



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0101576
(43) 공개일자 2022년07월19일

(51) 국제특허분류(Int. Cl.)
H04L 9/32 (2006.01) H04L 9/06 (2006.01)
(52) CPC특허분류
H04L 9/3218 (2013.01)
H04L 9/0643 (2013.01)
(21) 출원번호 10-2022-0003975
(22) 출원일자 2022년01월11일
심사청구일자 2022년01월11일
(30) 우선권주장
1020210003532 2021년01월11일 대한민국(KR)

(71) 출원인
충남대학교산학협력단
대전광역시 유성구 대학로 99 (궁동, 충남대학교)
(72) 발명자
류재철
대전광역시 유성구 어은로 57 한빛아파트 132-801
김근영
대전광역시 유성구 장대로 87 퍼스트캐슬 202호
김혁진
대전광역시 서구 대덕대로168번길 82 웰리움
(74) 대리인
이은철, 김재문

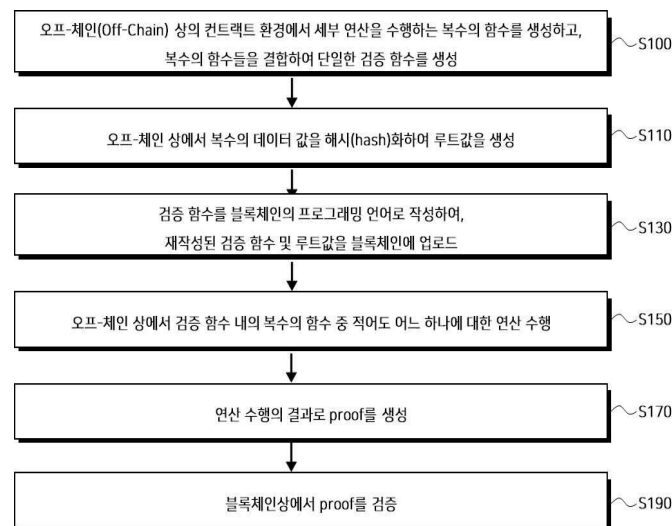
전체 청구항 수 : 총 3 항

(54) 발명의 명칭 프라이버시 보호가 가능한 블록체인 기반의 영지식 검증 방법

(57) 요약

본 발명은 컴퓨터 장치에 의해 수행되는 블록체인 기반의 영지식 검증 방법에 있어서, (a) 오프-체인(Off-Chain) 상의 컨트랙트 환경에서 세부 연산을 수행하는 복수의 함수를 생성하고, 복수의 상기 함수들을 결합하여 단일한 검증 함수를 생성하는 단계; (b) 오프-체인 상에서 복수의 데이터 값을 해시(hash)화하여 루트값을 생성하는 단계; (c) 상기 검증 함수를 블록체인의 프로그래밍 언어로 작성하여, 제작성된 상기 검증 함수 및 상기 루트값을 블록체인에 업로드하는 단계; (d) 오프-체인 상에서 상기 검증 함수 내의 복수의 상기 함수 중 적어도 어느 하나에 대한 연산이 수행되는 단계; (e) 상기 (d) 단계에서의 연산 수행의 결과로 proof를 생성하는 단계; 및 (f) 블록체인상에서 proof를 검증하는 단계를 포함하여, 프라이버시 보호가 가능하며 확장성을 가지는 것을 특징으로 한다.

대표도 - 도4



(52) CPC특허분류
H04L 9/50 (2022.05)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711102829
과제번호	2018-0-00251-003
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보보호핵심원천기술개발(R&D)
연구과제명	스마트 컨트랙트 프라이버시 보호 및 취약점 분석 기술개발
기 여 율	1/1
과제수행기관명	충남대학교 산학협력단
연구기간	2018.01.01 ~ 2020.12.31

명세서

청구범위

청구항 1

컴퓨터 장치에 의해 수행되는 블록체인 기반의 영지식 검증 방법에 있어서,

- (a) 오프-체인(Off-Chain) 상의 컨트랙트 환경에서 세부 연산을 수행하는 복수의 함수를 생성하고, 복수의 상기 함수들을 결합하여 단일한 검증 함수를 생성하는 단계;
- (b) 오프-체인 상에서 복수의 데이터 값을 해시(hash)화하여 루트값을 생성하는 단계;
- (c) 상기 검증 함수를 블록체인의 프로그래밍 언어로 작성하여, 제작성된 상기 검증 함수 및 상기 루트값을 블록체인에 업로드하는 단계;
- (d) 오프-체인 상에서 상기 검증 함수 내의 복수의 상기 함수 중 적어도 어느 하나에 대한 연산이 수행되는 단계;
- (e) 상기 (d) 단계에서의 연산 수행의 결과로 proof를 생성하는 단계; 및
- (f) 블록체인상에서 proof를 검증하는 단계를 포함하여, 프라이버시 보호가 가능한 영지식 검증 방법.

청구항 2

제 1 항에 있어서,

상기 (b) 단계의 루트값은,

- (b-1) 복수의 데이터 값에 대해 각각 해시(hash)화하여 복수의 제1 해시값을 생성하는 단계;
- (b-2) 복수의 상기 제1 해시값 중 인접한 두 개의 노드를 하나의 쌍으로 묶어 해시화하여 제2 해시값으로 변환하는 단계; 및
- (b-3) 두 개의 제n 해시값이 남을 때까지 상기 (b-2) 단계를 반복하며, 두 개의 상기 제n 해시값을 해시화하여 생성되는 것을 특징으로 하는 영지식 검증 방법.

청구항 3

제 1 항에 있어서,

상기 데이터값 중 적어도 어느 하나가 변경되는 경우,

- (g) 오프-체인 상에서, 변경된 상기 데이터 값에 대해 상기 검증 함수 내의 복수의 상기 함수 중 적어도 어느 하나에 대한 연산이 수행되고, 연산 결과에 대한 proof를 생성하는 단계;
- (h) 변경된 상기 데이터 값을 기반으로 변경된 수정 루트값을 생성하는 단계; 및
- (i) 블록체인에서 상기 (g) 단계의 proof를 검증하고, 검증 결과가 참일 경우 블록체인 내에 저장된 기존의 상기 루트값을 상기 수정 루트값으로 변경하는 단계를 더 포함하여,
변경된 데이터를 노출하지 않고 검증을 수행할 수 있는 것을 특징으로 하는 영지식 검증 방법.

발명의 설명

기술 분야

본 발명은 블록체인 기반의 영지식 검증 방법에 관한 것으로, 보다 상세하게는 함수 연산 수행 단계와 검증 단

[0001]

계를 블록체인과 오프-체인을 구분하여 검증을 수행할 수 있는 영지식 검증 방법에 관한 것이다.

배경 기술

- [0003] 공개 블록체인은 합의를 통해 블록을 생성하여 중앙화된 서버 없이 탈중앙화된 시스템으로 신뢰를 제공할 수 있다. 또한 스마트 컨트랙트에서는 사용자의 데이터와 실행할 함수를 저장하고 트랜잭션을 통해 해당 함수를 실행하여 사용자의 데이터를 변경시킬 수 있다. 그러나, 사용자의 데이터가 모든 노드에게 공개되고 어떠한 연산을 수행하는지 또는 어떠한 연산이 수행되었는지가 드러나 프라이버시 문제가 발생한다.
- [0004] 영지식 증명(Zero-Knowledge Proof)은 비밀 값을 드러내지 않고 증명하고자 하는 사실에 대해 검증을 가능하게 한다. 영지식 증명을 사용하여 연산을 증명하는 경우, off-chain에서 개인 정보에 대한 연산을 수행하고 개인 정보는 드러내지 않지만 연산에 대해 증명하는 증명값을 생성하여 블록체인에서 증명값을 제출해 계산이 올바르게 수행되었음을 검증하게 된다. 그 중 libSNARK를 이용해 제안된 ZoKrates와 zkay의 경우에는 memory와 storage의 값을 영지식 증명으로 처리하는 방법을 통해 프라이버시를 보장할 수 있다. 그러나, 어떤 연산을 수행하는지에 대한 정보는 함수를 통해 알 수 있어, 궁극적으로 프라이버시 문제를 해결하지 못하는 한계를 갖는다.
- [0005] 이에 본 발명은 연산이 수행된 함수와, 함수의 연산으로 변경된 정보가 노출되지 않아 프라이버시가 확보될 수 있는 검증 시스템에 대한 연구 개발을 진행하였다.

선행기술문헌

특허문헌

- [0008] (특허문헌 0001) 한국등록특허 제10-2289414호

발명의 내용

해결하려는 과제

- [0009] 본 발명은 블록체인 기반의 영지식 검증 방법에 관한 것으로, 종래의 영지식 증명이 갖는 문제인 증명을 하는 과정에서 사용자의 정보 또는 검증에 연산된 함수가 노출되어 사용자의 프라이버시가 보호되지 않는 문제를 해결하고자 한다.
- [0010] 본 발명이 해결하려는 과제들은 앞에서 언급한 과제들로 제한되지 않는다. 본 발명의 다른 과제 및 장점들은 아래 설명에 의해 더욱 분명하게 이해될 것이다.

과제의 해결 수단

- [0013] 상기 목적을 달성하기 위하여 본 발명은, 컴퓨터 장치에 의해 수행되는 블록체인 기반의 영지식 검증 방법에 있어서, (a) 오프-체인(Off-Chain) 상의 컨트랙트 환경에서 세부 연산을 수행하는 복수의 함수를 생성하고, 복수의 상기 함수들을 결합하여 단일한 검증 함수를 생성하는 단계; (b) 오프-체인 상에서 복수의 데이터 값을 해시(hash)화하여 루트값을 생성하는 단계; (c) 상기 검증 함수를 블록체인의 프로그래밍 언어로 작성하여, 제작성된 상기 검증 함수 및 상기 루트값을 블록체인에 업로드하는 단계; (d) 오프-체인 상에서 상기 검증 함수 내의 복수의 상기 함수 중 적어도 어느 하나에 대한 연산이 수행되는 단계; (e) 상기 (d) 단계에서의 연산 수행의 결과로 proof를 생성하는 단계; 및 (f) 블록체인상에서 proof를 검증하는 단계를 포함하는 것을 특징으로 한다.
- [0014] 바람직하게, 상기 (b) 단계의 루트값은, (b-1) 복수의 데이터 값에 대해 각각 해시(hash)화하여 복수의 제1 해

시값을 생성하는 단계; (b-2) 복수의 상기 제1 해시값 중 인접한 두 개의 노드를 하나의 쌍으로 묶어 해시화하여 제2 해시값으로 변환하는 단계; 및 (b-3) 두 개의 제n 해시값이 남을 때까지 상기 (b-2) 단계를 반복하며, 두 개의 상기 제n 해시값을 해시화하여 생성될 수 있다.

- [0015] 바람직하게, 상기 데이터값 중 적어도 어느 하나가 변경되는 경우, (g) 오프-체인 상에서, 변경된 상기 데이터값에 대해 상기 검증 함수 내의 복수의 상기 함수 중 적어도 어느 하나에 대한 연산이 수행되고, 연산 결과에 대한 proof를 생성하는 단계; (h) 변경된 상기 데이터 값을 기반으로 변경된 수정 루트값을 생성하는 단계; 및 (i) 블록체인에서 상기 (g) 단계의 proof를 검증하고, 검증 결과가 참일 경우 블록체인 내에 저장된 기존의 상기 루트값을 상기 수정 루트값으로 변경하는 단계를 더 포함하여, 변경된 데이터를 노출하지 않고 검증을 수행할 수 있다.

발명의 효과

- [0018] 본 발명에 따르면, 복수의 데이터 값을 해시화하여 최종적인 루트값을 생성하고, 루트값만을 블록체인에 업로드 시킴으로써, 데이터의 구체적인 값을 노출시키지 않을 수 있다. 특히, 특정 데이터 값이 변경되는 경우에도 블록체인에는 최종적인 루트값만 변경되어 저장되므로 어떤 데이터 값이 변경된 것인지 확인 불가능하여 사용자 프라이버시가 보호될 수 있다.
- [0019] 본 발명에 따르면, 블록체인에는 복수의 연산 함수가 결합된 단일한 검증 함수만 저장되어 있으므로, 검증에 있어서도 어떠한 함수에서 연산 수행되었는지 드러나지 않으므로 사용자 프라이버시가 보호될 수 있는 장점을 갖는다.
- [0020] 또한, 본 발명에 따르면 오프-체인에서 복수 회의 연산이 수행된 경우에도 결과적으로 생성되는 proof는 연산 수행횟수에 비례하지 않고 일정한 크기로 생성되므로, 블록체인에서 검증될 때에도 동일한 검증 비용이 요구된다. 따라서, 본 발명은 연산되는 횟수에 제한되지 않으므로 확장성이 크다는 장점을 갖는다.

도면의 간단한 설명

- [0023] 도 1은 본 발명의 실시예로서 오프-체인에서 생성된 proof가 블록체인 내에서 검증되는 과정을 나타낸다.
- 도 2는 본 발명의 실시예에 따른 복수의 함수를 결합하여 단일한 검증 함수가 생성되는 과정을 나타낸다.
- 도 3은 발명의 실시예에 따른 복수의 데이터 값으로부터 루트값이 생성되는 과정을 나타낸다.
- 도 4는 본 발명의 실시예에 따른 프라이버시 보호가 가능한 블록체인 기반의 영지식 검증 방법의 순서도를 나타낸다.
- 도 5는 본 발명의 실시예로서, 오프-체인에서 데이터 값이 변경된 경우 블록체인 내에서 검증이 수행되어 루트값이 변경되는 과정을 나타낸다.
- 도 6은 발명의 실시예로서, 데이터 값이 변경된 경우 검증이 수행되는 과정을 나타낸 순서도이다.

발명을 실시하기 위한 구체적인 내용

- [0024] 이하, 첨부된 도면들에 기재된 내용들을 참조하여 본 발명을 상세히 설명한다. 다만, 본 발명이 예시적 실시 예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일 참조부호는 실질적으로 동일한 기능을 수행하는 부재를 나타낸다.
- [0025] 본 발명의 목적 및 효과는 하기의 설명에 의해서 자연스럽게 이해되거나 보다 분명해질 수 있으며, 하기의 기재만으로 본 발명의 목적 및 효과가 제한되는 것은 아니다. 또한, 본 발명을 설명함에 있어서 본 발명과 관련된 공지 기술에 대한 구체적인 설명이, 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략하기로 한다.

- [0026] 도 1은 본 발명의 실시예로서 오프-체인(Off-Chain)에서 생성된 proof가 블록체인 내에서 검증되는 과정을 나타낸다. 도 2는 본 발명의 실시예에 따른 복수의 함수(110, 130)를 결합하여 단일한 검증 함수(10)가 생성되는 과정을 나타낸다.
- [0027] 본 발명의 검증 과정은 블록체인과 블록체인 시스템이 아닌 그 외부에서 트랜잭션이 발생하는 오프-체인을 통해 수행될 수 있다. 본 발명은 오프-체인을 이용함으로써 블록체인만으로는 해결하지 못했던 확장성 문제를 해결할 수 있다. 블록체인을 구성하는 각각의 블록은 그 사이즈가 제한적이며, statement 또는 연산 수행횟수에 따라 검증 비용이 증가하므로 확장성에 한계를 가진다. 그러나, 본 발명은 오프-체인에서 연산 수행 함수(110, 130)에서의 수행이 이루어져 오프-체인에서 proof를 생성하고, 블록체인 내에서는 proof를 검증하도록 한다. 따라서, 본 발명은 statement 또는 연산 수행 횟수에 관계없이 일정한 검증 비용을 요구하여 확장성이 있다.
- [0028] 본 발명은 (a)단계로서, 오프-체인(Off-Chain) 상의 컨트랙트 환경에서 세부 연산을 수행하는 복수의 함수를 생성하고, 복수의 함수(110, 130)들을 결합하여 단일한 검증 함수(10)를 생성할 수 있다(S10).
- [0029] 복수의 함수(110, 130)는 검증하고자 하는 대상에 대한 검증 기준을 포함할 수 있다. 예를 들어, 함수(110, 130)는 특정 나이(19세 이상)를 기준으로 한 성인 인증일 수 있다. 사용자가 성인 인증을 위해 나이를 기준으로 하는 함수(110, 130) 수행을 요청하는 경우, 사용자의 나이를 입력받아 연산을 수행할 수 있다. 함수(110, 130)의 종류 또는 개수는 본 발명에서 제한하지 않는다. 복수의 개별 함수(110, 130)는 결합되어 단일한 검증 함수(10)가 생성될 수 있다. 이때 ‘함수(110, 130)가 결합’ 된다는 의미는, 검증 함수(10)는 복수의 함수(110, 130) 중 어느 함수(110, 130)에서 연산이 수행되는지 노출되지 않도록 하는 방법을 포함할 수 있다. 즉, 검증 함수(10) 내에서 함수(110, 130)가 결합되는 방법에 제한이 있는 것은 아니며, 복수의 함수(110, 130)를 포함한다는 의미로 이해될 수 있다.
- [0030] 본 발명의 실시예에서 검증 함수(10)는 Mixing contract에서 검증 함수(10)는 Mixing contract 환경에서 생성될 수 있다. 도 1을 참고하면, 연산을 수행하는 복수의 함수(110, 130)들은 Mixing contract의 DSL(Domain Specific Language)에 각각 존재할 수 있다. Mixing contract의 DSL을 이용해 Mixing contract 코드가 작성될 수 있다. Mixing contract가 작성되면 Mixing contract는 블록체인에서 영지식 증명을 검증하기 위한 언어로 재작성될 수 있다. 블록체인의 프로그래밍 언어로 재작성된 검증 함수(10)는 (c)단계와 같이 블록체인에 업로드될 수 있다(S130).
- [0031] 본 발명의 일실시예로서 Mixing contract는 이더리움에서 사용되는 언어인 Solidity에 따라 재작성되어 Solidity 코드로 생성됨으로써, 블록체인에서 영지식 증명을 검증하기 위해 배포될 Solidity Contract가 생성될 수 있다. 따라서, 복수의 함수(110, 130)들은 Mixing contract의 DSL에 각각 존재하지만, 재작성된 스마트 컨트랙트에서는 함수(110, 130)를 구분할 수 있는 private 값을 통해 연산을 구분하는 하나의 검증 함수(10)만이 생성될 수 있다. 이를 통해 복수의 함수(110, 130)에 대한 정보는 Mixing contract DSL에서 구분하고 입력을 달리 할 수 있지만, 전체 노드들에게는 Solidity Contract에서 하나의 검증 함수(10)로 인식될 수 있다. 따라서, 검증 함수(10) 내에서 어느 함수(110, 130)의 연산이 수행되는지 노출되지 않는다.
- [0032] Mixing contract는 오프-체인 상에서 proof를 생성할 수 있다. 검증 함수(10) 내의 복수의 함수(110, 130) 중 적어도 어느 하나에 대한 연산이 수행될 수 있고, 연산 수행의 결과로 proof가 생성될 수 있다. 블록체인에서는 proof에 대한 검증을 수행할 수 있다. proof에 대한 검증이 참인 경우, 외부에서 파악할 수 없는 복수의 함수(110, 130) 중 적어도 어느 하나에 대한 연산에 대한 연산이 참임을 증명할 수 있다. 반면, proof에 대한 검증이 거짓인 경우에는 수행된 함수(110, 130)에 대한 연산이 거짓인 것으로 증명할 수 있다.
- [0033] 한편, 블록체인의 컨트랙트에 업로드 및 저장되는 값은 복수의 raw 데이터가 아니라 루트값일 수 있다. 도 3은 발명의 실시예에 따른 (b)단계에서 복수의 데이터 값으로부터 루트값이 생성되는 과정을 나타낸다(S110). 데이터는 오프-체인 상에서 복수 회 해시(hash)화될 수 있다. 해시는 단방향 암호화 기법으로, 해시 알고리즘을 통해 raw 데이터가 고정된 길이의 암호화된 문자열로 바뀔 수 있다. 특히 본 발명에서는 복수 회 해시가 수행되어, 단일 노드의 단일 해시값인 루트값을 생성할 수 있다.
- [0034] 보다 상세하게, 루트값이 생성되는 과정은 다음과 같다. n개의 데이터에 대해 각 데이터를 해시화하여 n개의 제1 해시값을 생성할 수 있다.
- [0035] n이 2 이상의 짝수인 경우, n개의 제1 해시값 중 인접한 두 개의 노드를 하나의 쌍으로 묶어 해시화하여 n/2개의 제2 해시값으로 변환할 수 있다. n이 3 이상의 홀수인 경우에는 도 3과 같이 나머지 1개의 데이터는 제1 해시값인 상태로 유지한다. 인접한 두 개의 노드를 하나의 쌍으로 묶어 해시화하는 과정을 두 개의 제n 해시값이

남을 때까지 반복한다. 두 개의 제 n 해시값을 해시화하여 최종적으로 하나의 루트값을 생성한다. n 이 3 이상의 홀수인 경우에는 나머지 하나의 제1 해시값과 하나의 제 n 해시값을 해시화하여 루트값을 생성할 수 있다. 다만, 도 3은 데이터에 대한 해시화 과정을 통해 루트값을 생성하는 일실시예로서, n 이 3 이상의 홀수인 경우 나머지 1개의 데이터에 대한 해시화 순서가 이에 제한되는 것은 아니다.

[0036] 도 4는 본 발명의 실시예에 따른 프라이버시 보호가 가능한 블록체인 기반의 영지식 검증 방법의 순서도를 나타낸다.

[0037] 본 발명의 블록체인 기반의 영지식 검증 방법은 컴퓨터 장치에 의해 수행되며, 오프-체인(Off-Chain) 상의 컨트랙트 환경에서 세부 연산을 수행하는 복수의 함수를 생성하고, 복수의 함수들을 결합하여 단일한 검증 함수를 생성하는 (a)단계(S100), 오프-체인 상에서 복수의 데이터 값을 해시(hash)화하여 루트값을 생성하는 (b)단계(S110), 검증 함수를 블록체인의 프로그래밍 언어로 작성하여, 재작성된 검증 함수 및 루트값을 블록체인에 업로드하는 (c)단계(S130), 오프-체인 상에서 검증 함수 내의 복수의 함수 중 적어도 어느 하나에 대한 연산이 수행되는 (d)단계(S150), (d)단계에서의 연산 수행의 결과로 proof를 생성하는 (e)단계(S170) 및 블록체인상에서 proof를 검증하는 (f)단계(S190)를 포함할 수 있다.

[0038] 도 5는 본 발명의 실시예로서, 오프-체인에서 데이터 값이 변경된 경우 블록체인 내에서 검증이 수행되어 루트값이 변경되는 과정을 나타낸다. 복수의 데이터 중 적어도 어느 하나의 데이터 값이 변경되는 경우, 변경된 데이터에 대한 함수(110, 130)의 연산 수행의 결과도 이전과 달라질 수 있다. 또한, 전술한 도 3을 참고하면, 변경된 데이터 값에 대한 해시값이 변경되고, 순차적인 해시화에 의해 루트값이 수정될 수 있다. 수정 루트값에 대한 proof를 오프-체인에서 생성하면, 전술한 바와 같이 블록체인에서 proof를 검증할 수 있다. 블록체인의 검증 결과, 참인 경우 컨트랙트의 기존 루트값을 수정된 루트값으로 변경시킬 수 있다. 이때, 실질적으로 데이터에 대한 연산은 Mixing contract DSL을 통해 특정 함수(110, 130)에 대해서만 수행(로컬에서만 수행)되지만, 블록체인에서는 검증 함수(10)를 통해 proof에 대한 검증만 수행되므로, 어떤 데이터값이 변경되었는지, 어떤 함수(110, 130)가 연산되었는지 드러나지 않는다.

[0039] 도 6은 발명의 실시예로서, 데이터 값이 변경된 경우 검증이 수행되는 과정을 나타낸 순서도이다. 오프-체인 상에서, 변경된 데이터 값에 대해 검증 함수(10) 내의 복수의 함수(110, 130) 중 적어도 어느 하나에 대한 연산이 수행되고, 연산 결과에 대한 proof를 생성하는 (g)단계(S310), 변경된 데이터 값을 기반으로 변경된 수정 루트값을 생성하는 (h)단계(S330), 블록체인에서 (g)단계의 proof를 검증하고, 검증 결과가 참일 경우 블록체인 내에 저장된 기존의 루트값을 수정 루트값으로 변경하는 (i)단계(S350)를 통해 변경된 데이터를 노출하지 않고 검증을 수행할 수 있다. 반면, 검증 결과가 거짓일 경우 루트값을 변경하지 않고 기존의 루트값을 유지할 수 있다.

[0040] 이상에서 대표적인 실시예를 통하여 본 발명을 상세하게 설명하였으나, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 상술한 실시예에 대하여 본 발명의 범주에서 벗어나지 않는 한도 내에서 다양한 변형이 가능함을 이해할 것이다. 그러므로 본 발명의 권리 범위는 설명한 실시예에 국한되어 정해져서는 안 되며, 후술하는 특허청구범위뿐만 아니라 특허청구범위와 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태에 의하여 정해져야 한다.

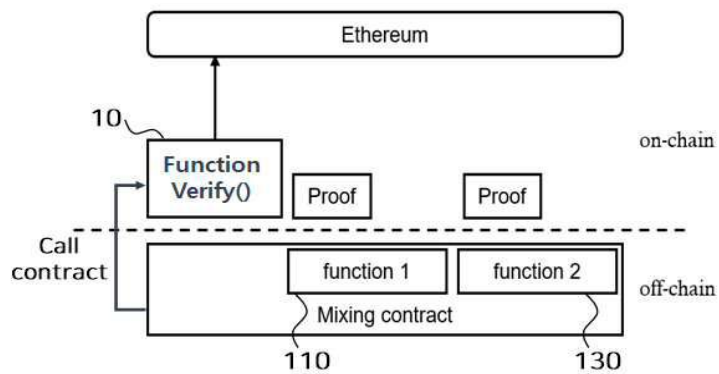
부호의 설명

[0043] 10 : 검증 함수

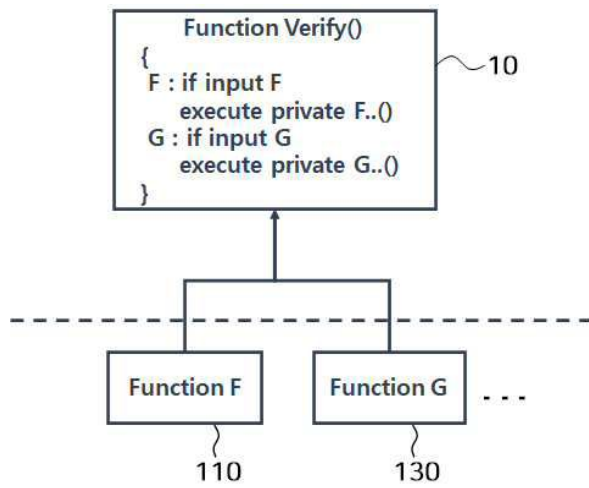
110, 130: 연산 수행 함수(함수)

도면

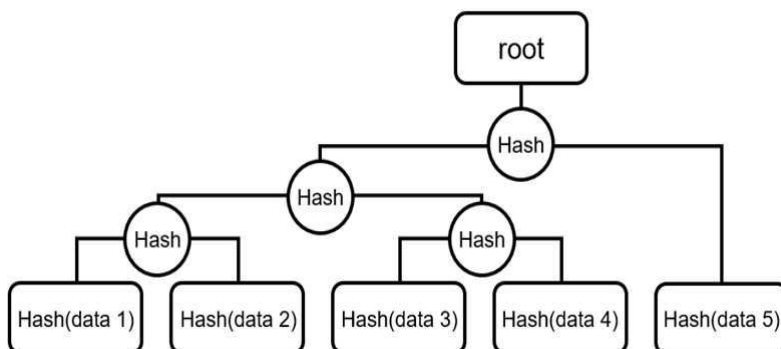
도면1



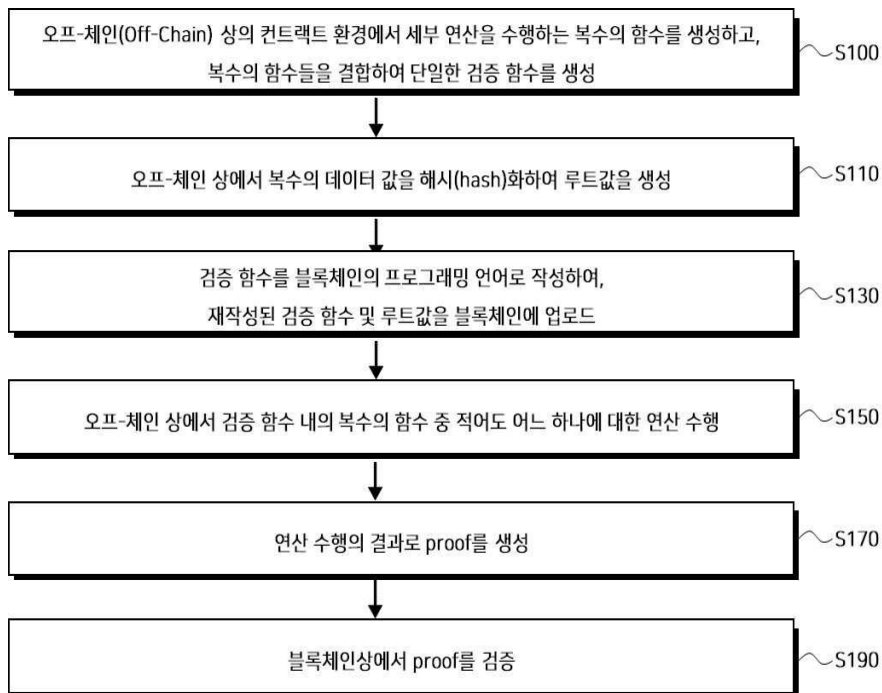
도면2



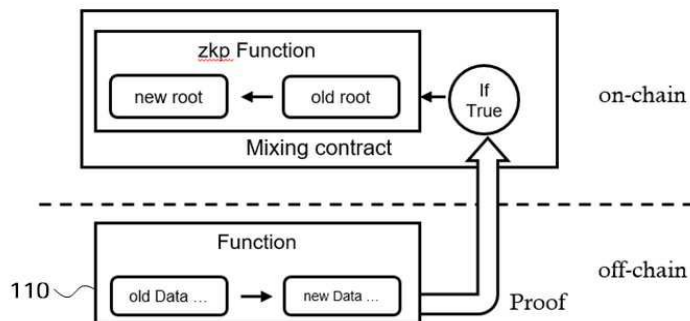
도면3



도면4



도면5



도면6

