Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.32, NO.5, Oct. 2022 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2022.32.5.899

안전한 블록체인 기반 서비스를 위한 개인키 관리 가이드라인*

노 시 완,[1†] 이 경 현[2‡]
[1,2]부경대학교 (대학원생, 교수)

A Private Key Management Guideline For Secure Blockchain-Based Services*

Siwan Noh,[1†] Kyung-Hyune Rhee[2‡]
[1,2]Pukyong National University (Graduate student, Professor)

요 약

블록체인 기반 탈중앙 서비스는 참여자들의 합의에 기반하여 시스템을 운영함으로써 중앙화된 서버 없이도 신뢰할 수 있는 서비스를 사용자에게 제공할 수 있다. 참여자들은 디지털서명 메커니즘을 사용하여 블록체인과 상호작용할 수 있지만 개인키 관리와 관련된 이슈는 여전히 해결되지 않는 문제로 남아있다. NIST SP800-57 암호키 관리 권고안 등에서는 사용자의 개인키 관리에 관한 내용을 기술하고 있지만, 이는 탈중앙 서비스 환경을 고려하지 않았기에 블록체인 환경에 적용하기에는 적절하지 않다. 본 논문에서는 개인키 관리와 관련된 지갑 애플리케이션의 기능을 정의하고 NIST SP800-57을 반영한 블록체인 개인키 관리 방안과 이를 만족하기 위한 관련 기술을 제시한다. 마지막으로 논문에서 정의한 내용을 바탕으로 퍼블릭 블록체인에서 일반 사용자를 대상으로 하는 개인키 관리 가이드라인을 제안한다.

ABSTRACT

A blockchain-based decentralized service can offer reliable services without the centralized server by operating the system based on the consensus among byzantine participants. Participants can interact with the blockchain network through a digital signature mechanism but the private key management issue remains unresolved. NIST SP800-57 provides a key-management guidance but this guidance is not appropriate for blockchain-based services because it does not consider a decentralized environment. In this paper, we define the core functions of the blockchain wallet application for private key management and present security protections according to NIST SP800-57, as well as related techniques to satisfy them. Finally, we propose the private key management guideline for secure blockchain-based decentralized services.

Keywords: Blockchain, Wallet, Private Key Management

I. 서 론

가트너의 하이프 사이클(hype cycle)에 따르면 블록체인 기반의 탈중앙 서비스는 비트코인의 성공으로 시작하는 기술의 촉발기(technology trigger), 다양한 암호화폐 및 블록체인 서비스가 등장하는 거품기(the peak of inflated expectations), 그리고 거품기에 등장했던 대다수 사업이 실패하는 환멸기(trough of disillusionment)를 지나 기술이 안정적으로 수익을 발생시킬 수 있는 성숙기(slope

Received(07. 19. 2022), Accepted(09. 15. 2022)
* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었으며 (IITP-2022-2020-0-01797) 일부는 2022년도 정부(교육부)의재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2021R1I1A3046590).
† 주저자, nosiwan@pukyong.ac.kr
‡ 교신저자, khrhee@pknu.ac.kr(Corresponding author)

of enlightenment)에 들어서고 있다[1]. 대표적으로 신원인증과정에서 신뢰기관의 개입이 필요 없는 분산신원증명(Decentralized IDentity, DID)기술[2]이 백신 접종 증명 등[3]에서 사용되고 있으며 중앙은행이 발급하는 디지털화폐(Central Bank Digital Currency, CBDC)에 대한 정부 차원의 연구 등[4]도 이루어지고 있다. 또한, 디지털 자산 영역에서는 미술품 등의 고유한 희소성이 있는 자산의 가치를 블록체인에 저장하는 대체불가 토큰(Non Fungible Token, NFT)[5], 그리고 이를 게임에 접목하여 사용자에게 게임플레이로 수익을 기대할 수 있도록 하는 플레이 투 언(Play to Earn, P2E) 시장 등[6]이 큰 관심을 받고 있다.

블록체인은 기존의 중앙화된 시스템에서 핵심적인 기능을 전적으로 담당하는 중앙서버를 제거하고 그 역할을 참여자들에게 분산시킨다. 이때 전체 시스템의 동작은 참여자들의 합의에 기반하도록 하여 악의적인 참여자에 의한 시스템 오동작을 방지할 수 있고, 따라서 중앙화된 단일지점 없이 신뢰할 수 있는 시스템 운영이 가능하다. 하지만 서버의 역할을 참여자들에게 전가함으로써 여러 문제가 발생하게 되는데, 개인키 관리[7]는 그 대표적인 사례로 관련된 피해사례가 지속해서 보고되고 있다[8].

암호화폐를 비롯한 모든 블록체인 기반 서비스에서 사용자는 개인키와 지갑(wallet) 애플리케이션을 통해 블록체인과 상호작용하여 서비스에서 제공하는 기능을 이용할 수 있기에, 안전한 블록체인 기반 서비스의 이용을 위해서는 사용자 지갑 애플리케이션이 사용자의 개인키와 관련하여 충분히 안전한 기능들을 사용자에게 제공해줄 수 있어야 한다. 하지만 현존하는 암호키 관리 권고안[9]은 그 대상이 기관(organization)으로 민간 사용자 및 탈중앙 환경을 고려하지 않았기에 기존의 암호키 관리 권고안을 블록체인 시스템에 그대로 적용하기에는 한계가 있다. 따라서 본 논문에서 기여하는 바는 다음과 같다.

- 첫 번째, 현시점에서 퍼블릭 블록체인 환경에서 지갑 애플리케이션의 개인키 관리에 대한 가이드라인이 존재하지 않으며 앞서 서술한 문제로 기존 암호키 관리 권고안을 그대로 적용하기는 어렵다. 따라서 본 논문에서는 대부분의 블록체인 기반 서비스에서 대중적으로 사용되고 타원곡선 디지털 서명(Elliptic Curve Digital Signature Algorithm, ECDSA)의 개인키 관리를 목적으로 하는 지갑을 기능적으로 정의하고 본 논문에서 목적으로 하는 환경을 고려한 키 관리 방안을 NIST SP800-57의 권고안을 기반으로 제시한다.
- 두 번째, 제시한 키 관리 방안을 준수하기 위해 필요한 관련 기술 및 그 동향을 소개한다.
- 마지막으로, 논문에서 정의한 키 관리 방안에 따라 안전한 블록체인 서비스를 위한 퍼블릭 블록체인 환경에서의 개인키 관리 가이드라인을 제안한다.

II. 배경지식

2.1 블록체인·분산원장 기술

블록체인/분산원장 기술위원회 ISO/TC307의 표준문서 ISO 22739[10]에서는 블록체인(blockchain)은 '오직 추가만 가능하고 암호학적인 링크를 사용하여 순차적인 체인으로 구성된 검증된 블록들의 분산원장'이고 분산원장 기술(Distributed Ledger Technologies, DLT)은 'DLT 노드들 사이에서 공유되고 합의 알고리즘을 사용하여 동기화되는 장부'로 정의하고 있다. 또한, 미국 표준기술 연구소(National Institute of Standards and Technology, NIST)의 내부보고서 8202[11]에서는 블록체인을 '분산기술을 사용하여 구현한 위·변조 방지 기능을 제공하는 디지털 장부(데이터베이스)'로 설명하고 있다. 즉, 블록체인은 시스템에 참여한 여러 노드들의 분산된 합의에 의해 관리되는 특수한 데이터베이스이다. Fig. 1은 퍼블릭 블록체인 시스템의 논리적인 구성을 나타낸 것으로 우리가 익히 블록체인 네트워크라고 부르는 블록체인 시스템은 물리적·네트워크 계층 위에서 동작하는 블록체인 계층에서 동작한다. 블록체인 계층은 블록체인 네트워크의 참가자들에 의해 유지되는 위·변조 방지의 컴퓨팅 플랫폼인 베이스 프로토콜 계층과 이더리움 가상머신과 같이 애플리케이션에서 지정한 명령어 집합으로 다양한 블록체인 애플리케이션을 수행하는 환경인 스마트계약 계층으로 구성되며 NFT 등으로 대표되는 블록체인 토큰은 스마트계약 계층을 통해 배포될 수 있다. 배포된 토큰은 디지털 자산으로서 검증 가능한 데이터를 교환하는 일종의 도구 역할을 하며 응용계층에서 사용자 인터페이스와 결합하여 블록체인 계층과 응용계층 사이에서 미들웨어로서 동작하게 되며 사용자는 지갑 애플리케이션을

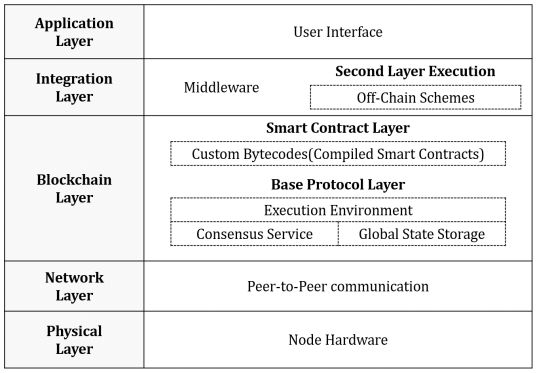

Fig. 1. Blockchain System Layer[11]

사용해 블록체인 기반 서비스를 이용할 수 있다.

2.2 블록체인 지갑

ISO 22739[10]에서는 지갑을 '개인키와 공개키를 1) 생성, 2) 관리, 3) 보관 혹은 사용하는 애플리케이션'으로 정의하고 있다. 즉, 지갑은 사용자와 블록체인 시스템을 연결하는 사용자 인터페이스(User Interface, UI)로서 상호작용을 위한 사용자의 키 쌍을 생성하고 이를 보관하면서 보관된 공개키와 개인키를 사용하여 각각 블록체인 데이터베이스에 대한 상호작용을 수행한다.

거래소 해킹으로 인한 디지털 자산의 탈취는 디파이(Decentralized Finance, DeFI) 해킹과 함께 대표적인 암호화폐 보안 사고 사례이다[8]. 대부분의 거래소는 사용자 가입 단계에서 거래에 사용할 개인키들을 생성하고 이렇게 생성된 사용자들의 개인키를 위탁 관리하는 방식으로 사용자들의 암호화폐 자산을 거래한다. 이때 사용자들의 개인키를 거래소 서버에서 동작하는 핫월렛에만 보관할 경우 해킹 등으로 개인키를 탈취당해 모든 사용자의 디지털자산을 분실할 수 있다. 2020년 11월 한국인터넷진흥원에서는 암호화폐 거래소 등 가상자산 거래를 영업으로 하는 가상자산 사업자에 특화하여 기업·기관의 정보보호 체계에 대한 인증인 정보보호 관리체계(ISMS)를 개선하였다. 가상자산사업자의 ISMS 심사에서는 기존 ISMS 항목 외에 추가로 가상자산 특화항목이 적용되었으나 ISMS 인증에서 다루는 영역은 가상자산 사업자의 관리적 보안 측면(월렛룸 출입통제, 주요직무 계정관리 등)에 중점을 두어 민간사용자를 고려하지 않은 한계가 있다. 2020년 국가정보원에서 발간한 국가공공기관 도입을 위한 블록체인 암호기술 가이드라인[12]에서는 국가공공기관에서 블록체인을 도입할 때 고려해야 할 암호기술 가이드라인을 제시하고 있다. 가이드라인의 2.2절 2-3항은 개인키 관리에 대한 권고안을 제시하고 있는데 안전한 하드웨어를 사용하여 키를 생성 및 저장할 것과 공개키를 계정관리 시스템에 등록할 것을 권고하고 있다. 하지만 하드웨어 외에 다른 키 관리 방식은 제시하지 않고 있고 하이퍼레저 등 중앙화된 계정관리 시스템이 존재하는 프라이빗 블록체인 환경만을 고려하고 있다는 한계가 있다.

III. 블록체인 지갑의 기능적 정의

NIST의 특별발간물(Special Publication, SP) 800-57[9]에서는 암호 메커니즘을 도입하려는 기관과 개발자가 적절한 메커니즘을 선택하고 사용할 수 있도록 다양한 암호키 메커니즘에 관련한 지식과 암호키 관리에 대한 권고안을 제공하고 있다. 권고안은 기밀성, 무결성 등의 보안 서비스, 이를 위한 암호 알고리즘, 키 관리를 위한 가이드(키 생명주기, 관리 주기 등)를 포함하고 있으며 특히, 암호키와 다른 파라미터(initial vectors 혹은 domain parameters)를 포함하는 키 재료(keying material) 및 키와 관련된 메타데이터(metadata)를 포함하는 키 정보(key information)에 대한 보호를 다루고 있어 본 논문에서 목표로 하는 암호키에 대한 보안과 밀접하게 관련이 있다. 하지만 권고안의 대상은 거대한 시스템을 운영하는 중앙화된 기관으로 본 논문에서 목표로 하는 민간 사용자 및 탈중앙 환경을 대상으로 적용하기에는 적절하지 않다. 따라서 이 장에서는 블록체인 기반 서비스에서 지갑 애플리케이션이 가져야 할 개인키와 관련된 공통적인 기능을 1) 키생성, 2) 키 저장, 그리고 3) 키 사용 3가지로 정의하고 본 논문에서 목적으로 하는 민간 사용자 및 탈중앙 환경에 따라 [9]의 권고안을 수정하여 각각의 기능별로 해당하는 보안 요소와 보호조치를 기술한다.

3.1 키 생성

3.1.1 NIST SP 800-57: 키 생성

[9]에서는 ECDSA에서 사용될 키 쌍의 안전한

생성에 적절한 타원곡선(elliptic curve) 및 기준점(base point) 등의 선택이 필요하다고 언급하고 있다. 하지만 알고리즘에 사용할 타원곡선 및 기준점의 선택은 Fig. 1과 같이 블록체인 계층에서 사전에 정의되어 시스템 참여자들에 도메인 파라미터로서 배포되므로 안전한 지갑 애플리케이션의 설계에서 필요한 요구사항으로 볼 수 없다. 하지만 이러한 타원곡선과 기준점이 선택된 후 개인키 생성과정에서 사용되는 난수(random number)는 사용자의 요청에 따라 지갑 애플리케이션에서 생성되므로 안전한 키 생성을 위해서는 적절한 난수의 생성이 필요하다.

안전한 암호키 생성에 대한 권고사항을 다룬 NIST SP800-133[13]에서는 키 쌍 생성에 검증된 암호 라이브러리를 사용하고 암호학적으로 안전한(cryptographically strong) 의사 난수생성기(Pseudo-Random Number Generator, PRNG)를 사용할 것을 권하였다. 또한, 기업용 이더리움 플랫폼을 개발하는 이더리움 기업 연합(Ethereum Enterprise Alliance, EEA)의 기업용 이더리움 클라이언트 개발의 규격을 정의한 문서[14]에서도 이더리움 클라이언트 구현 과정에서 보안 문제가 발생하기 쉬운 영역 중 하나로 키 쌍 생성을 언급하며 클라이언트 설계 및 보안 평가 시 개발자의 주의를 요구하였다. 즉, 비대칭 키 쌍의 생성은 블록체인 플랫폼에 따라서 시스템에서 정의하는 규격이 상이할 수 있으나 개인키의 생성과정에서 지갑 애플리케이션이 안전한 난수생성기를 사용해야 함은 자명하다고 볼 수 있다. Table 1.은 세계적으로 잘 알려진 블록체인 플랫폼 클라이언트별 키 생성에 사용되는 난수생성기에 대해 정리한 표이다. 각 클라이언트는 지갑 애플리케이션의 역할을 하며 키 생성에 암호학적으로 안전한 난수생성기를 사용함을 확인할 수 있다.

Table 1. RNG in existing wallet clients

Client	Platform	RNG
geth[15]	Ethereum	crypto/rand(Go)
ethers js [16]	Ethereum	crypto.randomBytes (Node.js)
GoQuorum[17]	Quorum	crypto/rand(Go)
Hyperledger Fabric[18]		crypto/rand(Go)
Hyperledger Besu[19]		SecureRandom (java)

3.1.2 키 생성 관련 기술

HD(Hierarchical Deterministic) 지갑은 난수로부터 생성된 하나의 마스터 시드(master seed)로부터 이론적으로 무수히 많은 주소를 생성하고 이들을 효율적으로 관리할 수 있는 기술이다. 블록체인과 사용자 사이의 상호작용은 〈블록체인 주소-디지털 서명〉의 유효성 검증을 기반으로 이루어진다. 이때 사용자가 상호작용마다 매번 동일한 주소를 사용할 경우 제3자에 의한 사용자 추적이 가능하기에 비트코인, 이더리움 등에서는 트랜잭션 생성마다 다른 주소를 사용할 것을 권하고 있으며 지갑에서 자동으로 새로운 주소를 생성하여 사용한다. 이때 지갑 애플리케이션은 여러 개의 주소와 이에 해당하는 개인키들을 관리하게 되므로 지갑이 여러 개의 블록체인 주소를 효율적으로 관리할 방안이 필요하게 되었다. HD 지갑은 하나의 마스터 시드로 알려진 단일 지점으로부터 여러 개인키와 블록체인 주소를 계층적 트리 형태로 구성하여 언제든 마스터 시드로부터 결정적(deterministic)으로 서명을 위한 개인키를 도출해낼 수 있도록 제공한다. 기존 키생성 알고리즘에서 사용자의 개인키는 각각 서로 다른 난수로부터 생성되어 독립적으로 구성되므로 키 사이의 연관성이 존재하지 않아 개별적인 관리가 필요하였다. 반면 HD 지갑은 하나의 동일한 난수로부터 여러 키를 도출해내고 이후 도출해낸 키들을 보관하는 대신 마스터 시드와 도출해낸 각각의 개인키에 해당하는 HD 패스(path)를 사용하여 이를 쉽게 도출해내는 것이 가능하다. HD 지갑은 비트코인 기능개선을 위한 제안을 담은 문서(Bitcoin Improvement Proposals, BIP)에서 제안된 여러 기술[20-22]을 기반으로 구성되어 있으며 각 기술에 대한 설명은 다음과 같다.

- BIP32 Hierarchical Deterministic Wallets[20]: BIP32는 HD 지갑의 기본적인 구조와 그 방법을 기술한 문서이다. BIP32는 계층적인 구조의 키생성 방법으로 하나의 루트 시드(root seed)를 사용하여 무수히 많은 주소를 생성할 수 있다. 루트 시드는 128~256비트의 랜덤한 수로 구성되어 있으며 사용자는 루트 시드로부터 계층적으로 파생된 트리를 구성하여 이론적으로 무한한 수의 주소를 생성할 수 있으며 사용자는 개인키 대신 오직 루트 시드만을 보관한다. 루트 시드는 장치내 안전한 영역에 보관되

어있으므로 사용자는 언제든지 원하는 키를 획득할 수 있다.

- BIP39 Mnemonic code for generating deterministic keys[21]: BIP39는 128~256bit의 랜덤한 난수인 루트 시드를 사용자가 관리하기 어렵고 필요시 이를 복구할 방안이 필요하기에 루트 시드를 특정한 단어들의 조합으로 변환해준다.
- BIP44 Multi-Account Hierarchy for Deterministic Wallets[22]: BIP32는 트리 형태로 구성되어 무한히 루트 시드로부터 자식 키 파생(child key derivation) 함수를 사용하여 자식 키(child key)를 생성하는 것이 가능하다. BIP44에서는 트리의 위치(location)를 패스로 정의한 HD 패스의 생성을 제공한다.

3.2 키 저장

3.2.1 NIST SP 800-57: 키 저장

Table 2.는 [9]에서 제시한 암호키를 위한 보호요구사항(protection requirement) 중 디지털 서명 개인키에 대한 내용만을 따로 정리한 것으로 표의 각 열에 대한 자세한 설명은 다음과 같다.

- Security Service: 암호키가 암호기술과 결합하여 제공할 수 있는 보안서비스를 나타내는것으로 디지털 서명 개인키는 디지털 서명 알고리즘을 통해 출처인증, 무결성 보장, 부인방지의 보안서비스를 사용자에게 제공할수있다.
- Security Protection: 암호키의 안전성을 위해 필요한 보호조치로 디지털 서명 개인키에 대한 조치로는 키에 대한 기밀성과 무결성이 있다.
- Association Protection: 암호키의 안전성을 위해 키와 연관되어 함께 보호되어야 하는 대상들을 의미하며 디지털 서명 개인키에 연관된 대상으로는 키의 사용 혹은 응용과정, 서명 과정에서 필요한 도메인 파라미터, 공개키가 있다.
- Assurances Required: 공개키 유효성의 보장 혹은 개인키 보유에 대한 보장의 필요 여부를 나타낸다. 디지털 서명 개인키는 개인키 보유에 대한 보장이 필요하며 키가 신뢰하는 제3자 등에 의해 생성되었거나 알려져있는 경우 개인키 사용 시점에서 실제 사용자에 의해 사용되었음에 대한 보장이 필요하다[23].
- Period Protection: 암호키에 대한 보호조치가 필요한 기간을 의미하며 디지털 서명 개인키의 경우 생성부터 폐기 시점까지 해당한다.

Table 2.에 따르면 디지털 서명 개인키의 보호에는 키의 기밀성과 무결성의 보장이 필요함을 알 수 있다. [9]에서 권고하는 저장소에서 키 정보의 보호조치는 기밀성과 무결성과 함께 가용성을 함께 제시하는데 이에 대한 자세한 설명은 다음과 같다.

- 가용성(availability): 키 정보는 암호키가 사용되는 동안 쉽게 사용할 수 있어야 한다. 이를 위해 키 정보의 복사본을 하나 이상 만들어 별도의 위치에 보관하는 것이 권고된다.
- 무결성(integrity): 무결성 보장은 키 정보에 대한 수정의 방지 혹은 탐지 그리고 승인되지 않은 수정에 대해 키 정보를 복원하는 것으로 물리적 메커니즘과 암호학적 메커니즘으로 이를 제공할 수 있다. 물리적 메커니즘은 1) 저장된 키 정보에 대한 임의 접근을 제한하는 검증된 암호 모듈 또는 운영체제, 2) 다른 시스템에 연결되어있지 않은 독립된 시스템 또는 매체, 그리고 3) 시스템 외부에 있는 적절한 접근 제어를 갖춘 물리적으로 안전한 환경을 사용하는 것이며, 암호학적 메커니즘은 1) 검증된 암호학적 무결성 보장 메

Table 2. Protection requirements for private signature key

Key Type	Security Service	Security Protection	Association Protection	Assurances Required	Period Protection
Private signature key	• Source authentication • Integrity authentication • Support for non-repudiation	Confidentiality Integrity	• Usage or application • Domain parameters (when used) • Public signature verification key	Possession	From generation until the end of the cryptoperiod

커니즘(메시지 인증 코드, 디지털 서명 등), 그리고 2) 특정한 암호학적 작업의 수행으로 키 정보의 변조를 탐지하는 방법의 사용이 있다. 오류 발생 시 키 정보의 복원에는 물리적으로 분리된 위치에 보관된 키 정보의 사본을 사용한다.

- 기밀성(confidentiality): 키 정보에 대한 기밀성 제공을 위해서는 1) 검증된 알고리즘을 사용한 암호화, 2) 암호화와 동일한 수준의 물리적 보호 기능, 혹은 3) 접근이 제한된 안전한 보관소에 의해 제공되는 물리적 보호 중 하나를 사용해야 한다.

3.2.2 가용성 보장 기술

키 복구(key recovery)는 지갑 애플리케이션이 동작하는 단말의 저장소(operational storage)의 분실 및 파손 등에 대해 대비하는 방법이다. 키 복구 메커니즘은 대표적으로 1) 개인키를 안전한 장소에 백업하는 개인키 백업 방법[21,24,25]과 2) 암호학적으로 안전하게 개인키를 분산저장·복구할 수 있는 비밀 분산 기법(Secret Sharing Scheme, SSS)을 사용하는 방법[26-30]이 존재한다.

종이 지갑(paper wallet)은 가장 기본적인 개인키 백업 방법으로 사용자의 개인키를 출력 및 이를 보관하여 해킹 등의 네트워크를 통한 공격을 원천적으로 차단할 수 있다. 니모닉 코드(mnemonic code)[21]는 이보다 발전하여 개인키를 사용자가 쉽게 관리할 수 있도록 개인키를 사전에 정해진 단어들의 조합으로 변환시킨다. 복잡한 개인키가 쉽게 기억될 수 있는 단어들의 조합으로 변환되기에 사용자는 보다 더 쉽게 개인키를 관리할 수 있으며 현재 많은 암호화폐 지갑에서 복구 수단으로 이를 사용하고 있다. Stanley는 이렇게 변환된 니모닉 코드를 보다 안전하게 관리하기 위해 양방향 마르코브 모델(2-directional markov model)을 적용하여 변환된 니모닉 코드를 기반으로 랜덤하지만 유의미한 영문서를 생성하는 방법을 제안하였다[24]. 양방향 마르코브 모델은 니모닉 코드 단어 앞뒤로 의미상 적절한 단어(BIP39에 정의된 영단어는 제외됨)를 추가하여 최종적으로 유의미한 문장을 생성하여 코드 자체를 보관하는 방법보다 안전하게 코드를 보관할 수 있다. Hosam은 개인키를 단어들의 조합으로 변환하는 대신 사용자의 개인키의 특징을 프랙탈(fractal) 트리 가지의 각도와 길이로 구분하여 최종적으로 개인키마다 고유한 프랙탈 구조로 변환되도록 하는 방법을 제안하였다[25]. 개인키를 프랙탈 트리로 변환하는 작업은 특정한 기하학적 모양의 프랙탈 트리를 그리는 것으로 이루어지며 각 가지 및 팔이 이루는 각도 및 길이는 사용된 개인키 비트열에 따라 다르다. 사용자는 생성된 트리를 개인키 대신 보관하고 필요시 트리 이미지를 사용하여 키를 복구할 수 있다.

하지만 개인 키 백업 방식은 키 자체, 혹은 키를 변환하여 보관하는 방식으로 여전히 사용자에게 그 안전성을 의존하는 한계가 있다. 비밀분산기법은 사용자의 개인키를 나눈 키 조각(share)을 특정 그룹원에게 분배하는 방법으로 개인키는 그룹원들로부터 충분한 키 조각들을 회수했을 때만 복구할 수 있어 블록체인 개인키 복구를 위한 방법으로 활발히 연구되고 있다[26-30]. 이때 키 조각은 일정 수 이상의 신뢰할 수 있고 안정적인(stable) 참여자들에게 분배되어야 하며 키 조각은 참여자의 공개키[26], 보안 질문[27], 사용자의 생체정보[28] 등으로 암호화되어 분배되며 사용자는 이후 언제든 키 조각들을 회수하여 복호화한 뒤 개인키를 복구할 수 있다. 하지만 참여자 그룹에 대한 공격으로 복구를 위한 충분한 수의 키 조각을 회수할 수 없는 상황을 고려하여 Li 등은 두 개의 임계점을 사용하는 메커니즘을 제안하였다[28]. 만약 참여자 그룹에 대한 공격으로 정상적인 임곗값인 t개의 키 조각을 회수할 수 없는 경우 v개(여기서 $t=2v$)의 키 조각만으로도 키 복구가 가능하다. 이때 키 조각의 분배에는 안전한 채널 및 공개키 인증 구조(Public Key Infrastructure, PKI)와 같은 중앙형 인증수단이 필요하다. Xiong 등은 블록체인 기반 공급망관리 프레임워크를 제안하면서 개인키의 분실/노출에 대응하기 위한 키 조각 분배 프로토콜을 함께 제안하였다[29]. [29]에서는 Password-Protected Secret Sharing과 Password-authenticated Secret Sharing을 사용하여 안전한 통신채널을 사용하지 않는 키 분배 방법을 제안하였다. Zheng 등은 인공지능 기술인 생성적 적대 신경망(Generative Adversarial Network, GAN)을 사용하여 그룹 내 참여자의 자격 등에 기반하여 그룹 내 차등적인 키 조각 분배 방법을 제안하였다[30]. 각 참여자는 보유한 능력에 따라 차등적으로 키 조각을 분배받으며 GAN은 학습을 통해 최적의 키 조각 분배 방법을 결정하여 여러 상황에 의해 사용자의 키 복구에

미치는 영향을 최소화할 수 있다.

3.2.3 무결성·기밀성 보장 기술

앞서 설명하였듯이 키 정보의 무결성과 기밀성 보장 기술은 크게 암호학적 보호 방법과 물리적 보호 방법의 사용이 권고된다. 암호학적 보호 방법은 사용자의 개인키를 다양한 암호 메커니즘을 사용하여 제3자로부터 보호하는 방법으로 패스워드를 사용한 암호화 방식이 대중적으로 사용되고 있다[7]. 이더리움 블록체인에서는 사용자의 개인키를 보호하기 위해 KeyStore 파일을 사용한다. 사용자의 개인키는 사용자가 설정한 패스워드를 기반으로 KeyStore 파일 형태로 암호화되어 존재하며 서명 단계에서는 이를 복호화하여 사용하게 된다[31]. KeyStore 파일은 UTC/JSON 포맷으로 다음과 같은 정보들을 포함한다.

- 암호화 알고리즘: 암호화에 사용한 알고리즘과 그 파라미터 정보. 주로 양방향 암호화 알고리즘인 AES(aes-128-ctr)를 사용한다.
- 키도출 알고리즘: 사용자 패스워드로부터 개인키 암호화에 사용할 암호화키를 도출하는데 사용한 알고리즘과 그 파라미터 정보. 주로 단방향 암호화 알고리즘인 Scrypt를 사용한다.
- 무결성 검증코드: 개인키 무결성 검증을 위한 메시지 인증코드. 암호화된 개인키와 개인키 암호화에 사용된 암호화키 일부에 대한 SHA3 해시값을 사용한다.

Keystore 파일에는 암호화된 개인키와 그 유효성을 검증하기 위한 무결성 검증코드(MAC)이 포함되어 있다. 개인키 복호화는 사용자가 입력한 패스워드로부터 재생성한 복호화키와 보관된 개인키의 암호문으로부터 계산한 MAC과 Keystore 파일에 저장된 MAC을 비교하여 두 값이 동일할 때 실시된다. 즉, 사용자가 유효한 패스워드를 입력했을 때 패스워드로부터 개인키 복호화를 위한 암호키를 획득할 수 있고 이를 사용, Keystore 파일 내의 암호화된 개인키를 복호화할 수 있다.

물리적 보호 방법은 물리적으로 접근이 제한된 환경에서 개인키를 보관하고 사용하는 것으로 개인키가 보관·사용되는 환경에 인가되지 않은 사용자가 접근하는 것을 차단할 수 있다. 물리적 보호 방법에 대한 자세한 설명은 3.3.2에서 기술한다.

3.3 키 사용

3.3.1 NIST SP 800-57 : 키 사용

[9]에서는 저장소에 보관된 암호키의 사용(usage) 혹은 응용(application)에 대해 '암호키가 주어진 암호 메커니즘과 함께 사용되거나 특정한 응용 프로그램과 함께 사용되는 것'으로 정의하고 있다. 이때 보호의 목적은 키 재료가 부정확하게(incorrectly) 사용되지 않았음을 보장하는 것으로 키가 올바른 키 재료(도메인 파라미터 등)와 함께 사용되며 동시에 이 관계의 무결성이 유지되는 것으로 설명하고 있다. 이는 암호키가 본래 목적과 다른 암호 메커니즘 등에서 연관되지 않은 키 재료와 함께 사용되는 것을 말하며 이를 위해 [9]에서는 다른 메커니즘 혹은 응용 프로그램과 키 재료를 분리할 것을 권고하였다.

본 논문에서는 NIST에서 정의한 키 사용의 개념을 블록체인 환경에 맞추어 사용자가 의도하지 않은 상호작용으로 정의한다. 이는 NIST의 정의인 본래 의도하지 않은 목적으로 암호키가 사용되는 것을 따라 탈중앙화된 환경에서 다른 사용자에 의해 개인키가 본래 사용자의 의도에서 벗어나 사용되는 것으로 간주한다. 이 절에서 다룰 의도되지 않은 상호작용은 사용자 단말의 해킹 등으로 인해 개인키가 직접 노출되거나 사용자 단말이 손상되어(compromised) 사용자가 의도하지 않은 상호작용이 발생하는 경우로 설명한다. 이 절에서는 이에 대해 [9]에 따른 격리된 실행환경과 본 논문에서 정의하고 있는 의도되지 않은 상호작용을 보호하기 위한 승인된 상호작용과 관련한 기술들을 소개한다.

3.3.2 격리된 실행환경

Chainalysis의 가상자산 범죄 보고서[8]에 따르면 2019년부터 2021년까지 개인키 도난으로 분실된 가상자산의 가치는 전체 분실된 자산의 30%에 달할 정도로 개인키 관리의 문제는 여전히 해결되지 않는 큰 문제 중 하나이다. 하드웨어 보안 모듈(Hardware Security Module, HSM)과 신뢰실행환경(Trusted Execution Environment, TEE)은 안전한 데이터 저장과 연산을 목적으로 하는 장치로 내부 데이터의 변조·추출·복제 방지 등을 보장할 수 있어 [9]에서 권고한 바와 같이 암호키를

다른 메커니즘 혹은 응용 프로그램과 격리된 환경에서 생성·보관·사용할 수 있는 환경을 사용자에게 제공할 수 있어 현재 블록체인의 개인키 관리와 관련된 여러 연구 및 실제 지갑 애플리케이션에서 개인키 보관 및 디지털 서명 생성에 사용되고 있다.

HSM은 암호 연산과 암호키 관리만을 위해 특화된 별도의 연산 장치로 플러그인 카드나 칩 등의 다양한 형태로 구현될 수 있다. Shbair 등은 이더리움 블록체인에서 개인키·공개키 및 서명 생성의 트랜잭션 처리를 보안 모듈에서 처리하는 방법을 softHSM과 Web3.js를 사용하여 구현하였다[32](softHSM은 PKCS#11 인터페이스를 통해 접근가능한 소프트웨어로 에뮬레이팅된 HSM으로 각종 테스트 등에 많이 활용된다). 개인키가 사용되는 모든 암호연산은 보안모듈 내에서 처리되고 개인키는 모듈 외부에서 추출하거나 볼 수 없도록 보장되어 개인키에 대한 보호는 충분하나 HSM이 가지는 제한적인 슬롯의 갯수로 인해 [32]에서는 BIP39[21]로 생성된 단어조합을 복호화하기 위한 마스터키를 HSM에 보관하고 개인키 사용시 HSM에서 마스터키를 사용하여 개인키를 재생성하는 방법을 제안하고 있다. González 등은 기업용 블록체인에서의 전자투표 시스템을 제안하였다[33]. [33]에서는 투표 권한을 가진 인증된 사용자에게 투표를 위한 익명과 함께 관계된 개인키를 HSM을 통해 발급하도록 하고 투표 트랜잭션의 서명에는 믹서(identity mixer)를 사용하여 투표자의 프라이버시 보장하였다.

TEE는 동일 장치의 메인 프로세서 내에 일반적인 응용 프로그램이 실행되는 영역(rich OS)과 별도로 격리된 신뢰영역(trusted OS)을 제공한다. TEE는 신뢰영역에 업로드된 데이터에 대한 기밀성과 무결성을 보장해줄 수 있어 개인키 관리[34-36] 뿐만 아니라 스마트계약 등 블록체인에서 실행하는 연산(on-chain)을 블록체인 외부에서(off-chain) 실행하면서 그 결과의 신뢰성을 보장하는 신뢰할 수 있는 연산을 위한 연구[37,38]에도 활용되고 있다. 블록체인 개인키 관리 영역에서는 개인키를 신뢰영역에 보관하고 서명 등 개인키를 사용하는 연산을 보안영역 내에서 실행하여 외부 공격에 대한 개인키의 보호를 제공하는 방안이 연구되고 있으며 [34-36]. 대표적인 사례인 블록체인 키스토어[34]는 삼성의 플래그십 스마트폰에 탑재된 블록체인 개인키 보관 서비스로 ARM TrustZone을 사용하여 사용자 단말에서 블록체인 개인키의 안전한 보관 및 사용을 제공한다. 키스토어는 TEE 영역에서 구현된 보안 소프트웨어인 Trusted Application에서 사용자의 개인키 생성 및 트랜잭션에 첨부하기 위한 디지털 서명의 생성 등 높은 수준의 보안을 요구하는 작업을 수행한다. 키스토어 서비스는 블록체인 상호작용을 위한 외부 애플리케이션(DApp 등)의 요청을 받은 rich OS 내의 키스토어 애플리케이션이 서명 생성 요청을 TA에 전달함으로써 개인키 데이터의 보관 및 이를 이용한 연산이 TEE 영역 내에서만 이루어져 개인키의 기밀성과 무결성을 보장할 수 있다. TEE는 또한 신뢰할 수 있는 연산 기능으로 블록체인의 성능 개선 연구에도 사용되고 있다. 이더리움 블록체인 등 기존 블록체인 스마트계약은 order-execute 아키텍처[39]를 채용하여 대다수의 네트워크 참여자가 동일한 실행결과를 얻을 때만 그 신뢰성을 보장할 수 있었지만 이로 인해 시스템의 성능 제한이 존재하게 되었다. Wang[37] 등은 Intel SGX enclave를 사용하여 스마트계약을 블록체인 외부 시스템의 보안영역에서 실행, 블록체인 시스템의 성능을 개선하고 동시에 실행결과의 기밀성을 보장하는 Hybridchain을 제안하였다. 이때 데이터는 Rich OS 영역에서는 암호화된 상태지만 보안영역에서 복호화되어 처리 후 다시 암호화되어 rich OS 영역으로 반환되므로 실행되는 데이터의 기밀성을 보장할 수 있다.

HSM과 TEE는 개인키의 라이프사이클에서 외부 공격자가 개인키에 접근하는 것을 차단할 수 있어 현 시점에서 블록체인 서비스에서 사용자 프라이버시 보호 및 개인키 관리에 가장 효율적으로 사용될 수 있다.

3.3.3 승인된 상호작용

블록체인 시스템에서 서명은 특정한 상호작용(message)이 특정한 사용자(identity)에 의해서 발생했다는 사실을 증명하고 검증하는 데에 사용된다[40]. 하지만 단일 사용자에 의한 서명 검증 메커니즘은 Table 3.과 같은 단일지점 공격에 취약하다는 단점이 있다. 임계 서명 기술(Threshold Signature Scheme, TSS)은 상호작용을 위한 권한을 여러 장소 혹은 사용자에게 분산함으로써 단일지점 공격에 대한 저항성을 가질 수 있어 블록체인 시스템에서 승인된 상호작용의 구현이 가능하다.

Table 3. The most common types of cryptocurrency-focused malware families[8]

Type	Description
Info stealers	Collect saved credentials, files, autocomplete history, and cryptocurrency wallets from compromised computers.
Clippers	Can insert new text into the victim's clipboard, replacing text the user has copied. Hackers can use clippers to replace cryptocurrency addresses copied into the clipboard with their own, allowing them to reroute planned transactions to their own wallets.
Cyptojackers	Makes unauthorized use of victim device's computing power to mine cryptocurrency
Trojans	Virus that looks like a legitimate program but infiltrates victim's computer to disrupt operations, steal, or cause other types of harm.

TSS는 상호작용에 필요한 권한을 n개의 장소(사용자 혹은 장치)에 분산하고 상호작용을 위해서는 적어도 t개의 권한($t \leq n$)이 필요하도록 하여 단일지점 공격으로 인한 영향을 줄일 수 있다[41]. Fig. 2는 블록체인 시스템에서 임계 서명 구현에 사용되는 (a)다중서명, (b)비밀분산기법, 그리고 (c)다자간계산의 기본 개념을 보여준다.

다중서명(multi signature)[42]은 상호작용에 대한 동의로서 Fig. 2의 (a)와 같이 그룹원들의 서명을 결합한 다중서명을 생성한다. 유효한 다중서명의 생성을 위해서는 전체 n개의 서명 중 최소한 t개의 서명을 획득해야 하며($t \leq n$) 일반적으로는 실용적인 측면에서 $t = n$으로 설정하여 서명에 모든 그룹원의 동의가 필요하도록 하고 있다. 하지만 이때 서명의 검증을 위해서는 서명에 참여한 그룹원들의 공개키 리스트 $L_{pk} = \{pk_1, \ldots, pk_n\}$가 필요하므로 그룹원의 수가 증가할수록 서명의 검증에 필요한 시간이 선형적으로 증가하며[43] 이를 블록체인에 기록함으로써 블록체인 네트워크에 부담을 초래할 수 있다. 또한, 다중서명을 지원하지 않는 블록체인 플랫폼에서 다중서명을 사용하기 위해서는 기존에 사용하는 서명 검증 메커니즘을 변경해야 하는 문제가 있었다[44]. 이를 해결하기 위해 Maxwell 등은 슈노르 서명을 사용하여 공개키들을 하나로 집약하여 검증자는 짧게 집약된 공개키로 서명을 검증할 수 있는 공개키 집약(public key aggregation)을 제안하였다[45] 이러한 방식은 기존 서명 검증 메커니즘을 대체할 수 있어 어떤 블록체인 플랫폼이든 쉽게 적용 가능하여 기존 방식이 가지던 유연성 측면의 문제를 해결할 수 있다. 또한, 서명 검증에 공개키 리스트 대신 이를 집약한 공개키 하나만을 필요로 하기에 검증에 필요한 비용 측면에서도 많은 절약을 할 수 있어 현재까지도 다중서명과 관련된 연구가 꾸준히 진행되고 있다[46-47].

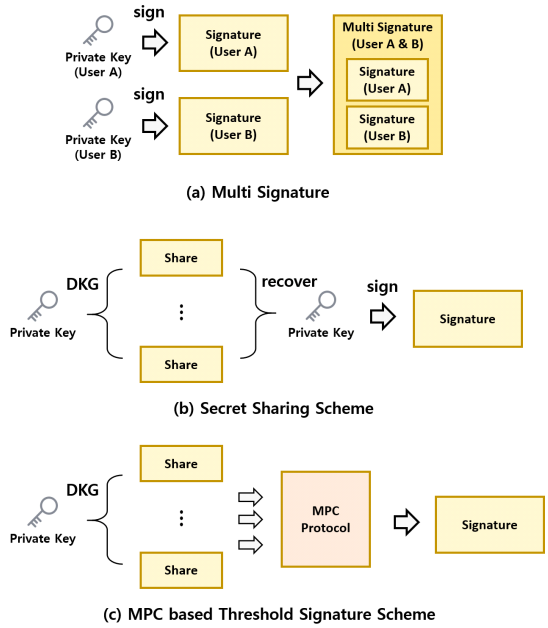

Fig. 2. An overview of the TSS

비밀분산기법은 신뢰하는 개체의 참여 없이 개체들 사이에 키 조각을 분배할 수 있는 분산 키 생성기술(distributed key generation)을 사용, 서명키를 암호학적으로 조각내어 그 조각들을 그룹원들에게 분배한다[26-30]. 키 복구를 위해서는 전체 n개의 조각 중 적어도 t개의 조각을 회수($t < n$), 키를 복구하여 사용하게 되며 Fig. 3의 (b)에서 보이듯 다

중서명과는 달리 서명생성을 위한 하나의 키가 존재하고 그룹원들에게는 키로부터 분할된 고유한 조각만이 할당되어 그룹원의 수와 관계없이 서명의 크기 및 검증시간이 일정하다. 하지만 키를 재조합하고 서명을 생성할 신뢰하는 단일지점(딜러)이 필요한 문제가 존재하였다. 다자간계산(Multi Party Computation, MPC)에 기반한 임계 서명[48,49]은 신뢰하는 딜러 문제를 그룹원 사이의 MPC 프로토콜을 사용하여 해결한다. Fig. 3의 (c)에서 나타나듯 그룹원들은 비밀분산기법과 같이 서명생성을 위해 필요한 비밀조각을 가지고 있으며 MPC 프로토콜을 통해 비밀조각들로부터 서명을 생성한다. 이 과정에서 그룹원 중 누구도 개인키를 획득할 수 없지만 유효한 서명을 계산하는 것이 가능하다.

임계 서명은 이렇듯 상호작용에 필요한 권한을 여러 개체에 분산하여 키 분실 등의 단일지점 장애로 인한 영향을 최소화할 수 있다. 이러한 권한의 분산은 시스템 운영 환경에 따라서, 같은 그룹의 여러 사용자, 사용자가 소유한 여러 단말 등으로 구현할 수 있다. TSS는 경우에 따라 서명의 생성과정에서 TEE나 HSM을 대체할 수 있는 기술로 하드웨어 기반 보안기술을 적용하기 어렵거나 하드웨어 기반과 함께 시스템에 추가적인 보안 성능을 향상시키기 위해 사용될 수 있다[50].

IV. 개인키 관리 가이드라인

3장에서는 블록체인 서비스에서 개인키 관리를 위해서 지갑 클라이언트가 가져야 할 핵심기능으로 키 생성, 키 저장, 그리고 키 사용 3가지를 정의하고 각각에 해당하는 보안요소와 조치 및 관련 기술들을 분석하였다. 이 장에서는 실제 여러 목적으로 사용되고 있는 블록체인 지갑 클라이언트들의 개인키 관리 동향을 파악하기 위해 3장에서 정의한 핵심기능 3가지에 맞춰 각 애플리케이션에서 개인키 관리를 위해 사용하는 기술을 분석한다. 그리고 앞서서 기술한 내용들을 바탕으로 퍼블릭 환경에서 민간사용자의 안전한 개인키 관리를 위한 가이드라인을 제시한다.

4.1 개인키 관리 기술 분석

Table 4.는 ECDSA를 사용하는 여러 암호화폐에서 사용되는 지갑 클라이언트의 키 생성, 키 저장, 그리고 키 사용 영역에서 실제로 사용하는 기술들을 정리한 것이다. 현시점에서 블록체인 기술은 암호화폐 외에도 유통관리, 신원증명 등 여러 분야에서 사용되고 있으나 암호화폐를 제외하면 아직까진 그 활용범위가 제한적이며 이러한 영역에서는 공개된 자료 역시 제한적이기에 이 장에서는 가장 많은 자료 수집이 가능한 암호화폐 영역에서의 조사를 수행하였다.

Table 4. Comparison of private key management techniques in existing wallet clients

Wallet	Type	Key Generation	Key Storage			Key Use
			Availability	Confidentiality	Integrity	
Bitcoin Core	Software	OpenSSL	File Backup	Password	Password	-
Metamask	Web	Web Crypto API	Mnemonic code	Password	Password	-
Ledger Nano	Hardware	TRNG, HD Wallet	Mnemonic code	Secure Element	Secure Element	Secure Element
TREZOR	Hardware	TRNG	Mnemonic code, SSS	Secure Element	Secure Element	Secure Element
Samsung Blockchain Keystore	Software	TRNG	Mnemonic code	TEE	TEE	TEE
Electrum	Software	Python secrets module	Mnemonic code	Password	Password	MultiSig
Bitstamp	Software	Weak RNG	File Backup	Password	Password	-

Table 4.의 키 저장 영역에서는 본 논문에서 앞서 정의한 바와 같이 가용성, 기밀성, 그리고 무결성 3개 세부항목으로 구분하였으며 2개 이상의 항목에 동일한 기술이 중복으로 표기되어있는 경우 이는 해당 기술이 2개 이상의 항목을 동시에 만족하는 경우이다.

- 키 생성: 조사한 모든 클라이언트가 암호학적으로 안전한 난수생성기를 사용하고 있다. 소프트웨어 타입의 클라이언트는 Python, Java, OpenSSL 등 여러 개발 언어에서 지원하는 암호 라이브러리에서 제공되는 난수생성기를 사용하였고 하드웨어 타입의 클라이언트는 물리적 처리를 통해 난수를 생성하는 하드웨어 True Random Number Generator(TRNG)를 사용하였다. 다만 삼성의 블록체인 키스토어는 소프트웨어 타입이지만 자사의 특정 라인업의 기기에서만 동작할 수 있도록 하여 키생성에 TRNG를 사용하였다. 암호화폐 거래소인 BitStamp의 경우에는 정확히 어떤 난수생성기를 사용했는지는 알려져있지 않으나 2015년 난수생성기와 관련된 문제로 사용자들에게 거래를 바로 중지할 것을 요청한 바 있다.
- 키 저장(가용성): 대부분 클라이언트에서 가용성을 위한 키 복구 솔루션으로 니모닉코드를 사용하였다. 세부적인 동작은 상이하지만 기본적으로는 사용자로 하여금 키 생성에 사용되는 시드를 BIP39를 사용하여 일련의 단어 조합으로 변경하고 이를 보관하도록 하였다. 일부에서는 신뢰기관이 참여하는 비밀분산기법을 사용하여 키 복구 서비스를 제공하거나 구글 드라이브 등 클라우드 저장소 서비스와 연계하여 암호화된 시드 파일을 외부 저장소에 보관하는 파일 백업 방식을 사용하기도 하였다.
- 키 저장(기밀성·무결성): 개인키의 기밀성과 무결성 보장을 위한 기술은 일반적으로 함께 제공되기에 대부분 클라이언트에서 동일한 기술을 사용하였다. 소프트웨어 타입에서 개인키는 암호화되어 저장소에 보관되고 서명생성 과정에서 사용자가 입력하는 패스워드로 이를 복호화하여 사용하는 패스워드 방식을 주로 사용하였다. 하드웨어 타입에서는 하드웨어에 개인키가 보관되는 격리된 실행환경을 구축하여 물리적으로 키에 대한 보안을 보장하였다. 보안 칩(Secure Element, SE)은 사용자의 데이터를 보호하고 관리하는 환경을 제공하는 칩으로 TEE와 유사한 기능을 제공한다. 삼성 블록체인 키스토어는 디바이스에 탑재되어있는 보안 솔루션인 Knox를 사용하여 개인키를 보호하여 키 생성 영역과 마찬가지로 소프트웨어 타입이나 키 생성 및 보관은 하드웨어에서 처리하고 있다.
- 키 사용: 키 사용 영역은 클라이언트 타입에 따른 적용 기술이 달라짐을 확인할 수 있다. 하드웨어 타입은 강한 물리적 보안으로 개인키가 하드웨어 내의 제한된 환경 내에서만 존재하도록 하여 별도의 암호기술을 사용하는 대신 격리된 실행환경을 통해 강한 보안을 제공하였다. 반면, 소프트웨어 타입은 네트워크에 연결되어 여러 공격에 취약하고 별도의 격리된 실행환경을 제공하기 어렵기에 임계 서명 등의 암호기술을 사용하여 안전한 키 사용 환경을 제공하고 있다.

4.2 개인키 관리 가이드라인 제안

이 절에서는 앞서 분석한 내용을 토대로 다양한 블록체인 시스템을 개발하려는 개발자 혹은 서비스 제공자의 관점에서 개인키 관리를 위해 고려해야 할 사항과 본 논문에서 기술한 내용을 바탕으로 하는 개인키 관리 가이드라인을 제안한다. 가이드라인은 3장에서 정의한 바와 같이 크게 3개 항목(키 생성, 키 저장, 키 사용)으로 나누어져 있으며 각 항목마다 2개의 세부항목을 가지고 있다. Table 5.는 제안하는 가이드라인이며 가이드라인은 총 6개 세부항목에 대해 사용자가 고려해야 할 준수사항과 이를 준수하기 위한 준수방안을 포함하고 있다. 가이드라인의 기술에 사용된 1) 해야한다, 2) 권고된다, 3) 할 수 있다 는 해당 사항에 대한 권고의 강도에 따라 사용되었으며 각각 의무사항, 권고사항, 선택사항을 의미한다. 가이드라인에서 제시하는 각종 암호기술은 특정한 암호 알고리즘을 지칭하는 것이 아니며 적절한 암호 알고리즘의 선택은 암호모듈 검증제도(KCMVP) 검증대상 암호 알고리즘을 사용할 것을 권고한다.

키생성 항목은 난수 생성(1-1), 계층적 키관리(1-2)의 두 세부항목으로 구성되어 있다. 난수생성은 블록체인 개인키의 중요한 보안요소로 지갑 애플리케이션의 타입에 따른 난수생성 방법을 소개하고 이러한 방법 중 하나를 개인키 생성에 사용하도록 한다(의무사항). 계층적 키관리는 사용자가 복수의 개

Table 5. Private key management guideline

Principles		Requirement
		Recommendation
Key Generation	(1-1) Random number generation	The wallet application developer shall use a cryptographic random number generator to generate private keys.
		☑ Software Wallet - Random number generators provided by cryptographic libraries(e.g., Python, Java, OpenSSL, etc.) shall be used. ☑ Hardware Wallet - TRNG(True Random Number Generator) shall be used to generate random numbers
	(1-2) Hierarchical key generation	A wallet application may provide a way to efficiently manage private keys
		☑ Private keys may be generated and managed hierarchically - Root seed shall be generated in compliance with the principles of 1-1. - Root seed should be backed up using BIP39
Key Storage	(2-1) Confidentiality and Integrity	The confidentiality and integrity of the root seed or private key shall be guaranteed.
		☑ The confidentiality and integrity can be guaranteed as follows. - Physical mechanisms • The private key may be stored in a physically secure area within the device where the wallet operates(trusted execution environment, hardware security module, etc. can be used) - Cryptographic mechanisms • The private key is encrypted with the password entered by the user, and the encryption key derived from the password should be used for the encryption of the private key • A message authentication code should be generated with a ciphertext to verify the integrity of the encrypted private key
	(2-2) Availability	The availability of the root seed or private key shall be guaranteed.
		☑ The availability can be guaranteed as follows. - Private key backup • The private key or root seed may be backed up using BIP39 - Secret sharing scheme • Group members who receive key shares can be friends or trusted service providers.
Key Usage	(3-1) Isolated Environment	The private key should be used in a physically isolated environment
		☑ The isolated environment may be provided as follows. - Trusted Execution Environment, Hardware Security Module, etc. • Digital signature generation can be processed in this environment. - If possible, an isolated environment should be used rather than the approved Interaction
	(3-2) Approved Interaction	If it is difficult to provide an isolated environment, the permission to use the private key should be distributed
		☑ The permission to use the private key can be guaranteed as follows. - Multi signature, Secret sharing scheme, Multiparty computation-based threshold signature, etc. • The distribution of permission may be implemented by multiple users in the organization, or multiple devices owned by the same user.

인키들을 사용하는 시스템 환경에서 사용자를 개인키를 효율적으로 관리하기 위한 방안의 제공을 권고하는 것으로 블록체인이 사용되는 시스템의 환경 등에 따라 개발자가 선택적으로 적용할 수 있다(선택사항). 키 저장 항목은 개인키의 기밀성 및 무결성(2-1), 가용성(2-2)의 두 세부항목으로 구성되어 있다. 두 세부항목 모두 개인키를 안전하게 저장하기 위한 의무사항을 NIST SP 800-57에서 기술된 내용에 근거하여 정의하고 있다(의무사항). 마지막으로 키 사용 항목은 격리된 실행환경(3-1), 승인된 상호작용(3-2)의 두 세부항목으로 구성되어 있으며 개인키 사용 시 발생할 수 있는 위협을 방지할 수 있는 기술들을 제시하고 있으며 시스템의 환경에 상관없이 가능하다면 적용하는 것을 권고하고 있다(권고사항).

V. 결 론

본 논문에서는 최근 몇 년 사이에 큰 성장을 거듭하여 암호화폐를 넘어 다양한 영역에서 실용적인 사용을 앞둔 블록체인 시스템에서 중요한 개인키 관리에 관해 다루었다. 본 논문에서 제시한 지갑 클라이언트 관점에서의 키 관리 고려사항은 개인키의 생성과 사용, 그리고 보관의 측면에서 NIST의 암호키 관리 가이드라인을 따르거나 이를 블록체인 시스템 환경에 맞춰 수정하여 기술되었으며 각 항목에서 요구하는 요구사항을 만족할 수 있는 대표적인 기술들이 함께 제시되었다. 기존 NIST의 암호키관리 가이드라인이 존재하나 이는 블록체인 시스템 환경을 고려하지 않았으므로 본 논문에서 정의한 지갑의 핵심 기능과 이에 기반한 개인키 관리 고려사항은 추후 블록체인 개인키 관리 기술 연구 및 지갑 클라이언트 개발에 기여할 것으로 기대된다.

References

[1] Gartner, "Hype Cycle for Blockchain 2021; More Action than Hype" https://blogs.gartner.com/avivah-litan/2021/07/14/hype-cycle-for-blockchain-2021-more-action-than-hype/ (accessed Apr. 18, 2022).

[2] W3C, "Decentralized Identifiers (DIDs) v1.0," Jul. 2022.

[3] "Digital vaccine passports aim to help South Koreans get back on the road", The Telegraph, Jun. 2021.

[4] Deloitte, "Are Central Bank Digital Currencies (CBDCs) the money of tomorrow?" https://www2.deloitte.com/lu/en/pages/banking-and-securities/articles/central-bank-digital-currencies-money-tomorrow.html. (accessed Apr. 18, 2022).

[5] L. Lesavre, P. Varin, and D. Yaga, "Blockchain Networks: Token Design and Management Overview." NIST IR 8301, Feb. 2021.

[6] Play to Earn Online, "Play To Earn Games: Earn NFTs & Play-To-Earn Crypto News." https://www.playtoearn.online/ (accessed Apr. 18, 2022).

[7] ITU-T, "X.1401: Security threats to distributed ledger technology," T-REC-X.1401, Nov. 2019.

[8] K. Grauer, W. Kueshner and H. Updegrave, "The 2022 Crypto Crime Report: Original data and research into cryptocurrency-based crime," Chainalysis, Feb. 2022.

[9] E. Barker, "Recommendation for key management: Part 1." NIST SP 800-57 Part 1, May 2020.

[10] ISO, "ISO 22739: Blockchain and distributed ledger technologies – Vocabulary," ISO 22739:2020, Jul. 2020.

[11] D. Yaga, P. Mell, N. Roby, and K. Scarfone, "Blockchain Technology Overview," NIST IR 8202, Oct. 2018.

[12] National Intelligence Service, "A Cryptographic Guideline for Blockchain Technology Adoption in the Pulbic Institution," Dec. 2020.

[13] E. Barker, A. Roginsky, and R. Davis, "Recommendation for cryptographic key generation." NIST SP 800-133, Jun. 2020.

[14] C. Nevile, "Enterprise Ethereum

Alliance Client Specification v7," Enterprise Ethereum Alliance, Apr. 2022.
[15] Go Ethereum, "Go Ethereum" https://geth.ethereum.org/ (accessed Apr. 18, 2022).
[16] ethers, "Ethers.js" https://docs.ethers.io/v5/ (accessed Apr. 18, 2022).
[17] GoQuorum, "GoQuorum Enterprise Ethereum Client" https://consensys.net/docs/goquorum/en/latest/ (accessed Apr. 18, 2022).
[18] Hyperledger Foundation, "Hyperledger Fabric" https://www.hyperledger.org/use/fabric (accessed Apr. 18, 2022).
[19] Hyperledger Foundation, "Hyperledger Besu" https://www.hyperledger.org/use/besu (accessed Apr. 18, 2022).
[20] P. Wuille, "Hierarchical Deterministic Wallets," BIP-0032, Feb. 2012.
[21] M. Palatinus, P. Rusnak, A. Voisine, and S. Bowe, "Mnemonic code for generating deterministic keys," BIP-0039, Sep. 2013.
[22] M. Palatinus and P. Rusnak, "Multi-Account Hierarchy for Deterministic Wallets," BIP-0044, Apr. 2014.
[23] E. Barker, C. M. Gutierrez, R. Cresanti, and W. Jeffrey, "Recommendation for obtaining assurances for digital signature applications." NIST SP 800-89, Nov. 2006.
[24] J. Stanley, "Steganographic Bitcoin seeds: Hiding cash in plain sight," https://incoherency.co.uk/blog/stories/steganographic-bitcoin-seeds.html (accessed Apr. 18, 2022).
[25] O. Hosam, "Hiding Bitcoins in Steganographic Fractals," Proceedings of the 2018 IEEE International Symposium on Signal Processing and Information Technology, pp. 512 - 519, Dec. 2018.
[26] R. Soltani, U. T. Nguyen, and A. An, "Practical Key Recovery Model for Self-Sovereign Identity Based Digital Wallets," Proceedings of the 2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress, pp. 320-325, Aug. 2019.
[27] H. P. Singh, K. Stefanidis, and F. Kirstein, "A Private Key Recovery Scheme Using Partial Knowledge," Proceedings of the 2021 11th IFIP International Conference on New Technologies, Mobility and Security, pp. 1-5, Apr. 2021.
[28] G. Li and L. You, "A Consortium Blockchain Wallet Scheme Based on Dual-Threshold Key Sharing," Symmetry, vol. 13, no. 8, pp. 1444, Aug. 2021.
[29] F. Xiong, R. Xiao, W. Ren, R. Zheng, and J. Jiang, "A Key Protection Scheme Based on Secret Sharing for Blockchain-Based Construction Supply Chain System," IEEE Access, vol. 7, pp. 126773 - 126786, Aug. 2019.
[30] W. Zheng, K. Wang, and F.-Y. Wang, "GAN-Based Key Secret-Sharing Scheme in Blockchain," IEEE transactions on cybernetics, vol. 51, no. 1, pp. 393 - 404, Jan. 2021.
[31] Ethereum Development with Go, "Ethereum Keystores" https://goethereumbook.org/en/keystore/ (accessed Apr. 19, 2022).
[32] W. M. Shbair, E. Gavrilov, and R. State, "HSM-based Key Management Solution for Ethereum Blockchain," Proceedings of the 2021 IEEE International Conference on

Blockchain and Cryptocurrency, pp. 1 – 3, May 2021.

[33] C. D. González, D. F. Mena, A. M. Muñoz, O. Rojas, and G. Sosa-Gómez, "Electronic Voting System Using an Enterprise Blockchain," Applied Sciences, vol. 12, no. 2, pp. 531, Jan. 2022.

[34] Samsung Developers, "Samsung Blockchain Keystore" https://developer.samsung.com/blockchain/keystore/overview.html (accessed Apr. 19, 2022).

[35] W. Dai, J. Deng, Q. Wang, C. Cui, D. Zou, and H. Jin, "SBLWT: A Secure Blockchain Lightweight Wallet Based on Trustzone," IEEE Access, vol. 6, pp. 40638 – 40648, Jul. 2018.

[36] W. Dai, Q. Wang, Z. Wang, X. Lin, D. Zou, and H. Jin, "Trustzone-based secure lightweight wallet for hyperledger fabric," Journal of Parallel and Distributed Computing, vol. 149, pp. 66 – 75, Mar. 2021.

[37] Y. Wang, J. Li, S. Zhao, and F. Yu, "Hybridchain: A Novel Architecture for Confidentiality-Preserving and Performant Permissioned Blockchain Using Trusted Execution Environment," IEEE Access, vol. 8, pp. 190652 – 190662, Oct. 2020.

[38] C. Muller, M. Brandenburger, C. Cachin, P. Felber, C. Gottel, and V. Schiavoni, "TZ4Fabric: Executing Smart Contracts with ARM TrustZone : (Practical Experience Report)," Proceedings of the 2020 International Symposium on Reliable Distributed Systems, pp. 31 – 40, Sep. 2020.

[39] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A.D. Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S.W. Cocco, and J. Yellick, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," Proceedings of the thirteenth EuroSys conference, pp. 1 – 15, Apr. 2018.

[40] Coinbase, "Why digital signatures are essential for blockchains" https://www.coinbase.com/cloud/discover/dev-foundations/digital-signatures (accessed Apr. 19, 2022).

[41] J. Han, M. Song, H. Eom, and Y. Son, "An efficient multi-signature wallet in blockchain using bloom filter." Proceedings of the 36th Annual ACM Symposium on Applied Computing, pp. 273-281, Mar. 2021.

[42] Gnosis, "Multisignature Wallet," https://github.com/Gnosis/MultiSigWallet (accessed Apr. 19, 2022).

[43] D.-P. Le, G. Yang, and A. Ghorbani, "A New Multisignature Scheme with Public Key Aggregation for Blockchain," Proceedings of the 2019 17th International Conference on Privacy, Security and Trust, pp. 1 – 7, Aug. 2019.

[44] G. Andresen, "Pay to Script Hash," BIP-0016, Jan. 2012.

[45] G. Maxwell, A. Poelstra, Y. Seurin, and P. Wuille, "Simple Schnorr multi-signatures with applications to Bitcoin," Designs, Codes and Cryptography, vol. 87, no. 9, pp. 2139 – 2164, Feb. 2019.

[46] R. Kojima, D. Yamamoto, T. Shimoyama, K. Yasaki, and K. Nimura, "A New Schnorr Multi-Signatures to Support Both Multiple Messages Signing and Key Aggregation," Journal of Information Processing, vol. 29, pp. 525 – 536, 2021.

[47] D. Boneh, M. Drijvers, and G. Neven, "Compact Multi-signatures for Smaller Blockchains," Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, pp. 435-464, Dec. 2018.
[48] Z. Jian, Q. Ran, and S. Liyan, "Securing Blockchain Wallets Efficiently Based on Threshold ECDSA Scheme Without Trusted Center," Proceedings of the 2021 Asia-Pacific Conference on Communications Technology and Computer Science, pp. 47-51, Jan. 2021.
[49] L. Zhou, L. Wang, Y. Sun, and P. Lv, "BeeKeeper: A Blockchain-Based IoT System With Secure Storage and Homomorphic Computation," IEEE Access, vol. 6, pp. 43472-43488, Jun. 2018.
[50] J. P. Aumasson and O. Shlomovits, "Attacking Threshold Wallets,", IACR ePrint 2020-1052, Sep. 2020.

〈저자소개〉

노 시 완 (Siwan Noh) 학생회원
2016년: 부경대학교 IT융합응용공학과 (학사)
2018년: 부경대학교 정보보호학과 (석사)
2018년~현재: 부경대학교 정보보호학과 (박사수료)
〈관심분야〉 블록체인 보안, 정보보호, 암호프로토콜

이 경 현 (Kyung-Hyune Rhee) 종신회원
1982년: 경북대학교 수학교육과 (학사)
1985년: 한국과학기술원 응용수학과 (석사)
1992년: 한국과학기술원 수학과 (박사)
1985년~1993년: 한국전자통신연구원 연구원, 선임연구원
1993년~현재: 부경대학교 컴퓨터공학부 교수
〈관심분야〉 정보보호, 블록체인 보안, 인공지능 보안