*Journal of The Korea Institute of Information Security & Cryptology* ISSN 1598-3986(Print)
VOL.33, NO.6, Dec. 2023 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2023.33.6.947

# 블록체인 기반 익명 전자 어음 시스템*

우 현 주,[1†] 김 효 승,[2] 이 동 훈[3‡]
[1,3]고려대학교 (대학원생, 교수), [2]한림대학교 (교수)

# Anonymous Electronic Promissory Note System Based on Blockchain*

HyunJoo Woo,[1†] Hyoseung Kim,[2] Dong Hoon Lee[3‡]
[1]Korea University (Graduate student, Professor), [2]Hallym University (Professor)

요 약

현재 우리나라는 실물 어음을 전자 어음으로 대체해 나가고 있으며, 이러한 변화는 신뢰 기관인 금융결제원이 주도하고 있다. 하지만 기존 시스템은 해킹의 위협, 기관 내부의 오류 등 보안상의 취약점이 존재한다. 이에 따라 본 논문에서는 블록체인 기반의 새로운 익명 전자 어음 시스템을 정의하였다. 제안하는 프로토콜에서 모든 어음 정보는 커밋되기 때문에 지급이 일어나기 전까지는 거래 정보가 노출되지 않으며, 지급이 완료된 후에는 어음 정보가 블록체인에 공개되기 때문에 자금 세탁, 탈세와 같은 불법적인 행위를 탐지할 수 있다. 또한 본 프로토콜은 우리나라 전자 어음 시스템의 중요한 기능인 분할 배서가 가능하기 때문에 실제 금융 거래에 적용하기 적합하다.

ABSTRACT

In Korea, traditional paper promissory notes are currently undergoing a transformation, being gradually replaced by electronic notes. This transformation is being steered under the Korea Financial Telecommunications Institute, a trusted authority. However, existing electronic systems have security vulnerabilities, including the risk of hacking and internal errors within the institute. To this end, we have defined a novel anonymous electronic promissory note system based on blockchain. We have constructed a concrete protocol and conducted security analysis of our protocol. Note that, in our protocol, every note information is committed so that the note remains undisclosed until the point of payment. Once the note information becomes public on the blockchain, it enables the detection of illicit activities, such as money laundering and tax evasion. Furthermore, our protocol incorporates a feature of split endorsement, which is a crucial functionality permitted by the Korean electronic note system. Consequently, our proposed protocol is suitable for practical applications in financial transactions.

**Keywords:** Anonymous, Promissory Note, Blockchain, Decentralized Financial

## I. 서 론

전자 어음이란, 전자 문서로 작성되고 전자어음관리기관에 등록된 약속 어음을 말한다.(전자어음법 제2조 제2호) 전자 어음은 기존의 실물 어음과는 다르게 발행, 배서, 권리 행사 및 소멸이 모두 온라인에서 이루어진다. 실물이 필요 없기 때문에 발행 비용이 절감되고, 어음을 분실할 위험도 적으며 위변조가 어렵다는 장점이 있다. 특히 전자 어음은 부도율이 0%에 가깝기 때문에[1], 기존 실물 어음을 사용할 때 위협이 되었던 연쇄 부도와 같은 문제가 크게 개선되었다.

Received(10. 31. 2023), Modified(12. 01. 2023), Accepted(12. 03. 2023)
* 이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No. 2021-0-00518, 블록체인 데이터 암호화 기반의 프라이버시 보호 기술개발)
† 주저자, woohj@korea.ac.kr
‡ 교신저자, donghlee@korea.ac.kr(Corresponding author)

현재 우리나라의 전자어음관리기관은 금융결제원으로, 은행을 통해 어음 정보를 사전에 등록하고 해당 어음 정보는 금융결제원에 집중되는 중앙 집중형 방식으로 동작하고 있다. 전자 어음은 2005년 첫 도입 이후 발행량이 점차적으로 늘어나 실물 어음을 대체하고 있다. 우리나라의 전자 어음 발행 금액은 2022년 1024조, 2023년 1분기 216조에 달하는 등 전자 어음의 등록, 발행 및 배서가 활발하게 이루어지고 있다[2].

하지만 이와 같은 중앙 집중형 금융 서비스에는 몇 가지 문제점이 존재한다. 우선 중앙 기관을 운영 및 관리하는 비용이 발생한다. 또한 중앙 기관을 신뢰한다는 가정 하에 전체 시스템이 동작하기 때문에, 중앙 기관에 오류가 생기거나 공격이 발생하게 되면 시스템을 사용하는 모든 사용자들이 위협을 받을 수 있다. 실제로 2020년, 금융결제원이 해킹을 당해 4만여 개의 공인인증서가 유출되었고[3], 올해 3월에는 금융결제원 단말기 시스템에 오류가 생겨서 사용자들이 3시간 동안 서비스를 이용하지 못하는 문제가 발생하였다[4].

이러한 중앙 집중형 방식의 한계를 해결하고자 Defi(Decentralized Finace)가 등장하였다. Defi는 신뢰 기관을 두지 않고 블록체인 네트워크 상에서 구현된 금융 서비스를 말한다. 현재 우리나라의 전자 어음 시스템은 모든 어음 정보가 금융결제원에 저장되는데, 만약 전자 어음 Defi 서비스가 상용화 된다면 더 이상 금융결제원이 어음 정보를 관리하지 않기 때문에 기존 기관을 운영, 관리하며 발생했던 비용이 절감된다. 사용자가 어음을 발행하고자 할 때, 배서하고자 할 때 은행에 지불하던 수수료 또한 지불하지 않아도 된다. 또한 블록체인 네트워크의 스마트 컨트랙트를 사용하면 어음 발행, 배서, 권리 행사 및 소멸이 완전히 자동화되기 때문에 기존 은행을 통해 전자 어음을 발급받는 방식보다 편리한 서비스 제공이 가능하다. 또 블록체인의 특성상 중앙 기관이 없어도 투명성을 만족하기 때문에 더 이상 금융결제원을 공격하는 방식으로 시스템을 위협할 수 없고, 전자 어음의 사본을 만드는 공격도 불가능하다.

### 1.1 블록체인 기반 전자 어음 동향

2018년, 블록체인을 이용하여 어음 지급이 자동(automatic)으로 이루어지는 전자 어음 시스템이 설계되었다[5,7]. 특히 폴란드 정부에서는 실제 환경에서 블록체인 기반 전자 어음 시스템의 적용이 용이하도록 관련 법적 제도 등을 정비하고 출간하였다[5]. 2021년에는 블록체인 네트워크를 활용하여 문서 소유권을 양도하는 연구가 등장하였고, 이를 응용한 전자 어음 시스템이 제안되었다[6]. 또한 FQX의 eNote는 블록체인을 사용하여 자동화된 전자 어음 시스템을 실제로 서비스하고 있다[7].

하지만 기존의 연구에는 두 가지 명확한 한계가 있다. 첫 번째, 배서(endorsement)의 기능이 제대로 구현되어 있지 않다. 전자 어음의 기능은 단순히 미래에 약속된 돈을 지급 받는 것으로 끝나지 않으며, 어음을 다른 사용자에게 판매하는 배서의 기능 또한 존재한다. 실제로 은행의 전자 어음 서비스 수익 대부분을 배서 수수료가 차지할 정도로 배서는 전자 어음의 매우 중요한 기능이다. 하지만 어음을 소유하고 있는 사용자가 배서의 권한을 가지고 있는 전자 어음과는 다르게, 기존 연구[5]에서 제시한 시스템은 어음을 발행한 사람이 배서를 진행한다. 이는 어음 발행자는 배서에 관여하지 않는 전자 어음 시스템과 상이하다. 또한 우리나라 전자 어음 제도에서 허용하는 분할 배서(split endorsement)를 지원하지 않는다[5,6,7].

두 번째, 계약 내용을 비밀로 유지(secrecy)하기 어렵다. 기존 연구들은 스마트 컨트랙트 특성상 작성한 내용이 블록체인 상에 모두 공개되기 때문에 전자 어음을 발급한 순간 계약 내용이 노출된다. 실제 전자 어음은 기업 간의 계약 시 많이 사용되는 것을 고려하면 중대한 취약점이다. 기존 연구는 이를 해결하기 위해 전자 어음 시스템을 프라이빗 블록체인에서 구현하거나[7], 단순히 암호화하였다[5,6]. Table 1은 기존 블록체인 기반 전자 어음 관련 연구와 제안하는 프로토콜의 차이를 정리한 것이다.

Table 1. Comparison with related works

| | [5] | [6] | [7] | ours |
|---|---|---|---|---|
| Automatic | ○ | × | ○ | ○ |
| Secrecy | △ | ○ | × | ○ |
| Endorsement | ○ | ○ | ○ | ○ |
| Split Endorsement | × | × | × | ○ |

○:supported, ×:not supported,
△:not supported but a method is suggested

### 1.2 제안하는 프로토콜의 기여도

본 논문에서는 새로운 블록체인 기반 전자 어음 시스템을 제안하고 구체적인 프로토콜을 설계한다. 설계 프로토콜은 계약 정보를 노출하지 않으면서, 현재 서비스 중인 우리나라 전자 어음 시스템의 기능을 모두 제공한다. 특히 기존 연구에서 다루지 않았던 전자 어음의 분할 배서 기능까지 지원하기 때문에, 제안하는 어음 시스템은 실제 금융 서비스에 적용하기 적합하다.

제안하는 프로토콜은 다음과 같은 특징을 가진다. 우선 어음 소유자는 어음 발행자로부터 사전에 약속한 날짜에 어음에 작성된 내용과 동일한 금액의 지급을 보장받는다. 전자 어음은 다른 사용자에게 배서가 가능하며, 실제 전자 어음 시스템과 동일하게 어음의 소유자가 배서를 수행한다. 지급이 이루어지기 전, 거래 당사자들을 제외한 사용자들은 거래 내용에 대해 어떠한 정보도 알 수 없기 때문에 안전한 거래가 가능하다. 실제 지급이 이루어지면 모든 사용자들은 그 거래 내용을 열람할 수 있기 때문에 어음이 자금세탁 등의 불법적인 행위에 사용되는 것을 방지한다.

## II. 배경 지식

### 2.1 블록체인

블록체인은 데이터를 여러 시스템에 분산하여 저장하는 분산 원장 기술로, 데이터가 중앙 집중화된 서버에 저장되는 것이 아니라 네트워크를 이루는 다수의 노드에 복제된다. 따라서 중앙 관리 기관이 없어도 데이터를 안전하고 투명하게 관리할 수 있다. 블록체인 기술을 기반으로 계약의 조건과 규정을 프로그래밍 하여 실행 가능한 형태로 만든 코드가 스마트 컨트랙트이며, 계약 당사자들의 거래를 자동으로 실행하고 관리한다.

블록체인은 데이터를 블록에 저장하며, 그 블록은 체인 형태로 이어져 있다. 한 블록의 데이터가 변조되면 체인이 끊어지기 때문에 모든 사용자는 그 데이터가 변조되었음을 알 수 있다. 또한 데이터를 변경하려면 전체 네트워크에서 과반수의 동의를 얻어야 한다.

제안하는 프로토콜에서 사용하는 블록체인 네트워크는 다음과 같은 특징을 가진다[8].

1. 탈중앙화(Decentralization) : 블록체인은 중앙 기관의 통제 없이 프로토콜과 참여자의 합의에 의해 이루어지며, 모두에게 공유되는 분산 장부를 사용하기 때문에 모든 참여자가 데이터를 열람 및 검증할 수 있다.

2. 변조 저항성(Tamper-Resistance) : 블록체인 네트워크의 합의에 의해 데이터가 기록되면 조작이나 수정이 어렵다.

3. 안전성(Secrecy) : 블록체인 네트워크를 위협하는 디도스 공격(DDos attack), 이중 지불 공격(double-spending attack), 과반수 공격(51% attack) 등에 안전하다.

### 2.2 zk-SNARK

zk-SNARK는 특정 정보를 가지고 있음을 증명하지만 해당 정보는 노출하지 않는 영지식 증명 프로토콜이다. zk-SNARK 기법은 아래와 같이 정의할 수 있다[9].

**Definition 1** (zk-SNARK). 도메인이 정해져 있는 하이 레벨 프로그램(high-level domain specific) $P(\vec{s},\vec{v})\rightarrow\vec{o}$가 주어지면, 증명자(prover)는 다음과 같은 함수들을 사용해서 $\vec{s}$를 드러내지 않고도 검증자(verifier)에게 $(\vec{s},\vec{v},\vec{o})$가 $P(\vec{s},\vec{v})\rightarrow\vec{o}$에 명시된 관계 $R$을 만족한다는 것을 보일 수 있다. 프로그램 $P$를 설정할 때 결정되는 $(\vec{s},\vec{v},\vec{o})$는 정수, 이진수, 문자열 모두 가능하며 여러 형태가 모인 튜플의 형식도 가질 수 있다.

$KGen^{SNARK}(P)\rightarrow(PK,VK)$ : 프로그램 $P$를 입력으로 받아 증명 키 $PK$와 검증 키 $VK$ 쌍을 생성하여 출력한다.

$Proof^{SNARK}(PK,P,\vec{s},\Phi)\rightarrow\pi$ : 프로그램 $P$와 비밀 입력 값 $\vec{s}$, 공개 파라미터 $\Phi=(\vec{v},\vec{o})$, 증명 키 $PK$를 입력받아 증명 스트링(proof string) $\pi$를 출력한다. $\vec{s}$, $\Phi=(\vec{v},\vec{o})$는 정수, 이진수, 문자열 모두 입력 가능하다.

$Verify^{SNARK}(VK,\pi,\Phi)\rightarrow b\in\{1,0\}$ : 증명 스트링 $\pi$, 파라미터 $\Phi=(\vec{v},\vec{o})$와 검증 키 $VK$를 입력으로 받아 입력값이 프로그램 $P$에 명시된 관계 $R$을 만족하면 1, 그렇지 않으면 0을 출력한다.

zk-SNARK는 다음과 같은 특징을 가진다.

1. 영지식(Zero-Knowledge) : 증명자가 증명 스트링 $\pi$를 사용하여 검증하는 과정에서 프라이빗 파라미터 $\vec{s}$ 정보가 유출되지 않는다.

2. 완전성(Completeness) : 정직한 증명자는 언제나 정확한 증명 스트링을 생성할 수 있고, 정직한 검증자는 언제든 올바른 증명 스트링을 확인할 수 있다.

3. 건전성(Soundness) : 정당하지 않은 값을 가지고 정확한 증명 스트링을 생성할 확률은 무시할 수 있을 정도로 작다.

### 2.3 해시 커밋먼트

해시 커밋먼트(Hash Commitment)는 해시 함수를 사용하여 설계한 커밋먼트 기법을 의미한다. 커밋먼트 기법을 이용하면 선택한 메시지를 커밋먼트 형태로 숨기고 나중에 그 값을 공개하였을 때, 검증자는 주어진 커밋먼트 값이 공개된 값을 이용하였는지 확인할 수 있다. 정수 $q$, 도메인의 크기를 $\ell$이라고 할 때, 해시 함수 $H: \{0,1\}^{*} \rightarrow \{0,1\}^{\ell}$ 이 주어지면 해시 커밋먼트는 다음 두 가지 함수로 구성된다.

$Commit(x;r) \rightarrow c$ : 커밋먼트 함수는 커밋된 값 $x \in \mathbb{Z}_q$와 랜덤한 값 $r \in \mathbb{Z}_q$을 입력으로 받아 커밋먼트 $c \leftarrow H(x;r) \in \{0,1\}^{\ell}$를 출력한다.

$Open(c,s,r) \rightarrow \{0,1\}$ : 커밋먼트 오프닝 함수는 커밋먼트 $c$, 커밋된 값 $x$, 랜덤 $r$을 입력으로 받아 $c = H(x;r)$이면 1, 그렇지 않으면 0을 출력한다.

본 연구의 프로토콜에 사용된 해시 커밋먼트는 다음과 같은 특징을 가진다.

1. 바인딩(Binding) : 메시지 $m$의 커밋먼트 $c$가 주어진 경우($Commit(m;r) = c$), 공격자가 $m \neq m'$이면서 $Commit(m;r) = Commit(m';r')$을 만족하는 메시지 $m'$을 찾을 확률은 무시할 수 있을 정도로 작다.

2. 하이딩(Hiding) : 공격자는 커밋먼트 $c$가 주어져도($Commit(m;r) = c$) 메시지 $m$에 대한 어떠한 정보도 알 수 없다.

## III. 제안하는 프로토콜

본 연구에서 제안하는 전자 어음 프로토콜의 전제조건은 다음과 같다.

- 거래 당사자들은 사전에 오프 체인에서 거래 내용의 합의를 마쳤으며, 거래 당사자들 모두 거래 내용에 대해 알고 있다.

블록체인 기반 전자 어음 시스템이 만족해야 하는 성질은 다음과 같다.

1. 정확성(correctness) : 어음 소유자는 어음에 작성된 내용과 동일한 금액을 어음 발행자로부터 사전에 약속한 날짜에 지급받음

2. 익명성(anonymity) : 실제 지급이 이루어지기 전, 거래 당사자들을 제외한 다른 사용자들은 거래 내용에 대해 어떠한 정보도 알 수 없음

### 3.1 프로토콜 개요

제안하는 어음 프로토콜을 수행하는 객체는 어음 발행자, 어음 소유자, 새로운 어음 소유자로 구분할 수 있다. 어음 발행자는 미래에 돈의 지급을 약속하며, 어음 소유자와 함께 사전에 오프체인에서 거래 내용의 합의를 마친 후 어음을 발행(issuing)한다. 어음 소유자는 미래에 돈을 지급 받을 것을 약속하며 지급(payment)받을 권리를 증명할 수 있는 어음을 소유한다. 어음 소유자가 그 권리를 다른 사용자에게 판매하는 것을 배서라고 하는데, 배서가 올바르게 이루어지면 기존 어음 소유자는 더 이상 해당 어음에 어떠한 권리도 가질 수 없게 되고, 해당 어음의 소유권은 새로운 어음 소유자에게로 넘어가게 된다.

본 기법의 전자 어음 프로토콜은 Fig. 1.과 같이 크게 두 가지 단계로 나눌 수 있다. 우선 ① 어음 발행자는 본인이 알고 있는 거래 내용을 커밋하여 스마

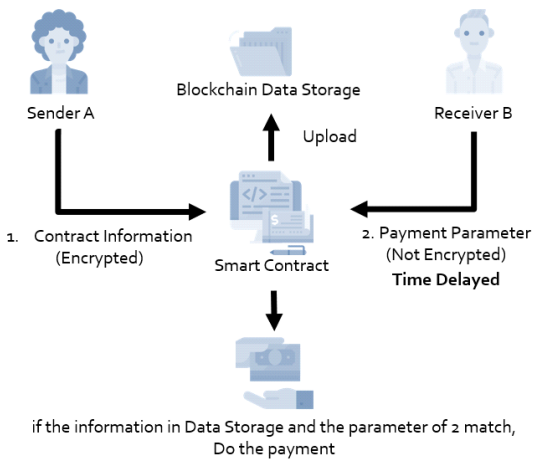


Fig. 1. Overview of the proposed protocol

트 컨트랙트에 입력한다. 이 커밋먼트 값은 블록체인 데이터 저장소에 업로드 된다. 이후 사전에 약속한 지급 날짜가 되면, ② 어음의 소유자는 본인이 알고 있는 거래 내용(어음 발행자와 어음 소유자의 정보, 지급 금액 등)을 스마트 컨트랙트에 입력한다. 어음 발행자가 사전에 업로드 한 커밋먼트 값과 어음 소유자가 입력한 거래 내용의 커밋먼트 값이 일치하는 경우, 지급이 진행된다.

본 시스템에서 어음 발행자가 스마트 컨트랙트에 입력한 값은 커밋먼트 값이기 때문에 다른 사용자들은 거래 내용에 대해 어떠한 정보도 알 수 없는 반면, 어음 소유자가 입력한 값은 평문이기 때문에 모든 정보가 공개된다. 하지만 정보가 공개된 시점에서는 이미 실제로 지급이 끝난 상태이기 때문에, 사전 거래 정보 누출은 일어나지 않는다. 또한 실제 거래가 발생하면 무조건 그 정보가 공개되기 때문에 자금 세탁 등의 불법적인 행위에 사용되는 것을 막을 수 있다.

만약 어음 발행자가 사전에 합의한 내용과 다르게 거래 내용을 변조하여 어음을 생성하였을 경우, 어음 소유자는 블록체인 데이터 저장소에 업로드 된 커밋먼트 값과 본인이 알고 있는 거래 내용을 비교하여 변조 여부를 사전에 파악할 수 있다. 반대로 어음 소유자가 지급 시에 거래 내용을 변조하였을 경우, 어음 발행자가 업로드 한 커밋먼트 값과 차이가 생기기 때문에 지급이 이루어지지 않는다. 따라서 어음의 거래 내용 오류 또는 변조를 통한 부적절한 지급 시도를 막을 수 있다.

### 3.2 프로토콜

논문에서 제안하는 전자 어음 프로토콜은 발행, 지급, 배서 총 세 개의 단계로 구성된다. 제안 프로토콜을 설명하기 위한 표기는 Table 2로 나타낼 수 있다. Fig. 2.는 제안하는 프로토콜의 전자 어음 발행 과정을 나타낸 것이고, Fig. 3.은 지급 과정, Fig. 4.는 배서 과정, Fig. 5.는 배서된 어음의 지급 과정을 그림으로 나타낸 것이다.

본 논문에서는 스마트 컨트랙트를 통해 어음 발행자의 블록체인 주소 $S$로부터 어음 소유자의 블록체인 주소 $R$에 $A$만큼의 금액을 지급하는 것을 다음과 같이 표기하였다.

$$Note.S \xrightarrow{Note.A} Note.R$$

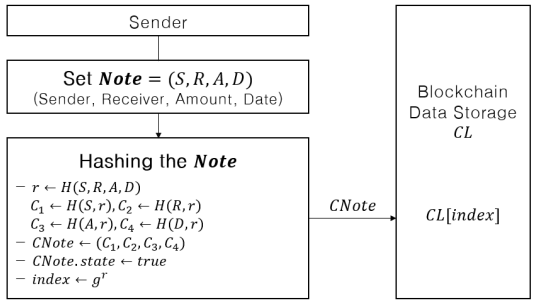


Fig. 2. The proposed protocol's issuing steps

**- 발행(issuing)**

1. 어음 발행자는 본인이 알고 있는 거래 정보(돈을 보내는 사용자, 받는 사용자, 금액, 날짜)를 토대로 어음 $Note=(S,R,A,D)$를 생성한다.

2. $r \leftarrow H(Note)$을 생성하고 이를 사용하여 $Note$를 커밋한다.

$$C_1 \leftarrow H(S,r),\ C_2 \leftarrow H(R,r)$$
$$C_3 \leftarrow H(A,r),\ C_4 \leftarrow H(D,r)$$
$$CNote \leftarrow (C_1, C_2, C_3, C_4)$$

3. $CNote$의 상태를 $true$로 설정한다.

4. $index \leftarrow g^r$을 계산하고 $CL[index]$에 $CNote$를 저장한다.

거래 당사자들은 모두 거래 내용을 알고 있기 때문에 $r \leftarrow H(Note)$를 생성할 수 있다. 따라서 어음 발행자가 $CNote$를 블록체인 데이터 저장소에 업로드하면, 어음 소유자는 본인이 알고 있는 $r$ 값과 $Note$ 값을 통해 현재 업로드된 $CNote$의 변조 여부를 지급이 일어나기 전에 판단할 수 있다.

Table 2. Notations used in this paper

| notation | descriptions |
|---|---|
| $S$ | note issuer's blockchain address |
| $R$ | note owner's blockchain address |
| $A$ | payment amount |
| $D$ | payment date |
| $Note$ | promissory note information $(S,R,A,D)$ |

| notation | descriptions |
|---|---|
| $C_1, C_2, C_3, C_4$ | hash of $S,R,A,D$ $(c = H(x;r))$ |
| $CNote$ | committed note $(C_1, C_2, C_3, C_4)$ |
| $CL$ | List of committed note |
| $msg.address$ | user's address who runs the smart contract function |
| $r$ | hash of $(S,R,A,D)$ |
| $\vec{s}, \vec{a}, \vec{d}$ | private inputs of zk-SNARK |
| $\Phi_1, \Phi_2, \Phi_3$ | public parameters of zk-SNARK |
| $\pi, \pi_1, \pi_2, \pi_3$ | verify string which is made with zk-SNARK $Proof$ function |
| $E$ | Elliptic Curve |
| $q$ | the number of points of Elliptic Curve |
| $Z_q$ | Elliptic Curve order |
| $g$ | group element |
| $pp$ | public parameter $pp = (E, Z_q, q, g)$ |
| $index$ | $index \leftarrow g^r$ |
| $state$ | 0 or 1, effectiveness of commits |

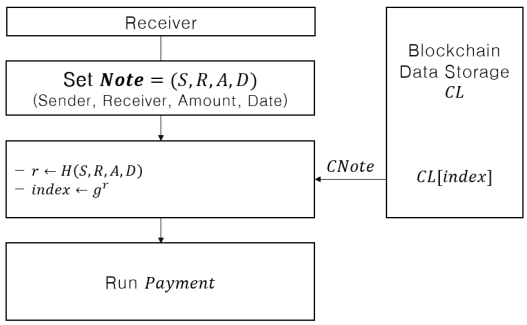


Fig. 3. The proposed protocol's payment steps

**- 지급(payment)**

1. 어음 소유자는 본인이 알고 있는 거래 정보(돈을 보내는 사용자, 받는 사용자, 금액, 날짜)를 토대로 어음 $Note = (S,R,A,D)$를 생성한다.

2. $r \leftarrow H(Note)$를 생성하고 $index \leftarrow g^r$을 계산하여 $CL[index]$에 저장된 $CNote$를 참조한다.

3. 생성한 값으로 $Payment(Note, r)$ 함수(Table 5)를 호출한다.

4. $CNote$의 상태 값이 1이고, 현재 블록체인 시간 정보와 $CNote$의 날짜 정보가 일치하며, 함수 $CNote$를 호출한 사용자의 블록체인 주소 $msg.address$가 $Note.R$과 같고, 어음 소유자가 입력한 거래 정보$(S,R,A,D)$와 $CNote$의 거래 정보가 일치하는 경우에만 지급이 정상적으로 이루어진다.

**- 배서(endorsement)**

제안하는 프로토콜의 배서는 다른 사용자에게 어음을 판매할 때 어음 소유자를 제외한 다른 어음 정보가 변하지 않았다는 것을 증명하기 위해 프로그램 $P_1$을 사용한다. Table 3은 프로그램 $P_1$을 자세히 기술한 것으로 커밋먼트 값 $C$와 $C'$가 각각 $(S,r)$와 $(S',r')$로부터 생성되었음을 확인하고 $S$와 $S'$가 동일함을 모두 확인한다. 모든 조건을 만족하면 1, 그렇지 않으면 0을 출력한다.

Table 3. Program 1

| Program $P_1$ |
|---|
| Inputs : $(S,r),(S',r'),C,C'$ |
| Output : $b$ |
| 1. $c_1 = (C == H(S,r))\,?\,1:0;$ ; |
| 2. $c_2 = (C' == H(S',r'))\,?\,1:0;$ |
| 3. $c_3 = (S == S')\,?\,1:0;$ ; |
| 4. $b = c_1 \wedge c_2 \wedge c_3;$ |
| 5. return $b$; |

실제 배서 과정에서는 $(S,r)$와 $(S',r')$를 커밋한 커밋먼트 값 $C_1, C_1'$이 사용된다. 따라서 전자 어음을 배서할 때 zk-SNARK를 사용하여 $(S,r)$, $(S',r')$ 값을 공개하지 않고 $P_1$을 통과할 수 있음을 증명한다.

배서과정의 자세한 내용은 다음과 같이 기술한다.

1. 어음 소유자는 오프 체인에서 배서하고자 하는 사용자와 새로운 거래 내용을 합의한다.

2. 어음 소유자는 새로운 거래 정보를 토대로 새로운 어음 $Note' = (S',R',A',D')$을 생성한다.

3. 새로운 $r' \leftarrow H(Note')$을 생성하고 이를 사용하여 $Note'$을 커밋한다.

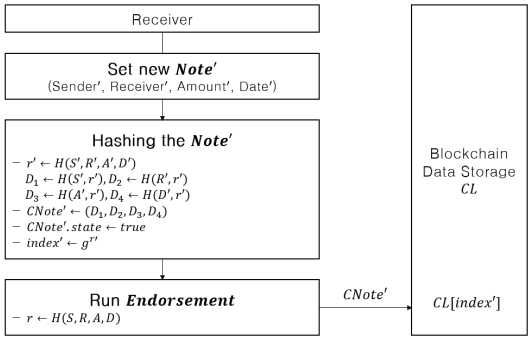


Fig. 4. The proposed protocol's endorsement steps

$$C_1' \leftarrow H(S', r'), C_2' \leftarrow H(R', r')$$
$$C_3' \leftarrow H(A', r'), C_4' \leftarrow H(D', r')$$
$$CNote' \leftarrow (C_1', C_2', C_3', C_4')$$

4. 기존의 거래 정보로 $r \leftarrow H(Note)$를 생성한다.

5. 영지식 증명에 사용할 키를 생성한다.

$$KGen^{SNARK}(P_1) \rightarrow (PK_{P_1}, VK_{P_1})$$

6. 비밀 입력 값 $\vec{s}, \vec{a}, \vec{d}$를 다음과 같이 설정한다.

$$\vec{s} \leftarrow ((S,r),(S',r')),\ \vec{a} \leftarrow ((A,r),(A',r'))$$
$$\vec{d} \leftarrow ((D,r),(D',r'))$$

7. $CNote = (C_1, C_2, C_3, C_4)$를 사용하여 공개 파라미터 $\Phi_1, \Phi_2, \Phi_3$를 다음과 같이 설정한다.

$$\Phi_1 \leftarrow (C_1, C_1'),\ \Phi_2 \leftarrow (C_3, C_3'),\ \Phi_3 \leftarrow (C_4, C_4')$$

8. 증명 스트링 $\pi = (\pi_1, \pi_2, \pi_3)$를 다음과 같이 생성한다.

$$Proof^{SNARK}(PK_{P_1}, P_1, \vec{s}, \Phi_1) \rightarrow \pi_1$$
$$Proof^{SNARK}(PK_{P_1}, P_1, \vec{a}, \Phi_2) \rightarrow \pi_2$$
$$Proof^{SNARK}(PK_{P_1}, P_1, \vec{d}, \Phi_3) \rightarrow \pi_3$$

9. $Endrosement(r, CNote, CNote', \pi, VK_{P_1})$ 함수(Table 5)를 호출한다.

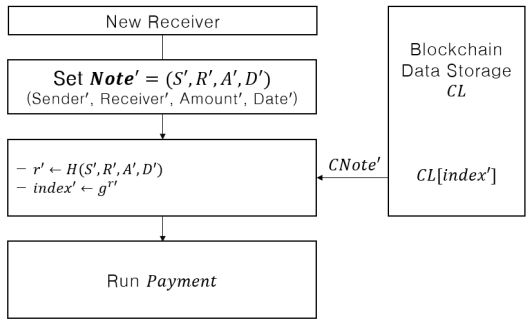


Fig. 5. The proposed protocol's payment steps with endorsed note

10. 기존의 $CNote$와 $CNote'$의 값을 비교하여 변조 여부를 확인한다.

(Table 5 $Endrosement$함수 3~5번째 줄 참조)

11. $CNote$가 유효하고 $CNote'$가 변조되지 않은 경우, 기존의 $CNote$의 상태를 $false$로 설정한다.

12. $CNote'$의 상태를 $true$로 설정한다.

13. $index' \leftarrow g^{r'}$을 계산하고 $CL[index']$에 $CNote'$을 저장한다.

14. 새로운 어음 소유자는 본인이 알고 있는 거래 정보를 토대로 어음 $Note' = (S', R', A', D')$을 생성한다.

15. 거래 정보를 통해 $r' \leftarrow H(Note')$을 생성하고 $index' \leftarrow g^{r'}$을 계산하여 $CL[index']$에 저장된 $CNote'$을 참조한다.

16. 생성한 값으로 $Payment(Note', r')$ 함수(Table 5 참조)를 호출한다.

17. 기존 지급 1~4 과정과 동일하게 진행된다.

단순히 어음의 소유권을 이전하는 배서가 진행되면 어음 정보에서 변화하는 것은 돈을 받는 사용자 $R$ 뿐이기 때문에, 이를 제외한 나머지 거래 정보가 배서를 하며 변화하지 않았는지 영지식 증명을 통해 확인한다.

만약 어음을 배서하여 더 이상 소유권을 가지고 있지 않은 사용자가 $Payment$ 함수를 통해 지급을 시도할 경우, 기존의 어음 $CNote$의 상태가 $false$이기 때문에 지급이 일어나지 않는다.

### 3.3 분할 배서

제안하는 프로토콜의 분할 배서는 다수의 사용자에게 어음을 판매할 때 새로 작성한 어음 지급 금액

Table 4. Program 2

| Program $P_2$ |
|---|
| Private inputs : $(S,r),(S_1,r_1),(S_2,r_2)$ |
| Public inputs : $C,C_1,C_2$ |
| Output : $b$ |
| 1. $c_1=(C==H(S,r))?1:0;$ : |
| 2. $c_2=(C_1==H(S_1,r_1))?1:0;$ |
| 3. $c_3=(C_2==H(S_2,r_2))?1:0;$ : |
| 4. $c_4=(S_1+S_2==S)?1:0;$ |
| 4. $b=c_1\wedge c_2\wedge c_3\wedge c_4;$ |
| 5. return $b;$ |

의 총 합이 기존의 지급 금액과 같다는 것을 증명하기 위해 프로그램 $P_2$을 사용한다. 본 논문에서는 가장 간단한 형태인 두 명의 사용자에게 분할 배서하는 프로토콜을 보였으나, 더 나아가 다수의 사용자에게도 분할 배서가 가능하다. Table 4는 프로그램 $P_2$을 자세히 기술한 것으로 커밋먼트 값 $C,C_1,C_2$가 각각 $(S,r),(S_1,r_1),(S_2,r_2)$로부터 생성되었음을 확인하고 $S_1$과 $S_2$의 합이 $S$가 맞는지 확인한다. 모든 조건을 만족하면 1, 그렇지 않으면 0을 출력한다.

Fig. 6.은 분할 배서의 전체적인 과정을 그림으로 나타낸 것이다. 분할 배서 과정의 자세한 내용은 다음과 같이 기술한다.

1. 어음 소유자는 본인의 어음 정보 $Note$를 기반으로 새로운 어음 $Note_1,Note_2$를 생성한다. 이때 $Note_1$과 $Note_2$의 금액의 합은 $Note$의 금액과 같아야 한다.

2. $r_1\leftarrow H(Note_1),r_2\leftarrow H(Note_2)$를 생성하고 이를 사용하여 $Note_1,Note_2$를 커밋한다.

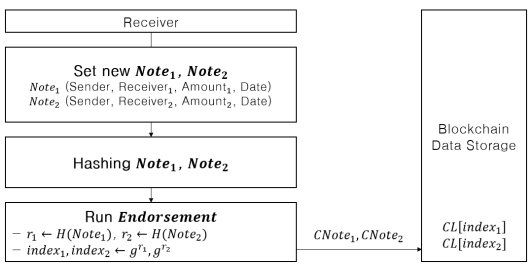


Fig. 6. The proposed protocol's split endorsement steps

$$D_1\leftarrow H(S_1,r_1),D_2\leftarrow H(R_1,r_1)$$
$$D_3\leftarrow H(A_1,r_1),D_4\leftarrow H(D_1,r_1)$$
$$E_1\leftarrow H(S_2,r_2),E_2\leftarrow H(R_2,r_2)$$
$$E_3\leftarrow H(A_2,r_2),E_4\leftarrow H(D_2,r_2)$$
$$CNote_1\leftarrow(D_1,D_2,D_3,D_4)$$
$$CNote_2\leftarrow(E_1,E_2,E_3,E_4)$$

3. 기존의 거래 정보로 $r\leftarrow H(Note)$를 생성한다.

4. 영지식 증명에 사용할 키를 생성한다.

$$KGen^{SNARK}(P_1)\rightarrow(PK_{P_1},VK_{P_1})$$
$$KGen^{SNARK}(P_2)\rightarrow(PK_{P_2},VK_{P_2})$$

5. 비밀 입력 값 $\vec{s},\vec{a},\vec{d}$를 다음과 같이 설정한다.

$$\vec{s}\leftarrow((S,r),(S_1,r_1),(S_2,r_2))$$
$$\vec{a}\leftarrow((A,r),(A_1,r_1),(A_2,r_2))$$
$$\vec{d}\leftarrow((D,r),(D_1,r_1),(D_2,r_2))$$

6. $CNote=(C_1,C_2,C_3,C_4)$를 사용하여 공개 파라미터 $\Phi_1,\Phi_2,\Phi_3$를 다음과 같이 설정한다.

$$\Phi_1\leftarrow(C_1,D_1,E_1)$$
$$\Phi_2\leftarrow(C_3,D_3,E_3)$$
$$\Phi_3\leftarrow(C_4,D_4,E_4)$$

7. 증명 스트링 $\pi=(\pi_1,\pi_2,\pi_3)$를 다음과 같이 생성한다.

$$Proof^{SNARK}(PK_{P_1},P_1,\vec{s},\Phi_1)\rightarrow\pi_1$$
$$Proof^{SNARK}(PK_{P_2},P_2,\vec{a},\Phi_2)\rightarrow\pi_2$$
$$Proof^{SNARK}(PK_{P_1},P_1,\vec{d},\Phi_3)\rightarrow\pi_3$$

8. $SplitEndrosement(r,CNotes,\pi,VK_{P_1},VK_{P_2})$ 함수(Table 6)를 호출한다.

$$CNotes=(CNote,CNote_1,CNote_2)$$

Table 5. Smart contract function of payment and endorsement

| Smart Contract | |
|---|---|
| Declaration : | |
| struct $Note$ { | |
| address $S$; | // note issuer's blockchain address |
| address $R$; | // note owner's blockchain address |
| uint $A$; | // payment amount |
| date $D$; }; | // payment date |
| struct $CNote$ { | |
| uint256 $C_1$; | |
| uint256 $C_2$; | |
| uint256 $C_3$; | |
| uint256 $C_4$; | |
| ECPoint $index$; | // location of $CNote$ in $CL$ |
| bool $state$; }; | // effectiveness of commits |
| Mapping(ECPoint → $CNote$) : $CL$ | |
| Function : | |
| $Payment(Note, r)$ | |
| 1. $index \leftarrow g^r$; | |
| 2. Require : $CL[index].state == 1$; | // if the $CNote$ is valid |
| 3. Require : $msg.address == note.R$; | // and the user who runs payment function is same with the note owner |
| 4. Require : $CL[index].C_1 == H(Note.S, r)$; | |
| 5. Require : $CL[index].C_2 == H(Note.R, r)$; | |
| 6. Require : $CL[index].C_3 == H(Note.A, r)$; | // and the payment information is same with $CNote$ |
| 7. Require : $CL[index].C_4 == H(T, r)$; | // compare the payment date with blockchain timestamp |
| 8. $note.S \xrightarrow{Note.A} Note.R$; | // do the payment |
| 9. Update : $CL[index].state \rightarrow 0$; | // now $CNote$ is invalid |
| 10. Return : 1; | |
| $Endrosement(r_1, oldCNote, CNote, \pi, VK_{P_1})$ | |
| 1. Require : $CL[oldCNote.index].state == 1$; | // if the $oldCNote$ is valid |
| 2. Require : $H(msg.address, r_1) == CL[oldCNote.index].C_2$; | // and the user who runs payment function is same with the old note owner |
| 3. Require : $Verify^{SNARK}(VK_{P_1}, \pi_1, CNote.C_1, oldCNote.C_1, 0)$; | |
| 4. Require : $Verify^{SNARK}(VK_{P_1}, \pi_2, CNote.C_3, oldCNote.C_3, 0)$; | |
| 5. Require : $Verify^{SNARK}(VK_{P_1}, \pi_3, CNote.C_4, oldCNote.C_4, 0)$; | // check if there is no change of the note issuer, payment amount, payment date |
| 6. Add : $CNote \rightarrow CL[CNote.index]$; | |
| 7. Update : $CL[CNote.index].state \rightarrow 1$; | |
| 8. Update : $CL[oldCNote.index].state \rightarrow 0$; | // now $oldCNote$ is invalid |
| 9. return : 1; | |

Table 6. Smart contract function of split endorsement

| Smart Contract |
|---|
| $SplitEndrosement(r, oldCNote, CNote_1, CNote_2, \pi, VK_{P_1}, VK_{P_2})$ |
| 1. Require : $CL[oldCNote.index].state == 1$; |
| 2. Require : $H(msg.address, r) == CL[oldCNote.index].C_2$; |
| 3. Require : $Verify^{SNARK}(VK_{P_1}, \pi_1, oldCNote.C_1, CNote_1.C_1, CNote_2.C_2)$ ; |
| 4. Require : $Verify^{SNARK}(VK_{P_2}, \pi_2, oldCNote.C_3, CNote_1.C_3, CNote_2.C_3)$ ; |
| 5. Require : $Verify^{SNARK}(VK_{P_1}, \pi_3, oldCNote.C_4, CNote_1.C_4, CNote_2.C_4)$ ; |
| 6. Add : $CNote_1 \rightarrow CL[CNote_1.index]$; |
| 7. Add : $CNote_2 \rightarrow CL[CNote_2.index]$; |
| 8. Update : $CL[CNote_1.index].state \rightarrow 1$; |
| 9. Update : $CL[CNote_2.index].state \rightarrow 1$; |
| 10. Update : $CL[oldCNote.index].state \rightarrow 0$; |
| 11. return : 1; |

9. 함수를 호출한 사용자가 어음 소유자인지 확인한다. (Table 6 *SplitEndrosement* 함수 2번째 줄 참조)

10. 기존의 $CNote$와 $CNote_1, CNote_2$의 값을 비교하여 변조 여부를 확인한다.

(Table 6 *SplitEndrosement* 함수 3, 5번째 줄 참조)

11. 두 어음의 금액 총 합이 기존 금액과 동일한지 검증한다.(Table 6 *SplitEndrosement* 함수 4번째 줄 참조)

12. 정당한 $CNote_1, CNote_2$인 경우, 기존의 $CNote$의 상태를 $false$로 설정한다.

13. $CNote_1, CNote_2$의 상태를 $true$로 설정한다.

14. $index_1 \leftarrow g^{r_1}, index_2 \leftarrow g^{r_2}$를 계산하고, $CL[index_1], CL[index_2]$에 $CNote_1, CNote_2$을 저장한다.

**Remark** (블록체인 전자 어음 담보). 어음 보증기관이 존재하지 않는 블록체인 기반 전자 어음은 부도의 가능성이 존재할 수 있다. 제안 프로토콜에서는 어음이 부도 처리되는 것을 방지하기 위하여 스마트 컨트랙트를 이용한다. 스마트 컨트랙트를 사용하면 담보의 기능을 구현하여 최소한의 보증을 받을 수 있기 때문이다.

블록체인 자산의 소유자를 스마트 컨트랙트로 설정하고, 어음을 생성할 때 스마트 컨트랙트 함수를 통해 지급이 제대로 이루어지면 소유자를 다시 어음 발행자로 설정, 이루어지지 않으면 자산의 소유자를 어음 소유자로 변경하도록 설정하는 방식으로 담보를 맡길 수 있다. 만약 지급인이 대금을 지불하지 않아 전자 어음이 부도가 나더라도 사전에 합의했던 블록체인 자산의 소유자가 어음 소유자로 변경되기 때문에 사용자는 보상을 지급받는다. 따라서 사용자는 어음을 생성하기 전, 거래 정보를 합의하면서 상대방의 신용이 충분치 않다고 생각되면 스마트 컨트랙트 함수를 통해 담보를 설정하는 방식으로 안전한 전자 어음 거래를 진행할 수 있다.

## IV. 안전성 분석

제안하는 전자 어음 프로토콜의 안전성은 구성 요소인 블록체인, 해시 커밋먼트, zk-SNARK의 안전성에 기대어 분석할 수 있다. 각 구성 요소를 대상으로 공격하는 경우로 정확성과 익명성의 공격 타입을

정의하고, 어떠한 타입의 공격도 성공하기에 현실적으로 불가능함을 증명한다. 이를 위하여 다음과 같은 하이브리드 증명 논리[10]를 이용한다.

먼저 다항 시간(polynomial-time) 공격자가 존재하는 실제 환경을 첫 번째 게임(Game 0)으로 정의하고, 공격자가 알지 못하게 점차 마지막 게임(Game F)로 변경한다. 마지막 게임에서 공격자는 공격 목표를 이루지 못하기 때문에, 현실의 공격이 이상적인 안전성을 만족하는 환경을 공격하는 것과 크게 다르지 않음을 증명할 수 있다.

**Theorem 1 (정확성)**. 제안된 시스템은 (1) 블록체인이 변조 저항성을 가지고, (2) 블록체인이 안전성을 가지며, (3) 해시 커밋먼트가 바인딩을 성질을 만족하고, (4)블록체인이 탈중앙화 성질을 가지며, (5)zk-SNARK가 완전성, (6)건전성을 가질 때 정확성을 만족한다.

*Proof.* 블록체인이 변조될 확률을 $\epsilon_1$, 블록체인의 안전성이 깨질 확률을 $\epsilon_2$, 해시 커밋먼트의 바인딩 성질이 깨질 확률을 $\epsilon_3$, 블록체인이 탈중앙화가 아닐 확률을 $\epsilon_4$, zk-SNARK의 완전성이 깨질 확률을 $\epsilon_5$, zk-SNARK의 건전성이 깨질 확률을 $\epsilon_6$로 정의한다. 또한 제안하는 프로토콜의 정확성을 깰 수 있는 공격자의 이점(advantage)을 $Adv_c$로 표기하며, Game $n$에서 가지는 정확성을 깨는 공격자의 이점을 $Adv_c(Game\,n)$과 같이 나타낸다.

공격자가 다항식 시간 안에 제안하는 프로토콜의 정확성을 깨기 위해 시도할 수 있는 공격은 다음과 같다.

Type 1. 블록체인의 데이터에 접근하여 어음 소유자, 금액, 날짜, 상태 등을 변조하는 공격

Type 2. 블록체인 과반수 공격을 통해 정당한 어음 소유자의 지급을 막거나(censored) 하나의 어음으로 여러 번 지급 받는(double spending) 공격

Type 3. 같은 커밋먼트 값을 가지는 거래 정보를 찾아 거래 정보를 위조하는 공격

Type 4. 사용자의 기기에 물리적으로 접근하여 비밀 입력 값으로 지급, 배서를 수행하는 부채널 공격, 물리적 공격

Type 5. 해당 커밋먼트 값과 같은 값을 출력하는 거래 정보를 전수 조사하여 찾은 후 해당 값으로 거래 정보를 바꾸는 공격

Type 6. 거래 정보(지급 금액, 지급 날짜 등)를 변조하고 정당한 증명 스트링을 생성하여 변조 어음을 배서하는 공격

Type 7. 공모, 해킹 등으로 커밋먼트 값 $r \leftarrow H(Note)$을 알아내 지급을 시도하는 공격

Type 8. 이미 거래가 완료된 어음을 가지고 다시 한 번 지급을 시도하는 공격

Type 9. 현재 시간 정보를 위조하여 사전에 합의한 날짜보다 이른 날짜에 지급을 시도하는 공격

Game 0. 공격자 A가 시스템의 정확성을 깨려고 하는 실제 게임

Game 1. Game 0와 동일하지만 Type 1의 공격이 이루어지지 않는 게임. 블록체인은 변조가 어렵기 때문에 저장된 어음 정보를 변조하여 거래를 변조하는 경우가 발생하지 않는다. 따라서 공격자가 Game 1에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,0) - Adv_c(Game\,1)| \leq \epsilon_1$$

Game 2. Game 1과 동일하지만 Type 2의 공격이 이루어지지 않는 게임. 블록체인의 안전성에 의하여 거래 트랜잭션 방해가 일어나지 않는다. 따라서 공격자가 Game 2에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,1) - Adv_c(Game\,2)| \leq \epsilon_2$$

Game 3. Game 2와 동일하지만 Type 3의 공격이 이루어지지 않는 게임. 같은 어음 데이터를 가지는 다른 거래 정보를 찾을 확률은 해시 커밋먼트가 바인딩 성질을 가지기 때문에 무시할 수 있을 정도로 작다. 따라서 공격자가 Game 3에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,2) - Adv_c(Game\,3)| \leq \epsilon_3$$

Game 4. Game 3와 동일하지만 Type 4의 공격이 이루어지지 않는 게임. 블록체인은 탈중앙화의 성질을 가지므로 부채널, 물리적 공격은 일어나지 않는다. 따라서 공격자가 Game 4에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,3) - Adv_c(Game\,4)| \leq \epsilon_4$$

Game 5. Game 4와 동일하지만 Type 5의 공격이 이루어지지 않는 게임. 다항식 시간 공격자이기 때문에 일어나지 않는다. 따라서 공격자가 Game 5에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,4) - Adv_c(Game\,5)| \leq 0$$

Game 6. Game 5과 동일하지만 Type 6의 공격이 이루어지지 않는 게임. zk-SNARK는 건전성을 가지기 때문에 일어나지 않는다. 따라서 공격자가 Game 6에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,5) - Adv_c(Game\,6)| \leq \epsilon_6$$

Game 7. Game 6과 동일하지만 Type 7의 공격이 이루어지지 않는 게임. 제안하는 프로토콜은 $Payment$를 호출한 사용자의 블록체인 주소 $msg.address$가 어음 정보에 저장되어 있는 어음 소유자 정보인 $Note.R$과 같은지 확인하는 과정을 거치기 때문에 공격자는 단순히 $r$을 소지하는 방법으로 부적절한 거래를 발생시킬 수 없다. 따라서 공격자가 Game 7에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,6) - Adv_c(Game\,7)| \leq \epsilon_4$$

Game 8. Game 7과 동일하지만 Type 8의 공격이 이루어지지 않는 게임. 제안하는 프로토콜은 어음이 지급 완료된 경우 해당 어음의 상태를 $false$로 설정하며, 지급 함수를 호출한다면 가장 먼저 어음의 상태를 확인하기 때문에 이미 거래가 완료되었다면 지급이 일어나지 않는다. 따라서 공격자가 Game 8에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,7) - Adv_c(Game\,8)| \leq \epsilon_1$$

Game F. Game 8과 동일하지만 Type 9의 공격이 이루어지지 않는 게임. 제안하는 프로토콜은 블록체인 타임스탬프를 사용하여 현재 시간을 받아오기 때문에 블록체인이 불변하고, 과반수 공격에 안전하면 일어나지 않는다. 따라서 공격자가 Game F에서 가지는 이점은 다음과 같다.

$$|Adv_c(Game\,8) - Adv_c(Game\,F)| \leq \epsilon_1 + \epsilon_2$$

따라서 최종적으로 가지는 공격자의 이점은 다음과 같다.

$$|Adv_c(Game\,0) - Adv_c(Game\,F)| \leq 3\epsilon_1 + 2\epsilon_2 + \epsilon_3 + 2\epsilon_4 + \epsilon_5 + \epsilon_6$$

$\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4, \epsilon_5, \epsilon_6$은 모두 무시할 수 있을 정도로 작기 때문에 Game F에서 정확성을 깨는 공격자가 가지는 이점은 Game 0와 구분할 수 없으며, Game F에서는 공격자의 어떠한 공격도 발생하지 않는다. 또한 공격이 일어나지 않는 상황에서는 zk-SNARK의 완전성에 의해 모든 증명 스트링이 정직하므로 제안하는 프로토콜의 정확성을 만족한다. 결과적으로 제안된 시스템은 정확성을 만족한다.

**Theorem 2 (익명성)**. 제안된 시스템은 (1)해시 커밋먼트가 하이딩 성질을 만족하고, (2)블록체인이 탈중앙화 성질을 가지며 (3)zk-SNARK가 영지식일 경우 익명성을 만족한다.

*Proof.* 해시 커밋먼트의 하이딩 성질이 깨질 확률을 $\delta_1$, 블록체인이 탈중앙화가 아닐 확률을 $\delta_2$, zk-SNARK의 영지식이 깨질 확률을 $\delta_3$로 정의한다. 또한 제안하는 프로토콜의 익명성을 깰 수 있는 공격자의 이점을 $Adv_a$로 표기하며, Game $n$에서 가지는 익명성을 깨는 공격자의 이점을 $Adv_a(Game\,n)$과 같이 나타낸다.

공격자가 다항식 시간 안에 제안하는 프로토콜의 익명성을 깨기 위해 시도할 수 있는 공격은 다음과 같다.

Type 1. 블록체인의 데이터로부터 거래 정보를 알아내는 공격

Type 2. 사용자의 기기에 물리적으로 접근하여 사용자 데이터와 부채널 정보를 토대로 거래 정보를 알아내는 부채널 공격, 물리적 공격

Type 3. 해당 커밋먼트 값과 같은 값을 출력하는 거래 정보를 전수 조사하여 찾는 공격

Type 4. 어음을 배서할 때 증명 스트링에서 비밀입력 값인 거래 정보를 알아내는 공격

Game 0. 공격자 A가 시스템의 익명성을 깨려고 하는 실제 게임

Game 1. Game 0와 동일하지만 Type 1의 공격이 이루어지지 않는 게임. 블록체인에 저장된 어음

정보는 커밋먼트 값이기 때문에 해시 커밋먼트가 하이딩 성질을 만족하면 거래 내용을 알아낼 수 없다. 따라서 공격자가 Game 1에서 가지는 이점은 다음과 같다.

$$|Adv_a(Game\,0) - Adv_a(Game\,1)| \leq \delta_1$$

Game 2. Game 1과 동일하지만 Type 2의 공격이 이루어지지 않는 게임. 블록체인은 탈중앙화 성질을 만족하므로 부채널, 물리적 공격은 일어나지 않는다. 따라서 공격자가 Game 2에서 가지는 이점은 다음과 같다.

$$|Adv_a(Game\,1) - Adv_a(Game\,2)| \leq \delta_2$$

Game 3. Game 2와 동일하지만 Type 3의 공격이 이루어지지 않는 게임. 다항식 시간 공격자이기 때문에 일어나지 않는다. 따라서 공격자가 Game 3에서 가지는 이점은 다음과 같다.

$$|Adv_a(Game\,2) - Adv_a(Game\,3)| \leq 0$$

Game F. Game 3와 동일하지만 Type 4의 공격이 이루어지지 않는 게임. zk-SNARK는 영지식이기 때문에 일어나지 않는다. 따라서 공격자가 Game F에서 가지는 이점은 다음과 같다

.

$$|Adv_a(Game\,3) - Adv_a(Game\,F)| \leq \delta_3$$

따라서 최종적으로 가지는 공격자의 이점은 다음과 같다.

$$|Adv_a(Game\,0) - Adv_a(Game\,F)| \leq \delta_1 + \delta_2 + \delta_3$$

$\delta_1$, $\delta_2$, $\delta_3$ 모두 무시할 수 있을 정도로 작기 때문에 Game F에서 익명성을 깨는 공격자가 가지는 이점은 Game 0와 구분할 수 없으며, Game F에서는 공격자의 어떠한 공격도 발생하지 않는다. 즉, 위와 같이 시뮬레이션 된 게임은 실제 게임과 구분할 수 없다. 따라서 제안된 시스템은 익명성을 만족한다.

## V. 결 론

본 논문에서는 거래 내용의 익명성을 보장하는 블록체인 기반 전자 어음 시스템 모델을 제안하였다. 제안하는 모델은 중앙 기관이 없어도 익명성, 정확성이 보장되며, 기존의 전자 어음과 동일하게 동작하기 때문에 우리나라의 전자 어음 시스템에 적용하기 적합하다.

하지만 제안 기술의 구성 요소인 블록체인, 해시 커밋먼트, zk-SNARK의 성질을 위협하는 것이 아닌 새로운 타입의 공격이 이루어진다면 제안하는 프로토콜의 안전성을 보장할 수 없게 된다. 따라서 모든 공격을 대상으로 엄밀한 안전성을 증명하는 연구가 추후 필요하다.

또한 제안하는 프로토콜에서 전자 어음을 배서할 때 발생할 수 있는 오류나 변조를 방지하기 위해서는 거래 정보의 커밋먼트 값을 공개하여 커밋된 값이 동일하다는 것을 증명해야 하는데, 이 경우 제공하고자 하는 익명성에 위협이 된다. 때문에 본 프로토콜은 zk-SNARK 영지식 증명을 사용하여 전자 어음 배서 시에 발생할 수 있는 오류와 변조를 방지하였다. 하지만 zk-SNARK를 많이 사용할수록 스마트 컨트랙트를 동작시킬 때 생성되는 수수료가 늘어나게 되므로, 이를 사용자들이 지불한다는 점은 제안하는 프로토콜의 단점이 될 수 있다. 따라서 제안하는 프로토콜의 효율성을 높이면서도 익명성을 지킬 수 있는 방법에 대한 추가적인 연구가 필요하다.

## References


[1] Kwon Min Kyung, "Risk Factors of DeFi and International Regulatory Trends," Capital Market Focus 01, Capital Market Institute, January 2022.

[2] Statistics Korea, "Current Usage of Electronic Bills," Bank of Korea Economic Statistics System (ECOS), 2022-2023.

[3] Kim Ik-hwan, "Over 40,000 Cases of 'Hacking' of Financial Payment Certificates at the Korea Financial Telecommunications & Clearings Institute," Hankook Economic Daily,

September 25, 2020
〔4〕 Kim Yu-dae, "Partial Errors in Korea Financial Telecommunications & Clearings Institute Card Terminals... Payment Disruptions at Gas Stations and More," KBS News Economy, February 26, 2023
〔5〕 Gov of Poland, "Electronic promissory notes on blockchain," Gov.pl, 2018
〔6〕 Batista, Danielle, et al. "Blockchains and provenance: How a technical system for tracing origins, ownership and authenticity can transform social trust." Building Decentralized Trust: Multidisciplinary Perspectives on the Design of Blockchains and Distributed Ledgers. 111-128, 2021.
〔7〕 FQX. eNITM Infrastructure, "Electronic Negotiable Instruments," fqx.ch 2020.
〔8〕 Zhang, Rui, Rui Xue, and Ling Liu. "Security and privacy on blockchain." ACM Computing Surveys (CSUR) 52.3 : 1-34, 2019.
〔9〕 Eberhardt, Jacob, and Stefan Tai. "Zokrates-scalable privacy-preserving off-chain computations." 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). IEEE, 2018.
〔10〕 Shoup, Victor. "Sequences of games: a tool for taming complexity in security proofs." cryptology eprint archive, 2004.

## 〈저자소개〉

우 현 주 (HyunJoo Woo) 학생회원
2021년 8월: 인천대학교 컴퓨터공학부 졸업
2021년 9월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 정보보호, 암호 프로토콜, PET 기술

김 효 승 (Hyoseung Kim) 종신회원
2010년 2월: 고려대학교 수학과 졸업
2021년 2월: 고려대학교 정보보호대학원 정보보호학과 박사 졸업
2021년 3월~2023년 8월: 고려대학교 4단계 BK21 스마트시티보안 교육연구단 연구교수
2023년 9월~현재: 한림대학교 정보과학대학 소프트웨어학부 조교수
〈관심분야〉 암호 프로토콜, 익명 인증, 영지식 증명, PET 기술

이 동 훈 (Dong Hoon Lee) 종신회원
1983년 8월: 고려대학교 경제학사 졸업
1987년 12월: Oklahoma University 전산학과 석사 졸업
1992년 5월: Oklahoma University 전산학과 박사 졸업
1993년 3월~1997년 2월: 고려대학교 전산학과 조교수
1997년 3월~2001년 2월: 고려대학교 전산학과 부교수
2001년 3월~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 암호 프로토콜, 암호이론, USN이론, 키 교환, 익명성 연구, PET 기술