J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

DApp 사용자의 프라이버시 보호 강화를 위한 공개형 블록체인 플랫폼 보안구조 강화방안☆

Strengthening security structure of open Blockchain platform to enhance privacy protection of DApp users

황 선 진[1] 고 동 현[1] 박 태 우[1] 최 윤 호[1*]
Seonjin Hwang DongHyun Ko Taeu Bahk Yoon-ho Choi

요 약

블록체인의 성장과 함께 이를 기반으로한 DApp(Distributed Application)이 주목받고 있다. DApp에 대한 관심이 커짐에 따라 시장 규모가 지속적으로 성장하고 있고 개발에 참여하는 개발자들이 늘어나고 있다. 많은 개발자들이 DApp 개발환경 구축의 어려움으로 인해 Infura와 같이 블록체인 노드를 중개해주는 API(Application Programming Inteface) 서비스를 이용하고 있지만, 중개 API 서비스를 이용할 경우 API 서비스 운영자는 DApp 사용자로부터 전달받은 트랜잭션의 계좌 주소와 DApp 사용자의 IP 주소의 1:1 매칭을 통해 사용자의 프라이버시를 침해할 수 있는 심각한 위험이 존재한다. 따라서, 본 논문에서는 기존 노드 탐색 프로토콜의 활용과 이를 이용한 암호화를 통해 사용자의 프라이버시를 보호할 수 있는 공개형 블록체인 플랫폼 구조 강화방안을 제안한다. 제안하는 구조를 통해 DApp 사용자는 API 서비스 운영자가 자신의 개인정보를 식별하지 못하게 방지함으로써 프라이버시를 보호할 수 있다. 기존의 공개형 블록체인 플랫폼들이 제공해주지 못했던 신뢰성 있는 DApp 사용 환경을 제공해줌으로써 프라이버시 침해 위험으로 인해 활성화되지 못하였던 DApp의 활성화와 사용자의 증가에 기여할 수 있을 것으로 기대된다.

☞ 주제어 : 공개형 블록체인 플랫폼, 프라이버시, 블록체인 보안, 이더리움

ABSTRACT

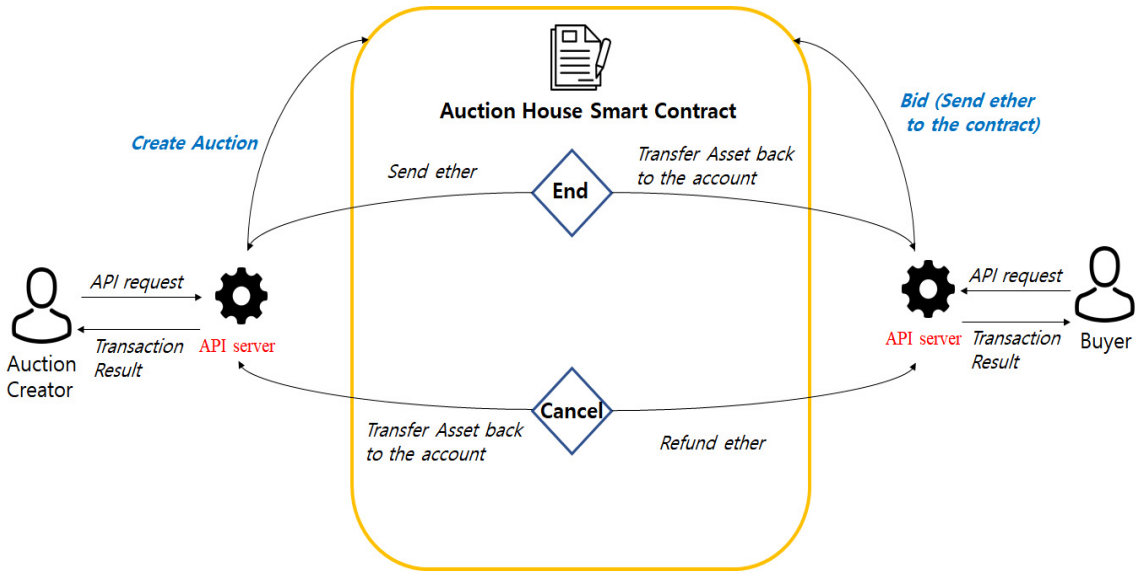
Along with the growth of Blockchain, DApp (Distributed Application) is getting attention. As interest in DApp grows, market size continues to grow and many developers participate in development. Many developers are using API(Application Programming Interface) services to mediate Blockchain nodes, such as Infura, for DApp development. However, when using such a service, there is a serious risk that the API service operator can violate the user's privacy by 1 to 1 matching the account address of the Transaction executed by the DApp user with the IP address of the DApp user. It can have an adverse effect on the reliability of public Blockchains that need to provide users with a secure DApp service environment. The proposed Blockchain platform is expected to provide user privacy protection from API services and provide a reliable DApp use environment that existing Blockchain platforms did not provide. It is also expected to help to activate DApp and increase the number of DApp users, which has not been activated due to the risk of an existing privacy breach.

☞ keyword : Blockchain Platform, Privacy, Blockchain Securicy, Ethereum

1. 서 론

다양한 산업, 연구 분야에서 블록체인에 대한 관심이 높아지고 있다. 특히 이더리움, 이오스 등 공개형 블록체인(Public Blockchain) 플랫폼은 누구나 이용할 수 있다는 장점으로 인해 허가형 블록체인(Private Blockchain)이 활용되지 못하는 부분에 활용될 수 있다는 점에서 그 가치를 주목 받고 있다. 최근, 블록체인은 암호화폐 뿐만 아니라 스마트 컨트랙트를 통한 분산 애플리케이션 DApp(Decentalized Application) 플랫폼으로 활용되고 있다. DApp이란, 블록체인 플랫폼 과 같은 분산 컴퓨팅 환경 상에서 동작하는 애플리케이션을 말하며 DApp의 수는 2017년 12월부터 2018년 12월까지 매월 평균 182% 지속적으로 성장하고 있다[1]. 이에 따라, DApp 개발에 대한 관심과 개발을 위한 도구들[2][3]이 인기를 얻고 있다.

1 School of Computer Science & Engineering, Pusan National University, Busan, 46241, Korea.
* Corresponding author (yhchoi@pusan.ac.kr)
[Received 4 November 2019, Reviewed 27 December 2019(R2 21 February 2020), Accepted 18 March 2020]
☆ 이 과제는 부산대학교 기본연구지원사업(2년)에 의하여 연구되었음. BK21플러스, IT기반 융합산업 창의인력양성사업단에 의하여 지원되었음.



(그림 1) API 서버를 통해 공개형 블록체인 네트워크에 연결되는 경매 DApp을 판매자와 구매자가 사용하는 예
(Figure 1) Example of an Auction DApp that creator and buyer connect to a public Blockchain network through API Server

인터넷 상에서, DApp을 구동시키기 위해서는 DApp과 블록체인 노드 간의 통신이 요구되는데 이를 위해 개발자는 블록체인 노드를 운영하여야 한다. 하지만, 블록체인 노드를 운영하기 위해서는 블록체인 네트워크 정보를 받아올 수 있는 안정적인 네트워크와 모든 트랜잭션 내역을 저장할 수 있는 충분한 저장공간 등이 필요하다. 개인 개발자가 이러한 제반 환경을 모두 갖추고 안정적으로 블록체인 노드 및 DApp을 운영하지 못하는 상황이 자주 발생한다. 이를 해결하기 위해, 블록체인과 관련된 서비스를 제공해주는 API를 이용한 DApp 개발이 증가하고 있다.

해외 블록체인 데이터 제공 서비스인 Fluence[4]가 이더리움의 DApp 프로젝트 개발자들을 대상으로 실시한 설문조사에서는 63%의 개발자가 블록체인 노드를 중개해주는 API 서비스인 Infura[5]를 사용하고 있다고 응답하였다[6]. Infura는 2017년 기준, 하루 60억건 이상의 API서비스 요청을 처리하고 있으며[7], Infura를 사용하는 DApp 개발 도구인 Metamask는 사용자 수 100만명을 돌파하였다[8].

많은 DApp 개발자들이 Infura와 같은 블록체인 노드를 중개해주는 API 서비스를 사용하고 있는 상황이지만, API 서비스 운영자는 DApp 사용자로부터 실행된 트랜잭션의 계좌 주소(표 1의 From)와 DApp 사용자의 IP 주소를 1:1 매칭하는 과정에서 사용자의 프라이버시를 침해할 수 있는 심각한 위협이 존재한다. 즉, API 서비스 운영자는 DApp 사용자의 API 요청과 IP를 매칭한 후, 이를 지속적으로 수집하여 사용자를 식별하는 것이 가능하다. 예를 들어, 블록체인 기반의 의료 어플리케이션에서 환자는 자신의 질병 기록이 노출될 수 있고 중고거래 어플리케이션에서 구매자는 자신의 구매상품과 구매가격 등과 같은 구매내역 등이 노출될 수 있다. 이는 사용자에게 안전한 DApp 서비스 환경을 제공해야 하는 공개형 블록체인(Public Blockchain)의 신뢰성에 치명적인 영향을 미칠 수 있다.

그림 1은 실제 공개형 블록체인 상에서 경매 DApp을 실행시켰을 때, 기존의 API 서비스 사용시 프라이버시 침해 가능성을 보여주는 예로서, 프라이버시 침해 가능성을 테스트하기 위한 경매 DApp 구성도이다. 그림 1에 보이는 것처럼 판매자(Creator)가 상품을 DApp을 통해 블록체인에 등록하고 구매자(Buyer)는 경매 금액을 DApp을 통해 지불함으로써, 경매에 참가한다. 경매 판매자 및 구매자는 DApp에 하드코딩 되어있는 API Server에 연결한다. 경매를 시작하기 위해 DApp에서 경매를 생성하면, 모든 요청과 결과는 API Server를 거치게 된다. 이 과정에서

API Server는 경매 판매자가 요청한 트랜잭션과 IP 주소를 통해 해당 판매자가 누구인지, 어떤 것을 판매하였는지 등을 모두 기록하고 식별하는 것이 가능하다. 또한, 경매 입찰을 위한 물품과 금액을 API Server를 경유하여 블록체인 네트워크에 보내게 될 경우, API Server는 구매자가 어떤 물품에 경매를 신청했는지 등을 트랜잭션과 해당 IP주소를 통해 식별할 수 있다.

본 논문에서 제안하는 공개형 블록체인 플랫폼 프라이버시 강화방안은 공개형 블록체인(Public Blockchain)의 DApp을 사용하면서 발생할 수 있는 프라이버시 침해 위협을 랜덤한 노드 ID 탐색과 암호화를 통하여 효율적으로 해결함으로써, DApp 사용자의 프라이버시를 안전하게 보호할 수 있다. 즉, DApp 사용자의 프라이버시를 보호하기 위한 공개형 블록체인 플랫폼 보안강화 방안을 적용함으로써, API 서비스 사용자에게 보다 안전한 DApp 사용 환경을 제공해 줄 수 있을 것으로 기대된다.

본 논문의 구성을 요약하면 다음과 같다. 2장에서는 일반적인 공개형 블록체인의 트랜잭션 구조와 기존의 대표적인 블록체인 익명성 연구를 요약하여 설명한다. 3장에서는 기존의 익명성 연구로 해결하지 못했던 프라이버시 침해 문제를 해결하기 위한 방안을 제안한다. 4장에서는 제안하는 방법의 구현 및 실험결과를 기술한다. 마지막으로, 5장에서는 결론을 맺는다.

2. 배경지식 및 관련연구

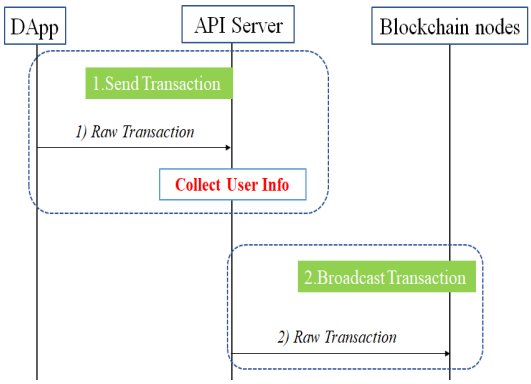
본 장에서는 공개형 블록체인의 트랜잭션 구조와 공개형 블록체인의 익명성 강화를 위한 대표적인 연구 결과에 대해 기술한다.

2.1 공개형 블록체인의 트랜잭션 구조

공개형 블록체인의 트랜잭션은 표 1과 같이 트랜잭션

(표 1) 트랜잭션 세부정보
(Table 1) Transaction details

해시	0x1a5136c82fbb90be9e2…
블록	8358702
타임스탬프	(Aug-16-2019 02:04:29)
송신자(From)	0x0f901bb0199efc9ec59f…
수신자(To)	0x6267873a5a0a1647a0b…
송금 금액	0.001 Ether
수수료	0.000037592 Ether



(그림 2) API 서비스 사용 시 발생할 수 있는 위협
(Figure 2) Threats to Conventional API Service

을 식별할 수 있는 해시, 트랜잭션이 포함된 블록의 위치를 나타내는 블록 , 트랜잭션의 발생시간인 타임스탬프, 트랜잭션의 출처를 나타내는 송신자(From) , 트랜잭션을 받는 사람을 나타내는 수신자(To), 얼마의 금액을 보낼 것인가를 나타내는 송금 금액, 트랜잭션의 수수료 등으로 구성된다. 서명(Signature) 및 넌스 값(nonce) 등은 생략하였다.

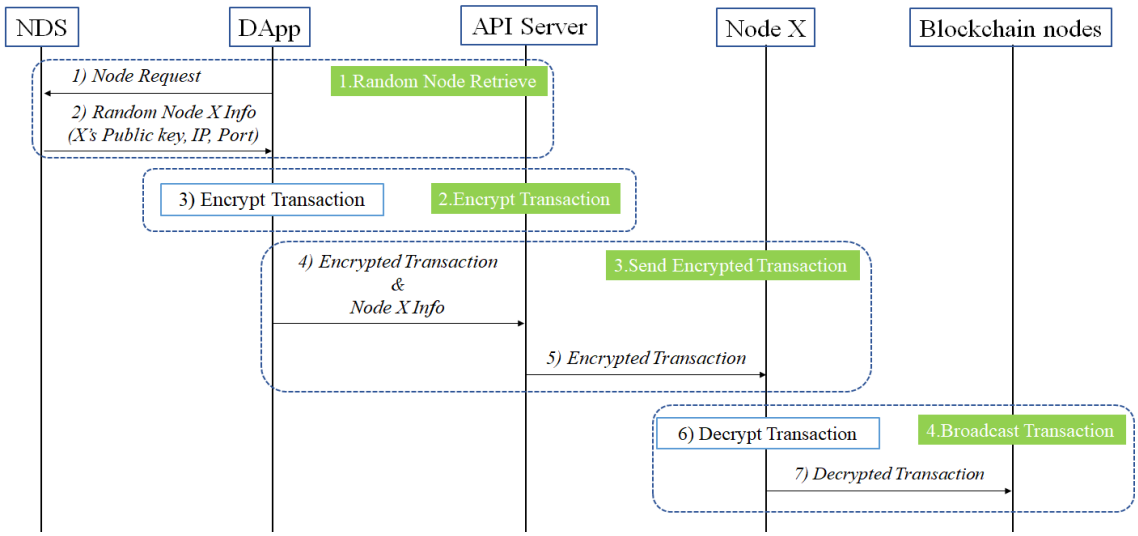
그림 2는 기존의 공개형 블록체인 API Service 사용 시에 발생할 수 있는 위협을 나타내고 있으며 DApp이 API Server에게 트랜잭션을 전송하는 Send Transaction 단계에서 API Server는 DApp 사용자의 IP 주소, 트랜잭션 내의 From과 To 등을 사용자를 특정할 수 있는 위협이 있다. 즉, API Server는 DApp들이 보낸 Raw Transaction들을 수집함으로써 특정 IP 주소의 DApp 사용자들이 언제, 어떤 어떤 물건을 얼마나 구매 했는지 등을 알 수 있는 것이다.

2.2 이더리움 노드 탐색 알고리즘

이더리움 플랫폼은 트랜잭션 및 블록 등의 블록체인 데이터를 공격자의 연결로부터 보호하기 위해 Kademlia 알고리즘을 기반으로한 RLPx 프로토콜을 사용한다. RLPx 프로토콜은 공격자로부터 노드 ID 위변조 공격을 방지하기 위해 ECIES 알고리즘을 기반으로 생성된 512 bit의 공개키를 노드 ID로 사용한다.

각 노드는 연결 전, 상호 간에 개인 키로 서명한 'Hello' 메시지를 주고 받음으로써 노드 ID가 위변조 되지 않았음을 확인할 수 있다.

RLPx 프로토콜은 초기 노드를 탐색하기 위해 클라이언트에 하드코딩 되어 있는 Bootstrap 노드들에게 FIND_NODE



(그림 3) 프라이버시가 보호 강화구조가 적용된 공개형 블록체인 플랫폼의 API 서비스
(Figure 3) API service process for Public Blockchain platform with enhanced privacy protection

메시지를 보내는 방법을 사용한다.

FIND_NODE 메시지를 받은 Bootstrap 노드들은 응답 메시지에 해당하는 NEIGHBORS로 랜덤한 노드 정보(IP 주소, UDP port, 노드 ID)들을 제공해주며, 초기 노드들은 제공받은 랜덤한 노드들에게 또 다시 FIND_NODE 메시지를 사용하여 새로운 노드들을 제공 받는다. 이를 반복함으로써, 초기 이더리움 노드들은 각자 다른 노드 정보를 가질 수 있다.

2.3 공개형 블록체인의 익명성 강화 연구

기존 비트코인의 트랜잭션에서 사용되는 여러 개의 입력주소(Input address)들은 동일한 사용자로 간주되어 프라이버시가 침해당할 수 있는 위험이 존재하였다[9]. 이를 해결하기 위해 다양한 트랜잭션들을 섞은 뒤, 다양한 출력 주소(Output address)를 생성하는 CoinJoin이 제안되었다[10]. 하지만, 믹싱(mixing) 서비스를 수행하는 과정에서 믹싱 서버가 비트코인 트랜잭션을 받았음에도 불구하고 이를 부인할 수 있는 문제와 믹싱 서비스 이용자가 믹싱 서비스를 이용해 믹싱된 비트코인 트랜잭션을 받았음에도 불구하고 이를 부인할 수 있는 문제가 존재하였다.

이를 해결하기 위해 다중 믹스(multiple mix)와 워런티(warranty)를 적용한 Mixcoin[11] 프로토콜이 제안되었다. Mixcoin 프로토콜은 기존 비트코인의 익명성을 강화하고 부인방지 문제를 해결하였지만, 믹싱 서비스 이용과정에서 믹싱 서비스의 운영자에게 사용자의 프라이버시가 노출될 수 있는 문제가 여전히 존재하였다. 이를 해결하기 위해 제안된 Blindcoin[12]은 믹싱 과정에서 blind signature[13]를 적용하였다. Blindcoin은 Blind signature를 통해 믹싱 과정 중 믹싱 서버의 프라이버시를 침해로부터 사용자를 보호하였다. 하지만, 여전히 블록체인과 같은 public log로의 접근이 필요하다는 단점이 존재한다. 비트코인 믹싱은 트랜잭션의 송신자와 수신자 사이에 연결될 수 있는 링크를 제거함으로써, 단순한 비트코인 트랜잭션의 익명성을 강화하는데 사용될 수 있다. 하지만 Mixcoin의 프로토콜의 경우, 현재 문제상황과 마찬가지로 API 서버로부터 프라이버시 침해가 일어날 수 있고 BlindCoin 프로토콜의 경우 블록체인 노드와의 직접적인 통신이 필요하기 때문에 문제 상황에 적용하기 어렵다.

대표적인 비트코인 믹싱 연구 외에도 Merge Avoidance[18], Ring Signatures[19], Zerocash[20], 등과 같은 블록체인 플랫폼 상의 프라이버시 연구들이 있었으나 이들은 허가형 블록체인(Private Blockchain)에 더 적합하거나 적용 시에 하드포크와 같은 대규모 변화가 필요하였다.

따라서 본문에서는 기존 연구의 한계를 보완하고 DApp 사용자의 프라이버시를 보호하기 위한 새로운 공개형 블록체인 플랫폼 보안 강화방안을 제안한다.

3. 제안 방안

본 장에서 제안하는 공개형 블록체인 플랫폼의 구성요소 및 동작과정을 설명한다.

3.1 구성요소 및 역할

제안하는 공개형 블록체인 플랫폼은 5가지 요소로 구성된다. (1) 사용자에게 랜덤한 노드 관련 정보를 제공해주는 Node Discovery Server(NDS), (2) 블록체인 네트워크에서 트랜잭션을 실행하려는 DApp, (3) 블록체인 API 서비스를 제공해주는 API Server, (4) NDS로부터 선택된 랜덤 노드인 Node X, (5) Node X와 연결된 나머지 블록체인 노드로 구성된다.

그림 3은 제안하는 플랫폼의 통신 과정을 나타내고 있으며, 플랫폼 구성요소의 역할 및 동작 과정은 다음과 같다.

- NDS(Node Discovery Server): NDS는 전체 이더리움 네트워크에서 랜덤하게 노드를 하나 선택하여 DApp 사용자에게 제공해주는 역할을 한다. 기존 이더리움 클라이언트가 다른 노드와 연결할 때 사용하는 노드 탐색 프로토콜[14]과 동일한 프로토콜을 사용하며, [14]는 Kademlia 프로토콜[15]을 기반으로 동작하므로 블록체인 네트워크에 존재하는 노드를 랜덤하게 탐색할 수 있다. NDS는 비트코인 지갑 클라이언트, Geth(Go-ethereum) 클라이언트 등에 포함될 수 있으며, 이들과 같이 사용자의 PC 내에서 운영할 수 있다. NDS는 블록체인 플랫폼의 지갑 클라이언트와 같이 오픈소스로 공개되어 있으며, 이더리움 네트워크 탐색처럼 적은 리소스로도 동작이 가능하다. 또한, 이더리움 사용자가 블록체인 네트워크를 탐색할 때 사용하는 알고리즘을 동일하게 사용하므로 추가적인 프라이버시 위협 없이 사용 가능하다.
- DApp: 서비스 사용자는 DApp에서 Node X의 ID(512bit 공개키)를 통해 트랜잭션을 암호화한다. 암호화 완료 후, 암호화된 트랜잭션 및 Node X의 정보를 API Server에게 전달한다.
- API Server: 서비스 사용자로부터 받은 Node X의 정보(Node ID, IP 주소, Port 번호)를 바탕으로 Node X에게 연결하여 암호화된 트랜잭션을 전달하는 역할을 한다.
- Node X: DApp 사용자로부터 선택된 Node로서, API Server로부터 들어온 연결을 수락하고 암호화된 트랜잭션을 전송받는다. 암호화된 트랜잭션을 자신의 개인키로 복호화 한 뒤, 자신과 연결되어있는 Blockchain nodes에게 트랜잭션을 전파한다.
- Blockchain nodes: 기존의 블록체인 상에서 동작하는 노드들을 의미한다. 기존의 블록체인 노드와 동일하게 트랜잭션을 전파 받으면 자신과 연결된 노드들에게 수신한 트랜잭션을 전파한다.

3.2 동작 과정

제안하는 공개형 블록체인 플랫폼 프라이버시 개선방안은 1) 랜덤노드 탐색(Random Node Retrieve) 단계, 2) 트랜잭션 암호화(Encrypt Transaction) 단계, 3) 암호화 트랜잭션 전송(Send Encrypted Transaction) 단계, 4) 트랜잭션 전파(Broadcast Transaction) 단계로 구성된다. 세부 동작과정은 다음과 같다.

3.2.1 랜덤노드 탐색(Random Node Retrieve)

1) Node request: 이더리움 네트워크 내의 랜덤한 노드 하나(Node X)를 요청한다.

2) Random node X Info: 이더리움 네트워크 내의 랜덤한 노드 정보들 중 하나를 DApp 사용자에게 제공해주며, 이더리움 노드 정보는 Node ID(512bit 공개키), IP 주소, Port 번호로 구성된다.

3.2.2 트랜잭션 암호화(Encrypt Transaction)

3) Encrypt Transaction: DApp 사용자는 트랜잭션을 Node X의 ID(512bit 공개키)를 통해 암호화한다. 암호화된 트랜잭션은 Node X의 개인키를 가진 Node X만이 복호화 가능하다. 암호화를 위해 타원곡선 암호 알고리즘(Elliptic Curve Cryptography)을 사용한다.

3.2.3 암호화 트랜잭션 전송(Encrypted Transaction sending)

4) Encrypted Transaction and Node X Info: DApp 사용자는 Digital Envelope를 만든 뒤, Node X의 정보, Digital Envelope와 암호화된 트랜잭션을 API Server에게 보낸다. 이 때, API Server는 트랜잭션을 받았으나 암호화되어 있으므로 트랜잭션의 내용을 볼 수 없다.

5) Encrypted Transaction: API service는 전송 받은 Node X의 정보(Node ID(512bit 공개키), IP 주소, Port 번호)를 바탕으로 Node X에게 연결하고 암호화된 트랜잭션을 전송한다.

3.2.4 트랜잭션 전파(Broadcast Transaction)

6) Decrypt Transaction: Node X는 API service 연결 후, 암호화된 트랜잭션 및 Digital Envelope를 받게 되고 Digital Envelope를 자신의 개인키(공개키에 대응되는 512bit 키)를 사용하여 복호화한다. Digital Envelope를 복호화하여 획득한 키를 사용하여 암호화된 트랜잭션을 복호화한다. 이때 Node X는 트랜잭션의 내용을 볼 수 있지만, 트랜잭션을 API service로부터 받았기 때문에 트랜잭션을 송신한 노드의 IP주소는 알지 못한다.

7) Decrypted Transaction: 복호화된 트랜잭션은 기존의 트랜잭션과 다르지 않기 때문에 Node X는 자신과 연결된 다른 노드들에게 전파하고 트랜잭션을 전파 받은 노드들은 트랜잭션을 계속해서 전파해간다.

3.3 동작 예제

제안하는 방안을 통해 암호화 되지 않은 트랜잭션 'T'를 전파하는 과정은 다음과 같다.

1. 트랜잭션 'T'를 암호화하기 위한 랜덤한 노드 ID(512 bit의 공개키)를 NDS를 통해 얻는다.(이 때 선택한 노드를 Node X라 가정한다.)
2. 랜덤한 노드 ID를 통해 트랜잭션 'T'를 'E(T)'로 암호화 한다. 이때 E(T)는 공개키에 해당하는 개인키를 가진 Node X만이 복호화 가능하다.
3. 'E(T)'를 API 서버에 전송한다.
4. API 서버는 전송받은 'E(T)'를 Node X에게 전송한다. 이때 API 서버는 Node X의 개인키가 없으므로 'E(T)'의 내용을 확인할 수 없다.
5. API 서버로부터 블록을 전파받은 Node X는 개인키를 통해 'E(T)'를 'T'로 복호화하고 'T'를 자신과 연결된 노드들에게 전파한다.

4. 실험 및 결과 분석

본 장에서는 제안 방안을 공개형 블록체인 플랫폼인 이더리움에 적용하고 이더리움 메인 넷(Main Network)과 프라이빗 넷(Private Network)에서 정상적으로 동작하는지 테스트하였다.

4.1 실험 환경

실험은 Intel Core i7-4790 CPU @ 3.60GHz, 16GB RAM, Windows 10 64bit 환경에서 진행되었으며 NDS, Node X의 Geth 버전은 1.8.27을 사용하였다. NDS는 이더리움 메인 넷에 연결하여 실제 노드 탐색을 수행하였으며, 프라이버시 침해 여부에 관한 테스트는 이더리움 프라이빗 넷을 직접 구축하여 수행하였다. 프라이빗 넷의 구성은 API Server, Node X, 복호화된 트랜잭션을 전파 받는 Node X까지 총 3개의 노드로 구성하였다. 실험을 위해 각 노드들을 그림 3과 같이 연결 명령어를 통해 연결하였다. 실험에 사용하였던 각 코드들은 https://github.com/unlockable/DAPP_Privacy에 공개하였다.

4.2 검증 결과

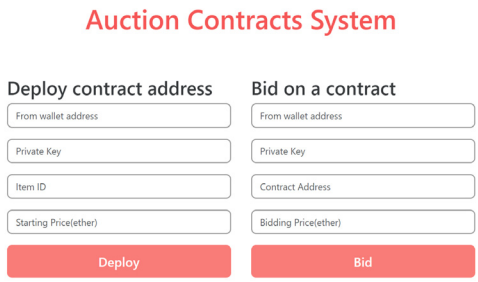
기존 이더리움의 노드 탐색 프로토콜을 수정하여 노드 탐색만을 수행하도록 하였다. 그림 4는 NDS를 통해 노드 탐색을 수행한 결과를 표시하고 있으며 노드 ID는 'enode://' 문자열로 시작한다. 이후, 랜덤한 512bit의 Node ID(공개키)가 이어지며 '@'를 구분자로 IPv4 주소와 Port가 뒤이어 나온다. NDS 구동 시에 NDS는 사용자의 Node ID를 기반으로 블록체인 네트워크 상의 노드들을 탐색해나가며[14], 사용자가 Node ID 요청 시에 탐색했던 노드 중 하나를 랜덤하게 선택하여 제공해준다.

그림 1의 시나리오를 기반으로 그림 5와 같은 경매 DApp을 구현하였다. 먼저, Deploy contract address에서 물건의 판매자는 자신의 지갑 주소(wallet address), 개인키, 판매할 item ID, 경매 시작 금액을 입력하여 계약을 생성한다.

마찬가지로 Bid on a contract를 이용하여, 구매자는 자신의 지갑 주소와 개인키, 생성된 계약 주소, 배팅 금액을 입력하고 경매에 참여한다.

DApp에서 트랜잭션을 생성한 후, 판매자와 구매자는 NDS를 통해 전달받은 Node X의 ID를 통해 트랜잭션을 암호화한다. 그 결과 그림 6에 나타난 것처럼 누구든지 확인할 수 있었던 기존 트랜잭션의 바이너리 코드(binary code)는 그림 7과 같이 Node X만이 복호화할 수 있는 정보로 암호화된다. 이후, 암호화된 트랜잭션을 API Server

(그림 4) NDS를 통한 노드탐색 결과
(Figure 4) Node Discovery Results Through NDS



(그림 5) 경매 DApp의 UI 프로토타입
(Figure 5) Auction DApp UI Prototype

```
Original transaction :
f908480a843b9aca00834c4b408080b907f66060604052
                    ⋮
3a5c580c1f90fc67ce23f0e2e977f6ad0fcb9a02daeef0
```

(그림 6) DApp의 Original 트랜잭션
(Figure 6) Original Transaction from DApp

에게 전달한다.

API Server는 API Service 사용자로부터 Node X의 연결정보와 암호화된 트랜잭션을 전달 받는다. API Server는 Node X의 개인키가 없기 때문에 암호화된 트랜잭션을 복호화할 수 없다. API Server는 단순히 Node X의 연결정보를 통해 Node X에게 연결만 할 뿐 내용을 볼 수 없다. 즉, 암호화된 트랜잭션을 전달한 후, 연결을 종료한다.

반면 Node X는 API Server로부터 전달받은 암호화된 트랜잭션을 자신의 개인키를 통해 복호화한 후, 그림 8과 같은 복호화된 트랜잭션을 얻어낸다. 이는 그림 6의 기존 트랜잭션과 동일한 트랜잭션이며, Node X는 이를 자신과 연결된 노드들에게 전파함으로써 전체 동작과정을 종료한다.

제안하는 블록체인 플랫폼 구조는 각 플랫폼의 구성요

```
Encrypted transaction:
 04368bd10e031de4bfa2d26268095f8f7e23b0055895ce
                    ⋮
 a38656c28ae7179dd945e98871af21525659772
```

(그림 7) Node X의 공개키를 통해 복호화된 트랜잭션
(Figure 7) Transaction Encrypted with Node X's public key

```
Decrypted Transaction :
 f908480a843b9aca00834c4b408080b907f66060604052
                    ⋮
 3a5c580c1f90fc67ce23f0e2e977f6ad0fcb9a02daeef0
```

(그림 8) Node X의 개인키를 통해 복호화된 트랜잭션
(Figure 8) Transaction Decrypted with Node X's private key

소들이 DApp 사용자의 트랜잭션과 IP 주소를 동시에 모두 수집하지 못하도록 구성함으로써, DApp 사용자의 프라이버시를 보호할 수 있었다. 구체적으로는 NDS는 사용자에게 랜덤한 Node 정보만을 제공해주기 때문에 DApp 사용자는 자신의 IP주소만을 NDS에 노출함으로써 다른 정보 유출을 막을 수 있다. API Server는 DApp으로부터 암호화된 트랜잭션을 전송 받기 때문에 사용자의 IP주소를 알 수 있으나 암호화된 트랜잭션의 내용은 볼 수 없다. 랜덤하게 선택된 Node X의 경우 트랜잭션을 복호화하여 트랜잭션의 내용을 볼 수 있으나 API Server로부터 트랜잭션을 전송 받았기 때문에 사용자의 IP주소를 알지 못한다.

5. 결 론

본 논문에서는 API 서비스를 통한 DApp 사용시에 트랜잭션의 계좌 주소와 DApp 사용자의 IP주소를 1:1 매칭하여 발생할 수 있는 프라이버시 침해 문제를 효율적으로 해결하기 위한 방법을 제시하였다. 제안하는 방법은 추가적인 트랜잭션 암호화와 NDS로 인해 추가적인 연산이 발생하지만, 사용자의 프라이버시를 안전하게 보호하면서 DApp을 사용할 수 있는 공개형 블록체인 플랫폼 동작환경을 제공해줄 수 있다. 즉, 기존의 프라이버시 침해 위협으로 인해 활성화되지 못하였던 DApp 활성화와 DApp 사용자 증가를 위해 제안하는 방안은 활용될 수 있을 것으로 기대한다.

참고문헌(Reference)

[1] Coingape, Dapp Development Seeing a Monthly Growth of 182% Amidst the Market Rout, Retrieved Oct. 31, 2019. [Online]. Available: https://coingape.com/dapp-development-monthly-growth-of-182/

[2] Metamask, Brings Ethereum to your browser, Retrieved Oct. 31, 2019. [Online]. Available: https://metamask.io/

[3] TRUFFLE SUITE, SWEET TOOLS FOR SMART CONTRACTS, Accessed: Oct. 31, 2019. [Online]. Available: https://trufflframework.com/

[4] Fluence, Decentralized database network, Accessed: Oct. 31, 2019. [Online]. Available: https://fluence.network/

[5] Infura, YOUR ACCESS TO THE ETHEREUM NETWORK, Accessed: Oct. 31, 2019. [Online]. Available: https://infura.io/

[6] Fluence, State of DApps Ecosystem, Technology and Adoption 2019, Accessed: Oct. 31, 2019. [Online]. Available: https://medium.com/fluence-network/dapp-survey-results-2019-a04373db6452

[7] Infura, Scaling INFURA: Not All API Calls are Equal, Accessed: Oct. 31, 2019. [Online]. Available: https://blog.infura.io/not-all-api-calls-are-equal-3659c119bb6c

[8] Chrome web store, Metamask, Accessed: Oct. 31, 2019. [Online]. Available: https://chrome.google.com/webstore/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=ko

[9] E. Androulaki, G. O. Karame, M. Roeschlin, T. Scherer, S. Capkun, "Evaluating User Privacy in Bitcoin", Financial Cryptography, 2013. https://doi.org/10.1007/978-3-642-39884-1_4

[10] G. Maxwell, "CoinJoin: Bitcoin privacy for the real world", bitcointalk.org, 2013. https://bitcointalk.org/index.php?topic=279249.0

[11] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J. A. Kroll, E. W. Felten, "Mixcoin: Anonymity for Bitcoin with accountable mixes", Financial Cryptography 2014. https://doi.org/10.1007/978-3-662-45472-5_31

[12] L. Valenta, B. Rowan, "Blindcoin: Blinded Accountable Mixes for Bitcoin", Workshop on Bitcoin Research, 2015. https://doi.org/10.1007/978-3-662-48051-9_9

[13] Chaum, David. "Blind signatures for untraceable payments." Advances in cryptology. Springer, Boston, MA, 1983. https://doi.org/10.1007/978-1-4757-0602-4_18

[14] Node Discovery Protocol v4, Accessed: Oct. 31, 2019. [Online]. Available: https://github.com/ethereum/devp2p/blob/master/discv4.md

[15] Petar Maymounkov and David Mazieres, "Kademlia: A peer-topeer information system based on the xor metric", In International Workshop on Peer-to-Peer Systems, pages 53 – 65. Springer, 2002. https://doi.org/10.1007/3-540-45748-8_5

[16] Satoshi Client Node Discovery, Accessed: Oct. 31, 2019. [Online]. Available: https://en.bitcoin.it/wiki/Satoshi_Client_Node_Discovery

[17] S. K. Kim, Z. Ma, S. Murali et al., "Measuring ethereum network peers", Proc. of IMC, 2018. https://doi.org/10.1145/3278532.3278542

[18] Mike Hearn, Merge avoidance. Accessed: Oct. 31, 2019. [Online]. Available: https://medium.com/@octskyward/merge-avoidance-7f95a386692f

[19] Bender, Adam, Jonathan Katz, and Ruggero Morselli. "Ring signatures: Stronger definitions, and constructions without random oracles." Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2006. https://doi.org/10.1007/11681878_4

[20] Sasson, Eli Ben, et al. "Zerocash: Decentralized anonymous payments from bitcoin." 2014 IEEE Symposium on Security and Privacy. IEEE, 2014. https://doi.org/10.1109/SP.2014.36

◐ 저 자 소 개 ◑

황 선 진(Seonjin Hwang)
2019년 부산대학교 전기컴퓨터공학부(공학사)
2019년~현재 부산대학교 대학원 전기전자컴퓨터공학과 석사과정
관심분야 : 블록체인, 네트워크 보안, 사용자 인증, 소프트웨어 보안
E-mail : unlockable7@gmail.com

고 동 현(DongHyun Ko)
2018년 부산대학교 정보컴퓨터공학부(공학사)
2018년~현재 부산대학교 대학원 전기전자컴퓨터공학과 석사과정
관심분야 : 네트워크 보안, 블록체인, 개인정보보호
E-mail : uyt1209@pusan.ac.kr

박 태 우(Taeu Bahk)
2017년 금오공과대학교 컴퓨터공학과(공학사)
2019년~현재 부산대학교 대학원 전기전자컴퓨터공학과 석사과정
관심분야 : 프라이버시 보호 데이터 배포(PPDP), 네트워크 보안, 퍼블릭 블록체인
E-mail : tu.bahk@daum.net

최 윤 호(Yoon-ho Choi)
2008년 서울대학교 전기컴퓨터공학부(공학박사)
2010년 펜실베니아 주립대학교 박사후 연구원
2012년 삼성전자 네트워크사업부 책임연구원
2014년 경기대학교 융합보안학과 조교수
2016년 부산대학교 전기컴퓨터공학부 조교수
2016년 ~ 현재 부산대학교 전기컴퓨터공학부 부교수
관심분야: 모바일 보안, 유무선 네트워크 침입탐지, IoT 보안 프로토콜, 경량 암호, 지능형 자동차 IT 보안 등
E-mail : yhchoi@pusan.ac.kr