그리고 특허청구범위로부터 분명해질 것이다.

도면의 간단한 설명

[0012] 도 1은 본 개시의 구현예들을 실행하는데 사용될 수 있는 예시적인 환경을 도시한다.

도 2는 본 개시의 구현예들에 따른 예시적인 개념적 아키텍처를 도시한다.

도 3은 본 개시의 구현예들에 따른 준동형 암호화 에 기초한 블록체인 트랜잭션의 프라이버시 보호형 유효성검사(validation)의 예시적인 방법을 도시한다.

도 4는 본 개시의 구현예들에 따른 준동형 암호화에 기초한 예시적인 블록체인 트랜잭션을 도시한다.

도 5는 본 개시의 구현예들에 따른 준동형 암호화 에 기초한 블록체인 트랜잭션의 프라이버시 보호형 유효성검사의 다른 예시적인 방법을 도시한다.

도 6는 본 개시의 구현예들에 따른 준동형 암호화에 기초한 다른 예시적인 블록체인 트랜잭션을 도시한다.

도 7은 본 개시의 구현예들에 따라 실행될 수 있는 예시적인 프로세스를 도시한다.

도 8은 본 개시의 구현예들에 따라 실행될 수 있는 다른 예시적인 프로세스를 도시한다.

다양한 도면들에서 유사한 참조 부호들은 동일한 요소들을 나타낸다.

발명을 실시하기 위한 구체적인 내용

[0013] 본 개시의 구현예들은 사용자 확인, 상호 작용 없이 그리고 트랜잭션 양들 또는 계좌 잔고들을 드러내지 않은, 블록체인 트랜잭션들의 프라이버시 보호형 검증을 위한 컴퓨터로 구현되는 방법들을 포함한다. 보다 구체적으로, 본 개시의 구현예들은 커미트먼트 스킴들에 기초한 블록체인 사용자들 간의 트랜잭션들, 그리고 트랜잭션 양, 계좌 잔고들 또는 다른 블록체인 노드들에 대한 커미트먼트들을 생성하기 위한 난수들을 드러내지 않은 준동형 암호화(HE)들을 유효성검사하는 것에 관한 것이다.

[0014] 본 개시의 구현예들을 위한 추가적인 맥락을 제공하기 위해, 위에서 소개된 바와 같이, 합의 네트워크들(예컨대, 피어-투-피어 노드들로 구성됨), 분산형 원장 시스템, 또는 간단히 블록체인으로 또한 지칭될 수 있는 블록체인 네트워크들은, 참여 엔티티들이 안전하고 변경 불가능하게 트랜잭션들을 수행할 수 있게 하고 데이터를 저장할 수 있게 한다. 블록체인은 공개 블록체인, 개인 블록체인 또는 컨소시엄 블록체인으로서 제공될 수 있다. 본 개시의 구현예들은 참여 엔티티들 사이에 공개되는 공개 블록체인을 참조하여 여기에서 더 상세하게 설명된다. 하지만, 본 개시의 구현예들은 임의의 적절한 유형의 블록체인에서 실현될 수 있다는 것이 고려된다.

[0015] 공개 블록체인에서, 합의 프로세스는 합의 네트워크의 노드들에 의해 제어된다. 예를 들어, 수백, 수천, 심지어 수백만 개의 엔티티들이 공개 블록체인에 참여할 수 있으며, 그들의 각각은 공개 블록체인에서 적어도 하나의 노드를 동작시킨다. 따라서, 공개 블록체인은 참여 엔티티들에 대한 공개 네트워크로 간주될 수 있다. 일부 예들에서, 엔티티들(노드들)의 대다수는 블록이 유효하고 블록체인에 부가되도록 모든 블록에 서명해야 한다. 예시적인 공개 블록체인은 피어-투-피어 지불 네트워크(암호 화폐 네트워크)인 비트코인 네트워크에서 사용되는 블록체인을 포함한다. 블록체인이라는 용어는 통상적으로 비트코인 네트워크와 관련이 있지만, 본 명세서에서 사용되는 바와 같이, 블록체인은 일반적으로 비트코인 네트워크를 특별히 참조하지 않는 분산형 원장들을 지칭한다.

[0016] 일반적으로, 공개 블록체인은 공개 트랜잭션들을 지원한다. 공개 트랜잭션은 블록체인 내의 모든 노드들과 공유되며 블록체인 원장은 모든 노드들에 걸쳐 복제된다. 즉, 모든 노드들이 블록체인과 관련하여 완벽한 상태로 합의된다. 합의(예컨대, 블록을 블록체인에 부가하는 것에 동의)를 달성하기 위해, 합의 프로토콜이 블록체인 네트워크 내에 구현된다. 예시적인 합의 프로토콜은 비트코인 네트워크에 구현된 작업 증명(POW: proof-of-work)을 포함하지만 이에 한정되는 것은 아니다.

[0017] 본 개시의 구현예들은 상기 맥락을 고려하여 본 명세서에서 보다 상세하게 설명된다. 보다 구체적으로, 위에서 소개된 바와 같이, 본 개시의 구현예들은 커미트먼트 스킴들에 기초한 블록체인 사용자들 간의 트랜잭션들, 그리고 트랜잭션 양, 계좌 잔고들 또는 다른 블록체인 노드들에 대한 커미트먼트들을 생성하기 위한 난수들을 드러내지 않은 HE를 유효성검사하는 것에 관한 것이다.

[0018] 본 개시의 구현예들에 따르면, 블록체인 트랜잭션들이 유효성검사될 수 있고, 트랜잭션 계좌 잔고, 트랜잭션 양, 또는 커미트먼트를 생성하는데 사용되는 난수를 드러내지 않은 커미트먼트에 기초하여 블록체인(원장)에 기록될 수 있다. 페더슨 커미트먼트(PC: Pedersen commitment)와 같은 커미트먼트 스킴은 난수를 사용하여 트랜잭션 양의 커미트먼트를 생성하는데 사용될 수 있다. 트랜잭션 양과 난수는 확률론적 또는 결정론적 HE를 사용하여 암호화될 수 있다. 트랜잭션 양 및 난수는 또한 HE의 속성들에 기초하여 트랜잭션을 유효성검사하기 위한 증명들로서의 한 세트의 값들을 생성하는데 사용될 수 있다. 트랜잭션의 커미트먼트, 암호화된 트랜잭션 양, 암호화된 난수 및 증명들은, 계좌 잔고, 트랜잭션 양 또는 난수가 드러나지 않은 채로 트랜잭션이 유효한지의 여부를 검증하기 위해 블록체인 노드에 의해 사용될 수 있다.

[0019] 도 1은 본 개시의 구현예들을 실행하는데 사용될 수 있는 예시적인 환경(100)을 도시한다. 일부 예들에서, 예시적인 환경(100)은 엔티티들이 공개 블록체인(102)에 참여할 수 있게 한다. 예시적인 환경(100)은 컴퓨팅 시스템들(106, 108) 및 네트워크(110)를 포함한다. 일부 예들에서, 네트워크(110)는 근거리 통신망(LAN: local area network), 광역 통신망(WAN: wide area network), 인터넷, 또는 이들의 조합을 포함하고, 웹 사이트들, 사용자 디바이스들(예컨대, 컴퓨팅 디바이스들) 및 백엔드(back-end) 시스템들을 접속시킨다. 일부 예들에서, 네트워크(110)는 유선 및/또는 무선 통신 링크를 통해 액세스될 수 있다.

[0020] 도시된 예에서, 컴퓨팅 시스템들(106, 108)은 각각 공개 블록체인(102)에서 노드로서의 참여를 가능하게 하는 임의의 적절한 컴퓨팅 시스템을 포함할 수 있다. 예시적인 컴퓨팅 디바이스들은 서버, 데스크탑 컴퓨터, 랩탑 컴퓨터, 태블릿 컴퓨팅 디바이스 및 스마트폰을 포함하지만, 이에 한정되는 것은 아니다. 일부 예들에서, 컴퓨팅 시스템들(106, 108)은 공개 블록체인(102)과 상호 작용하기 위한 하나 이상의 컴퓨터로 구현되는 서비스들을 호스팅한다. 예를 들어, 컴퓨팅 시스템(106)은 제 1 엔티티(예컨대, 사용자 A)의 컴퓨터로 구현되는 서비스들, 이를테면 제 1 엔티티가 하나 이상의 다른 엔티티들(예컨대, 다른 사용자들)과의 그 트랜잭션들을 관리하기 위해 사용하는 트랜잭션 관리 시스템을 호스팅할 수 있다. 컴퓨팅 시스템(108)은 제 2 엔티티(예컨대, 사용자 B)의 컴퓨터로 구현되는 서비스들, 이를테면 제 2 엔티티가 하나 이상의 다른 엔티티들(예컨대, 다른 사용자들)과의 그 트랜잭션들을 관리하기 위해 사용하는 트랜잭션 관리 시스템을 호스팅할 수 있다. 도 1의 예에서, 공개 블록체인(102)은 노드들의 피어-투-피어 네트워크로서 표현되고, 컴퓨팅 시스템들(106, 108)은 공개 블록체인(102)에 참여하는 제 1 엔티티 및 제 2 엔티티의 노드들을 각각 제공한다.

[0021] 도 2는 본 개시의 구현예들에 따른 예시적인 개념적 아키텍처(200)를 도시한다. 예시적인 개념적 아키텍처(200)는 엔티티 계층(202), 호스트형 서비스 계층(204) 및 공개 블록체인 계층(206)을 포함한다. 도시된 예에서, 엔티티 계층(202)은 3개의 엔티티들인 엔티티_1(E1), 엔티티_2(E2) 및 엔티티_3(E3)을 포함하며, 각각의 엔티티는 각각의 트랜잭션 관리 시스템(208)을 갖는다.

[0022] 도시된 예에서, 호스트형 서비스 계층(204)은 각 트랜잭션 관리 시스템(208)에 대한 블록체인 인터페이스들(210)을 포함한다. 일부 예들에서, 각각의 트랜잭션 관리 시스템(208)은 통신 프로토콜(예컨대, 하이퍼텍스트 전송 프로토콜 보안(HTTPS: hypertext transfer protocol secure))을 사용하여 네트워크(예컨대, 도 1의 네트워크(110))를 통해 각각의 블록체인 인터페이스(210)와 통신한다. 일부 예들에서, 각각의 블록체인 인터페이스(210)는 각각의 트랜잭션 관리 시스템(208)과 블록체인 계층(206) 간의 통신 접속을 제공한다. 보다 구체적으로, 각 블록체인 인터페이스(210)는 각각의 엔티티가 블록체인 계층(206)의 블록체인 네트워크(212)에 기록된 트랜잭션들을 수행할 수 있게 한다. 일부 예들에서, 블록체인 인터페이스(210)와 블록체인 계층(206) 간의 통신은 원격 프로시저 호출들(RPCs: remote procedure calls)을 사용하여 수행된다. 일부 예들에서, 블록체인 인터페이스들(210)은 각각의 트랜잭션 관리 시스템들(208)에 대한 블록체인 노드들(210)을 "호스트"한다. 예를 들어, 블록체인 인터페이스들(210)은 블록체인 네트워크(212)로의 액세스를 위한 애플리케이션 프로그래밍 인터페이스(API: Application Programming Interface)를 제공한다.

[0023] 본 명세서에 설명된 바와 같이, 블록체인 네트워크(212)는 블록체인(216)에 정보를 변경 불가능하게 기록하는 복수의 노드들(214)을 포함하는 피어-투-피어 네트워크로서 제공된다. 단일 블록체인(216)이 개략적으로 도시되어 있지만, 블록체인(216)의 다수의 카피가 제공되며, 블록체인(212)에 걸쳐 유지된다. 예를 들어, 각각의 노드(214)는 블록체인(216)의 카피를 저장한다. 일부 구현예들에서, 블록체인(216)은 공개 블록체인에 참여하는 둘

이상의 엔티티들 사이에서 수행되는 트랜잭션들과 연관된 정보를 저장한다.

[0024] 도 3은 본 개시의 구현예들에 따른, HE에 기초한 블록체인 트랜잭션의 프라이버시 보호형 유효성검사의 예시적인 방법(300)을 도시한다. 하이 레벨에서, 예시적인 방법(300)은 사용자 노드 A(302), 사용자 노드 B(도 3에 도시되지 않음) 및 합의 노드라고도 또한 지칭되는 블록체인 노드(304)에 의해 수행된다. 값의 전송과 같은 트랜잭션은 사용자 노드 A(302)로부터 사용자 노드 B로 이루어질 수 있다. 계좌 프라이버시를 보호하기 위해, 사용자 노드 A(302)는 난수 r 에 기초하여 PC와 같은 커미트먼트 스킴을 사용하여 트랜잭션 양 t 의 커미트먼트를 생성할 수 있다. PC를 사용하여 생성된 커미트먼트는 $PC(r, t)$ 로서 표현될 수 있다. 사용자 노드 A(302)는 또한 사용자 노드 B의 공개 키에 기초한 HE를 사용하여 난수를 암호화할 수 있다. 이것은 $HE(r)$로서 표현될 수 있다. $(PC(r, t), HE(r))$로서 표현된 트랜잭션 양 t 의 암호문(ciphertext)은 사용자 노드 B로 전송될 수 있다. 암호문을 수신한 후, 사용자 노드 B는 개인 키를 사용하여 난수 r을 해독할 수 있다. 사용자 노드 B는 난수 r을 사용하여 트랜잭션 양 t를 해독할 수 있다. 트랜잭션의 유효성을 증명하기 위해, 블록체인 노드(304)는 커미트먼트에서의 난수와, HE를 사용하여 암호화된 난수를 비교할 수 있다. 난수들이 일치하면, 트랜잭션은 트랜잭션 데이터의 영지식(zero-knowledge)을 이용하여 블록체인 노드(304)에 의해 유효한 것으로 결정된다. 예시적인 방법(300)에 대한 보다 상세한 내용은 도 3에 대한 다음의 설명에서 논의된다.

[0025] 306에서, 사용자 노드 A(302)는 제 1 난수에 기초하여 트랜잭션 양의 커미트먼트값을 생성하고, HE, 사용자 노드 A(302)의 공개 키를 사용한 제 2 난수, 그리고 사용자 노드 B의 공개 키를 사용한 제 3 난수에 기초하여 암호화한다. 제 1 난수, 제 2 난수 및 제 3 난수는 커미트먼트 스킴을 사용하여 트랜잭션 양 t 의 커미트먼트를 생성하는데 사용된 동일한 난수 r 일 수 있다. 일부 구현예들에서, 커미트먼트 스킴은 PC와 같은 이중 지수(double exponential) 형태를 가질 수 있다. PC를 비제한적인 예로서 사용하면, 제 1 난수 r 에 의해 생성된 커미트먼트값은 $PC(r, t) = g^{r}h^{t}$ 로서 표현될 수 있으며, 여기서 g 및 h는 타원형 곡선의 생성자(generator)들일 수 있고, $PC(r, t)$는 곡선 점들의 스칼라 곱이고, t는 확정되는 트랜잭션 양이다. Okamoto-Uchiyama(OU) HE 및 Boneh-Goh-Nissim HE와 같은 HE에 기초한 다른 커미트먼트 스킴들도 또한 커미트먼트값을 생성하는데 사용될 수 있음은 이해되는 것이다.

[0026] 사용자 노드 A(302)의 공개 키를 사용하여 암호화된 제 2 난수 r 의 암호화는 $HE_A(r)$로서 표현될 수 있다. 사용자 노드 B의 공개 키를 사용하여 암호화된 제 3 난수 r 의 암호화는 $HE_B(r)$로서 표현될 수 있다.

[0027] 일부 구현예들에서, 공개 키 HE 암호화는, 난수를 고정된 값으로 설정함으로써, Paillier HE, Benaloh HE, OU HE, Naccache-Stern HE, Damgard-Jurik HE 또는 Boneh-Goh-Nissim HE와 같은 확률론적 HE 스킴들로부터 획득될 수 있는 결정론적 HE일 수 있다. 일부 구현예들에서, $HE(a + b) = HE(a) + HE(b)$ 및 $HE(ab) = HE(b)^{a}$ 이되, 여기서 a 및 b 가 HE에 대해 사용되는 플레인텍스트(plaintext)인 선형 속성들을 만족시키는 결정론적 HE 스킴들은 본 개시에 대해 사용될 수 있다.

[0028] 일부 예들에서, $T = PC(r, t)$, $T' = HE\_A(r)$, 그리고, $T'' = HE_B(r)$이며, 트랜잭션 양의 암호문은 (T, T' 및 T") 로 표현될 수 있다. 예시적인 조건들이 충족되면 트랜잭션이 유효한 것으로 결정될 수 있다. 첫째, 트랜잭션 양 t는 0보다 크거나 같고, 사용자 노드 A(302)의 계좌 잔고 s_A 이하이다. 둘째, 트랜잭션이 사용자 노드 A(302)에 의해 승인된 것임을 증명하기 위해 사용자 노드 A(302) 개인 키의 개인 키에 의해 디지털 서명된다. 세째, 커미트먼트 $PC(r, t)$에서의 난수 r은 사용자 노드 A(302) 및 사용자 노드 B(302)의 공개 키들을 사용하여 암호문 $HE\_A(r)$ 및 $HE_B(r)$로 암호화된 r 과 동일하다.

[0029] 일부 구현예들에서, 암호문은 또한 $(PC(r', t'), HE_A(r'))$로서 표현될 수 있는 전송된 양(t")의 암호문, 그리고 $(PC(r", t"), HE_B(r"))$로서 표현될 수 있는 수신된 양(t")의 암호문으로 분리될 수 있다. 그러한 경우들에 있어서, 전송된 양 t'는, 트랜잭션을 유효성검사하기 위해, 수신된 양 t"와 동일하게 결정될 필요가 있다.

[0030] 308에서, 사용자 노드 A(302)는 하나 이상의 범위 증명서를 생성한다. 일부 구현예들에서, 범위 증명서들은 트랜잭션 양 t 가 0보다 크거나 같은 것을 나타내는 범위 증명서 $RP1$, 그리고 트랜잭션 양 t 가 사용자 노드 A의 계좌 잔고 이하임을 나타내는 범위 증명서 $RP2$를 포함할 수 있다.

[0031] 310에서, 사용자 노드 A(302)는 하나 이상의 선택된 난수에 기초한 HE를 사용하여 한 세트의 값들을 생성한다. 한 세트의 값들은 Pf 로 표시되며, 사용자 노드 A(302) 및 사용자 노드 B의 공개 키들을 각각 사용하여, 커미트먼트 $PC(r, t)$에서의 난수 r 이 암호문 $HE\_A(r)$ 및 $HE_B(r)$로 암호화된 r 과 동일함을 증명하는데 사용되는 증명들을 포함할 수 있다. 일부 구현예들에서, 두 개의 난수 $r1$ 및 $t1$은 (T1, T1', T1")로 표시된 $t1$ 의 다른 세

트의 암호문을 계산하도록 선택될 수 있으며, 여기서 T1 = $g^{r1}h^{t1}$, T1' = *HE_A*(*r1*), T1" = *HE_B*(*r1*) 이다. 2개의 부가적인 증명들 *r2* 및 *t2*는 *r2* = *r1* + *xr*, *t2* = *t1* + *xt* 로 계산될 수 있으며, 여기서 *x*는 T1, T1' 및 T1"의 해시(Hash)이다. 한 세트의 값들은 *Pf* = (T1, T1', T1", *r2*, *t2*) 로 나타낼 수 있다.

[0032] 312에서, 사용자 노드 A(302)는 그 개인 키를 사용하여 암호문(T, T', T"), 암호문(T1, T1', T1"), *r2*, *t2*, 범위 증명서들 *RP1* 및 *RP2*, 그리고 사용자 노드 A(302)와 사용자 노드 B의 공개 키들을 디지털 서명한다. 사용자 노드 A(302)에 의해 부가된 디지털 서명은 트랜잭션이 사용자 노드 A(302)에 의해 승인된 것을 나타내는 데 사용될 수 있다. 디지털 서명된 카피는 314에서 블록체인 네트워크에 제출된다.

[0033] 316에서, 블록체인 노드(304)는 사용자 노드 A(302)의 공개 키를 사용하여 디지털 서명을 검증한다. 블록체인 노드(304)는 블록체인 네트워크에서 트랜잭션의 유효성을 증명할 수 있는 합의 노드일 수 있다. 블록체인 노드(304)가 공개 키를 사용하여 사용자 노드 A(302)의 디지털 서명을 검증할 수 없다면, 디지털 서명은 부정확한 것으로 결정될 수 있고, 트랜잭션은 거부될 수 있다. 일부 구현예들에서, 블록체인 노드(304)는 또한 안티 이중 지출(anti-double spending) 메커니즘을 포함할 수 있다. 블록체인 노드(304)는 트랜잭션이 이미 실행되었거나 기록되었는지의 여부를 검증할 수 있다. 트랜잭션이 이미 실행되었다면 트랜잭션은 거부될 수 있다. 그렇지 않다면, 트랜잭션의 유효성검사가 진행될 수 있다.

[0034] 318에서, 블록체인 노드(304)는 하나 이상의 범위 증명서를 검증한다. 예를 들어, 트랜잭션 양 *t* 가 0보다 크거나 같음을 증명하는데 범위 증명서 *RP1* 이 사용될 수 있고, 트랜잭션 양 *t* 가 사용자 노드 A(302)의 계좌 잔고 이하임을 증명하는데 범위 증명서 *RP2* 가 사용될 수 있다.

[0035] 320에서, 블록체인 노드(304)는 제 1 난수, 제 2 난수 및 제 3 난수가 한 세트의 값들에 기초하여 동일하다고 결정한다. 일부 구현예들에서는, 결정은 전술한 바와 같이, 결정론적 HE의 속성들에 기초하여 예시적인 조건들 $g^{r2}h^{t2}$ = T^{x}T1, *HE_A*(*r2*) = T'^{x}T1' 및 *HE_B*(*r2*) = T''^{x}T1" 이 참인지의 여부를 결정하는 것을 포함한다. 참이면, 커미트먼트에서의 난수가, 사용자 노드 A(302) 및 사용자 노드 B의 공개 키들을 사용하여 준동형으로(homomorphically) 암호화된 난수와 동일하고 트랜잭션이 유효함을 나타낼 수 있다.

[0036] 322에서, 블록체인 노드(304)는 사용자 노드 A(302) 및 사용자 노드 B의 계좌 잔고들을 업데이트한다. 잔고 업데이트들은 사용자 노드 A(302) 또는 사용자 노드 B 중 어느 일방의 계좌 잔고들을 드러내지 않고서도 HE의 속성들에 기초하여 수행될 수 있다. 계좌 잔고들을 업데이트하는 것은 도 4를 참조하여 여기에서 더 상세히 설명된다.

[0037] 도 4는 본 개시의 구현예들에 따른, HE에 기초한 예시적인 블록체인 트랜잭션(400)을 도시한다. 예시적인 블록체인 트랜잭션(400)에 나타낸 바와 같이, 사용자 노드 A(402)는 트랜잭션 양 *t*를 사용자 노드 B(406)로 전송한다. 트랜잭션 이전에, 사용자 노드 A(402)는 s_A의 계좌 잔고를 가지며, 사용자 노드 B(406)는 s_B의 계좌 잔고를 갖는다.

[0038] 일 예로서, 도 3을 참조하여 여기에서 설명된 암호화 스킴들 및 트랜잭션 프로세스를 사용하여, 계좌 잔고 *s_A*는 PC에 기초한 난수 *r_A*를 사용하여 암호화될 수 있고, 난수 *r_A*는 HE에 기초하여 암호화될 수 있다. 계좌 잔고 s_A의 암호문은 (S_A, S'_A) = ($g^{r\_A}h^{s\_A}$, HE_A(*r_A*))로서 표현될 수 있으며, 여기서 g와 h는 계좌 잔고 *s_A*의 PC를 생성하는 타원형 곡선의 생성자들일 수 있다. 유사하게, 사용자 노드 B(406)의 계좌 잔고 *s_B*는 PC에 기초한 난수 *r_B*를 사용하여 암호화될 수 있다. 계좌 잔고 *s_B*의 암호문은 (S_B, S'_B) = ($g^{r\_B}h^{s\_B}$, HE_A(*r_B*))로서 표현될 수 있다.

[0039] 404에서, 사용자 노드 A(402)는 트랜잭션을 유효성검사하는데 사용되는 증명들에 디지털 서명을 부가할 수 있고, 디지털 서명된 카피를 블록체인 네트워크(408)에 제출할 수 있다. 도 3을 참조하여 전술한 바와 같이, 증명들은 트랜잭션 양(T, T', T")의 암호문, 하나 이상의 범위 증명서들(*RP1*, *RP2*) 및 다른 증명들(T1, T1', T1", *r2*, *t2*)을 포함할 수 있다.

[0040] 트랜잭션 이후에, 사용자 노드 A(402)의 계좌 잔고는 *s_A* - *t*'로서 표현될 수 있고, 사용자 노드 B(406)의 계좌 잔고는 *s_B* + *t*"로서 표현될 수 있으며, 여기서 *t*'는 사용자 노드 A(402)에 의해 전송된 양이고 t"는 사용자 노드 B에 의해 수신된 양이다. 트랜잭션 이후의 사용자 노드 A(402)의 계좌 잔고의 암호문은 (S_A / T, S'_A / T') 로서 표현될 수 있고 트랜잭션 이후에 사용자 노드 B(406)의 계좌 잔고의 암호문은 (S_B * T, S'_B * T") 로서 표현될 수 있다. S_A, S'_A, S_B, S'_B, T, T', T"는 이중 지수 형식를 갖는 HE를 사용하여 각각 암호화

되므로, 플레인텍스트 값들을 해독하지 않고서도 이들의 암호화된 형태로 가산 및 감산이 수행될 수 있다.

[0041] 도 5은 본 개시의 구현예들에 따른, HE에 기초한 블록체인 트랜잭션의 프라이버시 보호형 유효성검사의 다른 예시적인 방법(500)을 도시한다. 하이 레벨에서, 예시적인 방법(500)은 사용자 노드 A(502), 사용자 노드 B(도 5에 도시되지 않음) 및 합의 노드로서 지칭될 수 있는 블록체인 노드(504)에 의해 수행된다. 값의 전송과 같은 트랜잭션은 사용자 노드 A(502)로부터 사용자 노드 B로 이루어질 수 있다. 계좌 프라이버시를 보호하기 위해, 사용자 노드 A(502)는 난수 r 에 기초하여 PC와 같은 커미트먼트 스킴을 사용하여 트랜잭션 양 t 의 커미트먼트를 생성할 수 있다. PC를 사용하여 생성된 커미트먼트는 $PC(r, t)$로 표현될 수 있다. 사용자 노드 A(502)는 또한 OU와 같은 이중 지수 형태를 갖는 HE를 사용하여 트랜잭션 양 t 및 난수 r을 암호화할 수 있다.

[0042] 트랜잭션 양 t의 암호문은 블록체인 네트워크에 제출될 수 있다. 블록체인 노드(504)는, 암호문을 수신한 후, 사용자 노드 A(502) 및 사용자 노드 B의 공개 키들을 각각 사용하여 PC로 숨겨진 난수 r 이 OU로 암호화된 난수 r 과 일치하는지의 여부를 결정할 수 있다. 게다가, 블록체인 노드(504)는 사용자 노드 A(502) 및 사용자 노드 B의 공개 키들을 사용하여 PC로 숨겨진 트랜잭션 양 t 가 OU로 암호화된 트랜잭션 양 t 와 일치하는지의 여부를 결정할 수 있다. 난수들과 트랜잭션 양들 양방 모두가 일치하면, 트랜잭션 데이터의 영지식을 이용하여 블록체인 노드(504)에 의해 트랜잭션이 유효한 것으로 결정될 수 있다.

[0043] 506에서, 사용자 노드 A(502)는 제 1 난수에 기초하여 제 1 트랜잭션 양의 커미트먼트값을 생성하고, 제 1 트랜잭션 양 및 제 1 난수는 사용자 노드 A(502)의 공개 키를 사용하여 암호화된다. 제 2 트랜잭션 양 및 제 2 난수는 사용자 노드 B의 공개 키를 사용하여 암호화된다. 제 1 트랜잭션 양 및 제 2 트랜잭션 양은 동일한 양 t 일 수 있다. 제 1 난수 및 제 2 난수는 커미트먼트 스킴을 사용하여 트랜잭션 양 t 의 커미트먼트를 생성하는데 사용된 동일한 난수 r 일 수 있다. 일부 구현예들에서, 커미트먼트 스킴은 PC와 같은 이중 지수 형태를 가질 수 있다. 일 예로서 PC를 사용하면, 제 1 난수 r 에 의해 생성된 커미트먼트값은 $PC(r, t) = g^r h^t$ 로서 표현될 수 있으며, 여기서 g 와 h는 타원형 곡선의 생성자들이 될 수 있고, $PC(r, t)$는 곡선 점들의 스칼라 곱이고, t는 확정되는 트랜잭션 양이다. OU HE 및 Boneh-Goh-Nissim HE와 같은 HE에 기초한 다른 커미트먼트 스킴들도 또한 커미트먼트값을 생성하는데 사용될 수 있음은 이해되는 것이다.

[0044] 사용자 노드 A(502)는 또한 사용자 노드 A(502)의 공개 키를 사용하여 제 1 난수 및 제 1 트랜잭션 양을 암호화할 수 있고, 사용자 노드 B의 공개 키를 사용하여 제 2 난수 및 제 2 트랜잭션 양을 암호화할 수 있다. 일부 구현예들에서, 난수들 및 트랜잭션 양들의 암호화는 OU와 같은 확률론적 HE에 기초할 수 있다. 일 예로서 OU를 사용하면, 사용자 노드 A(502)의 공개 키를 사용하는 제 1 난수 및 제 1 트랜잭션 양의 암호화는 각각 $OU\_A(r) = u1^r v1^{y1}$ 및 $OU\_A(t) = u1^t v1^{y2}$ 로서 표현될 수 있으며, 여기서 $u1$ 및 $v1$은 타원형 곡선 상에서의 생성자들이고, $y1$ 및 $y2$는 $OU\_A(r)$ 및 $OU_A(t)$를 생성하는데 사용되는 난수들이다. 암호화된 제 2 난수 및 제 2 트랜잭션 양은 각각 $OU\_B(r) = u2^r v2^{z1}$ 및 $OU\_B(t) = u2^t v2^{z2}$ 로서 표현될 수 있으며, 여기서 $u2$ 및 $v2$는 타원형 곡선 상에서의 생성자들이고, $z1$ 및 $z2$는 각각 $OU\_B(r)$ 및 $OU_B(t)$를 생성하는데 사용되는 난수들이다. 확률론적 OU는 $OU(a + b) = OU(a) * OU(b)$라는 속성을 만족시키며, 여기서 a 및 b는 OU에 사용되는 플레인텍스트이다.

[0045] 트랜잭션 양 t 의 암호문은 $(PC(r, t), OU\_A(r), OU\_A(t), OU\_B(r), OU\_B(t))$ 로서 표현될 수 있다. 다음의 예시적인 조건들이 충족되면 트랜잭션이 유효한 것으로 결정될 수 있다. 첫째, 트랜잭션 양 t는 0보다 크거나 같고, 사용자 노드 A(502)의 계좌 잔고 s_A 이하이다. 둘째, 트랜잭션이 사용자 노드 A(502)에 의해 승인된 것임을 증명하기 위해 사용자 노드 A(502) 개인 키의 개인 키를 사용하여 디지털 서명된다. 세째, 커미트먼트 $PC(r, t)$에서의 난수 r은 사용자 노드 A(502) 사용자 노드 B(302)의 공개 키들을 사용하여 암호문 $OU\_A(r)$ 및 $OU_B(r)$로 암호화된 r 과 각각 동일하다. 네째, 커미트먼트 $PC(r, t)$에서의 트랜잭션 양은 각각 사용자 노드 A(502) 및 사용자 노드 B의 공개 키들을 사용하여 암호문 $OU\_A(t)$ 및 $OU\_B(t)$ 로 암호화된 t 와 각각 동일하다.

[0046] 일부 구현예들에서, 암호문은 또한 $(PC(r', t'), OU\_A(r'), OU\_A(t'))$ 으로서 표현될 수 있는 전송된 양(t')의 암호문, 그리고 $(PC(r", t"), OU_B(r"), OU_B(t"))$로서 표현될 수 있는 수신된 양($t"$)의 암호문으로 분리될 수 있다. 그러한 경우들에 있어서, 전송된 양 t'는, 트랜잭션을 유효성검사하기 위해, 수신된 양 $t"$와 동일하게 결정될 필요가 있다.

[0047] 508에서, 사용자 노드 A(502)는 하나 이상의 범위 증명서를 생성한다. 일부 구현예들에서, 범위 증명서들은 트랜잭션 양 t 가 0보다 크거나 같은 것을 나타내는 범위 증명서 $RP1$, 그리고 트랜잭션 양 t 가 사용자 노드 A의 계좌 잔고 이하임을 나타내는 범위 증명서 $RP2$를 포함할 수 있다.

[0048] 510에서, 사용자 노드 A(502)는 하나 이상의 선택된 난수에 기초한 HE를 사용하여 한 세트의 값들을 생성한다. *Pf* 로서 표시된 한 세트의 값들은 커미트먼트 *PC*(*r*, *t*)에서의 난수 *r* 이 암호문 OU_A(*r*) 및 OU_B(*r*)에서 암호화된 r과 동일함을 증명하는데 사용되는 증명들을 포함할 수 있으며 커미트먼트 *PC*(*r*, *t*)에서의 트랜잭션 양 *t*는 암호문 OU_A(*t*) 및 OU_B(*t*)에서 암호화된 *t* 와 동일하다. 일부 구현예들에서, 4개의 난수들 *r**, *t**, *z1** 및 *z2**는 (C, D, E)로서 표시되는 다른 세트의 암호문들을 계산하기 위해 선택될 수 있으며, 여기서 C = $g^{r^*}h^{t^*}$, D = $u2^{r^*}v2^{z1^*}$ 및 E = $u2^{t^*}v2^{z2^*}$이고, 여기서 *g*, *h*, *u2* 및 *v2*는 타원형 곡선의 생성자들이다. 4개의 부가적인 증명서들 *a*, *b*, *c* 및 *d*는 *a* = *r** + *xr*, *b* = *t** + *xt*, *c* = *z1** + *xz1* 및 *d* = *z2** + *xz2* 로서 계산될 수 있으며, 여기서 *x*는 *g*, *h*, *u2*, *v2*, C, D 및 E의 해시 함수이다. 그런 다음, 한 세트의 값들은 *Pf* = (C, D, E, *a*, *b*, *c*, *d*) 로서 나타낼 수 있다.

[0049] 512에서, 사용자 노드 A(502)는 그 개인 키를 사용하여 암호문 (*PC*(*r*, *t*), OU_A(*r*), OU_A(*t*), OU_B(*r*), OU_B(*t*)), 범위 증명서들 *RP1* 및 *RP2*, 그리고 한 세트의 값들 *Pf*를 디지털 서명한다. 사용자 노드 A(502)에 의해 부가된 디지털 서명은 트랜잭션이 사용자 노드 A(502)에 의해 승인된 것을 나타내는 데 사용될 수 있다. 디지털 서명된 카피는 514에서 블록체인 네트워크에 제출된다.

[0050] 516에서, 블록체인 노드(504)는 사용자 노드 A(502)의 공개 키를 사용하여 디지털 서명을 검증한다. 블록체인 노드(504)는 블록체인 네트워크 상에서 트랜잭션의 유효성을 증명할 수 있는 합의 노드일 수 있다. 블록체인 노드(504)가 사용자 노드 A의 공개 키를 사용하여 디지털 서명을 검증할 수 없다면, 디지털 서명은 부정확한 것으로 결정될 수 있고, 트랜잭션은 거부될 수 있다. 일부 구현예들에서, 블록체인 노드(504)는 또한 안티 이중 지출 메커니즘을 포함할 수 있다. 블록체인 노드(504)는 트랜잭션이 이미 실행되었거나 기록되었는지의 여부를 검증할 수 있다. 트랜잭션이 이미 실행되었다면 트랜잭션은 거부될 수 있다. 그렇지 않다면, 트랜잭션의 유효성검사가 진행될 수 있다.

[0051] 518에서, 블록체인 노드(504)는 하나 이상의 범위 증명서를 검증한다. 예를 들어, 트랜잭션 양 *t* 가 0보다 크거나 같음을 증명하는데 범위 증명서 *RP1* 이 사용될 수 있고, 트랜잭션 양 *t* 가 사용자 노드 A(502)의 계좌 잔고 이하임을 증명하는데 범위 증명서 *RP2* 가 사용될 수 있다.

[0052] 520에서, 블록체인 노드(504)는 제 1 트랜잭션 양이 제 2 트랜잭션 양과 동일한지의 여부, 그리고 제 1 난수가 한 세트의 값들에 기초한 제 2 난수와 동일한지의 여부를 결정한다. 일부 구현예들에서, 결정은 $g^a h^b$ = CT^x, $u2^a v2^c$ = DZ_B1^x 및 $u2^b v2^d$ = EZ_B2^x 인지의 여부를 결정하는 것을 포함하며, 여기서 T = $g^r h^t$는 제 1 트랜잭션 양 *t* 의 커미트먼트값이고, Z_B1 = $u2^r v2^{z1}$, Z_B2 = $u2^t v2^{z2}$이며, *z1* 및 *z2*는 확률론적 HE 스킴에 기초하여 제 2 트랜잭션 양 및 제 2 난수를 암호화하는데 사용되는 난수들이다. 참이면, 커미트먼트에서의 난수 및 트랜잭션 양이, 사용자 노드 A(502) 및 사용자 노드 B의 공개 키들을 사용하여 준동형으로 암호화된 난수들 및 트랜잭션 양들과 각각 동일하고 트랜잭션이 유효함을 나타낼 수 있다.

[0053] 522에서, 블록체인 노드(504)는 사용자 노드 A(502) 및 사용자 노드 B의 계좌 잔고들을 업데이트한다. 계좌 잔고 업데이트들은 사용자 노드 A(502) 및/또는 사용자 노드 B의 계좌 잔고들을 드러내지 않고서도 HE의 속성들에 기초하여 수행될 수 있다.

[0054] 도 6는 본 개시의 구현예들에 따른, HE에 기초한 다른 예시적인 블록체인 트랜잭션(600)을 도시한다. 예시적인 트랜잭션(600)에 나타낸 바와 같이, 사용자 노드 A(602)는 트랜잭션 양 *t*를 사용자 노드 B(606)로 전송한다. 트랜잭션 이전에, 사용자 노드 A(602)는 s_A의 계좌 잔고를 가지며, 사용자 노드 B(606)는 s_B의 계좌 잔고를 갖는다.

[0055] 일부 예들에서, 계좌 잔고 s_A는 도 5를 참조하여 본 명세서에 설명된 암호화 스킴들 및 트랜잭션 프로세스를 사용하는 PC 에 기초한 난수 r_A를 사용하여 숨길 수 있다. 난수 r_A 및 계좌 잔고는 OU에 기초하여 암호화될 수 있다. 계좌 잔고 s_A의 암호문은 (S_A, R_A, Q_A) = ($g^{r\_A}h^{s\_A}$, OU_A(*r_A*), OU_A(*s_A*))로서 표현될 수 있으며, 여기서 g와 h는 계좌 잔고 s_A의 Pc를 생성하는 타원형 곡선의 생성자들일 수 있다. 유사하게, 사용자 노드 B(606)의 계좌 잔고 s_B는 PC에 기초한 난수 r_B를 사용하여 암호화될 수 있다. 계좌 잔고 s_B의 암호문은 (S_B, S'_B) = ($g^{r\_B}h^{s\_B}$, OU_B(*r_B*), OU_B(*s_B*))로서 표현될 수 있다.

[0056] 604에서, 사용자 노드 A(602)는 트랜잭션을 유효성검사하는데 사용되는 증명들에 디지털 서명을 부가할 수 있고, 디지털 서명된 카피를 블록체인 네트워크(608)에 제출할 수 있다. 도 5를 참조하여 본 명세서에서 설명된

바와 같이, 증명들은 트랜잭션 양의 암호문 (*PC*(*r*, *t*), OU_A(*r*), OU_A(*t*), OU_B(r), OU_B(*t*)), 하나 이상의 범위 증명서들(*RP1*, *RP2*) 및 다른 증명들(C, D, E, *a*, *b*, *c*, *d*)을 포함할 수 있다.

[0057] 트랜잭션 이후, 사용자 노드 A(602)의 계좌 잔고는 *s_A* - *t* 로서 표현될 수 있고, 사용자 노드 B(606)의 계좌 잔고는 *s_B* + *t* 로서 표현될 수 있다. 트랜잭션 이후의 사용자 노드 A(602)의 계좌 잔고의 암호문은 (S_A / T, R_A / Y_A1, Q_A / Y_A2)로서 표현될 수 있으며, 여기서 Y_A1 = OU_A(*r*) 및 Y_A2 = OU_A(*t*) 이다. 트랜잭션 이후의 사용자 노드 B(606)의 계좌 잔고의 암호문은 (S_B * T, R_B * Z_B1, Q_B * Z_B2)로서 표현될 수 있으며, 여기서 Z_B1 = OU_B(*r*) 및 Z_B2 = OU_B(*t*)이다. S_A, S_B, R_A, R_B, Q_A, Q_B, Y_A1, Y_A2, Z_B1, Z_B2 및 T는 이중 지수 형식을 갖는 HE를 사용하여 암호화되므로, 플레인텍스트 값들을 해독하지 않고서도 이들의 암호화된 형태로 가산 및 감산이 수행될 수 있다.

[0058] 도 7은 본 개시의 구현예들에 따라 실행될 수 있는 예시적인 프로세스(700)를 도시한다. 설명의 명확성을 위해, 이하의 설명은 이 설명에서 다른 도면들의 맥락에서 방법(700)을 일반적으로 설명한다. 하지만, 예시적인 프로세스(700)는 예를 들어 임의의 시스템, 환경, 소프트웨어 및 하드웨어, 또는 시스템, 환경, 소프트웨어 및 하드웨어의 조합에 의해 적절하게 수행될 수 있다는 것은 이해될 것이다. 일부 구현예들에서, 예시적인 프로세스(700)의 단계들은 병렬로, 조합으로, 루프(loop)로, 또는 임의의 순서로 실행될 수 있다.

[0059] 702에서, 합의 노드는 제 1 난수에 기초하여 생성되고 제 1 계좌로부터 제 2 계좌로 전송될 트랜잭션 양의 커미트먼트값의 디지털 서명된 카피를 제 1 계좌로부터 수신한다. 합의 노드는 또한 제 1 계좌의 공개 키를 사용하여 암호화된 제 2 난수, 제 2 계좌의 공개 키를 사용하여 암호화된 제 3 난수, 하나 이상의 범위 증명서 및 하나 이상의 선택된 난수에 기초한 HE를 사용하여 생성된 한 세트의 값들을 제 1 계좌로부터 수신할 수 있다. 일부 구현예들에서, 커미트먼트값은 HE 기반의 커미트먼트 스킴을 사용하여 생성된다. 일부 구현예들에서, 제 2 난수 및 제 3 난수는 결정론적 HE 스킴에 기초하여 암호화된다.

[0060] 일부 구현예들에서, 한 세트의 값들은 (T1, T1', T1", *r2*, *t2*)로 표현되며, 여기서 *r2* = *r1* + *xr*, *t2* = *t1* + *xt*이고, *r1* 및 *t1*은 하나 이상의 선택된 난수를 나타내며, *r*은 제 1 난수를 나타내고, *t*는 잔고 전송의 양을 나타낸다. 일부 예들에서, T1 = $g^{r1}h^{t1}$, T1' = HE_A(*r1*), T1" = HE_B(*r1*)이며, 여기서 *g* 및 *h*는 타원형 곡선의 생성자들이고, HE_A(*r1*)는 제 1 계좌의 공개 키를 사용하여 *r1* 의 HE에 기초하여 생성되고, HE_B(*r1*)은 제 2 계좌의 공개 키를 사용하여 *r1* 의 HE에 기초하여 생성된다. 일부 예들에서 *x*는 T1, T1' 및 T"를 해싱하는 것에 기초하여 생성된다.

[0061] 704에서, 합의 노드는 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 제 1 계좌의 공개 키를 사용하여 디지털 서명된 카피에 대응하는 디지털 서명을 검증한다.

[0062] 706에서, 합의 노드는 하나 이상의 범위 증명서가 잔고 전송의 양이 0보다 크고 제 1 계좌의 잔고 이하임을 증명하는지의 여부를 결정한다.

[0063] 708에서, 합의 노드는 제 1 난수, 제 2 난수 및 제 3 난수가 한 세트의 값들에 기초하여 동일한지의 여부를 결정한다. 일부 구현예들에서, $g^{r2}h^{t2}$ = T^{x}T1, HE_A(*r2*) = T'^{x}T1' 및 HE_B(*r2*) = $T"^{x}$T1"이면, 제 1 난수, 제 2 난수 및 제 3 난수는 동일한 것으로 결정되며, 여기서 T = $g^{r}h^{t}$는 잔고 전송의 양의 커미트먼트값이고, T'= HE_A(*r*)이고, T" = HE_B(*r*)이며, HE_A(*r*)는 제 1 계좌의 공개 키를 사용한 r의 HE에 기초하여 생성되고, HE_B(*r*)는 제 2 계좌의 공개 키를 사용한 *r* 의 HE에 기초하여 생성되며, HE_A(*r2*)는 제 1 계좌의 공개 키를 사용한 *r2* 의 HE에 기초하여 생성되고, HE_B(*r2*)는 제 2 계좌의 공개 키를 사용한 *r2* 의 HE에 기초하여 생성되며, *x*는 *g*, *h*, T1, T1' 및 T1"를 해싱하는 것에 기초하여 생성된다. 일부 구현예들에서, T, T' 및 T"는 트랜잭션 양 *t* 의 양의 암호문을 형성한다.

[0064] 710에서, 합의 노드는 제 1 난수, 제 2 난수, 및 제 3 난수가 동일하면, 트랜잭션 양에 기초하여 제 2 계좌의 잔고 및 제 1 계좌의 잔고를 업데이트한다. 일부 구현예들에서, 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트하는 것은 HE에 기초하여 수행된다.

[0065] 도 8은 본 개시의 구현예들에 따라 실행될 수 있는 다른 예시적인 프로세스(800)를 도시한다. 설명의 명확성을 위해, 이하의 설명은 이 설명에서 다른 도면들의 맥락에서 예시적인 프로세스(800)를 일반적으로 설명한다. 하지만, 예시적인 프로세스(800)는 예를 들어 임의의 시스템, 환경, 소프트웨어 및 하드웨어, 또는 시스템, 환경, 소프트웨어 및 하드웨어의 조합에 의해 적절하게 수행될 수 있다는 것은 이해될 것이다. 일부 구현예들에서, 예시적인 프로세스(800)의 단계들은 병렬로, 조합으로, 루프로, 또는 임의의 순서로 실행될 수 있다.

[0066] 802에서, 합의 노드는 제 1 계좌로부터 제 2 계좌로의 전송을 위한 제 1 트랜잭션 양의 커미트먼트값의 디지털 서명된 카피를 제 1 계좌로부터 수신한다. 일부 예들에서, 커미트먼트값의 디지털 서명된 카피는 제 1 난수에 기초하여 생성된다. 컨센서스 노드는 또한 제 1 계좌의 공개 키를 사용하여 암호화된 제 1 트랜잭션 양 및 제 1 난수, 제 2 계좌의 공개 키를 사용하여 암호화된 제 2 양의 잔고 전송 및 제 2 난수, 하나 이상의 범위 증명서, 그리고 하나 이상의 선택된 난수에 기초한 HE를 사용하여 생성된 한 세트의 값들을 수신한다. 일부 구현예들에서, 커미트먼트값은 PC 스킴을 사용하여 생성된다. 일부 구현예들에서, 제 1 양의 잔고 전송 및 제 1 난수는 확률론적 HE 알고리즘에 기초한 제 1 계좌의 공개 키를 사용하여 암호화된다. 일부 예들에서, 제 2 양의 잔고 전송 및 제 2 난수는 확률론적 HE 알고리즘에 기초한 제 2 계좌의 공개 키를 사용하여 암호화된다. 일부 구현예들에서, 확률론적 HE 알고리즘은 Okamoto-Uchiyama HE 알고리즘이다.

[0067] 일부 구현예들에서, 한 세트의 값들은 (C, D, E, a, b, c, d)로 표시되며, 여기서 $a = r^* + xr$, $b = t^* + xt$, $c = z1^* + xz1$ 및 $d = z2^* + xz2$이고, 여기서 r^*, t^*, $z1^*$ 및 $z2^*$는 하나 이상의 선택된 난수를 나타내며, r은 제 1 난수를 나타내고, t는 제 1 양의 잔고 전송을 나타내고, $C = g^{r^*}h^{t^*}$, $D = u2^{r^*}v2^{z1^*}$, $E = u2^{t^*}v2^{z2^*}$, g, h, $u2$ 및 $v2$는 타원형 곡선의 생성자들이며, x는 C , D 및 E를 해싱하는 것에 기초하여 생성된다.

[0068] 804에서, 합의 노드는 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 제 1 계좌의 공개 키를 사용하여 디지털 서명된 카피에 대응하는 디지털 서명을 검증한다.

[0069] 806에서, 합의 노드는 하나 이상의 범위 증명서가 잔고 전송의 양이 0보다 크고 제 1 계좌의 잔고 이하임을 증명하는지의 여부를 결정한다.

[0070] 808에서, 합의 노드는 제 1 양이 제 2 양과 동일한지의 여부 및 제 1 난수와 제 2 난수가 한 세트의 값들에 기초하여 동일한지의 여부를 결정한다. 일부 구현예들에서, $g^ah^b = CT^x$, $u2^av2^c = DZ\_B1^x$ 및 $u2^bv2^d = EZ_B2^x$이면, 제 1 양과 제 2 양은 동일한 것으로 결정되고, 제 1 난수 및 제 2 난수는 동일한 것으로 결정되며, 여기서 $T = g^rh^t$는 잔고 전송의 양의 커미트먼트값이고, $Z\_B1 = u2^rv2^{z1}$, $Z\_B2 = u2^tv2^{z2}$ 이다. 일부 예들에서, $z1$ 및 $z2$는 확률론적 HE 스킴에 기초하여 제 2 트랜잭션 양 및 제 2 난수를 암호화하는데 사용되는 난수이다.

[0071] 810에서, 합의 노드는, 제 1 양과 제 2 양이 동일하고, 제 1 난수와 제 2 난수가 동일하다면, 제 1 양의 잔고 전송에 기초하여 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트한다. 일부 구현예들에서, 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트하는 것은 HE에 기초하여 수행된다.

[0072] 본 명세서에서 설명된 주제의 구현예들은 특정한 이점들 또는 기술적 효과들을 실현하도록 구현될 수 있다. 예를 들어, 본 개시의 구현예들은 블록체인 노드들의 계좌 잔고 및 트랜잭션 양이 트랜잭션 동안 비공개 상태로 되도록 허용한다. 자금 전송의 수신자는 트랜잭션을 확인하거나 커미트먼트를 검증하기 위한 난수를 사용할 필요가 없으며, 트랜잭션 유효성검사는 비-상호작용식(non-interactive)일 수 있다. 블록체인 노드는 HE 및 커미트먼트 스킴들에 기초하여 트랜잭션을 유효성검사하여 영지식 증명을 허용할 수 있다.

[0073] 설명된 방법론은 다양한 모바일 컴퓨팅 디바이스의 계좌/데이터 보안의 향상을 허용한다. 계좌들 및 트랜잭션 양들의 잔고는 HE에 기초하여 암호화될 수 있으며 커미트먼트 스킴들에 의해 숨겨진다. 이와 같이, 합의 노드는 계좌의 실제 계좌 잔고를 드러내지 않으면서 HE의 속성들에 기초한 트랜잭션 이후에 원장에서의 계좌 잔고들을 업데이트할 수 있다. 트랜잭션을 확인하기 위해 난수가 수신자에게 송신될 필요가 없기 때문에, 데이터 유출의 위험이 감소되고, 난수를 관리하기 위해 사용될 컴퓨팅 및 메모리 리소스들이 덜 필요하게 된다.

[0074] 본 명세서에서 설명된 구현예들 및 동작들은 본 명세서에 개시된 구조들 또는 이들 중 하나 이상의 조합들을 포함하는, 디지털 전자 회로로, 또는 컴퓨터 소프트웨어, 펌웨어 또는 하드웨어로 구현될 수 있다. 동작들은 하나 이상의 컴퓨터 판독가능 저장 디바이스 상에 저장된 데이터 또는 다른 소스들로부터 수신된 데이터에 대해 데이터 처리 장치에 의해 수행되는 동작들로서 구현될 수 있다. 데이터 처리 장치, 컴퓨터 또는 컴퓨팅 디바이스는 데이터를 처리하기 위한 장치, 디바이스들 및 머신들을 망라할 수 있으며, 예를 들어, 프로그래밍가능 프로세서, 컴퓨터, 시스템온칩(system on chip), 또는 이들의 다수의 것들 또는 조합을 포함한다. 장치는 예를 들어, 중앙 처리 유닛(CPU), 필드 프로그래밍가능 게이트 어레이(FPGA) 또는 주문형 집적 회로(ASIC)와 같은 특수 목적용 로직 회로를 포함할 수 있다. 장치는 또한 문제의 컴퓨터 프로그램에 대한 실행 환경을 생성하는 코드, 예를 들어 프로세서 펌웨어, 프로토콜 스택, 데이터베이스 관리 시스템, 운영 체제(예를 들어 일 운영 체제 또는 운영 체제들의 조합), 크로스 플랫폼(cross-platform) 런타임 환경, 가상 머신 또는 이들 중 하나 이상의 조합을 구성하는 코드를 포함할 수 있다. 장치 및 실행 환경은 웹 서비스들, 분산형 컴퓨팅 및 그리드 컴퓨팅

인프라스트럭처(infrastructure)들과 같은 여러 가지의 상이한 컴퓨팅 모델 인프라스트럭처들을 실현할 수 있다.

[0075] 컴퓨터 프로그램(예를 들어, 프로그램, 소프트웨어, 소프트웨어 애플리케이션, 소프트웨어 모듈, 소프트웨어 유닛, 스크립트 또는 코드로도 알려져 있음)은 컴파일러형 또는 해석형 언어들, 선언형 또는 절차형 언어들을 포함하는 임의의 형태의 프로그래밍 언어로 작성될 수 있으며, 스탠드얼론(stand-alone) 프로그램이나 모듈, 컴포넌트, 서브루틴, 객체 또는 컴퓨팅 환경에서 사용하기에 적합한 다른 유닛을 포함하는 임의의 형태로 배포될 수 있다. 프로그램은 다른 프로그램들 또는 데이터(예를 들어, 마크업(markup) 언어 문서에 저장된 하나 이상의 스크립트)를 유지하는 파일의 일부에, 문제의 프로그램에 전용되는 단일 파일에, 또는 다수의 조정형(coordinated) 파일들(예를 들어, 하나 이상의 모듈, 하위 프로그램 또는 코드의 일부를 저장한 파일들)에 저장될 수 있다. 컴퓨터 프로그램은 하나의 컴퓨터 상에서, 또는 일 사이트에 위치되거나 다수의 사이트들에 걸쳐 분산되어 통신 네트워크에 의해 상호 접속된 다수의 컴퓨터들 상에서 실행될 수 있다.

[0076] 컴퓨터 프로그램의 실행을 위한 프로세서들은, 예를 들어 범용 및 특수 목적용 마이크로 프로세서들과, 임의의 종류의 디지털 컴퓨터의 임의의 하나 이상의 프로세서 양방 모두를 포함한다. 일반적으로, 프로세서는 판독 전용 메모리 또는 랜덤 액세스 메모리 또는 양방 모두로부터 명령들 및 데이터를 수신할 것이다. 컴퓨터의 필수 요소들은 명령들과, 명령들 및 데이터를 저장하기 위한 하나 이상의 메모리 디바이스에 따라 액션들을 수행하기 위한 프로세서이다. 일반적으로, 컴퓨터는 또한 데이터를 저장하기 위해 하나 이상의 대용량 저장 디바이스들을 포함하거나, 또는 이들 디바이스들로부터 데이터를 수신하거나 이들 디바이스들에 데이터를 전송하거나 또는 양방 모두를 행하도록 동작 가능하게 커플링될 것이다. 컴퓨터는 다른 디바이스, 예를 들어 모바일 디바이스, 개인 휴대 정보 단말기(PDA: personal digital assistant), 게임 콘솔, GPS(Global Positioning System) 수신기, 또는 휴대용 저장 디바이스에 내장될 수 있다. 컴퓨터 프로그램 명령들 및 데이터를 저장하기에 적합한 디바이스들은, 예를 들어 반도체 메모리 디바이스들, 자기 디스크들 및 광 자기 디스크들을 포함하는 비 휘발성 메모리, 매체 및 메모리 디바이스들을 포함한다. 프로세서 및 메모리는 특수 목적용 로직 회로에 의해 보완되거나 이에 통합될 수 있다.

[0077] 모바일 디바이스들은 핸드셋, 사용자 장비(UE), 모바일폰(예를 들어, 스마트폰), 태블릿, 착용형 디바이스(예를 들어, 스마트 시계 및 스마트 안경), 인체 내부의 이식 디바이스(예를 들어, 바이오 센서, 달팽이관 임플란트), 또는 다른 유형들의 모바일 디바이스들을 포함할 수 있다. 모바일 디바이스들은 다양한 통신 네트워크들(이하에서 설명됨)에 무선으로 (예를 들어, 무선 주파수(RF: Radio Frequency) 신호들을 사용하여) 통신할 수 있다. 모바일 디바이스들은 모바일 디바이스의 현재 환경의 특성을 결정하는 센서들을 포함할 수 있다. 센서들은 카메라, 마이크로폰, 근접 센서, GPS 센서, 모션 센서, 가속도계, 주변 광 센서, 습도 센서, 자이로스코프, 나침반, 기압계, 지문 센서, 안면 인식 시스템, RF 센서(예를 들어, Wi-Fi 및 셀룰러 라디오(cellular radio)), 열 센서 또는 기타 유형의 센서를 포함할 수 있다. 예를 들어, 카메라는 이동식 또는 고정식 렌즈를 갖는 전방 또는 후방 카메라, 플래시, 이미지 센서 및 이미지 프로세서를 포함할 수 있다. 카메라는 안면 및/또는 홍채 인식을 위한 세부 내용을 캡처할 수 있는 메가픽셀(megapixel) 카메라일 수 있다. 카메라는 데이터 프로세서와, 메모리에 저장되거나 원격으로 액세스되는 인증 정보와 함께 안면 인식 시스템을 형성할 수 있다. 안면 인식 시스템 또는 마이크로폰, 동작 센서, 가속도계, GPS 센서 또는 RF 센서와 같은 하나 이상의 센서가 사용자 인증에 사용될 수 있다.

[0078] 사용자와의 상호 작용을 제공하기 위해, 구현예들은 디스플레이 디바이스 및 입력 디바이스, 예를 들어 사용자에게 정보를 디스플레이하기 위한 액정 디스플레이(LCD) 또는 유기 발광 다이오드(OLED)/가상 현실(VR)/증강 현실(AR) 디스플레이, 그리고 사용자가 컴퓨터에 입력을 제공할 수 있는 터치스크린, 키보드 및 포인팅 디바이스를 갖는 컴퓨터 상에서 구현될 수 있다. 다른 종류의 디바이스들이 사용자와의 상호 작용을 제공하는데 사용될 수 있는데; 예를 들어, 사용자에게 제공되는 피드백은, 예를 들어 시각 피드백, 청각 피드백 또는 촉각 피드백과 같은 임의의 형태의 감각 피드백일 수 있으며; 사용자로부터의 입력은 음향, 음성 또는 촉각 입력을 포함하는 임의의 형태로 수신될 수 있다. 또한, 컴퓨터는 문서들을 사용자에 의해 사용되는 디바이스와 송신 및 수신으로써; 예를 들어, 웹 브라우저로부터 수신된 요청들에 응답하여 사용자의 클라이언트 디바이스 상의 웹 브라우저에 웹 페이지들을 송신함으로써 사용자와 상호 작용할 수 있다.

[0079] 구현예들은 유선 또는 무선 디지털 데이터 통신(또는 이들의 조합)의 임의의 형태 또는 매체, 예를 들어 통신 네트워크에 의해 상호 접속된 컴퓨팅 디바이스들을 사용하여 구현될 수 있다. 상호 접속된 디바이스의 예들은 통신 네트워크를 통해 통상적으로 상호 작용하는 일반적으로 서로 멀리 떨어진 클라이언트 및 서버이다. 클라이언트, 예를 들어 모바일 디바이스는, 예를 들어 구매, 판매, 지불, 수여, 송신 또는 대여 트랜잭션들을 수행하

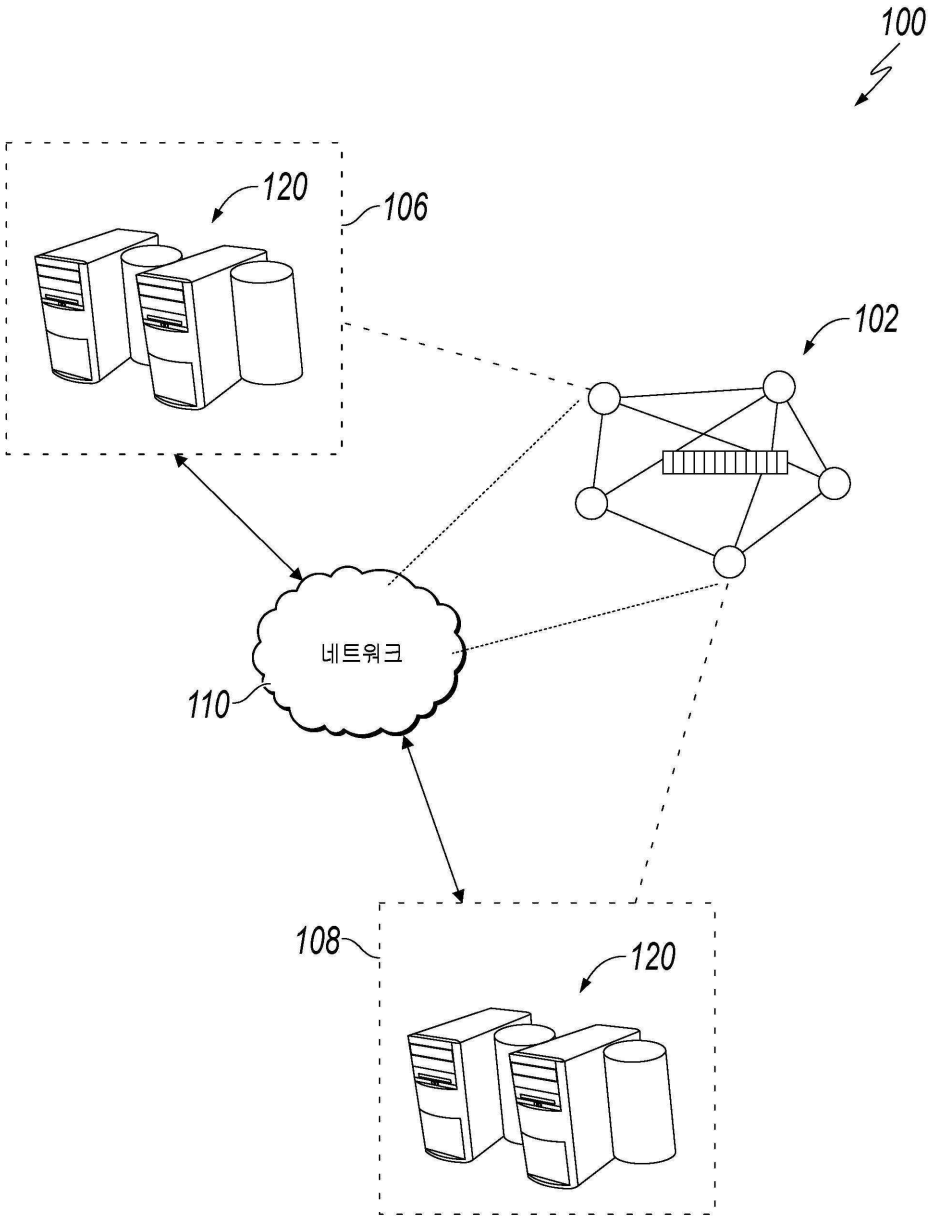
거나 이를 승인하는 서버와 함께 또는 서버를 통해, 트랜잭션 그 자체를 행할 수 있다. 이러한 트랜잭션들은 액션과 응답이 시간적으로 근접하도록 실시간으로 이루어질 수 있는데; 예를 들어 개인이 실질적으로 동시에 발생하는 액션과 응답을 인지하고 개인의 액션에 따르는 응답에 대한 시간차가 1밀리초(ms) 미만 또는 1초 미만이거나, 또는 응답이 시스템의 처리 제한 사항들을 고려한 의도적 지연을 갖지 않는다.

[0080] 통신 네트워크들의 예들은 LAN(Local Area Network), RAN(Radio Access Network), MAN(Metropolitan Area Network) 및 WAN(Wide Area Network)을 포함한다. 통신 네트워크는 인터넷의 전부 또는 일부, 다른 통신 네트워크 또는 통신 네트워크들의 조합을 포함할 수 있다. 정보는 LTE(Long Term Evolution), 5G, IEEE 802, 인터넷 프로토콜(IP) 또는 다른 프로토콜들 또는 프로토콜들의 조합을 포함하는 다양한 프로토콜들 및 표준들에 따라 통신 네트워크 상에서 전송될 수 있다. 통신 네트워크는 접속된 컴퓨팅 디바이스들 사이에서 음성, 비디오, 생체 인식 또는 인증 데이터 또는 기타 정보를 전송할 수 있다.

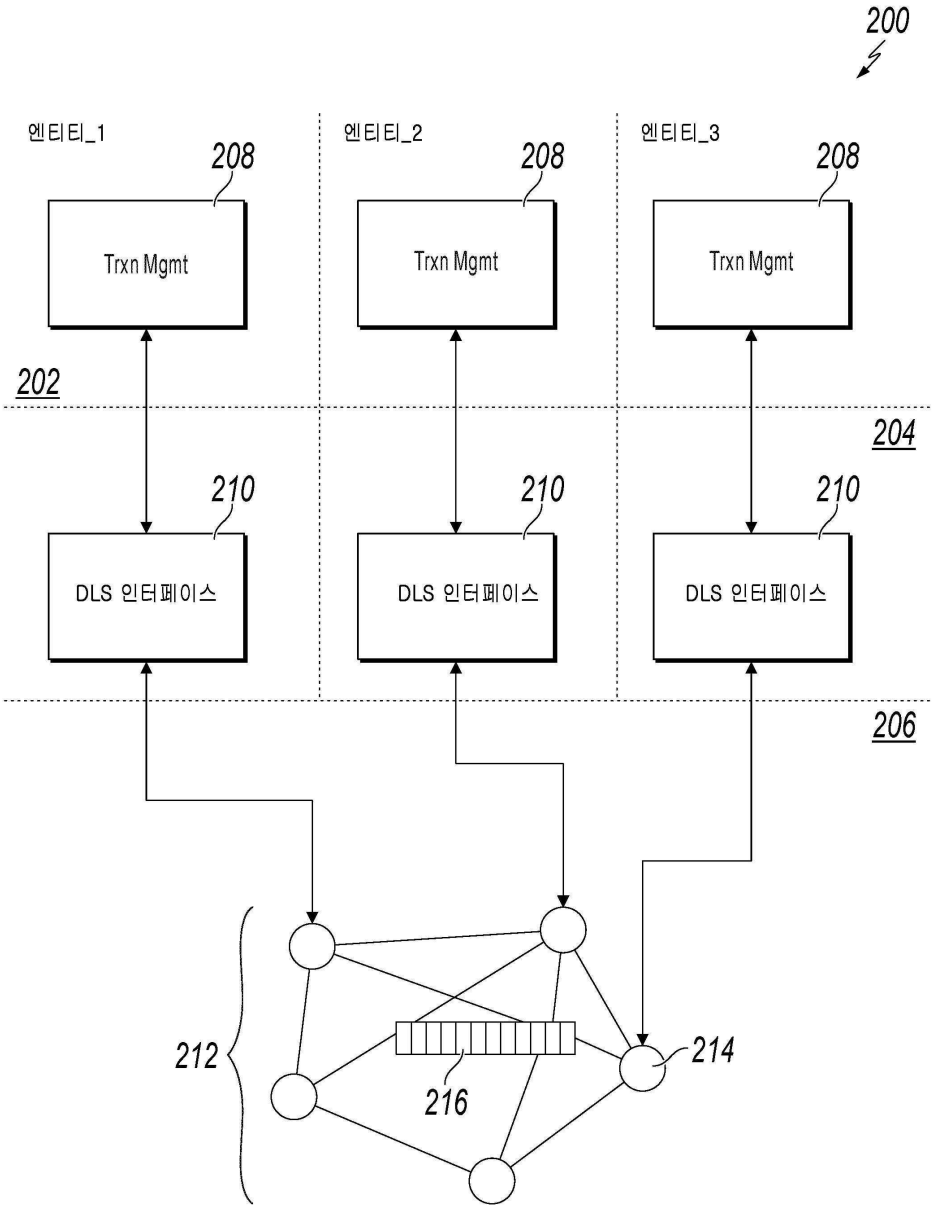
[0081] 개별 구현예들로서 설명된 특징들은, 조합으로, 단일 구현예로 구현될 수 있는 반면, 단일 구현예로서 설명된 특징들은 다수의 구현예들, 개별적으로 또는 임의의 적합한 하위 조합으로 구현될 수 있다. 특정 순서로 설명되고 주장된 동작들은 특정 순서가 수행되어야 한다거나 모든 예시된 동작들이 수행되어야 한다는 것을 요구하는 것으로 이해되어서는 아니된다(일부 동작들은 선택 사항일 수 있음). 필요에 따라, 다중 작업(multitasking) 또는 병렬 처리(또는 다중 작업과 병렬 처리의 조합)가 수행될 수 있다.

도면

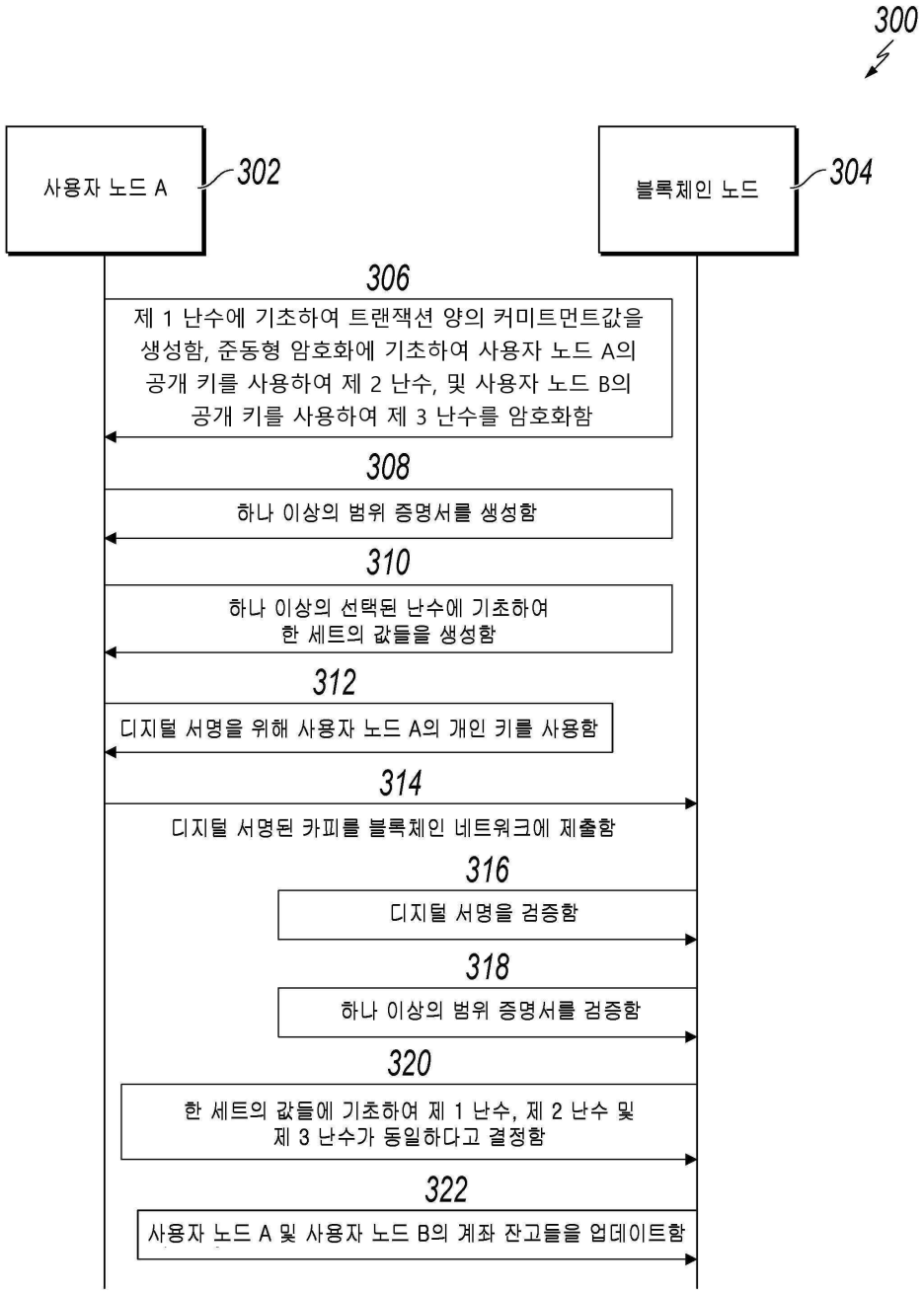
도면1



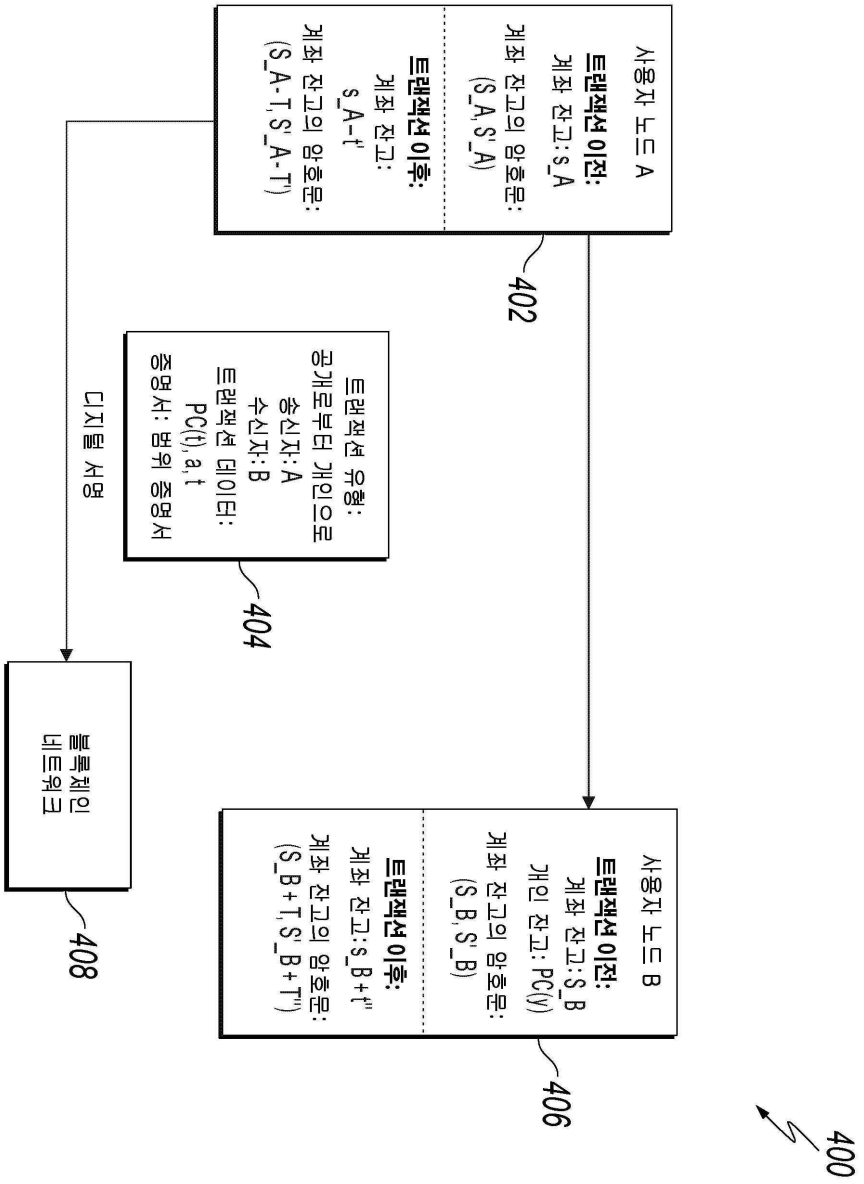
도면2


200
엔티티_1
엔티티_2
엔티티_3
208
208
208
Trxn Mgmt
Trxn Mgmt
Trxn Mgmt
202
204
210
210
210
DLS 인터페이스
DLS 인터페이스
DLS 인터페이스
206
212
214
216

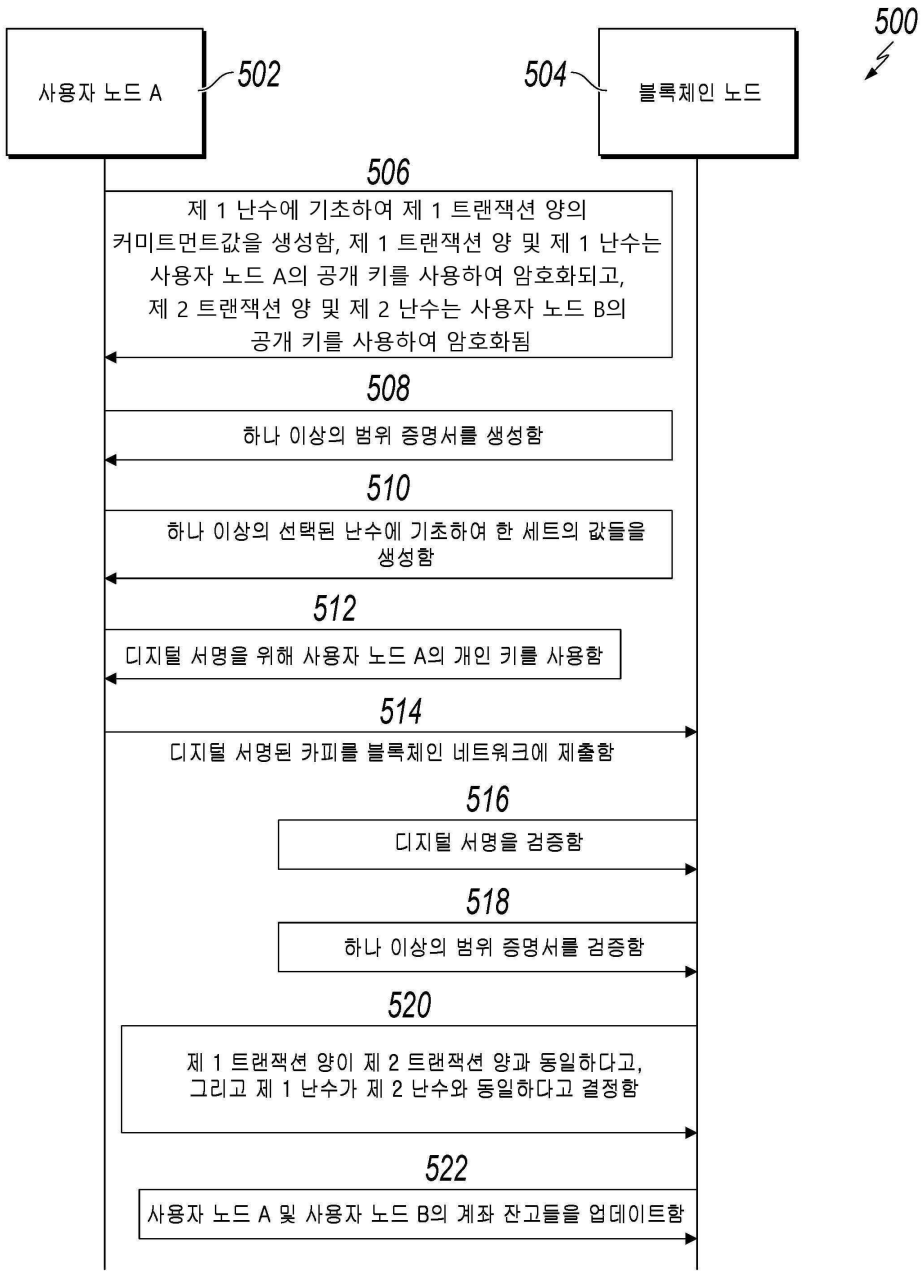
도면3


300
사용자 노드 A
302
블록체인 노드
304
306
제 1 난수에 기초하여 트랜잭션 양의 커미트먼트값을 생성함, 준동형 암호화에 기초하여 사용자 노드 A의 공개 키를 사용하여 제 2 난수, 및 사용자 노드 B의 공개 키를 사용하여 제 3 난수를 암호화함
308
하나 이상의 범위 증명서를 생성함
310
하나 이상의 선택된 난수에 기초하여 한 세트의 값들을 생성함
312
디지털 서명을 위해 사용자 노드 A의 개인 키를 사용함
314
디지털 서명된 카피를 블록체인 네트워크에 제출함
316
디지털 서명을 검증함
318
하나 이상의 범위 증명서를 검증함
320
한 세트의 값들에 기초하여 제 1 난수, 제 2 난수 및 제 3 난수가 동일하다고 결정함
322
사용자 노드 A 및 사용자 노드 B의 계좌 잔고들을 업데이트함

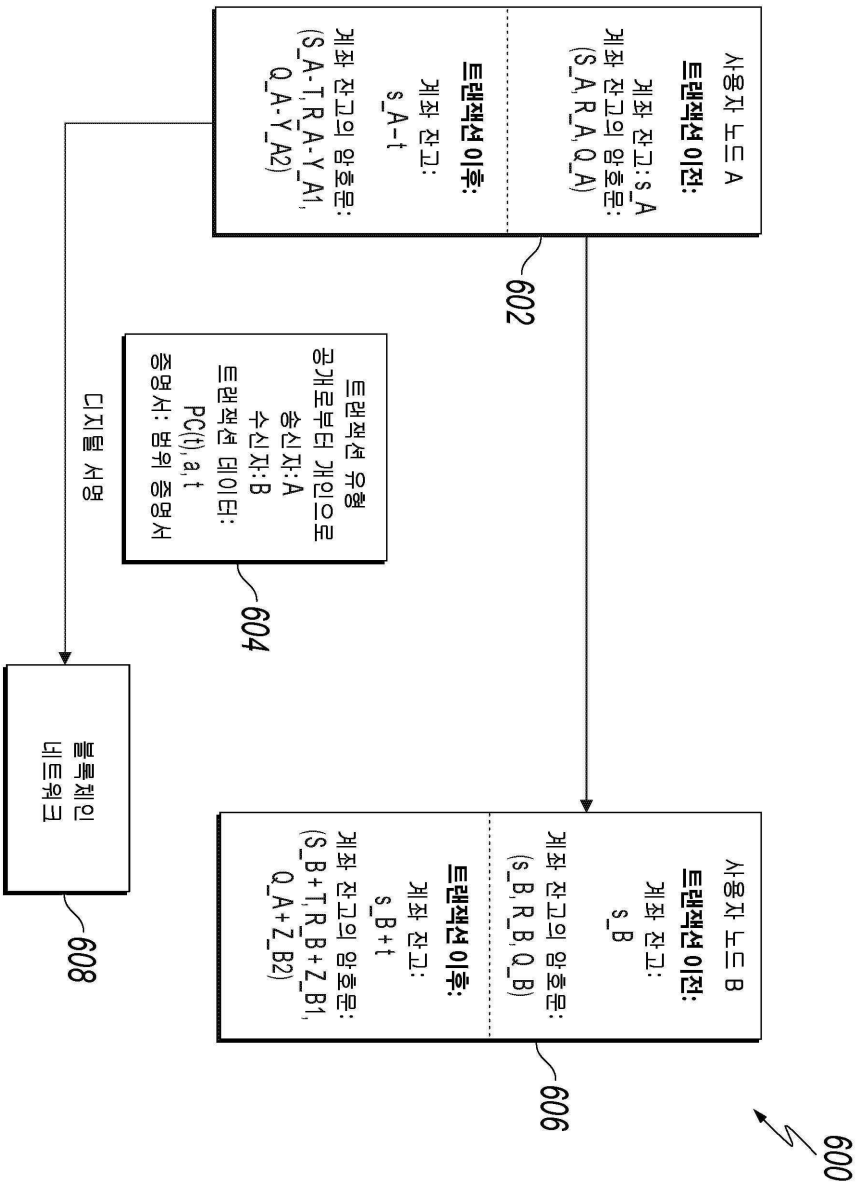
도면4


사용자 노드 A
트랜잭션 이전:
계좌 잔고: s_A
계좌 잔고의 암호문:
(S_A, S'_A)
트랜잭션 이후:
계좌 잔고:
s_A-t'
계좌 잔고의 암호문:
(S_A-T, S'_A-T')
402
트랜잭션 유형:
공개로부터 개인으로
송신자: A
수신자: B
트랜잭션 데이터:
PC(t), a, t
증명서: 범위 증명서
404
디지털 서명
블록체인
네트워크
408
사용자 노드 B
트랜잭션 이전:
계좌 잔고: S_B
개인 잔고: PC(y)
계좌 잔고의 암호문:
(S_B, S'_B)
트랜잭션 이후:
계좌 잔고: s_B+t''
계좌 잔고의 암호문:
(S_B+T, S'_B+T'')
406
400

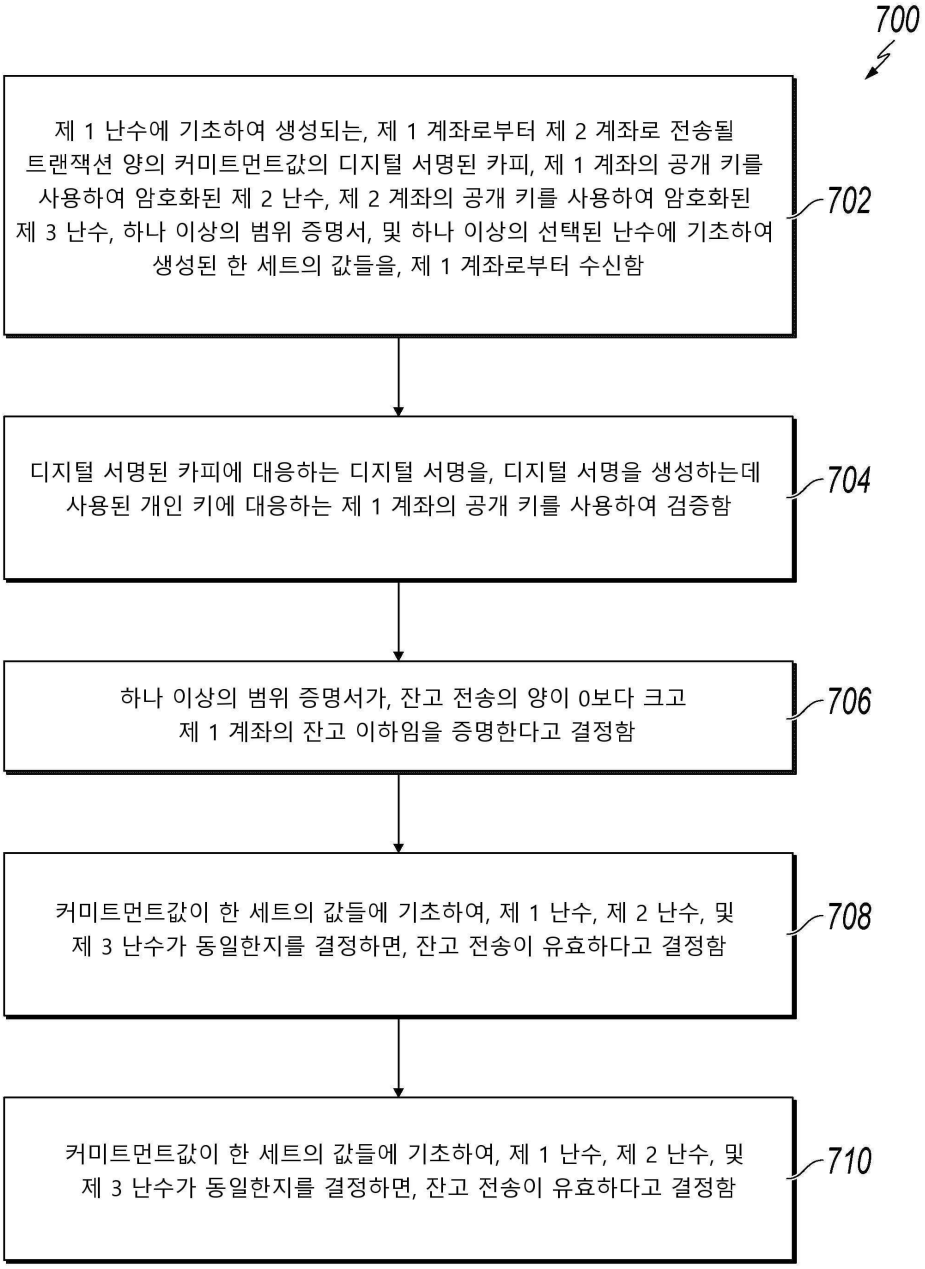
도면5


500
사용자 노드 A
502
504
블록체인 노드
506
제 1 난수에 기초하여 제 1 트랜잭션 양의 커미트먼트값을 생성함, 제 1 트랜잭션 양 및 제 1 난수는 사용자 노드 A의 공개 키를 사용하여 암호화되고, 제 2 트랜잭션 양 및 제 2 난수는 사용자 노드 B의 공개 키를 사용하여 암호화됨
508
하나 이상의 범위 증명서를 생성함
510
하나 이상의 선택된 난수에 기초하여 한 세트의 값들을 생성함
512
디지털 서명을 위해 사용자 노드 A의 개인 키를 사용함
514
디지털 서명된 카피를 블록체인 네트워크에 제출함
516
디지털 서명을 검증함
518
하나 이상의 범위 증명서를 검증함
520
제 1 트랜잭션 양이 제 2 트랜잭션 양과 동일하다고, 그리고 제 1 난수가 제 2 난수와 동일하다고 결정함
522
사용자 노드 A 및 사용자 노드 B의 계좌 잔고들을 업데이트함

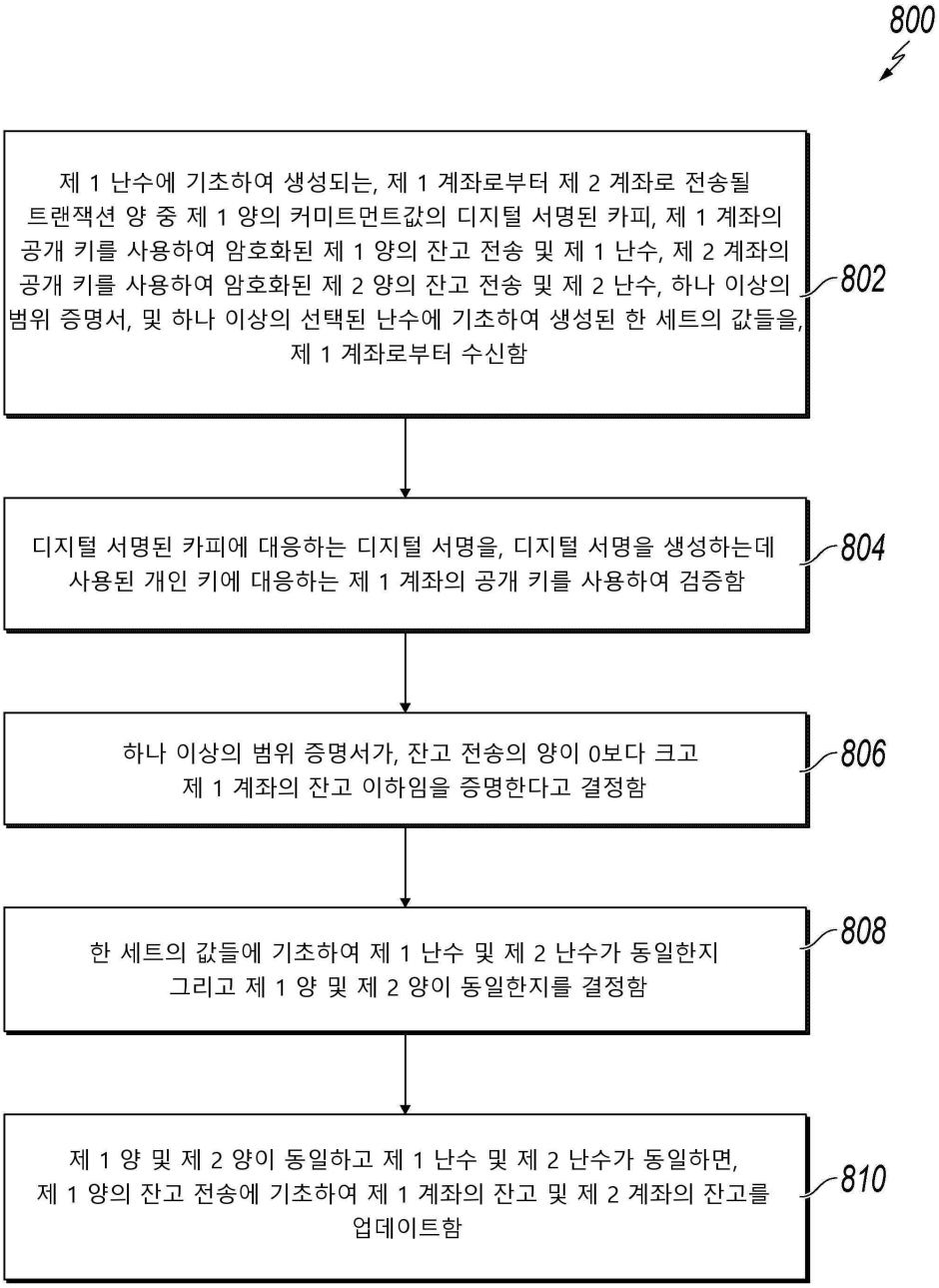
도면6


사용자 노드 A
트랜잭션 이전:
계좌 잔고:s_A
계좌 잔고의 암호문:
(S_A,R_A,Q_A)
트랜잭션 이후:
계좌 잔고:
s_A-t
계좌 잔고의 암호문:
(S_A-T,R_A-Y_A1,
Q_A-Y_A2)
602
트랜잭션 유형
공개로부터 개인으로
송신자:A
수신자:B
트랜잭션 데이터:
PC(t), a, t
증명서: 범위 증명서
604
디지털 서명
블록체인
네트워크
608
사용자 노드 B
트랜잭션 이전:
계좌 잔고:
s_B
계좌 잔고의 암호문:
(s_B,R_B,Q_B)
트랜잭션 이후:
계좌 잔고:
s_B+t
계좌 잔고의 암호문:
(S_B+T,R_B+Z_B1,
Q_A+Z_B2)
606
600

도면7


700
제 1 난수에 기초하여 생성되는, 제 1 계좌로부터 제 2 계좌로 전송될 트랜잭션 양의 커미트먼트값의 디지털 서명된 카피, 제 1 계좌의 공개 키를 사용하여 암호화된 제 2 난수, 제 2 계좌의 공개 키를 사용하여 암호화된 제 3 난수, 하나 이상의 범위 증명서, 및 하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들을, 제 1 계좌로부터 수신함
702
디지털 서명된 카피에 대응하는 디지털 서명을, 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 제 1 계좌의 공개 키를 사용하여 검증함
704
하나 이상의 범위 증명서가, 잔고 전송의 양이 0보다 크고 제 1 계좌의 잔고 이하임을 증명한다고 결정함
706
커미트먼트값이 한 세트의 값들에 기초하여, 제 1 난수, 제 2 난수, 및 제 3 난수가 동일한지를 결정하면, 잔고 전송이 유효하다고 결정함
708
커미트먼트값이 한 세트의 값들에 기초하여, 제 1 난수, 제 2 난수, 및 제 3 난수가 동일한지를 결정하면, 잔고 전송이 유효하다고 결정함
710

도면8


800
제 1 난수에 기초하여 생성되는, 제 1 계좌로부터 제 2 계좌로 전송될 트랜잭션 양 중 제 1 양의 커미트먼트값의 디지털 서명된 카피, 제 1 계좌의 공개 키를 사용하여 암호화된 제 1 양의 잔고 전송 및 제 1 난수, 제 2 계좌의 공개 키를 사용하여 암호화된 제 2 양의 잔고 전송 및 제 2 난수, 하나 이상의 범위 증명서, 및 하나 이상의 선택된 난수에 기초하여 생성된 한 세트의 값들을, 제 1 계좌로부터 수신함
802
디지털 서명된 카피에 대응하는 디지털 서명을, 디지털 서명을 생성하는데 사용된 개인 키에 대응하는 제 1 계좌의 공개 키를 사용하여 검증함
804
하나 이상의 범위 증명서가, 잔고 전송의 양이 0보다 크고 제 1 계좌의 잔고 이하임을 증명한다고 결정함
806
한 세트의 값들에 기초하여 제 1 난수 및 제 2 난수가 동일한지 그리고 제 1 양 및 제 2 양이 동일한지를 결정함
808
제 1 양 및 제 2 양이 동일하고 제 1 난수 및 제 2 난수가 동일하면, 제 1 양의 잔고 전송에 기초하여 제 1 계좌의 잔고 및 제 2 계좌의 잔고를 업데이트함
810