(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0083931
(43) 공개일자 2022년06월21일

(51) 국제특허분류(Int. Cl.)
H04L 9/32 (2006.01) *H04L 9/06* (2006.01)
H04L 9/08 (2006.01)
(52) CPC특허분류
H04L 9/3218 (2013.01)
H04L 9/0643 (2013.01)
(21) 출원번호 10-2020-0173535
(22) 출원일자 2020년12월11일
심사청구일자 2020년12월11일

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

전체 청구항 수 : 총 9 항

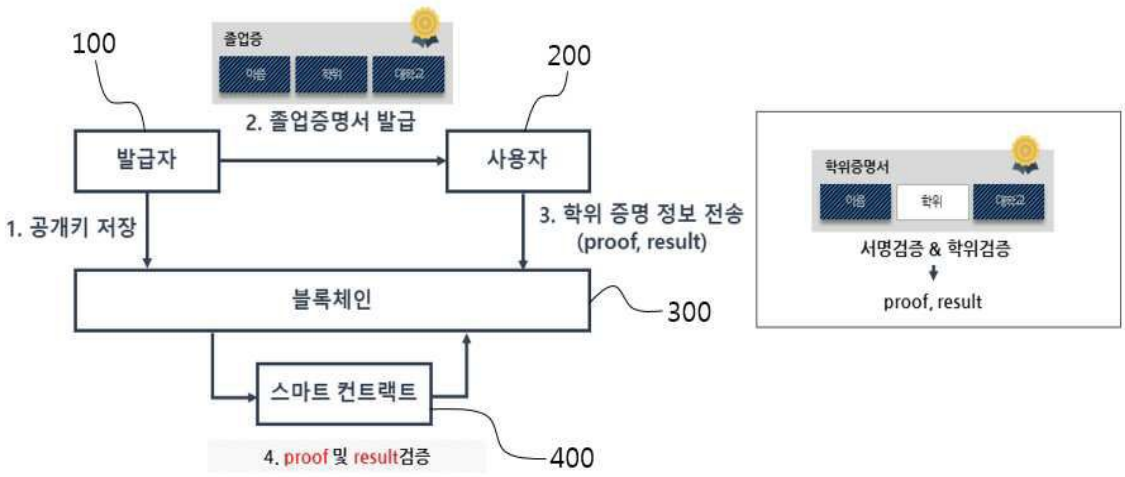
(54) 발명의 명칭 **블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템과 그 방법**

(57) 요 약

본 발명은 사용자가 프레젠테이션(Presentation)에 대한 내용을 드러내지 않으면서도 발급자(Issuer)로부터 전달받은 정보를 검증할 수 있는 블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템과 그 방법에 관한 것이다. 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생

(뒷면에 계속)

대 표 도 - 도2



성 방법은 발급자 단말(Issuer)에서 서명에 사용되는 개인키와 대응하는 공개키를 블록체인에 업로드하는 단계(S10)와, 상기 발급자 단말이 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 단계(S20)를 포함한다. 또한, 사용자 단말(Holder)에서 자신이 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트(400)를 호출하는 단계(S30)와, 스마트 컨트랙트가 상기 블록체인에 업로드된 공개키를 토대로 사용자 단말이 생성한 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 포함한다.

(52) CPC특허분류
H04L 9/0825 (2013.01)
H04L 9/50 (2022.05)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711116785
과제번호	2020-0-00321-001
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보보호핵심원천기술개발(R&D,정보화)
연구과제명	5G 서비스 환경에서 프라이버시가 보장되는 자기통제형 분산 디지털 신원 관리 및 보안 기술 개발
기 여 율	1/1
과제수행기관명	한국전자통신연구원
연구기간	2020.04.01 ~ 2020.12.31

명 세 서

청구범위

청구항 1

발급자 단말(Issuer)에서 서명에 사용되는 개인키와 대응하는 공개키를 블록체인에 업로드하는 단계(S10);

상기 발급자 단말이 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 단계(S20);

사용자 단말(Holder)이 자신이 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트를 호출하는 단계(S30); 및

스마트 컨트랙트가 상기 블록체인에 업로드된 공개키를 토대로 사용자 단말이 생성한 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 포함하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법.

청구항 2

제1항에 있어서,

상기 스마트 컨트랙트를 호출하는 단계(S30)에서 사용자 단말(Holder)이 생성하는 증명(proof)과 결과(result)는 영지식 증명(Zero-knowledge proof)을 기반으로 생성되는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법.

청구항 3

제1항에 있어서,

상기 스마트 컨트랙트를 호출하는 단계(S30)는

스마트 컨트랙트에서의 영지식 증명(Zero-knowledge proof)을 사용하기 위해 검증 컨트랙트를 생성하여 블록체인에 배포하고,

상기 블록체인의 온체인(on-chain)에서 검증을 수행하기 위해 제이슨(json, Javascript Object Notation) 데이터를 입력으로 받아 증명(proof)을 생성하는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법.

청구항 4

제1항에 있어서,

상기 증명(proof) 및 결과(result)를 검증하는 단계(S40)는

상기 발급자 단말에서 발급한 크리덴셜을 토대로 사용자 단말이 증명하고자 하는 값에 대한 해쉬(Hash) 값을 생성하는 단계(S41)와,

상기 발급자 단말의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성하는 단계(S42)와,

상기 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)을 검증하는 단계(S43)와,

상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트가 제시한 특정 조건을 만족하는지에 대해 검증하는 단계(S44) 및

상기 (S43) 단계의 증명(proof) 검증 결과와, (S44) 단계의 특정 조건에 대한 검증 결과값을 리턴(return)하여 결과(result)를 검증하는 단계(S45)를 포함하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법.

청구항 5

블록체인(Blockchain);

발급자 단말(Issuer)에게 크리덴셜(Credential)의 발급을 요청하고, 사용자가 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트를 호출하는 사용자 단말(Holder);

서명에 사용되는 개인키와 대응하는 공개키를 상기 블록체인에 업로드하고, 상기 사용자 단말의 요청에 따라 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 발급자 단말(Issuer); 및

상기 블록체인(Blockchain) 상에서 크리덴셜(Credential)의 증명(Proof)과 결과(result)의 검증을 수행하는 스마트 컨트랙트(smart contract)를 포함하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템.

청구항 6

제5항에 있어서,

상기 사용자 단말은

솔리디티(solidity)를 이용하여 인증서 타입(type)을 제공하며, 블록체인의 오프체인(off-chain)에서 크리덴셜(Credential) 값을 통해 증명(Proof)을 생성하고, 영지식 증명(Zero-knowledge proof)을 기반으로 검증 컨트랙트를 생성하는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템.

청구항 7

제6항에 있어서,

상기 사용자 단말은 블록체인의 온체인(on-chain)에서 검증을 수행하기 위해 제이슨(json, Javascript Object Notation) 데이터를 입력으로 받아 증명(proof)을 생성하는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템.

청구항 8

제5항에 있어서,

상기 사용자 단말은

발급자 단말에서 발급한 크리덴셜을 토대로 사용자 단말이 증명하고자 하는 값에 대한 해쉬(Hash) 값을 생성하며, 상기 발급자 단말의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성하는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템.

청구항 9

제5항에 있어서,

상기 스마트 컨트랙트는

사용자가 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)의 검증을 수행

하고,

상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트가 제시한 특정 조건을 만족하는지에 대해 검증을 수행하며,

상기 증명(proof)의 검증 수행 결과와, 특정 조건을 만족하는지에 대한 검증 수행 결과 값을 리턴(return)하여 결과(result)의 검증을 수행하는 것을 특징으로 하는 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템.

발명의 설명

기 술 분 야

[0001] 본 발명은 블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템과 그 방법에 관한 것으로서, 더욱 상세하게는 사용자가 프레젠테이션(Presentation)에 대한 내용을 드러내지 않으면서도 발급자(Issuer)로부터 전달받은 정보를 검증할 수 있는 블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템과 그 방법에 관한 것이다.

배 경 기 술

[0003] DID(Decentralized Identity)가 발전함으로써 종래의 중앙화된 신원 인증 시스템에서 분산 신원 관리 시스템으로 변경이 진행되고 있다. 종래의 신원 인증 시스템에서는 1) 정보 검증의 중앙화 문제와, 2) 증명서의 모든 정보가 공개되는 문제, 3) 사용자의 증명서 관리에 어려움이 있는 문제 등이 존재한다.

[0004] 따라서, 중앙 시스템에 의해 통제되지 않으며 개개인이 자신의 정보에 완전한 통제권을 가질 수 있도록 하는 디지털화된 신원 관리 체계인 분산 ID(DID)가 발전되고 있다.

[0005] 도 1은 DID(Decentralized Identity) 검증 과정을 나타내는 도면이다. 즉, 도 1은 현재 DID를 토대로 주민등록증과 같이 사용자를 인증할 수 있는 정보인 크리덴셜(Credential)을 이용하여 검증자(Verifier)에게 검증받는 과정을 나타낸다.

[0006] 도 1과 같이 사용자의 신원을 확인하기 위한 검증자(Verifier)는 하나의 서버로 이루어져 있어 사용자(Holder)가 제출한 신원인증을 그대로 전달받고, Verifiable Data Registry(Blockchain)에서 인증서를 발급해준 발급자 단말(Issuer)의 검증 정보를 받아와 사용자가 보낸 정보를 검증할 수 있다.

[0007] 하지만, 도 1의 시스템을 중앙화된 서버 클라이언트 환경이 아닌 투명한 블록체인 환경에서 컨트랙트(contract)를 통해 사용자의 신원을 확인한다고 가정할 때, 사용자가 제출한 신원 정보가 블록체인에 모두 공개되기 때문에 사용자의 정보가 노출될 수 있는 프라이버시 문제가 발생한다.

[0008] 또한, 종래의 검증 컨트랙트 생성 도구인 조크라테스(ZoKrates)의 경우 입력 값에 대한 제한이 있어 int 형태의 값을 입력해야 하며, 종래의 인증서 형태는 입력으로 받을 수 없는 문제가 있다.

[0009] 또한, 증명(Proof) 생성시 발급자(Issuer)가 생성해 주었다는 점을 입증할 수 없고, 입력 과정에서 외부의 값이 맞게 들어왔는지에 대한 오라클 문제(Oracle problem)가 발생한다.

선행기술문헌

특허문헌

[0011] (특허문헌 0001) 대한민국 등록특허 제10-2157208호(2020년 09월 18일 공고)

발명의 내용

해결하려는 과제

[0012] 본 발명이 이루고자 하는 기술적 과제는 종래의 단점을 해결한 것으로서, 사용자가 프레젠테이션(Presentation)을 그대로 제출하지 않으면서도 특정 조건에 해당하는 신원을 만족하고 증명할 수 있도록 하는데 그 목적이 있다.

[0013] 또한, 발급자 단말(Issuer)에서 발급하는 크리덴셜(Credential)의 증명(Proof)을 위한 입력 값의 제한을 극복하고자 하는데 그 목적이 있다. 또한, 증명(Proof) 생성시 발급자가 생성해 주었다는 점을 입증함으로써 데이터의 신뢰성을 증대하고자 하는데 그 목적이 있다.

과제의 해결 수단

[0015] 이러한 기술적 과제를 이루기 위한 본 발명의 일 측면에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법은 발급자 단말(Issuer)에서 서명에 사용되는 개인키와 대응하는 공개키를 블록체인에 업로드하는 단계(S10)와, 상기 발급자 단말이 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 단계(S20)를 포함한다.

[0016] 또한, 사용자 단말(Holder)에서 자신이 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트(400)를 호출하는 단계(S30)와, 스마트 컨트랙트가 상기 블록체인에 업로드된 공개키를 토대로 사용자 단말이 생성한 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 포함한다.

[0017] 이때, 상기 증명(proof) 및 결과(result)를 검증하는 단계(S40)는 상기 발급자 단말에서 발급한 크리덴셜을 토대로 사용자 단말이 증명하고자 하는 값에 대한 해쉬(Hash) 값을 생성하는 단계(S41)와, 상기 발급자 단말의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성하는 단계(S42)를 포함한다.

[0018] 또한, 상기 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)을 검증하는 단계(S43)와, 상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트가 제시한 특정 조건을 만족하는지에 대해 검증하는 단계(S44) 및 상기 (S43) 단계의 증명(proof) 검증 결과와, (S44) 단계의 특정 조건에 대한 검증 결과값을 리턴(return)하여 결과(result)를 검증하는 단계(S45)를 포함한다.

[0019] 또한, 본 발명의 다른 측면에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템은 크리덴셜(Credential)을 발급해주는 발급자 단말(Issuer)과, 상기 발급자 단말(Issuer)에게 크리덴셜(Credential)의 발급을 요청하는 사용자 단말(Holder)과, 블록체인(Blockchain)과, 상기 블록체인(Blockchain) 상에서 크리덴셜(Credential)의 증명(Proof)과 결과(result)의 검증을 수행하는 스마트 컨트랙트(smart contract)로 구성된다.

[0020] 상기 사용자 단말(Holder)은 사용자가 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트를 호출한다.

[0021] 또한, 상기 사용자 단말은 발급자 단말에서 발급한 크리덴셜을 토대로 사용자 단말이 증명하고자 하는 값에 대한 해쉬(Hash) 값을 생성하며, 상기 발급자 단말의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성한다.

[0022] 또한, 상기 발급자 단말(Issuer)은 서명에 사용되는 개인키와 대응하는 공개키를 상기 블록체인에 업로드하고, 상기 사용자 단말의 요청에 따라 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급한다.

[0023] 또한, 상기 스마트 컨트랙트는 사용자가 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)의 검증을 수행하고, 상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트가 제시한 특정 조건을 만족하는지에 대해 검증을 수행한다. 또한, 상기 스마트 컨트랙트는 증명(proof)의 검증 수행 결과와, 특정 조건을 만족하는지에 대한 검증 수행 결과 값을 리턴(return)하여 결과(result)의 검증을 수행한다.

발명의 효과

[0025] 이상에서 설명한 바와 같이, 본 발명에 따른 블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템과 그 방법은 사용자가 발급받은 신원 정보에 대한 증명(Proof)을 생성해줌으로써

사용자는 프레젠테이션(Presentation)을 그대로 드러내지 않으면서도 특정 조건에 해당하는 신원을 만족하고 있음을 증명할 수 있는 효과가 있다.

[0026] 또한, 크리덴셜(Credential)의 증명(Proof)을 위한 입력 값에 제한이 없어 제이슨(json, Javascript Object Notation) 데이터의 입력이 가능하고, 인증서 형태를 입력으로 받을 수 있다.

[0027] 또한, 영지식 증명(Proof) 생성시 cert 타입(type)을 지원하여 외부 데이터에 대한 값뿐만 아니라 발급자 단말(Issuer)에서 전달해준 값이 맞는지에 대한 검증도 진행하기 때문에 신뢰할 수 있는 기관에서 생성한 값임을 증명할 수 있어 데이터의 신뢰성을 증대할 수 있는 효과가 있다.

도면의 간단한 설명

[0029] 도 1은 DID(Decentralized Identity) 검증 과정을 나타내는 도면이다.

도 2는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템을 나타내는 구성도이다.

도 3은 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법을 나타내는 순서도이다.

도 4는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 프로토콜을 나타내는 도면이다.

도 5는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 알고리즘을 나타내는 도면이다.

도 6은 본 발명의 실시 예에 따른 온체인(on-chain) 검증 절차를 나타내는 도면이다.

도 7은 도 3의 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 세부적으로 나타내는 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0030] 아래에서는 첨부한 도면을 참고로 하여 본 발명의 실시 예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시 예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면부호를 붙였다.

[0031] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 "…부", "…기", "…모듈" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 또는 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0032] 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시 예를 설명함으로써, 본 발명을 상세히 설명한다.

[0033] 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0034] 도 1은 DID(Decentralized Identity) 검증 과정을 나타내는 도면이고, 도 2는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)을 나타내는 구성도이다.

[0035] 본 발명의 실시 예에 따른 블록체인 상에서 개인정보보호를 위한 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)은 종래의 영지식 증명 검증 컨트랙트 생성 도구인 조크라테스(ZoKrates)를 수정하여 사용자가 발급받은 신원 정보에 대한 증명(Proof)을 생성해줌으로써 사용자가 프레젠테이션(Presentation)에 대한 내용을 드러내지 않으면서도 발급자 단말(Issuer)(100)로부터 전달받은 정보를 검증할 수 있고, 누구나 증명(proof) 및 검증 컨트랙트를 생성할 수 있으며, 공개적으로 검증할 수 있기 때문에 블록체인(Blockchain)(300)의 투명한 생태계 유지가 가능하다.

[0036] 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)은 도 2에서 도시된 바와 같이 크리덴셜(Credential)을 발급해주는 발급자 단말(Issuer)(100)과, 발급자 단말(Issuer)(100)에게 크리덴

셜(Credential)의 발급을 요청하는 사용자 단말(Holder)(200)과, 블록체인(Blockchain)(300)과, 블록체인(Blockchain)(300) 상에서 상기 크리덴셜(Credential)의 증명(Proof)과 결과(result)의 검증을 수행하는 스마트 컨트랙트(smart contract)(400)로 구성된다.

[0037] 발급자 단말(Issuer)(100)은 크리덴셜(Credential)의 발급을 위한 서명에 사용되는 개인키와 대응하는 공개키(public key)를 블록체인(Blockchain)(300)에 업로드한다. 또한, 발급자 단말(Issuer)(100)은 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급한다.

[0038] 예를 들어, 상기 사용자 어드레스(Address)에 매핑이 된 정보는 사용자의 이름, 학위 및 대학교와 같은 신원정보가 될 수 있다.

[0039] 또한, 사용자 단말(Holder)(200)은 자신이 증명하고자 하는 값과 동시에 발급자 단말(100)로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트(400)를 호출한다.

[0040] 즉, 사용자 단말(Holder)(200)은 발급자 단말(100)로부터 크리덴셜(Credential) 또는 증명서를 발급받고, 컨트랙트(contract)를 확인하여 자신이 제출해야 할 정보를 획득하고 그에 맞는 증명(Proof)과 검증에 대한 결과(result)를 생성하여 스마트 컨트랙트(400)로 제출한다.

[0041] 여기에서, 상기 증명(Proof)은 영지식 증명(Zero-knowledge proof)을 기반으로 생성된다.

[0042] 또한, 사용자 단말(Holder)(200)은 스마트 컨트랙트(400)에서의 영지식 증명을 사용하기 위해 검증 컨트랙트를 생성하여 블록체인(300)에 배포한다. 또한, 사용자 단말(200)은 블록체인(300)의 온체인(on-chain)에서 검증을 수행하기 위해 제이슨(json, Javascript Object Notation) 데이터를 입력으로 받아 자동으로 증명(proof)을 생성한다.

[0043] 즉, 사용자 단말(200)은 발급자 단말(100)에서 발급한 크리덴셜을 토대로 사용자 단말(200)이 증명하고자 하는 값(이름, 학위, 대학교)에 대한 해쉬(Hash) 값을 생성하며, 발급자 단말(100)의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성한다.

[0044] 스마트 컨트랙트(400)에서는 블록체인(300)에 저장된 발급자 단말(Issuer)(100)의 공개키와 전달받은 증명(Proof) 및 결과(result)에 대한 검증을 수행한다.

[0045] 즉, 스마트 컨트랙트(400)는 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)의 검증을 수행한다. 또한, 스마트 컨트랙트(400)는 상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트(400)가 제시한 특정 조건을 만족하는지에 대해 검증을 수행한다.

[0046] 또한, 스마트 컨트랙트(400)는 증명(proof)의 검증 수행 결과와, 특정 조건을 만족하는지에 대한 검증 수행 결과 값을 리턴(return)하여 결과(result)의 검증을 수행한다.

[0047] 이때, 스마트 컨트랙트(400)는 증명(proof)의 검증 수행 결과와, 특정 조건을 만족하는지에 대한 검증 수행 결과를 판단하여 모두 만족하는 경우에 트루(T, True) 결과값을 리턴한다.

[0048] 예를 들어, 해당 사용자의 학위가 석사 이상인가? 라는 검증을 진행할 때 사용자 단말(200)은 사용자 자신의 대학교와 정확한 학위 정보(석사 또는 박사)를 드러내지 않고도 석사 이상임을 스마트 컨트랙트(400) 상에서 투명하게 검증할 수 있다.

[0049] 도 3은 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법을 나타내는 순서도이고, 도 4는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 프로토콜을 나타내는 도면이며, 도 5는 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 알고리즘을 나타내는 도면이다.

[0050] 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법은 발급자 단말(Issuer)(100)에서 서명에 사용되는 개인키와 대응하는 공개키를 블록체인(300)에 업로드하는 단계(S10)와, 발급자 단말(100)이 사용자의 어드레스(Address)에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 단계(S20)를 포함한다.

[0051] 예를 들어, 상기 어드레스(Address)에 매핑이 된 정보는 사용자의 이름, 학위 및 대학교와 같은 정보가 될 수 있다.

[0052] 또한, 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법은 사용자 단말(Holder)(200)에서 자신이 증명하고자 하는 값과 동시에 발급자 단말(100)로부터 만들어진 크리덴셜임을 증명하기 위해 내부적으로 생성된 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트(400)를 호출하는 단계(S30)를 포함한다.

[0053] 즉, 상기 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트(400)를 호출하는 단계(S30)에서 발급자 단말(Issuer)(100)로부터 크리덴셜 또는 증명서를 받은 사용자 단말(Holder)(200)은 컨트랙트(contract)를 확인하여 자신이 제출해야 할 정보를 획득하고, 그에 맞는 증명(Proof)과 검증에 대한 결과를 생성하여 스마트 컨트랙트(400)로 제출한다.

[0054] 이때, 상기 증명(Proof)은 영지식 증명(Zero-knowledge proof)을 기반으로 생성된다.

[0055] 또한, 도 4에서 도시된 바와 같이 스마트 컨트랙트(400)에서의 영지식 증명을 사용하기 위해 검증 컨트랙트를 생성하여 블록체인(300)에 배포하고, 제이슨(json, Javascript Object Notation) 데이터를 입력으로 받아 자동으로 증명(proof)을 생성해 블록체인(300)의 온체인(on-chain)에서 검증을 수행한다.

[0056] 종래의 영지식 증명 검증 컨트랙트 생성 도구인 조크라테스(ZoKrates)는 블록체인(300)의 외부 데이터인 제이슨(json) 형태를 입력으로 받지 못하고, 자체 생성한 코드를 통해 증명(proof)과 검증 컨트랙트를 생성한다.

[0057] 반면에, 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)과 그 방법은 제이슨(json) 형태의 외부 데이터로 값들을 입력 받을 수 있으므로 종래에 사용하고 있던 DID의 크리덴셜(Credential)을 그대로 입력 받을 수 있는 장점이 있다.

[0058] 이를 통해 사용자는 직접적으로 영지식 증명 검증 컨트랙트를 구현하지 않아도 되며, 영지식 증명(Proof)을 직접 만들 필요 없이 본 발명의 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)과 그 방법을 통해 쉽게 구현할 수 있다.

[0059] 또한, 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)과 그 방법은 도 5와 같이 솔리디티(solidity)를 통해 인증서 형태의 타입(type)을 지원해주며, 인증서 검증 함수를 제공한다.

[0060] 따라서, 사용자가 직접적으로 컨트랙트 및 증명(Proof)을 만들 필요 없이 사용자 단말(200)에서 명령 함수만을 입력하여 도 5의 우측과 같이 블록체인(300)의 오프체인(off-chain)에서 크리덴셜(Credential) 또는 증명서 값을 통해 증명(Proof)을 생성하고 검증 컨트랙트 또한 제시할 수 있다.

[0061] 또한, 제이슨(json) 데이터에 대한 발급자 단말(Issuer)(100)의 서명 검증 값까지 확인할 수 있는 장점이 있다.

[0062] 이때, 스마트 컨트랙트(400)에서는 사용자의 구체적인 정보를 파악할 수 없으며, 스마트 컨트랙트(400)가 제시한 특정 조건을 만족하는지에 대한 여부를 파악할 수 있다.

[0063] 또한, 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 방법은 스마트 컨트랙트(400)가 상기 (S10) 단계에서 업로드된 발급자 단말(Issuer)(100)의 공개키를 토대로 사용자 단말(200)이 생성한 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 포함한다.

[0064] 즉, 스마트 컨트랙트(400)에서는 저장된 발급자 단말(Issuer)(100)의 공개키와 전달받은 증명(Proof) 및 결과에 대한 검증을 수행할 수 있다.

[0065] 도 6은 본 발명의 실시 예에 따른 온체인(on-chain) 검증 절차를 나타내는 도면이고, 도 7은 도 3의 증명(proof) 및 결과(result)를 검증하는 단계(S40)를 세부적으로 나타내는 순서도이다.

[0066] 상기 증명(proof) 및 결과(result)를 검증하는 단계(S40)는 도 6 및 도 7에서 도시된 바와 같이 발급자 단말(100)에서 발급한 크리덴셜을 토대로 사용자 단말(200)이 증명하고자 하는 값(이름, 학위, 대학교)에 대한 해쉬(Hash) 값을 생성하는 단계(S41)와, 발급자 단말(100)의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성하는 단계(S42)와, 상기 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)을 검증하는 단계(S43)를 포함한다.

[0067] 또한, 상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트(400)가 제시한 특정 조건을 만족하는지에 대해 검증하는 단계(S44)와, 상기 (S43) 단계 및 (S44) 단계의 결과값을 리턴(return)하여 결과(result)를 검증하는 단계(S45)를 포함한다. 즉, 상기 결과(result)를 검증하는 단계(S45)는 상기 (S43) 단계 및 (S44) 단계의 결과를 모두 만족하는지 여부를 판단하여 모두 만족하는 경우에 결과값을 리턴한다.

[0068] 이때, 상기 (S43) 단계와, (S44) 단계 및 (S45) 단계는 도 6에서 도시된 바와 같이 영지식 증명(Zero-knowledge proof)을 기반으로 검증을 수행한다. 즉, 검증의 수행 결과로 T(참, True) 또는 F(거짓, False) 여부 외에는 사용자의 신원 정보가 노출되지 않는다.

[0069] 이와 같이 본 발명의 실시 예에 따른 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템(10)과 그 방법은 이름, 주민번호, 주거지역, 자격 현황, 직업 현황, 자산 현황 등과 같은 사용자의 신원내용을 확인하는 모든 곳에서 사용자의 프라이버시를 유지하며, 발급자 단말(100)이 발급하는 정보를 검증할 수 있는 효과가 있다.

[0070] 또한, 현재 DID 시장이 활발히 성장하고 있는 추세에 발급받은 크리덴셜(Credential)에 대한 내용을 드러내지 않으면서도 발급자 단말(200)로부터 전달받은 정보를 검증할 수 있고, DID와 연동이 되어 구현가능한 장점이 있다.

[0071] 또한, 누구나 증명(proof) 및 검증 컨트랙트를 생성할 수 있으며, 공개적으로 검증할 수 있기 때문에 블록체인(Blockchain)(300)의 투명한 생태계 유지가 가능한 효과가 있다.

[0072] 이상으로 본 발명에 관한 바람직한 실시예를 설명하였으나, 본 발명은 상기 실시예에 한정되지 아니하며, 본 발명의 실시예로부터 당해 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의한 용이하게 변경되어 균등하다고 인정되는 범위의 모든 변경을 포함한다.

부호의 설명

[0074] 10 : 영지식 증명 생성 및 검증 스마트 컨트랙트 생성 시스템

100 : 발급자 단말(Issuer)

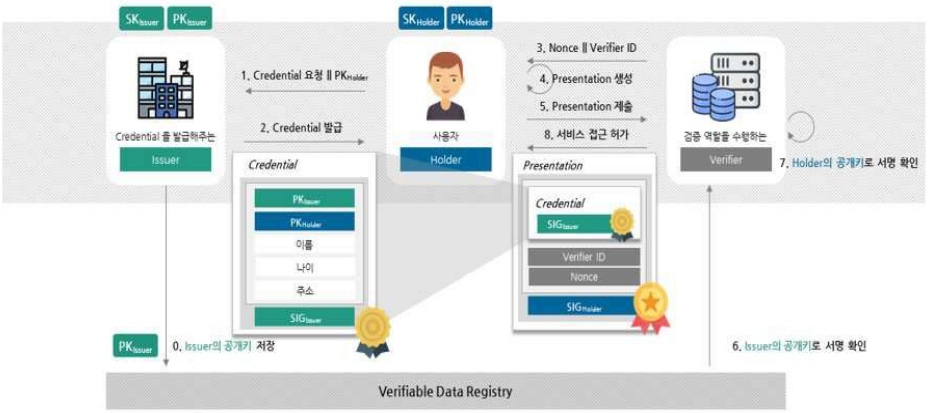
200 : 사용자 단말(Holder)

300 : 블록체인(Blockchain)

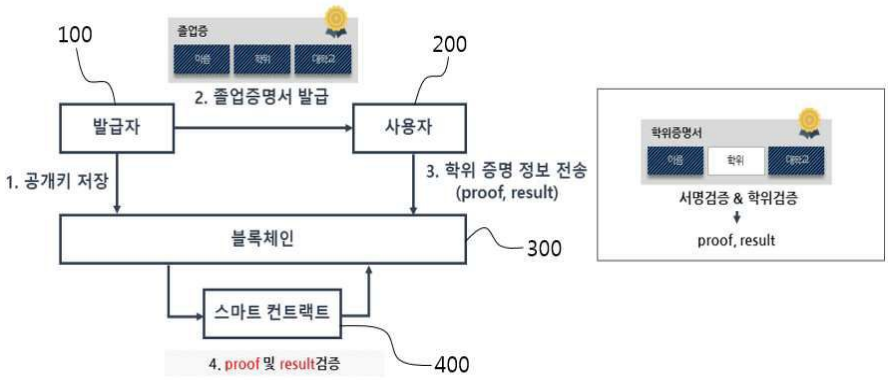
400 : 스마트 컨트랙트(smart contract)

도면

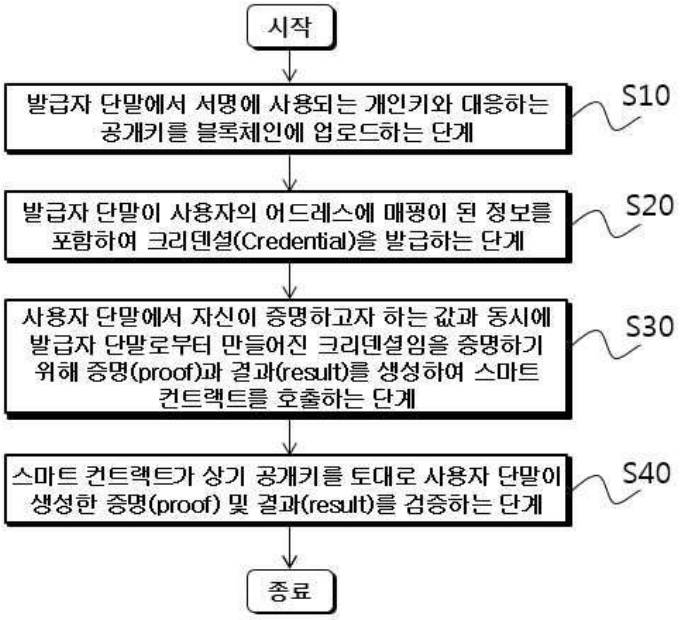
도면1



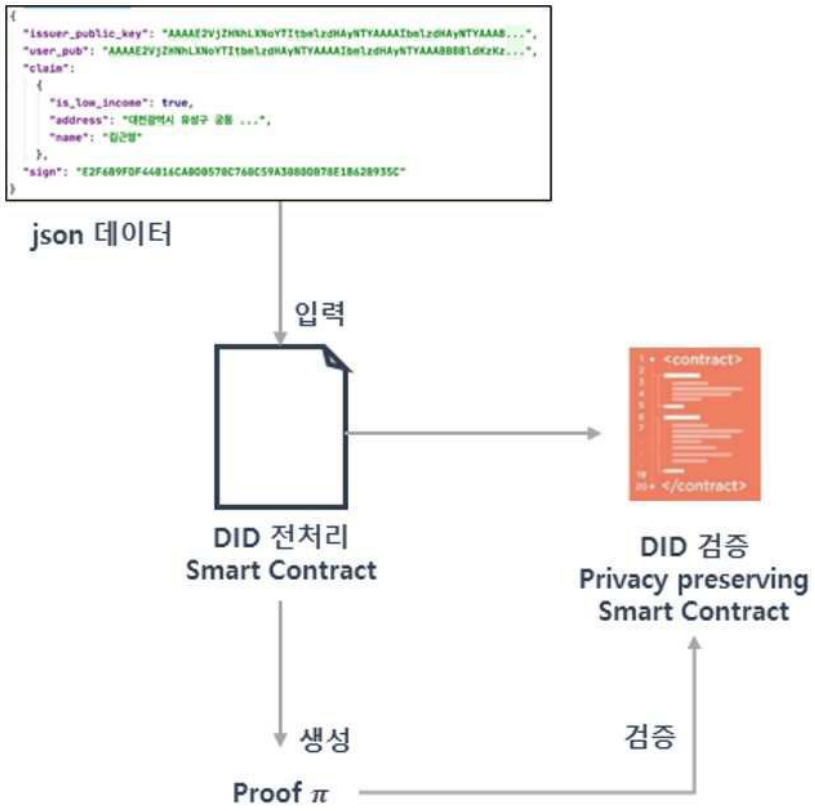
도면2


100
200
2. 졸업증명서 발급
발급자
사용자
1. 공개키 저장
3. 학위 증명 정보 전송
(proof, result)
블록체인
300
스마트 컨트랙트
400
4. proof 및 result검증
서명검증 & 학위검증
proof, result

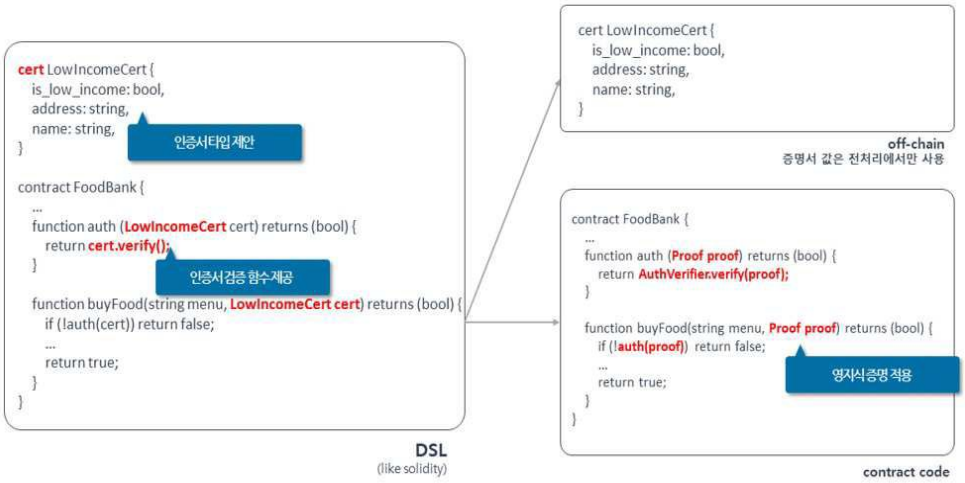
도면3


시작
발급자 단말에서 서명에 사용되는 개인키와 대응하는 공개키를 블록체인에 업로드하는 단계
S10
발급자 단말이 사용자의 어드레스에 매핑이 된 정보를 포함하여 크리덴셜(Credential)을 발급하는 단계
S20
사용자 단말에서 자신이 증명하고자 하는 값과 동시에 발급자 단말로부터 만들어진 크리덴셜임을 증명하기 위해 증명(proof)과 결과(result)를 생성하여 스마트 컨트랙트를 호출하는 단계
S30
스마트 컨트랙트가 상기 공개키를 토대로 사용자 단말이 생성한 증명(proof) 및 결과(result)를 검증하는 단계
S40
종료

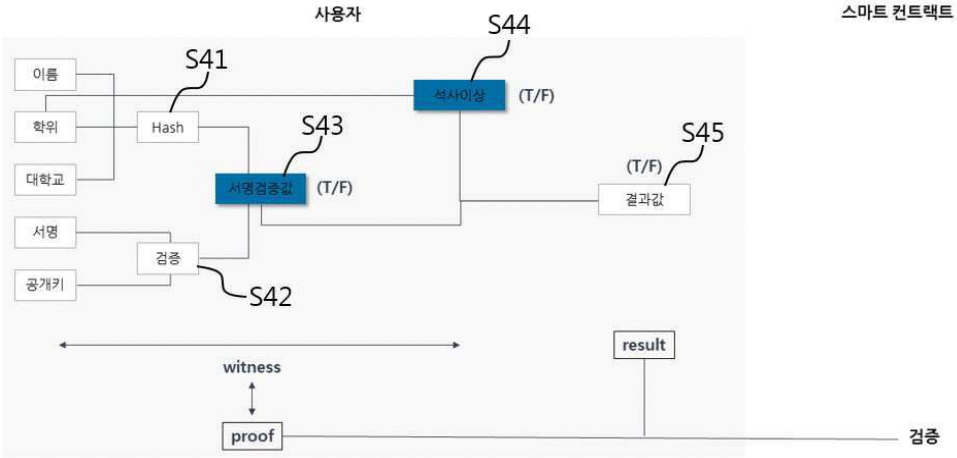
도면4


{
"issuer_public_key": "AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAAB...",
"user_pub": "AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBBldKzKz...",
"claim":
{
"is_low_income": true,
"address": "대전광역시 유성구 궁동 ...",
"name": "김건형"
},
"sign": "E2F689FDF44816CA0B0570C768C59A38B80B78E18628935C"
}
json 데이터
입력
<contract>
</contract>
DID 전처리
Smart Contract
DID 검증
Privacy preserving
Smart Contract
생성
검증
Proof π

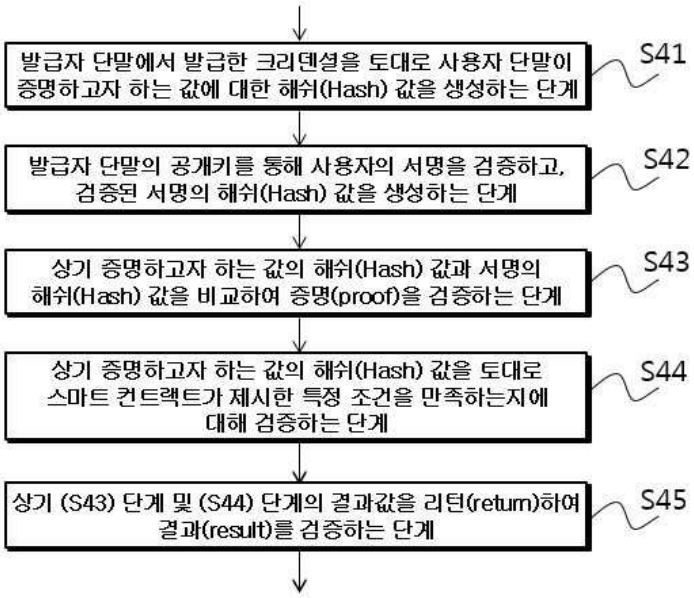
도면5


cert LowIncomeCert {
is_low_income: bool,
address: string,
name: string,
}
인증서타입 제안
contract FoodBank {
...
function auth (LowIncomeCert cert) returns (bool) {
return cert.verify();
}
인증서검증 함수제공
function buyFood(string menu, LowIncomeCert cert) returns (bool) {
if (!auth(cert)) return false;
...
return true;
}
}
DSL
(like solidity)
cert LowIncomeCert {
is_low_income: bool,
address: string,
name: string,
}
off-chain
증명서 값은 전처리에서만 사용
contract FoodBank {
...
function auth (Proof proof) returns (bool) {
return AuthVerifier.verify(proof);
}
function buyFood(string menu, Proof proof) returns (bool) {
if (!auth(proof)) return false;
...
return true;
}
}
영지식증명 적용
contract code

도면6


사용자
스마트 컨트랙트
이름
학위
대학교
서명
공개키
Hash
검증
서명검증값
(T/F)
S41
S42
S43
S44
S45
석사이상
(T/F)
(T/F)
결과값
witness
proof
result
검증

도면7


발급자 단말에서 발급한 크리덴셜을 토대로 사용자 단말이 증명하고자 하는 값에 대한 해쉬(Hash) 값을 생성하는 단계
S41
발급자 단말의 공개키를 통해 사용자의 서명을 검증하고, 검증된 서명의 해쉬(Hash) 값을 생성하는 단계
S42
상기 증명하고자 하는 값의 해쉬(Hash) 값과 서명의 해쉬(Hash) 값을 비교하여 증명(proof)을 검증하는 단계
S43
상기 증명하고자 하는 값의 해쉬(Hash) 값을 토대로 스마트 컨트랙트가 제시한 특정 조건을 만족하는지에 대해 검증하는 단계
S44
상기 (S43) 단계 및 (S44) 단계의 결과값을 리턴(return)하여 결과(result)를 검증하는 단계
S45