Journal of KIIT. Vol. 18, No. 11, pp. 131-138, Nov. 30, 2020. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2020.18.11.131

비밀공유 기법을 적용한 스마트 컨트랙트 플랫폼에 대한 연구

손애선*[1], 유수빈*[2], 조정화*[3], 유수민*[4]

A Study on Smart Contract Platform using Secret Sharing Scheme

Ae-Seon Son*[1], Soo-Bin Yoo*[2], Jung-Hwa Jo*[3], and Su-Min Yoo*[4]

본 연구는 문화체육관광부 및 한국저작권위원회의 2019년도 저작권연구개발사업의 연구결과로 수행되었음.
(2019-SC-9500)

요 약

스마트 컨트랙트는 분산 원장 환경에서 데이터를 기록하여 데이터의 무결성과 유효성이 검증된다는 점과 작성된 코드에 의하여 설정된 조건이 충족되면 자동으로 이행되는 특성 때문에 신뢰성을 요구하는 다양한 자동화 시스템에 적용되고 있다. 스마트 컨트랙트가 활발하게 사용되고 있는 분야 중 하나는 계약 체결과 관련된 분야이다. 하지만 블록체인이 가진 분산 원장 환경의 특성상 거래되는 데이터가 모든 네트워크 참여자들에게 공유되기 때문에 기밀성이 요구되는 데이터는 저장하지 못한다는 문제가 있다. 본 논문은 스마트 컨트랙트 기반의 계약 플랫폼에 비밀공유 기법을 이용한 계약 내용을 별도의 데이터베이스에 저장하는 방식을 적용하여 기밀성과 무결성을 보장함으로써 비대면으로 계약서를 작성하는 과정에서 신뢰성 있는 계약 체결이 가능하도록 하는 것에 궁극적 목표를 둔다.

Abstract

Smart Contracts are applied to various automation systems that require reliability because of the fact that integrity and validity of data is verified by recording data in a distributed ledger environment, and the characteristics that are automatically executed when conditions set are satisfied by the written code. One of the areas where smart contracts are used is to sign a contract. However, due to the nature of the distributed ledger environment of the Blockchain, there is a problem that data requiring confidentiality cannot be stored because the data that is transacted is shared with all network participants. This paper aims to make it possible to conclude a reliable contract in the process of writing a contract non-face-to-face by applying a method of storing contract contents in off-chain using a secret sharing technique in a contract platform based on smart contracts.

Keywords

secret sharing scheme, blockchain, smart contract, information security, online contract platform

* 경주스마트미디어센터 연구원 (*[2]교신저자)
- ORCID[1]: https://orcid.org/0000-0001-5153-2929
- ORCID[2]: https://orcid.org/0000-0001-8095-7862
- ORCID[3]: https://orcid.org/0000-0002-1144-7153
- ORCID[4]: https://orcid.org/0000-0001-5643-1422

· Received: Sep. 25, 2020, Revised: Nov. 04, 2020, Accepted: Nov. 07, 2020
· Corresponding Author: Soo-Bin Yoo
Strategic Planning Dept. Gyeongju Smart Media Center, 587-18 Gyeonggam-ro, Gyeongju, 38118, Korea
Tel.: +82-54-781-2943, Email: yoosoobin@silgam.or.kr

Ⅰ. 서 론

최근 인공지능, 빅데이터, 클라우드, 블록체인 등의 4차 산업혁명 기반 기술들이 각종 분야에서 활용되고 있다. 그중 블록체인 기술은 지난 2018년부터 공공서비스 혁신을 위한 사업에 이용되어 오고 있다. 2020년에는 과학기술정보통신부와 한국인터넷진흥원 주도하에 '블록체인 공공선도 시범사업·민간주도 국민 프로젝트'를 시행 중이며[1], 다양한 분야에서 과제가 선정되어 스마트 컨트랙트를 이용한 블록체인 기반의 플랫폼 구현이 활발하게 이루어지고 있다.

스마트 컨트랙트는 블록체인 기술 기반의 이더리움 환경에서 구현되며, 제3기관의 공증 없이 다양한 형태의 계약을 체결할 수 있다. 스마트 컨트랙트는 특정 조건이 충족되면 이행되는 특징을 갖기 때문에 자동화 시스템에 적합하며, 데이터가 한 번 기록되면 변경이 불가능하여 데이터의 신뢰성이 요구되는 플랫폼에 적용할 수 있다.

2019년 8월 고용주가 퇴직한 종업원의 임금을 체불하여 고발당하자 근로계약서를 위조해 해당 종업원을 상대로 소송을 제기한 사건이 있었다[2]. 이처럼 계약 당사자 중 한 명이 계약 내용을 위조하게 되면 둘 중 어느 계약서가 합의하에 작성된 계약서인지 판단이 어려워지는 경우가 발생할 수 있다. 계약서의 조작 여부를 판단하기 위해서는 법적 소송이 불가피한 상황이다. 이에 계약서 위변조 문제의 원천을 제거하고자 스마트 컨트랙트 기술을 이용한 계약서 작성 플랫폼의 구현을 고안하게 되었다.

하지만 기존의 스마트 컨트랙트는 하나의 노드에 저장할 수 있는 데이터 용량의 제한과 데이터 용량에 따른 수수료 문제 때문에 대용량의 데이터를 저장하는 것에 한계가 있다.

데이터 용량 제한과 같은 블록의 확장성 문제의 해결법은 온체인(On-chain) 솔루션과 오프체인(Off-chain) 솔루션으로 구분한다[3]. 온체인이란 블록체인에 데이터를 기록하는 방식을 의미하며, 블록 크기를 늘리는 것이라고 볼 수 있다. 블록의 크기를 늘리면 수수료도 줄어들고 많은 데이터의 저장이 가능해진다. 하지만 기존 프로토콜 구조를 변경하는 방법이기 때문에 하드포크가 필수적이다[3]. 오프체인은 블록체인이 아닌 다른 곳에 데이터를 저장하는 것을 의미한다. 대용량의 데이터를 별도의 데이터베이스에 저장함으로써 스마트 컨트랙트 거래에서 빠른 처리 속도를 보이지만 해킹의 위험이 있기 때문에 별도의 암호화가 필요하다.

본 논문에서는 스마트 컨트랙트 시스템의 한계 극복을 위해 비밀공유 기법을 적용한 데이터를 별도의 데이터베이스에 저장하는 오프체인 방식을 이용한 스마트 컨트랙트 기반의 계약 플랫폼 구현 방안을 제안하고자 한다. 제안하는 플랫폼에서는 블록체인 네트워크의 분산 원장 환경에서 거래가 이루어지기 때문에 데이터의 무결성과 신뢰성이 보장되고, 비밀공유 기법을 이용하여 데이터가 저장되어 기밀성을 확보할 수 있다.

Ⅱ. 관련 연구

2.1 기존 암호화 방식의 문제점

기존의 암호화 알고리즘의 종류는 크게 대칭키 암호화 방식과 비대칭키 암호화 방식으로 나뉜다.

대칭키 암호화 방식은 동일한 비밀키 하나를 이용하여 암호화와 복호화를 진행하는 방식으로, 인원이 늘어날수록 키의 수가 급증하여 관리하기 어렵다는 단점이 있다. 대칭키 암호화 방식의 문제를 해결하기 위해 제안된 비대칭키 암호화 방식은 암호화할 때는 공개키를 이용하고, 복호화할 때는 개인이 소유하고 있는 각자 다른 비밀키를 이용하는 방식이다. 이는 비밀키를 개인이 관리하기 때문에 비밀키 유실 시 복호화가 불가능하다는 문제점이 있다.

이 같은 기존 암호화 방식의 문제를 해결하기 위해 다수의 참여자가 암호화된 데이터를 조각으로 나누어 분배된 값을 소유하는 비밀공유 기법을 해당 플랫폼에 적용하고자 한다.

2.2 비밀공유 기법

1979년 Adi Shamir에 의해 제안된 비밀공유 기법은 하나의 비밀정보를 여러 개의 비밀조각으로 나누어 합법적인 참가자들에게 이를 분배하고, 일정

수 이상의 참가자들이 모여야 비밀정보를 복원할 수 있는 방식이다[4]. 이는 '$(t,n)-threshold$' 기법으로 지칭되는데, 이 때 t는 비밀정보의 복원에 필요한 비밀조각의 개수이며, n은 암호화에 참여하는 인원수를 의미한다. $(t,n)-threshold$ 기법은 초기화 과정, 비밀조각인 공유값을 분배하는 과정과 비밀정보를 복원하는 과정으로 구성된다[4].

초기화 과정에서는 n보다 작은 소수 p에 대하여 Z_p상의 0이 아닌 서로 다른 정수 n개를 선택한다. 집합 Z_p는 정수를 p로 나누었을 때 나올 수 있는 나머지 값들의 집합을 의미한다. 선택한 값을 공개값 x_i로 표기하여 참가자 P_i에 각각 대응시켜준다. 여기서 i는 1부터 n사이의 값을 만족한다.

비밀정보(S)는 Z_p상의 원소로 가정하고, Z_p상에서 $t-1$개의 원소들을 선택하여 이를 식 (1)에 해당하는 다항식의 계수로 사용한다. 식 (1)을 이용해 공유값 $y_i = f(x_i)$ 값을 생성하고, P_i에게 분배한다.

$$f(x) = S + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1} (\mathrm{mod} p) \quad (1)$$

비밀정보의 복원은 t명 이상의 참가자 P_i로부터 수집한 (x_i, y_i)쌍들을 Lagrange 보간 다항식에 대입하여 이루어진다. 복원에 사용되는 Lagrange 보간 다항식은 식 (2)와 같다.

$$f(x) = \sum_{j=0}^{t-1} \left(y_j \prod_{f=0, f\neq j}^{t-1} \frac{x - x_f}{x_j - x_f} \right) \mathrm{mod} p \quad (2)$$

비밀공유 기법에서는 참여자들이 갖고 있는 비밀조각이 유실되더라도 일정 인원 이상의 비밀조각만 있으면 비밀정보의 복구가 가능하다. 한 명의 독단적인 결정에 의해서 이루어지는 것이 아닌 여러 명의 합의가 필요한 업무를 수행하거나, 참여자 중 일부 인원만으로 정보를 확인해야 하는 상황에 적합하다고 볼 수 있다.

이 같은 특징으로 인해 비밀공유 기법은 다수가 참여하는 계약 체결에 효과적으로 적용할 수 있다.

2.3 블록체인

블록체인은 peer-to-peer 환경에서 클라이언트들에게 데이터 사본을 공유하여 신뢰성을 보장하는 분산원장 네트워크이다[5]. 블록체인은 블록이 사슬처럼 연결되어 있으며, 각 블록은 해시 값으로 식별되어 이전 블록의 해시를 참조함으로써 연결된다. 블록체인에서 블록이 연결되는 구조는 그림 1과 같다.

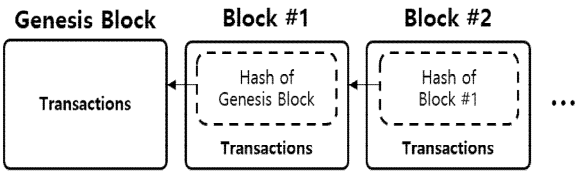

그림 1. 블록체인의 블록 연결 구조[6]
Fig. 1. Block connection structure in Blockchain

위와 같은 블록의 연결구조로 인해 블록체인에 저장되는 데이터의 위변조를 방지할 수 있다. 만약 블록 1의 데이터가 변경된다면 블록 1의 해시 값이 바뀌게 되고, 블록 2가 참조하고 있는 블록 1의 해시 값과 달라 연결 구조가 끊어지게 된다. 그로 인해 블록 1은 유효하지 않은 블록으로 판단되어 데이터의 위변조가 일어났음을 판단할 수 있다.

또한 블록체인은 거래를 관리하고 처리하는 중앙 서버 없이 거래 당사자들 간의 직접적인 거래를 가능하게 한다[7]. 블록체인 네트워크에서 중앙 서버 없이도 정보보호가 이루어질 수 있는 이유는 분산원장 환경에서의 작업증명을 통한 데이터 기록 방식 덕분이다[8]. 작업증명은 블록체인에서 거래 기록의 유효성을 검증하기 위해 필요한 과정이며. 작업 증명 과정은 그림 2와 같다[7].

① When a transaction is generated, it is broadcast to all users.
② Each user collects new transactions into a block.
③ Each user attempts a proof-of-work for the block.
④ If a user succeeds in proof-of-work, the block is propagated to all users.
⑤ All users approve the block if it is valid.
⑥ All users indicate that the block has been approved.

그림 2. 블록체인에서 작업증명이 이루어지는 과정
Fig. 2. Proof of work process in Blockchain

블록체인은 작업증명 과정을 통해 모든 사용자가 거래내역을 공유하고 있기 때문에 거래내역의 변조가 일어나더라도 참여자들의 거래내역과 대조하여 위변조 사실을 검증할 수 있다.

2.4 스마트 컨트랙트

1994년 닉 재보(Nick Szabo)는 스마트 컨트랙트의 개념을 처음 제안한 인물로 이 용어를 “당사자들이 다른 약속에 따라 수행하는 프로토콜을 포함하여 디지털 형식으로 지정된 일련의 약속”이라고 정의했다[9]. 닉 재보는 스마트 컨트랙트를 이용하여 코드로 작성된 계약 내용이 자체적으로 동작할 수 있도록 하여 신뢰할 수 있는 거래 중개자의 필요성과 악의적이거나 우발적인 예외의 발생을 최소화할 것을 제안했다[6]. 당시에는 기술의 한계로 실제 개발되지 못하고, 2015년 비탈릭 부테린(Vitalik Buterin)이 이더리움을 개발함으로써 구현되었다[10].

스마트 컨트랙트는 고유한 주소를 가지며, 이를 컨트랙트 계정(Contract Account, CA)이라고 한다. 개인키로 컨트랙트나 계정에 대한 접근을 제어하는 외부 소유 계정(Externally Owned Account, EOA)과는 다르게 CA는 스마트 컨트랙트 코드로 제어를 수행한다.

스마트 컨트랙트는 EOA로부터의 트랜잭션에 의해 호출된 경우에만 실행되며, 스마트 컨트랙트 자체적으로 실행되지 않는다. 스마트 컨트랙트 코드가 활성화되는 과정은 그림 3과 같다.

이처럼 스마트 컨트랙트 코드를 작동시키기 위해서는 외부소유계정의 생성이 필수적이며, 본 플랫폼에서는 Geth 클라이언트를 이용하여 계정을 생성할 수 있도록 하였다.

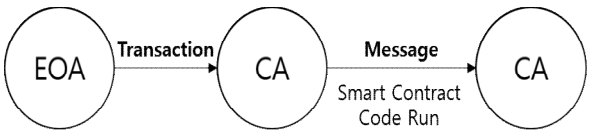

그림 3. 외부소유계정에 의해 스마트 컨트랙트 코드가 활성화되는 과정
Fig. 3. Process of activating the smart contract code by an externally owned account

III. 제안하는 기법

계약자들은 계약서 작성 시 계약 내용과 함께 계약에 참여하는 인원 n과 암호화된 계약 내용 확인에 필요한 최소 인원 t를 입력한다. 입력된 계약 내용은 비밀공유 기법을 거쳐 별도의 데이터베이스에 저장된다. 암호화된 계약 내용을 바탕으로 비밀공유가 진행되고, $n+1$번째의 비밀공유 값을 블록체인 노드에 저장하게 된다. 비밀공유를 통해 나온 공유값은 참여자들에게 각각 분배된다.

계약 내용을 확인할 때에는 계약서 작성 시 입력했던 n, t의 값과 함께 참여자들에게 분배된 공유값을 입력하여 Lagrange 보간법에 따라 복원을 수행한다. 복원된 다항식에 $n+1$의 값을 대입하여 나온 결과 값이 블록체인 노드에 저장된 값과 일치한다면 비밀정보의 값을 알아낼 수 있어 별도의 데이터베이스에 저장된 계약 내용을 불러올 수 있다.

본 절에서는 위의 과정에 대한 구현 방안에 대해 설명한다.

3.1 계약 내용에 대한 기밀성 보장

비밀공유 기법을 사용하기 위해서는 암호화할 내용에 대하여 연산을 수행하기 위한 변환이 필요하다. 계약 당사자 간의 정보가 담긴 데이터와 계약 내용과 관련된 계약 데이터를 SHA-256 해시 함수를 이용하여 암호화한다. 해시값은 32bit 크기를 갖는 byte형 배열로 변환된다. 변환된 해시값은 비밀정보가 되며, 계약 데이터 저장 시 파일명으로 사용하기 위해 String 형으로의 변환이 필요하다.

형변환은 Integer 객체를 통해 이루어진다. 이때 byte형은 8bit의 크기를 갖기 때문에 32bit의 크기를 갖는 Integer형에 따라 비트가 확장된다. 그 과정에서 2의 보수법 처리에 따라 가장 앞자리 비트가 0인 경우는 확장된 비트 값들이 0이 되고, 1인 경우에는 값이 1로 채워진다. 비트가 1이 되는 경우에는 전혀 다른 값이 되기 때문에 0xff와 AND 비트 연산을 수행하여 올바른 값으로 되돌리는 작업이 필요하다. 해당 연산 과정은 그림 4와 같다.

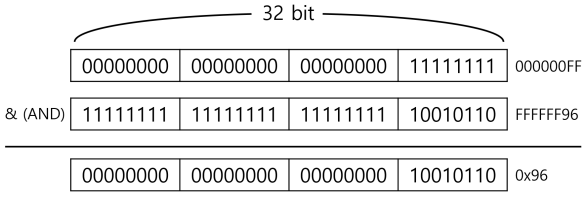

그림 4. 0xFF와 32bit로 비트가 확장된 0x96의 AND 비트 연산 결과
Fig. 4. Result of the bitwise AND operation of 0xFF and 0x96 bit extended to 32 bit

해시값은 32byte이기 때문에 바이트마다 형변환한 결과를 이어 붙여서 String형의 비밀정보로 저장한다. 이 과정에서 0부터 15 범위에 있는 숫자의 경우, 한 자릿수의 결과값이 나오게 된다. 그렇게 되면 바이트 당 연산을 수행한 모든 결과 값을 이어 붙였을 때 제대로 된 결과값을 얻을 수 없다. 그렇기 때문에 한 자릿수의 결과값이 나오게 되면 앞자리에 0을 다시 붙여서 값을 저장해준다.

다음의 과정은 2장의 2절에서 설명한 비밀공유 기법 수행 과정을 따른다.

초기화 과정에서는 각 참여자 P_i에게 공개값 x_i를 랜덤(Random)하게 분배한다. 공유값 분배 과정에서 다항식 값을 계산할 때는, 연산 속도의 향상을 위해 byte 단위로 연산을 수행하였다. 이에 따라 8bit 크기를 갖는 수의 연산을 위해 모듈러(Modular) 연산을 수행할 p의 값을 2^8으로 설정하였다.

다항식 계수가 될 값은 랜덤 함수를 이용하여 생성하는데 랜덤 함수에 이용될 값의 범위는 8차 기약다항식의 값을 최대로 설정한다. 기약다항식은 더 이상 인수분해 할 수 없는 다항식으로, n차 기약다항식으로 모듈러 연산을 수행하면 $(n-1)$차 다항식을 결과로 얻을 수 있다. 해당 기법에서 사용될 8차 기약다항식은 $x^8+x^4+x^3+x+1$로 설정하였으며, 이는 이진수로 표현하면 100011011_2이 되므로 283의 값을 가진다.

다음으로 p를 이용해 1부터 283 사이의 값을 갖는 수에 대한 모듈러 연산을 수행하여 나온 계수 값을 다항식에 대입하여 공유값 y_i를 생성한다. byte 단위로 연산을 수행하여 나온 y_j값들은 32byte 크기를 갖는 배열에 저장하여 하나로 연결하여 참여자들에게 분배된다.

3.2 계약 내용의 기록

스마트 컨트랙트를 이용해서 계약 내용 작성 시 입력받았던 참여자수와 계약 내용 복원에 사용될 최소 인원수, 다항식 계산 시 $n+1$의 값을 넣어 계산했던 비밀공유 값을 저장한다.

스마트 컨트랙트 코드는 solidity 언어로 작성하였다. 스마트 컨트랙트 코드를 통해 입력받은 값은 블록체인 네트워크를 통해 블록에 저장되어 더 이상 수정 및 변경이 불가능하다. 계약 내용의 암호화 및 데이터 저장 과정은 그림 5와 같다.

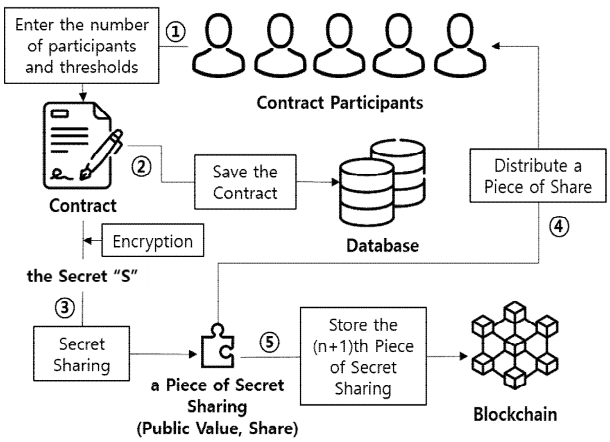

그림 5. 비밀공유 값 분배 후 계약서 저장 과정
Fig. 5. Process of storing the contract after distribution of secret sharing values

3.3 계약서 내용의 확인

비밀공유 값 분배 시 사용된 다항식을 다시 생성하기 위해서는 계약 참여 인원 n과 계약 내용 확인에 필요한 최소 인원 t, 최소 인원만큼의 공유값이 필요하다. 해당 데이터를 모두 입력으로 넘겨주면 복원 과정이 진행된다.

복원에는 Lagrange 보간 다항식이 사용되며, 생성된 보간 다항식에 $n+1$의 값을 넣어 계산한 값이 스마트 컨트랙트에 저장된 해시값과 동일한지 검증한다. 두 값이 동일하다면 합법적인 참여자들만이 복원에 참여했음이 검증된 것이므로 원래의 비밀정보 값을 계산하여 해당 비밀정보 값과 일치하는 파일명을 찾아 계약 데이터를 보여준다. 비밀공유 복원 과정을 이용한 계약 내용 복원 과정은 그림 6과 같다.

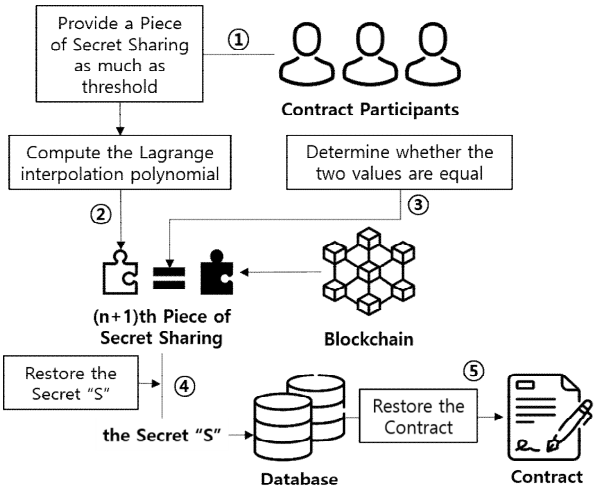

그림 6. 비밀공유 복원을 통한 계약서 복원 과정
Fig. 6. Process of restoring the contract using the secret sharing

IV. 연구 결과

4.1 기존 블록체인 기반 플랫폼의 문제점

블록체인은 분산원장 환경의 특성으로 인해 보안 문제에 취약함을 보여 왔다. 보안 문제로 자주 거론되는 것에는 51% 공격과, 내부 참여자 간의 담합 문제, 프라이버시 침해 문제, 참여자 권한 오남용 문제 등이 있다[11].

51% 공격과 참여자 담합 문제는 참여자 신원의 불분명성으로 인해 발생하는 문제이기 때문에, 사설 네트워크에서 플랫폼을 운영하여 모니터링을 지속함으로써 문제 발생을 방지할 수 있다.

프라이버시 침해 문제는 비밀공유 기법을 통해 암호화한 계약 내용의 비밀조각만을 블록체인에 저장하고, 실제 계약 내용은 오프체인에서 관리하여 옳은 비밀조각을 제출한 참여자들만이 계약 내용을 확인할 수 있도록 함으로써 해결 가능하다.

비밀공유 기법을 이용하면 다른 계약 참여자들의 동의가 있어야 계약 내용을 확인할 수 있기 때문에 권한 오남용 문제도 통제할 수 있다.

4.2 제안하는 기법에 대한 성능 분석

블록체인 네트워크의 성능 평가 기준은 초당 트랜잭션 수(Transaction Per Second, TPS)로, 블록 하나에 담긴 트랜잭션의 수와 블록 생성 시간을 이용하여 측정할 수 있다. 본 플랫폼의 네트워크 환경은 일반적인 이더리움 네트워크 환경과 동일하기 때문에 비밀공유 기법의 연산 수행 속도에 대해서만 성능 분석을 수행하였다.

표 1은 참여자 수를 100, 임계값을 20으로 설정하였을 때의 비밀공유 연산에 걸리는 시간을 나타낸 것이다. 수행을 반복할수록 유사한 결과 값이 나타나 5번의 수행 시간만을 기록하였다. 비밀공유 연산 수행 시간은 1~2초 정도로, 평균적인 웹 사이트의 응답 시간과 유사하게 측정되었다.

표 1. 비밀공유 연산 수행 시간
Table 1. Secret sharing operation execution time

Number of performances	Secret sharing operation execution time(ms)
1	1873
2	1852
3	1914
4	1836
5	1841
Average value	1863.2

V. 결 론

본 논문에서 제안하는 기법은 스마트 컨트랙트의 문제점으로 지적되어온 분산원장 환경이 갖는 공개 원칙에 따른 기밀성 부족 문제와 데이터의 확장성 문제를 해결할 수 있다는 강점이 있다.

비밀공유 기법을 적용한 스마트 컨트랙트 기술 기반의 계약 플랫폼에서는 블록체인에 데이터를 저장하기 때문에 계약을 진행하면서 생길 수 있는 계약서 위변조 가능성이 줄어들고, 위변조가 발생하였을 경우에도 이를 검증할 수 있다. 오프체인에 실제 데이터를 저장하고 계약 내용을 암호화한 해시값만을 스마트 컨트랙트를 이용하여 블록체인에 저장함으로써 데이터 저장 용량에 따라 급증하는 스마트 컨트랙트의 수수료 문제도 해결할 수 있다. 또한 오프체인에 데이터를 저장함으로써 블록체인에 저장할 데이터의 크기는 줄어들어 블록체인 노드에 저장하는 과정에서 빠른 처리 속도를 보여 신속한 계약 체결이 가능하다.

향후에는 스마트 컨트랙트를 이용한 계약 과정에서 계약 당사자 간의 서명을 대체할 수 있는 신원 증명 방안에 대해 연구하고자 한다.

References

[1] Ministry of Science and ICT holds the 2020 Blockchain pilot project initiation report meeting. https://www.gov.kr/portal/ntnadmNews/2156088 [accessed: Sept. 21, 2020]

[2] Employer who reported overdue wages to sue for falsification of labor contract. https://www.yna.co.kr/view/AKR20190808098800057 [accessed: Sept. 21, 2020]

[3] Chanjun Choi, Yeon-Joo Lim, Young-Jun Song, and Jong-Hyouk Lee, "Analysis of Off-Chain Solutions for Ethereum Scalability Issues", Proceedings of Symposium of the Korean Institute of communications and Information Sciences, Pyeongchang, Korea, pp. 208-209, Jan. 2019.

[4] A. Shamir, "How to Share a Secret", Communications of the ACM, Vol. 22, No. 11, pp. 612-613, Nov. 1979.

[5] S. Rouhani and R. Deters, "Security, Performance, and Applications of Smart Contracts: ASystematic Survey", IEEE Access, Vol. 7, pp. 50759-50779, Apr. 2019.

[6] K. Christidis and M. Devetsiokiotis, "Blockchains and Smart Contracts for the Internet of Things", IEEE Access, Vol. 4, pp. 2292-2303, May 2016.

[7] Korea Univ. Blockchain society KUBL, "Ethereum Basic", Book STAR, pp. 40-41, Nov. 2017.

[8] S. Underwood, "Blockchain Beyond Bitcoin", Communications of the ACM, Vol. 59, No. 11, pp. 15-17, Nov. 2016.

[9] Andreas M. Antonopoulos, "Mastering Ethereum: Building Smart Contracts and DApps", JPub, pp. 31-32, pp. 141-143, May 2019.

[10] "A Next-Generation Smart Contract and Decentralized Application Platform", https://github.com/ethereum/wiki/wiki/White-Paper. [accessed: Aug. 05, 2019]

[11] Heeyoul Kim, "Analysis of Security Threats and Countermeasures on Blockchain Platforms", Journal of Korean Institute of Information Technology, Vol. 16, No. 5, pp. 103-112, May 2018.

저자소개

손 애 선 (Ae-Seon Son)

2018년 2월 : 금오공과대학교 컴퓨터공학과(공학사)
2019년 6월 ~ 현재 : (재)경주스마트미디어센터 연구원
2020년 9월 ~ 현재 : 동국대학교 전자통신공학과(공학석사)
관심분야 : 블록체인, 인공지능

유 수 빈 (Soo-Bin Yoo)

2012년 2월 : 동국대학교 정보통신공학과(공학사)
2016년 8월 : 동국대학교 전자통신공학과(공학석사)
2016년 12월 ~ 현재 : (재)경주스마트미디어센터 주임연구원
2020년 3월 ~ 현재 : 동국대학교 전자통신공학과(공학박사)
관심분야 : 인공지능, 머신러닝, 데이터통신

조 정 화 (Jung-Hwa Jo)

2015년 2월 : 대구대학교 사회학과(사회학사)
2019년 6월 ~ 현재 : (재)경주스마트미디어센터 연구원
2020년 9월 ~ 현재 : 동국대학교 전자통신공학과(공학석사)
관심분야 : 블록체인, 인공지능, 데이터통신

유 수 민 (Su-Min Yoo)

2019년 2월 : 울산과학대학교
컴퓨터정보학부 소프트웨어
개발전공(공학전문학사)
2019년 6월 ~ 현재 :
(재)경주스마트미디어센터 연구원
관심분야 : 블록체인, 인공지능