Journal of the Korea Academia-Industrial cooperation Society
Vol. 19, No. 4 pp. 563-570, 2018

https://doi.org/10.5762/KAIS.2018.19.4.563
ISSN 1975-4701 / eISSN 2288-4688

신뢰성 향상을 위한 이더리움 블록체인 기반의 온라인 투표 시스템

김철진
인하공업전문대학 컴퓨터시스템과

An Online Voting System based on Ethereum Block-Chain for Enhancing Reliability

Chul-Jin Kim
Dept. of Computer Systems and Engineering, Inha Technical College

요 약 기존의 온라인 투표가 보안 위협에 대한 불확실성 때문에 공적 선거에 활용되지 못하고 있으며 오프라인 투표로 인해 막대한 비용이 소요되고 있다. 이에 대한 대안으로 블록체인 기술이 대두되고 있다. 블록체인 기술을 온라인 투표에 적용하면 투표자 정보 및 집계 정보를 분산 관리하므로 투명성과 기밀성이 보장될 수 있을 것이다. 블록체인은 투표 정보에 대해 분산관리하므로 기존의 중앙 서버 기반의 온라인 투표 시스템보다 보안 위협으로 부터 안전할 것이다. 이와 같이 블록체인 기술이 공적 선거에 적용되어 투표 정보의 투명성과 기밀성이 보장된다면 투표로 인한 비용을 획기적으로 감소시킬 수 있을 것이다. 본 연구에서는 블록체인 기술 중에 이더리움 기술을 온라인 투표시스템에 적용 개발하고자 한다. 이더리움은 확장성이 뛰어난 블록체인 기술로서 솔리디티 언어 기반의 스마트 컨트랙트를 제공한다. 이더리움의 스마트 컨트랙트를 이용하여 온라인 투표 컨트랙트를 개발하고 각 투표자에게 컨트랙트를 배포한다. 각 투표자는 배포 받은 컨트랙트에 투표하며 투표한 집계는 다른 투표자들에게 분산 저장된다. 실험에서는 저장된 투표 집계 정보에 대해 일관성을 검증한다.

Abstract Existing online voting is not being used for public elections due to uncertainty about security threats, and offline voting costs a lot of money. As an alternative, blockchain is emerging. Applying blockchain technology to online voting will ensure transparency and confidentiality, because voter information and aggregate information are distributed and managed. Since a blockchain distributes the voting information, it will be more secure than existing central server‐based online voting systems. If blockchain technology is applied to public elections, and the transparency and confidentiality of the voting information is guaranteed, the cost of voting will be greatly reduced. This paper tries to apply to an online voting system the Ethereum platform from among the blockchain technologies. Ethereum is a highly scalable blockchain technology that provides a smart contract based on the Solidity language to develop an online voting contract and to distribute the contract to each voter. Each voter votes on the contract that has been distributed, and the votes are distributed to other voters. The experiment verifies the consistency of the stored voting information.

Keywords : Block Chain, Ethereum, Smart Contrct, Solidity, Online Voting System

1. 서론

현재 세계적으로 블록체인(Block Chain) 기술에 대한 연구가 활발하게 진행되고 있으며 사물인터넷, SNS[1], 의료[2], 저작권, 등 다양한 분야에 적용하고 있다[3]. 특히 신뢰성, 보안성 논란으로 공적 선거를 온라인으로 시

*Corresponding Author : Chul-Jin Kim (Inha Technical College)
Tel : +82-32-870-2338 email : cjkim@inhatc.ac.kr
Received February 5, 2018 Revised (1st February 26, 2018, 2nd March 2, 2018)
Accepted April 6, 2018 Published April 30, 2018

행되는데 확신할 수 없었으나, 블록체인 기술의 보안성에 대한 확인으로 공적 선거를 온라인으로 시행하려는 시도가 진행 중이다[4,5].

본 논문에서는 분산원장 기술을 적용한 기존의 서버 기반의 온라인 투표 시스템[4]에 대응하는 블록체인 기반의 온라인 투표시스템을 개발한다. 블록체인 기술 중에 이더리움(Ethereum) 플랫폼을 이용하며, 투표자는 이더리움 클라이언트 모듈을 설치하여 투표한다. 투표 시스템의 개발은 이더리움의 솔리디티(Solidity) 언어 기반 스마트 컨트랙트(Smart Contract)를 이용한다.

1장에서는 본 연구의 동기를 설명하고, 2장에서는 관련 연구로서 블록체인 기술과 본 연구의 핵심 기술인 이더리움, 그리고 이더리움의 스마트 컨트랙트에 대해 파악하며, 기존의 분산원장 기술을 적용한 온라인 투표시스템의 보안 위협 및 대응 방안에 대해 분석한다. 3장에서는 이더리움 네트워크 구성과 스마트 컨트랙트 개발을 수행한다. 4장에서는 온라인 투표 결과가 분산 저장됨을 확인하며, 5장에서 결론 및 향후 연구과제를 제시한다.

2. 관련연구

2.1 블록체인

블록체인 기술은 거래정보의 원장(Ledger)을 모든 노드(사용자)들이 동일하게 분산 관리하는 기술이다. 원장의 거래정보를 블록이라고 하며 일정 시간 단위(비트코인은 10분, 이더리움 12초)로 새로운 거래정보를 취합하여 블록을 생성한다. 생성된 블록은 이전 블록에 연결되며, 이러한 과정이 모든 연결된 블록(블록 체인)들에게 이루어진다[6]. 새로 생성된 블록이 이전 블록들의 거래 내용을 포함해야 하므로 기존 거래정보에 대한 유효성을 검증 한다. 이것을 작업 증명(Proof of Work) 이라고 한다. 새로운 블록을 공유한 후 처음 블록부터 마지막 블록까지 작업증명이 완료되어야 만 해당 새로운 블록을 확정한다.

nonce	549304
Transaction	...
Before Hash (Address)	3452dabcd43232e
Current Hash (Address)	a940f3bed899dec9

Fig. 1. Block Information

블록의 구조는 Fig. 1과 같이 거래 내역(Transaction)과 이전 블록의 주소값(Before Hash), 현재 블록의 주소값(Current Hash), 그리고 작업 증명을 위한 넌스(nonce)로 구성된다.

작업 증명은 현재 블록의 해시(Current Hash)을 찾기 위해 넌스(nonce) 값을 찾는 과정이다. 이렇게 넌스를 찾으면 암호화폐가 지급되며 이러한 과정을 채굴(Mining)이라고 한다.

2.2 이더리움

이더리움은 프로그래밍이 가능한 블록체인으로 다양한 블록체인 서비스를 개발하여 추가할 수 있다. 이더리움은 분산 네트워크 플랫폼으로 클라이언트 프로그램으로 운영 가능하다[6,7].

이더리움의 화폐 단위는 ether로서 거래를 위한 수수료로 사용된다. 이더리움에서는 금융 거래에서는 프로그래밍이 필요 없지만, 스마트 컨트랙트를 위해서 프로그래밍을 지원한다. 발신자와 수신자 사이에 트랜잭션(금융거래)이나 컨트랙트(비금융거래) 형태로 전달하며, 전달 시 시스템을 보호하는 차원에서 수수료인 ether를 함께 전달한다. 화폐의 종류는 일반 거래를 위한 ether, 소액 결제를 위한 finney, 거래 시 수수료 지급을 위한 szabo, wei 가 있다[6].

개발되는 컨트랙트는 블록에 포함되어 블록체인의 다른 블록에 전달되어 검증 시 실행된다[Fig. 2].

2.3 스마트 컨트랙트

스마트 컨트랙트는 이더리움에서 탈중앙화 어플리케이션(DApp, Decentralized Application)을 개발하기 위한 프로그램이다. 다른 암호화폐가 주로 화폐로서의 기능을 제공하지만, 이더리움은 스마트 컨트랙트를 이용하여 다양한 응용 분야에 적용 가능하다.

스마트 컨트랙트는 이더리움 종속적인 언어인 솔리디티 언어를 이용하여 개발할 수 있으며 Fig. 2와 같이 블록 내에 스마트 컨트랙트가 포함된다. 블록의 거래 내역이 공유되는 것과 같이 컨트랙트도 계정 간에 공유되어 분산저장 된다.

스마트 컨트랙트 실행은 분산 환경의 각 이더리움 클라이언트(Ethereum Client) 내에 가상환경(EVM, Ethereum Virtual Machine) 위에서 실행될 수 있도록 바이트 코드(Byte code)로 컴파일 되어 배포된다.

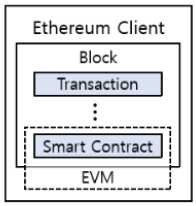


Fig. 2. Smart Contract Runtime Environment in Ethereum

2.4 분산 원장 기술 적용 온라인 투표에 대한 보안 위협 및 대응 방안

연구 [8]은 분산 원장 기술(DLT, Distributed Ledger Technology)에 기반한 온라인 투표를 위한 시스템 모델과 표준화 동향, 그리고 보안 위협 및 대응 방안을 제시한다. 온라인 투표에서 발생할 수 있는 보안 위협인 데이터 기밀성 위협, 데이터 무결성 위협, 서비스 가용성 위협, 등을 제시하였으며 대응 방안은 분산 원장 기술이 적용된 시스템 모델 기반으로 제시하였다.

연구 [8]은 분산 원장 기술을 적용한 온라인 투표 시스템 모델을 제시하였으나, 서버(투표 서버, 선거인 명부 서버, 데이터베이스 서버, 등) 기반의 서비스 모델로서 블록체인 분산원장 기술[9]의 온라인 투표 시스템과 차이를 보인다.

이에 본 연구에서는 중앙 서버 기반의 온라인 투표 시스템 관리로 인한 보안 위협에 대응할 수 있는 이더리움 블록체인 기반의 탈중앙화된 온라인 투표 시스템을 개발한다.

3. 블록체인 기반의 온라인 투표 시스템

본 연구에서는 이더리움 블록체인 기술을 활용하여 온라인 투표 시스템을 개발 한다. 블록체인 네트워크를 구성하고 온라인 투표 컨트랙트(솔리디티 기반)를 개발하여 투표자 간에 투표 결과가 분산 저장되도록 한다. 이렇게 분산 저장된 집계 결과의 일관성으로 인해 신뢰성이 보장될 수 있을 것이다.

3.1 블록체인(이더리움) 네트워크 구축

이더리움은 분산 네트워크 플랫폼으로서 클라이언트 프로그램만 있으며, Java나 Python 등 다양한 언어로 개발된 이더리움 플랫폼이 있다. 본 연구에서는 구글의 Go 언어로 개발된 Go Ethereum(Geth)[10]플랫폼을 구축하여 투표시스템을 적용한다.

– 계좌생성

이더리움에서는 계좌를 생성하여 사용자를 구분할 수 있다. 아래와 같은 명령어를 통해 16진수 형태의 주소로 계좌가 생성된다.

```
C:\Geth>geth --datadir "c: \ethereum \data"
         account new
```

```
C:₩Geth>geth --datadir "c:₩ethereum₩data" account new
Your new account is locked with a password. Please giv
t this password.
Passphrase:
Repeat passphrase:
Address: {1e4b994987e3886f2fffff198999867399fa9a34}
```

Fig. 3. Account Address

계좌는 사용자 별로 만들 수 있으며 본 투표 시시스템의 투표자가 계좌가 될 수 있을 것이다. 계좌 정보 리스트는 Fig. 4와 같이 16진수 주소값으로 관리된다.

```
C:\Geth>geth --datadir "c:\ethereum\data"
         account list
```

```
c:₩geth>geth -datadir "c:₩ethereum₩data" account list
Account #0: {eb2f3c12d73a49a40b72f29eada2e10a42b89728}
TC--2018-01-25T03-07-27.266796100Z--eb2f3c12d73a49a40b7
Account #1: {666b9869c4b7be918bf58c69135547ec439c5747}
TC--2018-01-25T03-21-58.752558300Z--666b9869c4b7be918bf
Account #2: {111c98f5fc33250b5633388c71145948e6bdfa9e}
TC--2018-01-25T03-22-17.586985800Z--111c98f5fc33250b563
```

Fig. 4. Account List of Voters

– 최초블록 생성

블록들 간의 블록체인을 형성하기 위해 최초 블록이 만들어져야 하며 최초블록을 제니시스(Genesis) 블록이라고 한다. 이더리움에서는 제니시스 블록을 생성하기 위해 Fig. 5와 같이 JSON파일 형태로 정의한다.

```
{
 "nonce" : "0x0000000000000054",
 "timestamp" : "0x0",
 "parentHash" : "0x00000000000...0000000000000",
 "extraData" : "0x00",
 "gasLimit" : "0x400000",
 "difficulty" : "0x3000",
 "mixhash" : "0x00000000000000...0000000000000",
 "coinbase" : "0x4000000000000000000000000000",
 "alloc" : {  ...  }
}
```

Fig. 5. Genesis Creation File in Ethereum (ElectionGenesis.json)

최초블록인 제니시스 블록을 생성하기 위한 파일에 포함되는 정보는 다음과 같다.

- nonce : 작업증명을 수행한 값으로 현재 블록을 찾기 위해 mixhash 값과 조합을 통해 계산
- timestamp : 현재 블록이 얻어진 시간
- parentHash : 전 블록의 해시값으로 제니시스 블록은 최초의 블록이므로 0
- gasLimit : 하나의 블록이 담을 수 있는 gas의 최대 크기. 거래가 이루어 질 수 필요한 화폐
- difficulty : 블록 생성을 위한 계산의 난이도

Fig. 5의 제니시스 블록 생성 파일을 아래의 초기화 작업에 의해 제니시스 블록이 생성된다.

```
C:\Geth>geth --datadir "c:\ethereum\data" init "c:\ethereum\ElectionGenesis.json"
```

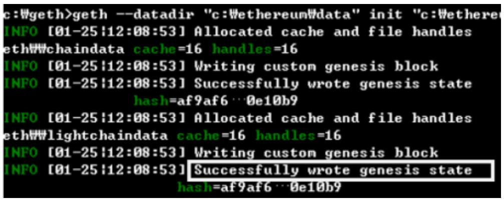


Fig. 6. Creating Genesis Block(ElectionGenesis.json)

– 이더리움 네트워크 실행

본 연구는 공용 네트워크가 아닌 사설 네트워크에서 이더리움을 실행한다[11]. Fig. 7에서와 같이 Geth 명령을 통해 사설 네트워크가 실행되며 안정적으로 서비스가 구동되며 명령 대기 상태(“>”)가 된다

```
C: \geth>geth --identity "PrivateNetwork" --datadir "c:\ethereum \data" --port "30303" --rpc --rpcaddr 0.0.0.0 --rpcport "8123" --rpccorsdomain "*" --nodiscover --networkid 1900 --nat "any" --rpcapi "db,eth,net,web3,miner" console
```

Fig. 7. Running Ethereum Network

– 채굴

컨트랙트의 거래를 가능하게 하기 위해 암호를 해독하는 과정인 채굴이 실행되어야 한다. Fig. 8에서와 같이 채굴을 실행 시키면 암호를 해독하는 과정이 실행되어 보상으로 ether를 얻게 된다.

```
> miner.start()
```

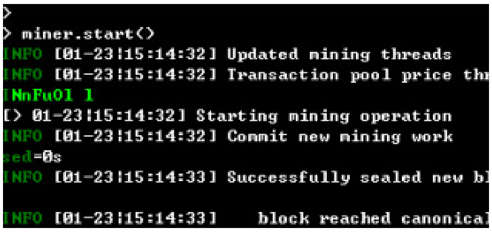


Fig. 8. Mining in Ethereum Network

지금까지 분산장부를 통한 온라인 투표 시스템을 개발하기 위한 이더리움 플랫폼을 구축하였다.

3.2 온라인 투표 시스템 개발

이더리움에서는 암호화폐 외에 비금융 분야의 정보(계약서,SNS,저작권, 등)에 대한 거래를 적용하기 위해 스마트 컨트랙트라는 확장 기술을 제공한다. 본 연구의 온라인 투표 시스템에 대해 솔리디티 프로그래밍 언어를 이용하여 온라인 투표 컨트랙트를 개발한다.

– 온라인 투표 컨트랙트 개발

이더리움 기반의 온라인 투표 시스템의 아키텍쳐는 Fig. 9에서과 같이 이더리움 클라이언트(Go Ethereum)의 블록 내에 온라인 투표에 해당하는 선거 컨트랙트(Election Contract)를 개발 배포하여 노드들 간에 블록의 일관성을 유지시킨다.

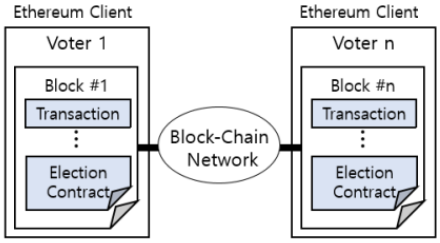


Fig. 9. Online Voting System Architecture based on Ethereum

온라인 투표 컨트랙트를 개발하기 위한 IDE로서 Fig. 10에서와 같이 이더리움 지갑(Ethereum Wallet)에 해당하는 Mist 브라우저[12]를 이용하며, Fig. 11에서와 같이 온라인 투표 컨트랙트를 개발 및 컴파일 할 수 있다.

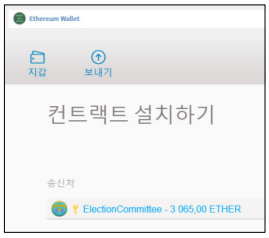


Fig. 10. Contract Creation in Mist Browser(Ethereum Wallet)

Fig. 11. Development and Compile in Mist Browser

Fig. 12는 솔리디티 프로그래밍 언어를 이용하여 온라인 투표 컨트랙트를 개발한 코드이다. 솔리디티로 개발된 온라인 투표 컨트랙트는 이더리움 네트워크 상에서 사용자 계정에 전달되는 프로그램이다.

```
pragma solidity ^0.4.18;

contract ElectionContract {
   …
   function vote(string candidate) public {
      vCount[candidate]++;
      voter[msg.sender] ····· = true;
   }
   function voteClosed() public returns(bool) {
      if(voter[msg.sender]) return true;
      else return false;
   }
   function voteCount(string candidate) private
            returns(uint) {
       if(voterRole[msg.sender])
            return vCount[candidate];
       else return 0;
   }
   …
}
```

Fig. 12. Developed Online Voting Contract by Solidity

온라인 투표 스마트 컨트랙트의 기능은 투표 기능(vote()), 투표 확인 기능 (voteClosed()), 집계 확인 기능 (voteCount()) 등으로 구성된다. 투표를 하게 되면 후보자의 성명(candidate)을 투표자 계정에서 입력하며 해당 투표자의 vCount 가 집계된다. 투표자가 중복 투표를 하지 않도록 투표를 한 사용자 계정(msg.sender)을 true로 저장하여 투표했음을 기록한다. 각 계정이 가지고 있는 컨트랙트에서는 투표가 된 경우 true로 기록되어 있어 중복 투표를 할 수 없다(voteClosed()). 투표 집계(voteCount())에 대한 확인은 후보자의 성명을 입력하면 현재의 투표 집계(vCount)를 확인할 수 있다. 본 논문에서 집계현황은 가시성을 private으로 했기 때문에 확인할 수 없으나 검증을 public으로 공개하여 확인한다.

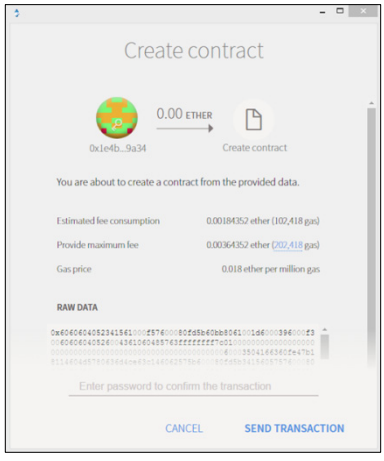


Fig. 13. Creating Online Voting Contract

생성된 온라인 투표 컨트랙트를 Fig. 13과 같이 생성되어 투표자(Account)들에게 배포(Deploy)된다. Fig. 14에서와 같이 현재 투표자의 전자지갑에 온라인 투표 컨트랙트(Election Contract)가 배포된 것을 확인할 수 있다.

Fig. 14. Deployed Online Voting Contract

4. 실험 및 평가

지금까지 이더리움 플랫폼 기반의 온라인 투표 시스템을 개발하기 위해 이더리움 플랫폼을 구축하였으며, 온라인 투표 컨트랙트를 개발하여 투표자의 전자지갑에 배포하였다.

본 실험에서는 구축된 이더리움 플랫폼 환경과 개발된 온라인 투표 컨트랙트를 기반으로 블록체인 기반의 온라인 투표가 분산원장으로서 역할을 통해 신뢰성을 확보할 수 있는지 검증 한다.

– 투표자 등록

Mist 브라우저를 통해 투표자 계정을 등록할 수 있으며 Fig. 15와 같이 선거관리 위원회(Election Committee) 계정과 투표자(Voter 1, Voter 2) 계정을 등록한다.

Fig. 15. Election Committee and Voter Account

선거관리 위원회는 투표자들에게 온라인 투표 컨트랙트를 전송하기 위해서 ether를 송금한다. 이렇게 ether가 송금된 사용자들과 블록체인을 형성하여 투표권을 부여받을 수 있다. Fig. 16에서와 같이 선거관리 위원회에서 투표자들에게 10 ether를 전송한다.

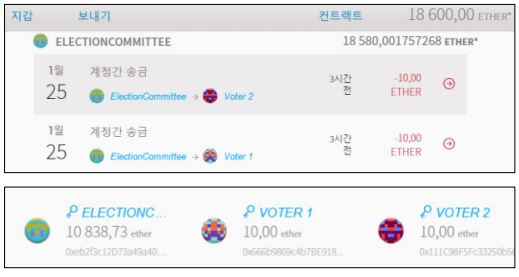


Fig. 16. Chain Organization between Accounts

– 투표 컨트랙트 전송

투표자와 체인을 형성한 상태에서 투표 컨트랙트(Election Contract)를 전송하면 Fig. 17과 같이 투표 정보(투표 상태, 후보자 정보(성명), 투표 집계)가 제공된다. 실제 선거에서 투표집계는 비공개호 해야 한다.

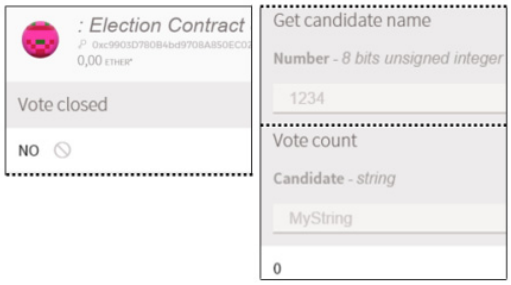


Fig. 17. Executed Election Contract

컨트랙트는 후보자를 등록하거나 투표를 하는 기능을 포함하고 있다. Fig. 18에서 같이 권한에 따라 후보자를 등록하거나 일반 투표자는 투표하는 기능을 제공 받는다.

Fig. 18. Functions of Election Contract

선거관리 위원회에서는 투표 컨트랙트 기능의 'register' 기능을 통해 후보자를 등록할 수 있다. 등록이 되면 다른 투표자들에게 전송하여 장부의 일관성을 맞춘다. Fig. 19에서과 같이 거래를 전송하면 등록된 모두 투표자의 블록에 후보자 정보가 추가된다.

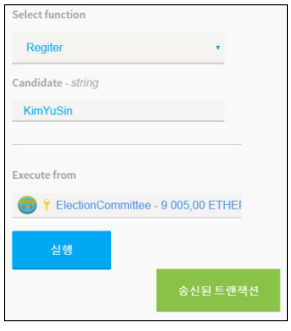


Fig. 19. Transfering Candidate Information to Block of Voters

– 투표

투표자들의 블록으로 전송된 후보자 정보를 통해 투표를 수행할 수 있다. Fig. 18의 컨트랙트 기능 중에 'Vote' 기능을 통해 Fig. 20과 같이 투표를 수행한 후 전송할 수 있다. 현재의 투표 컨트랙트에 포함된 선거관리 위원회와 투표자들에게 모두 전송되어 분산 저장된다.

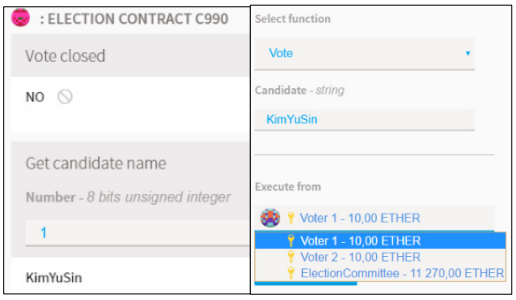


Fig. 20. Vote Function of Election Contract

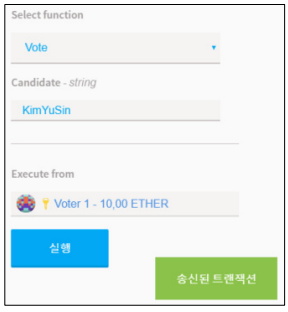


Fig. 21. Transfering Voting Transaction to Block of Voters

Fig. 21과 같이 투표 트랜잭션에 대해 다른 투표자의 블록으로 전송하며 모든 투표 컨트랙트에 포함된 투표자들에게 동일하게 일관성을 유지 시킨다. Fig. 22는 투표 트랜잭션에 대한 전송 기록이다.

Fig. 22. Transfer History of Voting Transaction

– 투표 집계 일관성

Fig. 23과 같이 투표 컨트랙트에 포함된 투표자 간에 투표 블록에 대한 정보가 공유되므로 일관성이 유지됨을 확인할 수 있다. 만약 투표자가 중복해서 투표하더라도 블록에 투표한 기록이 있기 때문에 중복 집계되지 않는다.

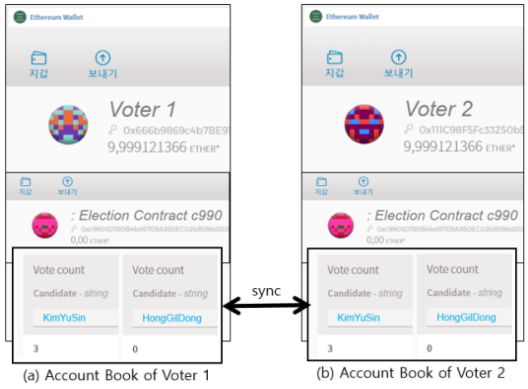


Fig. 23. Block Consistency between Voters

투표자가 투표를 조작하는 것은 Fig. 23에서와 같이 블록체인 작업 증명에 의해 조작이 불가능하다. 그러나 만약 개인 증명이 되지 않은 사용자에 의한 불법 투표 위협에 대해서는 선거관리 위원회에 의해 가상화폐를 송금 받은 투표자만 블록체인이 형성되어 투표권이 부여되므로, 해당 투표 컨트랙트와 블록이 형성되지 않은 투표자는 Fig. 24에서와 같이 투표자 명단에 포함될 수 없으며 불법 투표가 불가능하다.

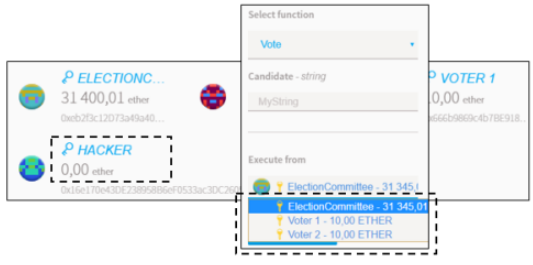


Fig. 24. Non-Voting Voter in Election Contract

5. 결론

본 연구에서는 블록체인 기술 중에 이더리움을 이용하여 온라인 투표 시스템을 개발하였다. 온라인 투표 시

스템을 개발하기 위해 솔리디티 언어 기반의 스마트 컨트랙트를 개발하였으며 해당 투표 컨트랙트를 투표자 간에 배포하여 투표 집계가 신뢰성 있게 유지됨을 검증하였다. 그러나 솔리디티 기반의 스마트 컨트랙트를 블록체인 내의 계정으로 배포가 된 후에는 수정이 불가능하여 유지보수 측면에서 어려움이 있을 것으로 판단된다.

향후에는 이더리움의 스마트 컨트랙트를 개발하기 위한 솔리디티 언어와 기존 개발 언어와의 차이에 대한 분석과 변환 알고리즘을 연구하여 스마트 컨트랙트 개발의 접근성과 유지보수성을 높일 수 있는 방안을 제시하고자 한다.

References

[1] Steemit, "https://steemit.com/", (accessed Jan., 10. 2018)

[2] H. S. Park, J. W. Chung, and U. M. Kim, "A Study On Shared EMR(Electronic Medical Record By Block Chain(Ethereum)", *Proceedings of KIIT Summer Conference*, 436-437, December 2017.

[3] Y. S. Ko and H. S. Choi, "Changing Business Paradigm and Its Application - Focused on the Block Chain Technology", *Korea Science & Art Forum*, 27, 2017.

[4] Internet newspaper of CryptoCoinsNews, “https://www.cryptocoinsnews.com/blockchain-tech-enables-utah-republicans-vote-candidate/,” Mar. 2016. (accessed Jan., 12. 2018)

[5] Internet newspaper of CoinDesk, “http://www.coindesk.com/libertarian-party-texas-logs-votes-presidential-electors-blockchain/,” Apr. 2016.
(accessed Jan., 12. 2018)

[6] R. James, "A Next-Generation Smart Contract and Decentralized Application Platform", "https://github.com/ethereum/wiki/wiki/White-Paper", 2017.
(accessed Jan., 15. 2018)

[7] A. Watanabe, Y. Matsumoto, Y. Nishimura, and T. Shimizu, "Block Chain, Introduction to Application Development", 2017.

[8] K. Park, C. O. Kim, and H. Y. Youm, "Countermeasures against Security Threats to Online Voting Using Distributed Ledger Technology", *Journal of the Korea Institute of Information Security & Cryptology*, vol. 27, no. 5, pp. 1201-1216, 2017.
DOI: http://doi.org/10.13089/JKIISC.2017.27.5.1201

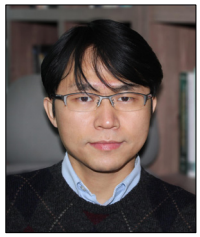
[9] Internet homepage of ISO/TC 307 Blockchain and distributed ledger technologies, “https://www.iso.org/committee/6266604.html,” Sep. 2016.(accessed Jan., 3. 2018)

[10] Go Ethereum(Geth), "https://ethereum.github.io/go-ethereum/downloads", (accessed Dec., 4. 2017)

[11] S. H. Jo, J. B. Lee, J. Y. Park, D. G. Lee, and H. IN, ETHEREUM BASIC, BookStar, 2017.

[12] Mist and Ethereum Wallet, "https://github.com/ethereum/mist/releases", Jan 2018. (accessed Dec., 4. 2017)

김 철 진(Chul-Jin Kim) **[종신회원]**



- 2004년 2월 : 숭실대학교 대학원 컴퓨터학과 (공학박사)
- 2004년 3월 ~ 2009년 2월 : 삼성전자 책임연구원
- 2009년 3월 ~ 현재 : 인하공전 컴퓨터시스템과 부교수

<관심분야>
객체/컴포넌트/서비스 커스터마이제이션, 모바일 서비스, 아키텍쳐 리팩토링, 블록체인, 이더리움