모바일 투표 Dapp 실행 및 보안 이슈

KHERLEN NARANTUYA · 박준범 · 박종서*
한국항공대학교

A Mobile Voting DApp Implementation and Security Issues

KHERLEN NARANTUYA · Park Jun Beom · Park Jong Sou*
Korea Aerospace University
E-mail : col.herlen@gmail.com / mvpup777@naver.com / jspark@kau.ac.kr

요 약

블록체인과 비트코인의 등장 이후에 스마트 컨트랙트를 갖춘 퍼블릭 블록체인 이더리움이 시작되면서 decentralization 이 전세계적으로 가속화 되고 있다. 개발자들은 이더리움의 블록체인 개발 플랫폼을 활용하여 분산화된 P2P 네트워크에서 실행되는 "분산화된 응용 프로그램 (DApp)" 을 개발할 수 있고, IoT 부터 모바일까지 다양한 유형의 기기가 블록체인 분산 환경에 참여 할 수 있다. 블록체인 과 스마트 컨트랙트와 상호 작용 할 수 있는 방법은 많이 있지만, 사용자들은 편의성과 접근성의 장점으로 인해 모바일방식을 선호하는 경향이 있다. 그렇기에 저자는 Android 기반 투표 DApp 을 개발하였고, 그에 관련된 이슈를 연구하였다. 현재는 DApp의 개발방법이 적절하게 연구되고 표준화 되어 있지 않기 때문에, 사용자 친화적인 DApp을 개발하기 위한 효율적인 방법을 연구하였다. 특히 DApp은 블록체인과 상호작용 하기 위해서 일정량의 수수료를 소비해야하기 때문에 Smart Contract 코드의 수수료 문제 및 코드의 Security 문제에 대해서 집중적으로 조사하였고, 본 논문에서는 이를 소개하고자 한다.

ABSTRACT

Since the advent of blockchain and bitcoin, decentralization has been accelerating around the world as a public blockchain ethereum with smartcontract has begun. Developers can use Ethereum's blockchain development platform to develop "distributed applications" (DApp) running on a decentralized P2P network, and various types of devices from IoT to mobile can participate in a block-chain distributed environment have. Using Ethereum's blockchain development platform, developers can develop "Decentralized Application (DApp)" that run on a decentralized P2P network and various types of devices from IOT to mobile can participate in distributed blockchain environments. There are many ways to interact with the blockchain and the smart contract, but users tend to prefer the mobile methods due to their convenience and accessibility advantages. Therefore, the author developed an Android based voting DApp and researched related issues. Since the current development methods of DApp are not adequately researched and standardized, efficient methods for developing user-friendly DApp were studied. Because DApp has to spend a certain amount of fees to interact with blockchain, it has intensively investigated the gas problem of Smart Contract code and the security problem of code, and author would like to introduce it in this paper.

키워드

Blockchain, DApp, Mobile, Security, Ethereum, Smart Contract

Ⅰ. 서 론

Blockchain은 최근 몇 년 동안 대중들로부터 큰 관심을 받고 있다. 블록체인의 기술적 장점은 금융, 의료, 사업, 시장, 심지어 무결성과 기밀성이 필수적인 공공 서비스에도 적용될 수 있다. 전통적으로 데이터나 자산의 이동은 검증되거나 승인되기 위해 제3자를 거쳐야 하는데, 이는 경우에 따라서는 적절하지 않거나 추가적인 비용이 발생할 수 있다. 하지만 블록체인 기술은 이러한 문제에 대한

* Corresponding author

편리한 해결책이 될 수 있다. 블록체인 제안 방법론은 간단히 말해서, P2P 토폴로지를 기반으로 하는 네트워크의 분산된 노드로 구성된다. Satoshi Nakamoto라는 인물이 "비트코인: 2008년 Peer-to-Peer Electronic Cash System" 연구 논문[1]을 발표하였고, 2008년 말 비트코인 암호화를 위한 최초의 블록체인이 구축됐다. 비트코인의 뒤를 이어서 이더리움이라는 public blockchain 이 2015년에 등장하였다. 이더리움 에서는 비트코인의 블록체인 이론에 더해 Nick Szabo의 Smart Contract 개념을 추가하였다. 그 결과 이더리움의 스마트 컨트랙트의 특성을 통하여 분산 어플리케이션(DApp) 이라는 새로운 용어를 만들었다. DApp은 제 3자 또는 중계자의 역할이 필요없으며, 사용자들이 공개적으로 수수료만 지급하여 사용할 수 있다. 본 논문에서는 이러한 DApp의 수수료 문제 및 보안 이슈에 대해서 조사하였다.

II. 모바일 투표 DApp

2-1 Keystore / Wallet File

일반적으로 DApp들 에서는 이더리움 블록체인에서 배포된 스마트 컨트랙트와 상호작용 하기 위해서 이더리움 계정을 가져야한다. 이더리움 계정은 공개키와 개인키의 암호 쌍이다. 우선 무작위 256비트 숫자를 생성하고, 이 숫자에 ECC(Eliptic Curve Cryptography)[2] 를 사용하여 512비트의 공개키를 생성한다. 다음 단계에서는 공개키를 Keccak256 암호화 하여 해시값의 160비트 만을 가져온다.

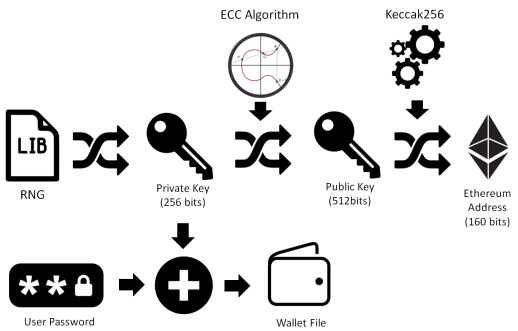

그림 1. 이더리움 계좌 주소 및 지갑 생성

2-2 Android DApp

투표 DApp에서 스마트 컨트랙트는 솔리디티 언어를 통해서 구성하였고, DApp 은 안드로이드 스튜디오를 통하여 개발하였다. 블록체인과 App 의 연결은 Web3.j 라이브러리[3]를 통하여 상호작용한다. 그림 2는 Main Activity 와 비동기 작업 간의 흐름을 보여준다. 이 두 흐름 사이의 상호작용을 보다 효율적으로 처리하기 위해, 비동기 작업 내에 사용자 정의 인터페이스 와 그 구현을 추가적으로 구성하였다.

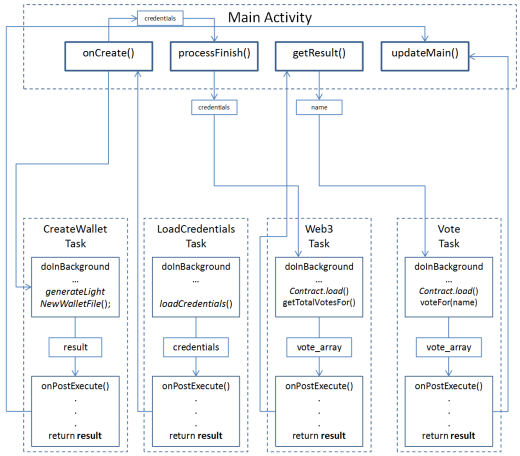

그림 2. Main Activity와의 상호작용

2-3 DApp User Interface

투표 DApp은 사용자의 이더리움 계정과 연결되어야 하므로 시작하면서 지갑/키스토어 파일에 대한 암호를 제공하여야한다. 사용자의 지갑과 싱크가 되고나서 투표를 하면서, 일정량의 수수료, Gas를 보내면 결과가 블록체인에 기록된다.

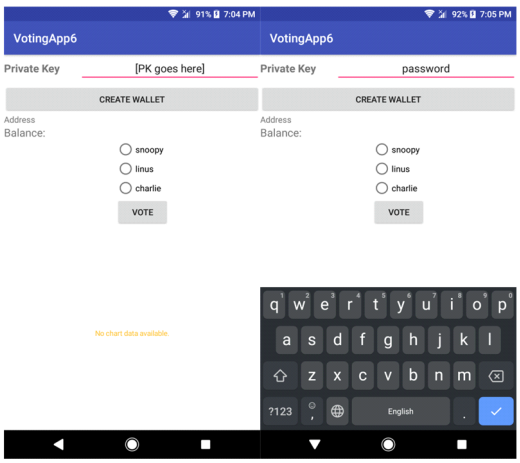

그림 3. DApp의 User Interface

III. DApp의 보안 이슈

3-1 Handling Keystore/Wallet File

이더리움의 지갑파일은 JSON 형식으로 되어있으며, 사용자의 암호화된 개인키를 포함하고 있다.

이러한 지갑 비밀번호는 계정 보안에 결정적인 영향을 미친다. 안드로이드 환경으로 구축된 DApp에서는 그림4와 같은 데이터를 저장하는 여러 옵션이 있다. 사용자 암호를 일반 텍스트 형식으로 저장하는 것은 좋은 방법이 아니나, 해시와 Private 설정의 조합은 신뢰할 수 있고 효율 적인 방법이 될 수 있다.

Storage type	Description
Internal File Storage	System provides a private directory on the file system for each app.
External File Storage	Physical removable storage space, accessible by other apps
Shared Preferences	Suitable for simple, small amount of data. It allows read/write persistent key-value pairs of primitive data types.
Databases	Android provides SQLite databases and they're accessible only by your app.

그림 4. Option to store sensitive data

3-2 Wrapper Class Issue

일반적으로 DApp 은 wrapper object를 통하여 다른 플랫폼 에서도 스마트 컨트랙트와 상호작용하게 된다. 자동 생성된 wrapper class는 안드로이드 프로젝트 폴더에 포함되어있으며, wrapper 에서는 트랜잭션을 호출하거나 이벤트 및 기능에 대한 일반적인 작업을 지원한다.

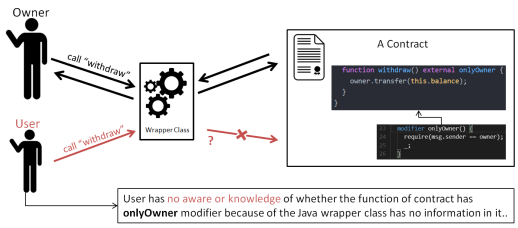

그림 5. Wrapper Issue

Ⅳ. DApp의 수수료 문제

DApp의 함수 호출에 있어서 아래그림에서는 성공한 함수 호출과 실패한 함수 호출에 대한 가스 소비량을 보여준다. 이더리움의 Remix[4]에서 스마트 컨트랙트를 컴파일하며, 프라이빗 블록체인 Ganache[5] 에 배포하여 테스트 하였다. 적당한 가스가 소비되지 않을 경우에는 함수 호출은 실패하며, 이러한 가스 소모량을 줄이기 위해서는 스마트 컨트랙트를 배포할 때 솔리디티의 최적화 및 추가 데이터 저장에 대한 부분을 조정하여야 한다.

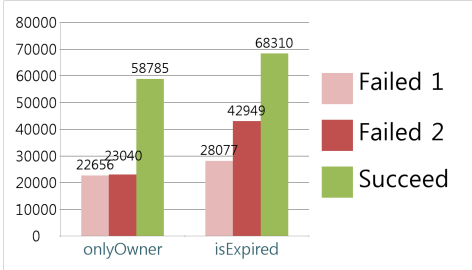

그림 6. Gas 소비량

Ⅴ. 결 론

이 논문에서는 안드로이드 환경의 모바일 분산 어플리케이션(DApp) 에 대한 연구를 실시하였다. 블록체인, 스마트 컨트랙트와 상호작용하는 많은 방법이 있지만, 아직 대부분의 방법은 사용자들이 효율적이고 간편하게 개발하기 위한 문서와 지원을 제공하지 않으며, 개발 표준은 충분히 연구되지 않았다. 본 연구에서는 Web3.j 자바 라이브러리를 이용하여 모바일 어플리케이션을 개발하고, 그 안에서 발생할 수 있는 보안상의 문제를 분석하였다. 스마트 컨트랙트 상에서의 모든 작동은 비동기적으로 처리되어야하며, 이는 구현상에 있어서 적절하게 조절되어야한다. 더욱이 DApp을 통해 사용자의 자산에 접근하기 때문에, 정보가 담긴 파일은 보다 안전하게 보관하여야한다. 하지만 이러한 보안이 강화된 wrapper class 로 구현 시에는 과도한 가스 소모를 야기할 수 있다. 이러한 문제를 피하기 위해서 솔리디티로 구현한 스마트 컨트랙트상에서 modifer 기능을 구현하거나 조정하여야한다. 또한 class 생성과정에 있어 적절한 경고와 주의를 기울여야 한다.

References

[1] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", Oct. 2008.

[2] D Hankerson, A Menezes, *Elliptic curve cryptography*, Boston, Springer , 2011.

[3] Lightweight Java library for integration with Ethereum clients Available : https://web3j.io/

[4] Testing, debugging, deploying of Smart Contracts on webpage : https://remix.readthedocs.io/en/latest/

[5] Private Ethereum Blokchain tool by truffleframework : https://truffleframework.com/ganache