JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 20, No. 12, pp. 2485-2493, Dec. 2019

블록체인 기반 서비스 환경에서의 개인키 백업 및 복원 프레임워크

윤 태 연[1] · 문 종 섭[2*]
[1]고려대학교 정보보호학과
[2]고려대학교 전자 및 정보공학과

Private Key Backup and Recovery Framework in Blockchain-based Service Environment

Ta-Yeon Yoon[1] · Jong-Sub Moon[2*]
[1]Graduate School of Information Security, Korea University, Seoul Anam-ro 145, Korea

[요 약]

블록체인 기술은 중앙기관 없이 데이터를 장부에 분산 저장하여 관리할 수 있는 기술로 많은 응용 분야에서 블록체인 적용을 위한 연구가 활발하게 진행되고 있다. 하지만 블록체인 기술은 개인키 분실에 대한 문제가 존재한다. 개인키를 분실할 경우 자신의 모든 데이터에 대한 접근 권한을 잃게 되므로 개인키 관리가 중점적으로 연구되고 있다. 그러나 기존의 연구는 일부 정보만으로 개인키에 대한 일부 정보가 노출되거나 제3의 기관에 대한 완전한 신뢰가 있어야 하는 한계점이 존재한다. 본 논문에서는 이러한 한계점을 개선하여 블록체인을 이용한 개인키 백업 및 복원 프레임워크를 제안한다. 백업은 개인키 정보 노출 방지를 목적으로 키를 분할하여 안전하게 보관한다. 복원은 신뢰성 있는 제3의 기관을 활용하나 의존성 및 기능을 최소화한다.

[Abstract]

Blockchain is a technology that can store and manage data distributed in ledger without central authority. Research is actively conducted to apply blockchain in many applications. However, blockchain technology has no solution in case of losing private keys. Losing private key in your wallet means losing access to all of your data. However, current researches have limitations that some information on the private key can be exposed by some information or full trust in a third party is required. In this paper, we propose a framework for backing up and recovering a private key using blockchain characteristics. Backups are secured by dividing a key to prevent the disclosure of the key information. This Framework utilizes a trusted third party organization with minimization of dependencies and functionality.

색인어 : 키 백업, 키 복원, 블록체인

Key word : Key Backup, Key Recovery, Blockchain

http://dx.doi.org/10.9728/dcs.2019.20.12.2485

Received 23 October 2019; **Revised** 20 November 2019
Accepted 15 December 2019

***Corresponding Author; Jong-Sub Moon**

Tel: +※ 개인정보 표시제한
E-mail: picardcapt1212@korea.ac.kr

Ⅰ. 서 론

4차 산업혁명은 정보통신기술(ICT)의 융합으로 이루어낸 혁명 시대를 의미하며 블록체인은 이를 대표하는 기술 중 하나이다. 블록체인 기술은 생산성 향상과 효율성 확보 측면에서 비용 절감 효과 발생, 거래 효율성 증대라는 장점으로 빠르게 성장할 것으로 예측된다. 블록체인 기술을 활용한 플랫폼 개발은 현재 금융, 물류·유통·제조, 공공 서비스 등 다양한 분야에 적용되고 있다[1].

블록체인 기술은 P2P (P2P:peer-to-peer) 네트워크에 참여하는 모든 참여자가 공동으로 거래 장부를 소유하고 기록, 보관할 수 있는 분산장부 기술(DLT: Distributed Ledger Technology)이다. 블록체인 네트워크에 참여하는 개인과 개인의 거래가 발생할 때마다 데이터는 장부에 기록되어 블록으로 만들어진다. 이 블록들은 순차적으로 연결되어 사슬 구조를 형성하여 블록체인이 생성된다. 블록체인은 거래 장부를 누구에게나 공개하는 투명성을 가지고 있어 거래 내역을 추적할 수 있다. 이 때문에 데이터 위·변조에 강하게 대처할 수 있다는 장점이 있다.

데이터 위·변조 관점에서 보면 블록체인을 기술은 보안의 3요소 중 무결성이 매우 뛰어나다. 블록체인 네트워크의 참여자가 거래 내역을 공유하고 대조하는 과정을 통해 거래 내역의 진위 판별이 가능하다. 하지만 블록체인의 기밀성 측면에서 블록체인의 키 관리는 보안 위협을 초래한다. EU 산하 정보보호기구인 ENISA에 의하면 블록체인 관련 보안 위협 중 하나로 개인키 관리를 지적했다[2]. 블록체인 기술은 PKI(PKI: Public Key Infrastructure) 구조를 이용한 개인키, 공개키 쌍을 생성하여 사용자에게 인증키로 부여한다. 개인키는 거래를 발생시키고 거래가 유효한지 검증할 때 사용하는 중요한 요소로 개인키에 대한 관리가 블록체인 기술에서 중점적으로 다루어야하는 분야 중 하나이다. Slaughter And May 보고서에 따르면 블록체인 기술의 장점은 거래 장부에 대한 사이버 공격을 막을 수 있는 것이라고 하면서도, 개인키에 대한 사이버 공격의 위험성을 지적하였다[3]. 개인키에 대한 보안은 블록체인 기술의 핵심적인 요인이라고 기술하고 있다.

블록체인 네트워크는 개인키를 통해 전자 자산에 접근할 수 있다. 이러한 개인키 보관은 소프트웨어/하드웨어 지갑, 소셜 네트워크에 기반한 지갑, 제3의 기관을 사용하여 보관된다. 하지만 개인키가 도난당하거나 분실할 경우 자신의 전자 자산에 대한 소유권을 주장하기 어렵게 된다. 기존 서버-클라이언트 환경의 서비스는 개인이 개인키를 별도로 관리할 필요가 없다. 개인키를 분실하더라도 서버에 본인인증을 하고 되찾을 수 있다. 하지만 블록체인 환경의 경우 본인인증을 해줄 서버가 존재하지 않아 본인인증으로 개인키를 복원할 수 없다. 블록체인 환경에서 개인키 복원은 다양한 경우에 필요하다. 개인이 개인키를 보관하던 저장 매체에 접근이 불가한 경우, 개인키를 관리하던 제3자가 개인키를 분실한 경우[4], 전자 자산을 유산으로 물려줄 경우가 그 예이다. 현재, 키 보호 및 복원 방안에 관한 연구는 키 위탁, 비밀 공유 방식, 생체 인증 방식으로 이루어지고 있다.

현재까지 연구된 키 보호 및 백업 방안은 비밀키 공유 방식을 이용한다는 점과 물리적 공격으로 인한 개인키 분실이 발생하면 개인키를 복원하지 못한다는 점에서 한계를 가지고 있다. 이에 본 논문에서는 신뢰할 수 있는 기관의 역할을 최소화하면서 온라인, 오프라인을 모두 활용하여 개인키를 안전하게 보관하는 방법과 개인키 분실 이외의 물리적인 공격에도 개인키를 복원하는 방안을 제시한다. 저장 매체에 보관되었던 개인키를 블록체인의 노드에 분산 저장하여 개인키의 기밀성을 향상시키고 개인키를 백업하고 복원하는 과정에서 개인키에 대한 정보가 노출되지 않도록 하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 현재까지 키 복원에 사용된 기법을 소개한다. 3장에서는 기존의 개인키 복원 방안의 한계점을 개선한 백업 및 복원 프레임워크를 제시한다. 4장에서는 3장에서 제안한 프레임워크의 안전성 검증에 대하여 설명하고 마지막 5장에서는 결론과 앞으로의 연구 방향을 제시한다.

Ⅱ. 관련 연구

이 장에서는 개인키를 보호하기 위한 기법 및 연구의 특징을 소개하고 각 기법에 대한 한계점을 살펴본다.

2-1 키 위탁

키 위탁(Key Escrow)이란, 거래 및 전자 서명 등에 사용되는 사용자의 키를 안전하게 보관하기 위해 신뢰할 수 있는 제3의 기관에 키를 맡기는 방안이다[5]. 사용자가 개인적으로 키를 관리할 경우 외부에 쉽게 노출되는 파일에 저장하거나 클라우드 환경에 저장하여 개인키 정보가 쉽게 노출될 수 있다. 이를 방지하기 위해 신뢰할 수 있는 기관에 키를 보관하여 자신의 개인키에 불특정 다수가 접근하는 것을 방지할 수 있다. 사용자가 자신의 개인키를 분실할 경우 제3의 기관은 사용자 인증 과정을 통해 개인키를 사용자에게 제공한다. 하지만 키 위탁은 사용자 개인키의 보안 수준이 전적으로 제3의 기관에 의존하므로 제3의 기관에 대한 신뢰도가 매우 중요하다. 키 위탁은 제3의 기관이 외부에서 공격을 당하거나 내부자의 정보 유출로 인해 사용자의 키 정보가 노출될 가능성이 있다는 한계가 있다.

2-2 비밀키 공유 방식

비밀정보를 안전하게 보관하는 방법으로 가장 일반적으로 쓰이는 방법은 여러 개의 보관 장소에 비밀정보를 백업해두는 방법이다. 하지만 비밀정보를 여러 곳에 백업하는 방법은

비밀정보에 대한 정보 유출 가능성이 커지는 위험이 있다. 이러한 정보 유출 가능성을 감소시키기 위해 샤미르는 비밀키 공유 방식 (Secret Sharing Scheme)을 제안했다[6]. 이 방식은 분할된 정보 조각이 일부 노출되어도 안정적으로 키 관리 체계를 구축하는 방법이다. 비밀키 공유 방식은 비밀정보 D를 n개의 서로 다른 정보 조각으로 나눈다. 분할된 전체 정보 조각 중 정해진 개수 k개 이상이 모여야 비밀정보 D를 복원할 수 있다. 반면에 k – 1개 이하의 조각만으로는 비밀정보 D에 대한 어떠한 정보도 얻는 것이 불가능하다. 이것을 (k, n) 비밀 공유라 부른다. 비밀키 공유 방식은 암호를 사용하지 않고 유한체(Finite Field)상의 다항식 계산 복잡도에 의존하는 방식이다. 비밀키 공유 방식은 보안상의 이유로 주기마다 새로운 조각으로 비밀정보를 분할 할 필요가 있다. k개 이상의 데이터 조각을 모아 키를 복원할 수 있는 비밀키 공유 방식은 개인키에 대한 정보가 유출되면 안 되는 블록체인 환경에 적합하지 않다.

2-3 개인키 보호 기법 연구

개인키를 보호하기 위한 방법으로 생체정보 기반의 서명 스킴 연구가 제시되었다[7]. 생체정보 기반의 서명 스킴은 사용자의 지문을 일대일로 대응하여 개인키로 사용하는 방안으로 사용자의 허가 없이는 변조할 수 없다는 점과 분실된 개인키를 언제든지 생체정보를 통해 복구할 수 있다는 점을 강조했다. 또한, 생체정보를 기반으로 한 암호화 및 복원 메커니즘도 제시되었다[8]. 하지만 블록체인의 특징인 익명성을 제한한다는 점과 사용자의 생체정보를 수집하는데 필요한 높은 비용을 고려했을 때 현실적으로 제한될 수 있다. 생체정보를 활용하지 않고 평문 텍스트에 개인키를 숨겨 보호하는 방안[9] 및 스테가노그래피 기법을 이용해 개인키를 숨기는 방안도 존재한다[10]. 하지만 이러한 방안은 개인키를 복원할 수 없다는 한계가 존재한다.

Ⅲ. 개인키 백업 및 복원 프레임워크

이 장에서는 본 논문에서 제안하는 개인키 백업 및 복원 방안에 관해 기술한다. 이를 위해 백업 및 복원 프레임워크의 개요를 소개하고 개인키 백업, 복원 절차에 대해 살펴본다.

3-1 개인키 백업 및 복원 개요

본 논문에서 제안하는 프레임워크의 참여 주체는 그림 1.과 같다.

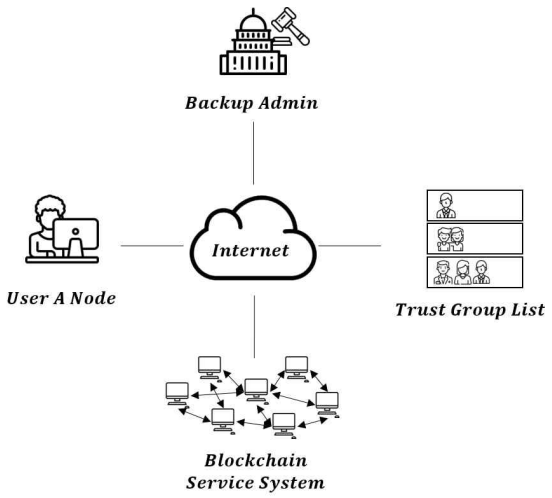

그림 1. 개인키 백업 프레임워크 참여 주체
Fig. 1. Participants in private key backup framework

- User A Node: 자신의 개인키를 블록체인 시스템에 백업하려는 사용자로 개인키 백업을 실행하는 시스템.
- Trust Group List(TGL): 사용자 A가 지정한 신뢰 그룹 목록으로, 사용자 A의 개인키를 복원 권한이 있는 사용자 목록임. 신뢰 그룹 목록에 등록된 사용자들은 개인키가 백업되는 블록체인에 사용자로 등록되어 있어야함.
- Backup Admin(BA): 사용자 A가 지정한 신뢰할 수 있는 제3의 기관인 백업 관리자로, 대칭키 복원에 관여하고 적법한 복원 요청자임을 나타내는 증명서를 발급하는 기관.
- Blockchain Service System: 사용자 A가 사용하는 블록체인 환경으로 분할된 키 정보를 보관하는 블록체인 기반 서비스 시스템.

사용자 A가 자신의 노드에서 개인키 백업을 위해 신뢰 그룹 목록, 백업 관리자를 지정한다. 이후 사용자 A의 개인키는 설정에 따라 복제 및 분할되어 블록체인 시스템에 분산 저장된다. 백업 관리자는 복원 요청자의 복원 권한 및 기타 정책적인 검증을 수행하고 개인키 복원에 필요한 정보를 제공한다. 본 논문에서는 백업 관리자의 정책적 측면에 대해서 기술하지 않는다.

제안하는 프레임워크에 필요한 용어는 아래와 같다.

- A_{priv}: 사용자 A의 개인키.
- $A_{priv}^{'}$: 대칭키 k로 암호화한 A의 개인키.
- BA_{priv}: 백업 관리자의 개인키.
- BA_{pub}: 백업 관리자의 공개키.
- TGL(Trust Group List): 신뢰 그룹 목록. $TGL=\{G_1,$

$G_2, ..., G_n\}$

- G_i: 신뢰 그룹 목록의 각 그룹으로, 복원 권한이 있는 사용자들의 집합. $G_i = \{TU_{i,1}, TU_{i,2}, ..., TU_{i,m_i}\}$. 각 그룹별로 m_i은 달라질 수 있음.
- U_{id}: 개인키 백업을 사용하는 사용자 아이디.
- $TU_{i,j}$: i번째의 신뢰 그룹 목록에 등록된 j번째 사용자
- $TU_{i,j_{priv}}$: i번째 신뢰 그룹 목록에 등록된 j번째 사용자의 개인키
- $TU_{i,j_{pub}}$: i번째 신뢰 그룹 목록에 등록된 j번째 사용자의 공개키
- k: 랜덤하게 생성된 대칭키
- $k_i^{'}$: k를 $TU_{i,1_{pub}}$로 암호화한 결과.
- $k_i^{''}$: $k^{'}$를 BA_{pub}로 암호화한 결과 .
- B_i: $A_{priv}^{'}$에 crc, $padding$을 추가한 데이터.
- $B_{i,j}$: $A_{priv}^{'}$를 G_i의 각 사용자에 대응하는 분할 블록.
- $B_{i,j}^{'}$: $B_{i,j}$을 G_i의 각 사용자의 공개키로 암호화 한 블록.
- $B_{i,j}^{''}$: $B_{i,j}^{'}$에 복원에 필요한 블록 정보(U_{id}, G_i, seq)를 추가한 데이터.
- seq: 분할한 블록의 순서 정보.
- crc: 오류 검출을 위한 CRC 코드 정보.
- doc: 복원에 필요한 제출 서류로, 복원 요청자가 백업 관리자에게 제출하는 서류.
- $Vcert$: 백업 관리자가 적법한 요청자에게 제공하는 인증 증명서로, 복원 요청자가 신뢰 그룹 내에 등록된 적법한 요청자임을 증명할 수 있는 증명서.
- $Node$: 블록체인 네트워크에 연결된 사용자의 단말 시스템.
- $BNum$: 블록체인 네트워크에 분산 저장할 데이터의 개수.
- $agent$: 복원 프로세스를 수행할 대리인으로 신뢰 그룹 목록에 등록된 사용자.

3-2 개인키 백업

개인키 백업은 총 8단계로 이루어진다. 백업에 사용되는 데이터는 A_{priv}, k, $TU_{i,j_{pub}}$, BA_{pub}이다. 그림 2.는 위의 데이터를 이용해 사용자 A의 개인키 백업 과정을 도식화한 것이다. 각 단계에 대한 자세한 설명은 아래와 같다.

1) Prepare Backup

사용자 A는 TGL, $BackupAdmin$, $BNum$을 선정하여 백업을 준비한다. 사용자 A는 신뢰할 수 있는 사용자를 직접 선정하고 개인키 복원 권한을 부여한 TGL을 작성한다. TGL은 G_i으로 구성되며 사용자 A는 G_i에 신뢰할 수 있는 사용자를 등록한다. 사용자 A는 그룹 개수를 n으로 지정할 수 있으며 그룹 내 사용자는 1명 이상의 사용자 m_i명을 등록할 수 있다. m_i 값은 그룹마다 다르게 설정할 수 있다. 예를

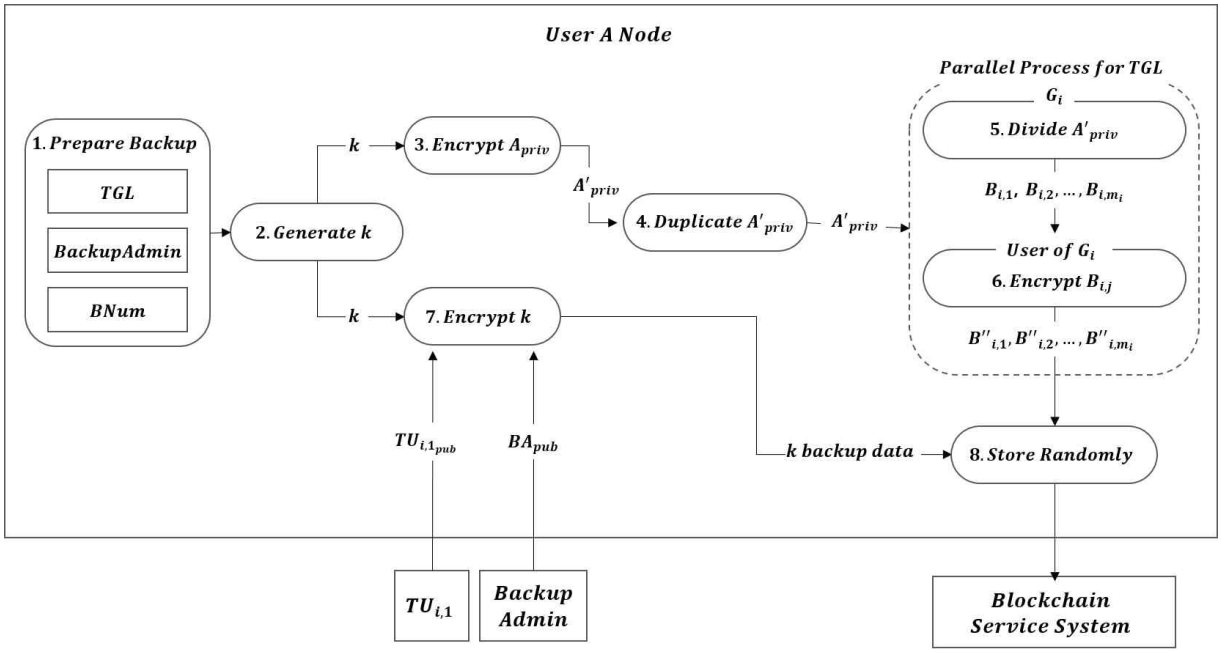

그림 2. 개인키 전체 백업 과정
Fig. 2. Private Key Backup Process

들어 $G_1=\{B,C\}$, $G_2=\{B,D\}$, $G_3=\{E,F,G\}$,... 와 같이 설정할 수 있다.

TGL은 (1) 수식으로 표현할 수 있다. i는 TGL내부의 그룹 개수이고 j는 그룹 내의 사용자 수를 의미한다. 각 그룹의 사용자 개수는 달라질 수 있으며 그룹별 사용자 수는 m_i로 표현한다. TGL에 대한 도식화는 그림 3.과 같다.

$$TGL=\{G_i : TU_{i,j}\} \quad (1)$$
$$i:(1\le i\le n)$$
$$j:(1\le j\le m_i)$$

$$TGL$$
$$G_1 : TU_{1,1}, TU_{1,2}, \dots, TU_{1,j}, \dots, TU_{1,m_1}$$
$$G_2 : TU_{2,1}, TU_{2,2}, \dots, TU_{2,j}, \dots, TU_{2,m_2}$$
$$\dots$$
$$G_i : TU_{i,1}, TU_{i,2}, \dots, TU_{i,j}, \dots, TU_{i,m_i}$$
$$\dots$$
$$G_n : TU_{n,1}, TU_{n,2}, \dots, TU_{n,j}, \dots, TU_{n,m_n}$$

그림 3. 신뢰 그룹 목록
Fig. 3. Trust Group List

2) Generate k

사용자 A의 개인키 A_{priv}를 암호화하기 위해 대칭키 k를 랜덤하게 생성한다. 이 과정은 사용자 A의 노드에서 이루어진다.

3) Encrypt A_{priv}

랜덤하게 생성한 k를 이용해 A_{priv}를 암호화하여 $A_{priv}^{'}$를 생성한다.

4) Duplicate $A_{priv}^{'}$

TGL의 G_i단위로 키 백업 및 복원을 수행하기 위해 $A_{priv}^{'}$를 n개를 복제하여 사용한다.

5) Divide $A_{priv}^{'}$

$A_{priv}^{'}$를 분할하기 전에 crc를 계산한다. $A_{priv}^{'}$과 crc를 추가한 길이가 m_i의 배수가 아닐 경우 패딩값을 추가하여 B_i를 생성한다. B_i의 구조는 그림4.와 같다.

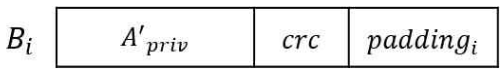

그림 4. B_i 생성 과정
Fig. 4. Generate B_i

생성된 B_i값을 모든 $G_1, G_2, \dots, G_n$의 각각의 사용자 수에 맞게 분할한다. 각 그룹의 사용자 수 m_i으로 분할한 B_i의 구조는 그림 5.와 같다.

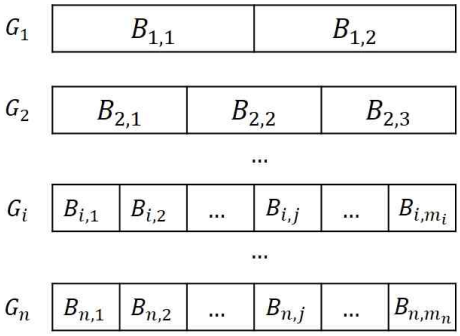

그림 5. $A_{priv}^{'}$ 분할 과정
Fig. 5. Divide $A_{priv}^{'}$

6) Encrypt $B_{i,j}$

$TU_{i,j_{pub}}$를 이용해 $B_{i,j}$를 암호화하여 $B_{i,j}^{'}$를 생성한다. $B_{i,j}^{'}$ 생성 과정은 그림 6.과 같다. 모든 G_i에 대하여, 이 과정을 수행한다.

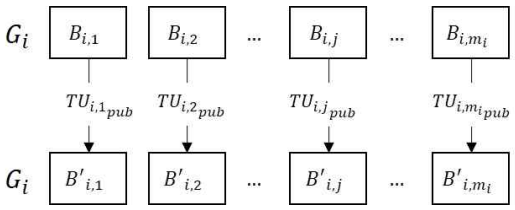

그림 6. $B_{i,j}$ 암호화 과정
Fig. 6. Encrypt $B_{i,j}$

$B_{i,j}^{'}$를 생성한 후 복원에 필요한 정보를 블록에 추가하여 $B_{i,j}^{''}$를 생성한다. 복원에 필요한 정보는 U_{id}, G_i, seq이다. 생성된 $B_{i,j}^{''}$의 구조는 그림 7.과 같다.

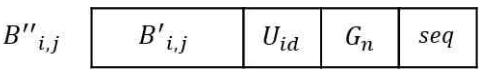

그림 7. $B_{i,j}^{''}$ 생성 과정
Fig. 7. Generate $B_{i,j}^{''}$

7) Encrypt k

대칭키 k는 분할하지 않고 이중으로 암호화한다. 각 그룹 첫 번째 사용자 $TU_{i,1_{pub}}$를 이용하여 $k1$를 암호화하여 $k_i^{'}$을 생성한다. 그리고 $k_i^{'}$을 BA_{pub}로 암호화하여 $k_i^{''}$을 생성한다. 모든 G_i에 대하여 이 과정을 수행하며 k를 암호화하는 과정은 그림 8.와 같다.

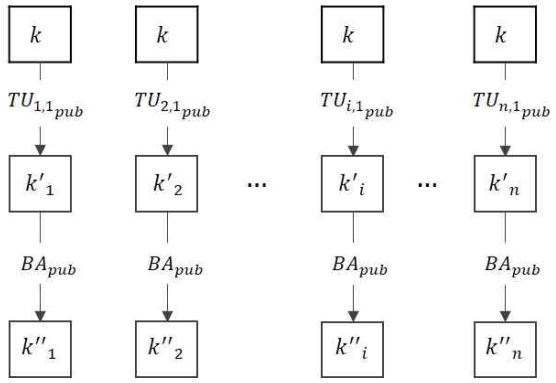

그림 8. k 암호화 과정
Fig. 8. Encrypt k

$k_i^{''}$을 생성한 후 U_{id}, G_i를 추가하여, 복원 단계에서 사용할 수 있는 데이터 (k backup data)를 그림 9.와 같이 구성한다.

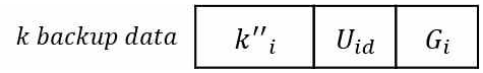

그림 9. k backup data 생성 과정
Fig. 9. Generate k backup data

브로드캐스팅을 이용해 3-2-1)절에서 지정한 $BNum$ 개수의 $B_{i,j}^{''}$와 (k backup data)를 블록체인 시스템에 분산 저장한다.

예를 들어 $BNum$값이 500이면 $B_{i,j}^{''}$와 k backup data를 각각 500개의 노드에 분산 저장한다.

3-3 개인키 복원

개인키 복원은 총 7단계로 이루어진다. 복원에 사용되는 데이터는 $B_{i,j}^{''}$, (k backup data), BA_{priv}, $TU_{i,j_{priv}}$이다. 그림 10.은 위의 데이터를 이용해 사용자 A의 개인키 복원 과정을 도식화한 것이다. 각 단계에 대한 자세한 설명은 아래와 같다.

1) Verify $agent$

복원 프로세스를 수행하는 $agent$는 백업 관리자에게 필요한 모든 doc을 제출한다. 백업 관리자는 doc을 검토하여 $agent$가 사용자 A의 TGL에 등록된 사용자인지 검토한다.

doc은 $agent$가 사용자 A와의 관계를 증명할 수 있는 인증서가 될 수 있다. 또는 사용자 A가 자신의 전자 자산을 유산으로 남기기 위해 TGL을 작성한 경우 doc은 A의 사망 진단서가 된다. 사용자 A가 사망했다고 가정하면 $agent$는 $TU_{i,j}$을 대표하여 A의 사망 진단서와 함께 본인 증명 인증서를 백업 관리자에게 제출한다. 사용자 A 본인은 TGL에

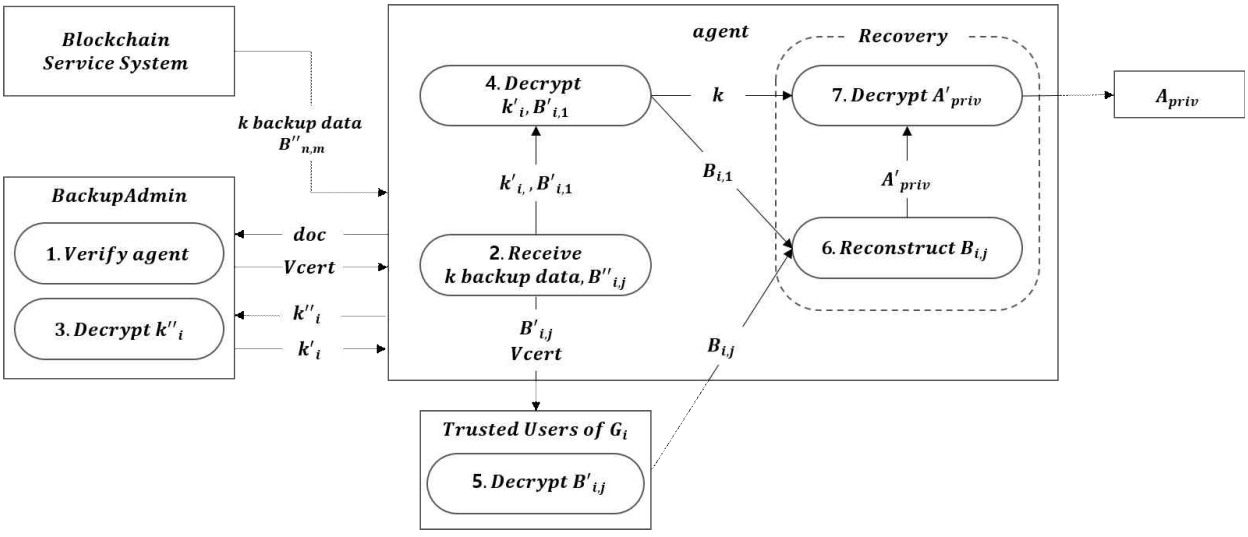

그림 10. 개인키 전체 복원 과정
Fig. 10. Private Key Recovery Process

등록할 수 없다. 따라서 사용자 본인이 개인키를 분실한 경우에는 사망 진단서 대신 신분 증명서를 기반으로 동일한 프로

8) Store Randomly

세스를 적용한다. 이러한 요건은 복원을 위한 정책적인 부분이므로 자세한 방법은 본 논문에서 생략한다.

$agent$가 제출한 모든 doc이 검증되고 $agent$가 TGL에 등록된 경우, 백업 관리자는 $Vcert$를 생성한다. $Vcert$는 $agent$가 TGL에 등록되어 있으며 함께 등록된 G_i의 $TU_{i,j}$을 나타내는 내용을 담고 있다. 백업 관리자는 $Vcert$를 $agent$의 공개키로 암호화하고 백업 관리자의 개인키로 전자 서명하여 $agent$에게 전달한다.

2) Receive k backup data, $B_{i,j}^{''}$

$agent$는 자신의 $Node$에서 A의 개인키 복원을 수행한다. $agent$는 블록체인으로부터 U_{id}와 G_i 데이터를 이용하여 k backup data와 $B_{i,j}^{''}$을 확보한다.

3) Decrypt $k_i^{''}$

$agent$는 (k backup data)에서 U_{id}, G_i를 기반으로 $k_i^{''}$를 추출한다. $agent$는 $k_i^{''}$와 $Vcert$를 온라인 또는 오프라인으로 백업 관리자에게 제출한다. 백업 관리자는 $Vcert$를 검토하여 타당할 경우 BA_{priv}를 이용해 $k_i^{''}$를 복호화하여 $k_i^{'}$를 추출한다. 백업 관리자는 추출된 $k_i^{'}$를 $agent$에게 전달한다. $k_i^{''}$를 복호화하는 과정은 그림 11.와 같다.

그림 11. $k_i^{''}$ 복호화 과정

Fig. 11. Decrypt $k_i^{''}$

4) Decrypt $k_i^{'}$, $B_{i,1}^{'}$

$agent$는 U_{id}, G_i, seq를 기반으로 $B_{i,1}^{''}$에서 $B_{i,1}^{'}$을 도출한다. $agent$는 TGL에 등록된 $TU_{i,1}$에게 $Vcert$, $B_{i,1}^{'}$, $k_i^{'}$를 전송하여 복호화를 요청한다.

$TU_{i,1}$은 $agent$가 $Vcert$에 등록되어 있고 $TU_{i,1}$도 등록된 것을 확인한 후 자신의 개인키를 이용해 $B_{i,1}^{'}$를 복호화하여 $B_{i,1}$을 생성한다. 또한, 자신의 개인키로 $k_i^{'}$를 복호화하여 k를 생성한다. $TU_{i,1}$은 $agent$의 공개키로 $B_{i,1}$, k를 암호화하여 전달한다. $B_{i,1}^{'}$와 $k_i^{'}$를 복호화하는 과정은 그림 12.와 같다.

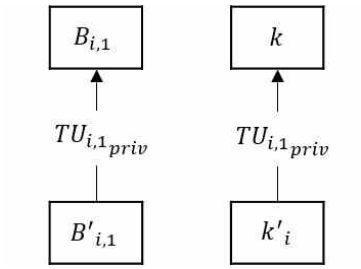

그림 12. $B_{i,1}^{'}$, $k_i^{'}$ 복호화 과정

Fig. 12. Decrypt $B_{i,1}^{'}$, $k_i^{'}$

5) Decrypt $B_{i,j}^{'}$

$agent$는 U_{id}, G_i, seq를 기반으로 각각의 $B_{i,j}^{''}$에서 $B_{i,j}^{'}$을 도출한다. $agent$는 $Vcert$와 $B_{i,2}^{'}$, $B_{i,3}^{'}$, ..., $B_{i,j}^{'}$, ..., $B_{i,m_i}^{'}$를 $TU_{i,2}, TU_{i,3}, ..., TU_{i,j}, ..., TU_{i,m_i}$에게 각각 전송하고 복호화를 요청한다.

각각의 $TU_{i,j}$은 $Vcert$를 검토한다. $agent$가 $Vcert$에 등록되어 있고 모든 $TU_{i,j}$도 등록된 것을 확인한 후 자신의 개인키를 이용해 $B_{i,j}^{'}$를 복호화하여 $B_{i,j}$를 생성한다. 각각의 $TU_{i,j}$는 복호화한 $B_{i,j}$를 $agent$의 공개키로 암호화하여 $agent$에게 전송한다. $B_{i,j}^{'}$를 복호화하는 과정은 그림 13.과 같다.

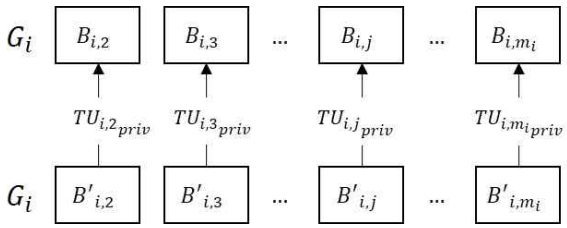

그림 13. $B_{i,j}^{'}$ 복호화 과정

Fig. 13. Decrypt other $B_{i,j}^{'}$

6) Reconstruct $B_{i,j}$

$agent$는 $B_{i,j}$ 재조립하여 seq, crc를 기반으로 B_i를 확보한다. B_i을 재조립하는 과정은 그림 14.와 같다.

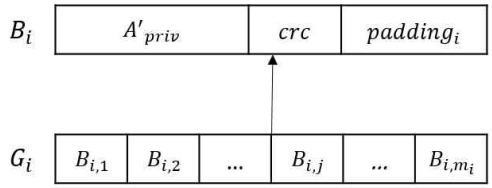

그림 14. 개인키 백업 과정
Fig. 14. Reconstruct $B_{i,j}$

7) Decrypt $A^{'}_{priv}$

$agent$는 $TU_{i,1}$이 복원한 k를 이용하여 $A^{'}_{priv}$ 복호화를 수행해 A의 개인키 A_{priv}를 확보한다.

Ⅳ. 안전성 검증

이 장에서는 본 논문에서 제시한 블록체인 기반 서비스 환경에서 개인키 백업 및 복원 프레임워크에 대한 안전성을 검증한다.

첫째, 사용자 A의 개인키를 백업하기 위해서 TGL을 구성하고 TGL 내의 사용자에게만 개인키 복원 권한을 부여한다. 악의적 사용자가 $B^{''}_{i,j}$을 확보하여도 $TU_{i,j}$의 개인키가 없으므로 사용자 A의 개인키를 복원할 수 없어 안전하다.

둘째, 악의적 사용자가 $B^{'}_{i,j}$을 확보하여도 복호화를 하기 위해서는 k가 필요하다. 하지만 악의적 사용자가 TGL에 등록되어 있지 않으면 k를 받지 못하므로 사용자의 개인키를 복원할 수 없다.

셋째, 제3의 기관인 백업 관리자는 자신의 개인키를 이용하여 $k^{''}_i$를 복호화하는 역할을 수행한다. 즉, 백업 관리자가 사용자 A의 개인키 백업 및 복원에 관여하지만 사용자 A의 개인키에 대한 어떠한 정보도 저장하고 있지 않다. 이는 백업 관리자의 역할을 최소화하면서 기존에 있던 키 위탁의 한계점을 개선한 것으로 볼 수 있다.

넷째, 각 신뢰 그룹의 사용자들은 백업 관리자가 발행한 $Vcert$를 사용하여, TGL 내에 등록된 사용자들을 서로 인증한다. $Vcert$에 등록된 사용자가 아닐 경우 $Vcert$를 발급받지 못하며, $B^{'}_{i,j}$ 복호화를 할 수 없다. 따라서 사용자 A의 개인키 복원은 TGL에 등록된 사용자들의 협력으로만 가능하므로 개인키 복원 과정이 안전하다.

마지막으로, 사용자 A의 개인키를 암호화하는 대칭키를 이중으로 암호화하여 저장하므로 복원 과정에서 백업 관리자를 포함한 누구에게도 대칭키에 대한 정보가 노출되지 않는다.

Ⅴ. 결 론

블록체인 시스템이 다양한 분야에 적용되면서 사용자의 개인키 관리 중요성이 대두되고 있어 이에 관한 연구와 대책이 요구된다. 본 논문에서는 블록체인 시스템을 이용한 사용자의 개인키 백업 및 복원 프레임워크를 제안하였다. 제안하는 프레임워크를 이용해 사용자의 개인키를 분실 및 상속의 경우를 대비해 사용자의 개인키를 안전하게 저장하고 복원할 수 있다. 이 방식은 백업 및 복원 과정에서 사용자의 개인키에 대한 어떠한 정보도 유출하지 않고 백업할 수 있음을 보였고 제3의 기관의 역할을 최소화할 수 있다는 것을 보였다.

블록체인 기술을 활용한 서비스는 다양한 분야에 적용될 것이다. 그러므로 적용할 서비스를 고려하여 개인키의 백업 및 복원 프로토콜 메시지 구조를 설계하거나 블록 분할 방안에 관한 향후 연구에서 발전시킬 수 있을 것이다.

참고문헌

[1] Kyung Wan Kuk, “Application Case by Blockchain Technology and Industry”, Institute of Information & COmmunications Technology Planning & Evaluation, Technical Report, 2019.

[2] ENISA, “ENISA Opinion Paper on Cryptocurrencies in the EU Version1 September 2017”, European Union Agency for Network and Information Security, 2017.

[3] Paul Symons, Ilse Peeters, Jorma Yli-Jaakkola, Andre-Marc Delhez, Angus Scott, James Mead, Ben Kingsley, “Blockchain settlement November 2016”, Slaughter and May and Euroclear, 2016.

[4] CCN. $190 Million in Crypto Gone Forever, How Canada’s Biggest Bitcoin Exchange Lost it All [Internet]. Available: https://www.ccn.com/190m-gone-how-canada-biggest-bitcoin-exchange-lost-it/.

[5] Wikipedia. "Key escrow". Available: https://en.wikipedia.org/wiki/Key_escrow.

[6] Shamir, Adi. "How to share a secret." *Communications of the ACM,* Vol22, No 11, pp 612-613, Nov 1979.

[7] Kaga, Y., Fujio, M., Naganuma, K., Takahashi, K., Murakami, T., Ohki, T., & Nishigaki, M., “A secure and practical signature scheme for blockchain based on biometrics. In International Conference on Information Security Practice and Experience.” International Conference on Information Security Practice and Experience, Information Security Practice and Experience, pp 877 – 891. Dec 2017.

[8] Mehmet Aydar, Salih Cemil Cetin, Serkan Ayvaz, Betul Aygun, "Private Key Encryption and Recovery in Blockchain", arXiv, 1907.04156v1 [cs.CR], Jul 2019.

[9] James Stanley, Steganographic Bitcoin seeds: Hiding cash in plain sight [Internet]. Available:

https://incoherency.co.uk/blog/stories/steganographic-bitcoin-seeds.html.

[10] Hosam, Osama. “Hiding Bitcoins in Steganographic Fractals.” IEEE, 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), pp 512 - 519, Dec 2018.

윤태연(Ta-Yeon Yoon)

2018년 : 동국대학교 컴퓨터공학과 (학사)
2018년 ~ 현 재 : 고려대학교 정보보호대학원 정보보호학과 석사과정

※관심분야 : 정보 보호, 시스템 보안, 블록체인, 키 관리

문종섭(Jong-Sub Moon)

1981년 : 서울대학교 계산통계학과 학사
1983년 : 서울대학교 계산통계학과 석사
1991년 : Illinois Institute of Technology 전산학과 박사
1993년 3월 ~ 현 재: 고려대학교 전자 및 정보공학부 교수
2001년 2월 ~ 현 재: 고려대학교 정보보호대학원 겸임교수

※관심분야 : 정보 보호, 운영체제, 침입탐지