JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 28 No. 12, pp. 115-127, December 2023
https://doi.org/10.9708/jksci.2023.28.12.115

A Proposed Authentication Scenario based on SBT implementation for Decentralized User Authentication

Sung-Woo Cho*, Jung-Won Seo*, Soo-Yong Park*

*M.S Student, Dept. of Computer Science, Sogang University, Seoul, Korea
*Ph.D. Candidate, Dept. of Computer Science, Sogang University, Seoul, Korea
*Professor, Dept. of Computer Science, Sogang University, Seoul, Korea

[Abstract]

In this paper, we propose a SBT user authentication method for decentralized environment like blockchain. Due to tranparancy of blockchain, it is difficult to secure the privacy of person information, so it is necessary to use a new authentication method. In order to solve this problem, research using SBT for user authentication is being conducted, however most studies have implemented SBT in form of removing function which is related to trasmission from NFT without applying the technical features requried for SBT. The proposed scheme of this paper is implemented SBT which secured safetey and usabillity with using locking token function of ERC-5192. Also, based of implemented SBT scheme propose a user authentication process. To verify our approach, we set a hypothetical user authentication scenario based on the proposed user authentication process and deployed a smart contract that satisfies the 19 function call scenarios that occur in that scenario.

▸ **Key words**: Blockchain, User Authentication, SBT, Web3.0, Smart Contract

[요 약]

본 논문에서는 블록체인과 같은 탈중앙 환경에서의 사용자 인증을 위한 SBT 기반의 사용자 인증 방안을 제안한다. 블록체인의 투명성으로 인해 기존에 사용하던 인증 수단을 사용하면, 개인정보 프라이버시 확보가 어려우므로 새로운 인증 수단이 필요하다. 이러한 문제를 해결하기 위해서 사용자 인증 수단으로 SBT를 활용한 연구가 진행되고 있으나, 아직 정립되지 않고 NFT의 전송과 관련된 함수를 제거한 형태로 SBT를 구현한 연구들이 대부분인 상황이다. 본 논문의 접근 방안은 ERC-5192 표준의 토큰 잠금 기능을 활용해 사용성과 안정성을 확보한 SBT 토큰을 구현하였다. 또한 구현한 SBT 토큰을 기반으로 한 인증 프로세스를 제안했다. 이러한 접근 방안을 검증하기 위해 제안한 인증 프로세스를 기반으로 한 가상의 사용자 인증 시나리오를 설정하고 해당 시나리오에서 발생하는 19개의 함수 호출 시나리오를 만족하는 스마트 컨트랙트를 작성했다.

▸ **주제어**: 블록체인, 사용자 인증, SBT, Web 3.0, 스마트 컨트랙트

• First Author: Sung-Woo Cho, Corresponding Author: Soo-Yong Park
*Sung-Woo Cho (csw2479@gmail.com), Dept. of Computer Science, Sogang University
*Jung-Won Seo (jungwonrs@gmail.com), Dept. of Computer Science, Sogang University
*Soo-Yong Park (sypark@sogang.ac.kr), Dept. of Computer Science, Sogang University
• Received: 2023. 11. 02, Revised: 2023. 11. 27, Accepted: 2023. 12. 04.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

단순 거래 기능만을 제안했던 비트코인[1]을 시작으로 스마트 컨트랙트(Smart Contract), NFT(Non-Fungible Token) 등의 서비스를[2] 지원하는 이더리움이(Ethereum)[3] 등장하면서 블록체인을 활용하여 다양한 기술 분야에서 연구가 진행되고 있다. 단순 데이터 분산 저장, 무결성 검증만을 하는 것이 아닌, 탈중앙화 금융 서비스 De-Fi(Decentralized Finance), 탈중앙화 애플리케이션 Dapp(Decentralized Application)[4] 등의 서비스에도 블록체인을 사용하고 있다. 특히 블록체인의 핵심 특징인 탈중앙화 되어있는 네트워크 구조와 해시 함수를 통해 기존 중앙화 구조에서 발생하는 단일 지점 장애(Single Point Failure), DDoS(Distributed Denial of Service)와 같은 공격을 방지할 수 있어 다양한 보안 관련 기술에 적용하고자 하는 연구 또한 진행되고 있다[5].

기존의 사용자 인증을 위해 사용하는 인증 수단은 사용자의 개인정보를 기반으로 인증 수단을 생성하고 이를 중앙화 된 서버 혹은 검증 기관에 저장하고 관리하며 사용자의 개인 인증을 수행한다. 이러한 중앙화 된 사용자 인증 방식은 사용자들에게 익숙하지만, 웹상에서의 서비스의 종류가 다양해지고 있는 지금 각 기관에 사용되어야 할 인증 수단이 중앙화된 형태로 각각 존재하고 다른 절차를 통해 인증이 필요한 부분에서 어려움을 겪는다[6]. 이러한 기본적인 형태의 사용자 인증 방식을 개선하기 위해 SSO (Single Sign on), 사용자 중심 인증 등 다양한 인증 방안이 사용되고 있지만, 여전히 인증 과정에서의 중앙화 및 중개인이 개입하게 되며, 사용자 개인 정보 유출 등의 문제가 여전히 발생한다. 블록체인과 웹 환경의 발전으로 탈중앙화, 탈 독점화 된 형태의 인터넷 Web3.0 생태계가 구축되고 있는 지금, 기존에 존재하던 인증 수단들은 전반적인 인증 프로세스가 중앙화되어있어 Web3.0 서비스에 적합하지 않아 새로운 인증 체계에 대한 연구가 진행되고 있다[7].

기존 사용자 인증 수단을 관리하기 위한 방안으로, 사용자의 인증 수단을 탈중앙화된 블록체인에 저장하여 관리하고 인증하는 방안들이 제안되고 있다[8, 9]. 그러나 탈중앙화된 환경에서 데이터를 투명하게 공유하는 블록체인 기술은 기존의 사용자 인증 수단을 저장하고 관리하게 되면, 투명성으로 인해 인증 절차에 필요한 인증 수단 및 개인정보를 블록체인에서 모든 참여 노드가 저장하게 되므로 적합하지 않아, 블록체인의 탈중앙 형태로 관리하기 위한 새로운 사용자 인증 수단을 사용하는 방안이 필요한 상황이다.

블록체인에서 관리하기에 적합한 사용자 인증 수단으로써 대표적인 블록체인 기반 기술인 NFT를 활용하는 연구들이 존재한다[10, 11]. 대체 불가능한 특징을 가지는 NFT는 각 토큰의 식별자가 존재하며, 해당 토큰은 대체되지 않고 유일하게 존재함으로써, 사용자 개인을 식별할 수 있는 기능을 할 수 있다. NFT는 블록체인의 해시 함수 활용과 스마트 컨트랙트를 통한 발급 절차를 통해 무결성 및 탈중앙성을 확보할 수 있는 인증 수단이지만, 기본적인 NFT는 디지털 자산 혹은 데이터의 소유권을 보장하고 거래하기 위한 기술로써, 거래기능을 가지고 있다. 이는, 사용자 인증 수단으로 사용하기에는 타인에게 전송을 가능하게 하고 이를 거래에 사용할 수 있는 부분에서 기존의 사용자 인증 수단을 대체하기에 적합하지 않다.

NFT가 가진 기술적 특징을 보완하기 위해 이더리움 재단의 설립자 비탈릭 부테린은 'Decentralized Society: Finding Web3`s Soul'[12] 을 통해 Decentralized Society(DeSoc)의 개념과 탈중앙화된 Web3.0 환경에서의 증명서(Credential)를 확보하기 위해 전송 불가능(Non-Transfer)한 Soul-Bound Token(SBT)을 제안했다. Soulbound는 특정 사용자에게 귀속되어 전송 불가능한 기능을 나타내며, SBT를 통해 Web3.0 환경에서의 사용자를 인증 할 수 있는 방안을 제시해 주었고 이와 관련된 여러 가지 연구가 진행되고 있다. Web3.0 생태계의 핵심 기술 중 하나인 메타버스(Metaverse) 상에서 사용자 아바타 인증을 위한 SBT, DID 기반의 블록체인 지갑을 제안하는 연구가 있다[13]. 또한 NFT의 전송기능을 제거해서 구현한 SBT를 구현한 연구[14, 15]에서는 코로나 백신 접종 증명서, 의료 데이터 등을 SBT를 통해 관리하고 인증하는 프로세스를 구현했다.

더 나아가 Tumati, Tarun Vihar이 진행한 연구[16]에서는 기존 ERC-721로 발행된 NFT를 ERC Extension을 사용하여 기존 NFT 표준을 준수하는 것을 기반으로 SBT를 구현해 교육기관의 학생증을 SBT로 발행하는 접근 방안을 제안했다.

그러나 현재 진행되고 있는 SBT를 인증 수단으로 활용하는 연구들은 기존 NFT에 전송(Transfer) 관련 함수를 제외하는 것으로 SBT를 구현한 연구가 대부분이다[13, 14, 15]. 추가적인 기능 없이 단순히 전송 함수들을 제외한 스마트 컨트랙트를 통해 발행된 SBT는 Transfer 관련 함수 및 이벤트를 사용할 수 없어 기능적인 측면에서 결함을 가지며, 블록체인에서 SBT를 따로 인식할 수 없고, 특정 사용자에게 귀속되어 전송 불가능한 Soulbound의 기능을 온전히 구현하지 못한다. 단순 전송이 되지 않는

NFT를 SBT로 사용하는 것은 무리가 있어 여러 가지 Web3.0 서비스에 적용하기 위해 Soulbound 기능이 구현된 SBT를 사용자 인증 수단으로 활용해야 한다.

본 논문에서는 Soulbound를 구현하기 위해 Ethereum Request for Comment(ERC) Extensions를 활용한 기존의 전송기능이 삭제되지 않은 SBT를 제안하고, 이렇게 구현된 SBT를 통해 사용자 인증을 위한 시나리오를 제안하고자 한다. 제안하는 접근 방안은 ERC-5192의 토큰 잠금 기능을 추가함으로써 Soulbound를 구현할 수 있고 이를 블록체인 네트워크를 통해 확인할 수 있다.

본 연구에서 제안하는 사용자 인증을 위한 SBT 제안은 다음과 같은 기여를 가진다.

1) ERC-721 표준을 준수하는 토큰을 기반으로 SBT를 구현
2) 이벤트를 통한 locked state 구현 및 블록체인에서 SBT 인식
3) ERC Extensions를 통한 Soulbound 구현
4) 본 논문의 SBT를 기반으로 사용자 인증 시나리오 제안

II. Preliminaries

본 장에서는 해당 논문의 접근 방안과 관련된 배경지식 및 SBT를 발행하여 인증 수단으로 사용한 이전 연구를 소개한다.

2.1. Background

2.1.1. Blockchain

블록체인은 여러 가지 암호학 기술 및 네트워크 기술을 통해서 탈중앙화된 구조를 가진 네트워크 기술이다. 블록체인의 핵심적인 요소기술은 블록이라는 데이터를 암호화하기 위한 해시 함수, 네트워크를 탈중앙화된 형태로 구성하기 위한 P2P(Peer-to-Peer) 네트워크, 합의 알고리즘이 있다. 블록체인의 특징을 활용하여 무결성, 투명성과 같은 보안 요소를 확보할 수 있다. 블록체인 네트워크를 통해서 처리하는 데이터는 공개키 기반의 서명을 통해서 저장되며, 블록이라는 자료구조에 저장된다. 블록체인에서 처리하는 모든 데이터는 분산되어 저장되고 이는 무단 위 변조를 불가능하게 하여, 데이터의 무결성과 신뢰성을 보장받을 수 있다. 블록체인에 저장되고 기록되는 모든 데이터는 블록체인 네트워크를 구성하는 모든 네트워크 노드 간의 합의 과정을 거친 데이터를 기록하므로 악의적인 노드의 공격을 방지할 수 있다.

블록체인의 기술을 통해 확보할 수 있는 보안 부분에서의 강점을 이용하여 거래장부를 암호화해 기록하고 장부를 분산 저장하는 금융 시스템[1, 2], 공급망에 적용하기 위한 분산 네트워크[17], 블록체인 네트워크 기반 사물인터넷(Internet of Things, IoT)[18], 디지털 콘텐츠 저작권 확보[19] 등 다양한 분야에서 블록체인을 사용하고 있다.

2.1.2. Ethereum Request for Comment(ERC)

Ethereum Request for Comment(ERC)는 이더리움 블록체인 네트워크의 개선안을 제안하는 Ethereum Improvement Proposal(EIP)[20]에서 토큰, Uniform Resource Identifier(URI) 체계, 지갑, 스마트 컨트랙트 라이브러리 등과 같은 애플리케이션 수준에서의 표준 및 규약을 뜻한다. 이더리움의 스마트 컨트랙트는 솔리디티(Solidity) 언어를 통해 여러 가지 기능을 구현할 수 있는데, 안정적인 기능 구현과 규격을 위해서 ERC 표준의 Application Programming Interface(API)를 따라서 스마트 컨트랙트를 작성하는 것을 권장한다.

대표적인 ERC 표준으로는 이더리움 네트워크 및 이를 기반으로 서비스되는 Decentralized application(Dapp)에서 사용할 수 있는 토큰의 표준안인 ERC-20이 있다. 모든 ERC가 등록되었다고 해서 표준처럼 사용할 수 있는 것이 아닌, 이더리움 사용자, 개발자 등에 의해 사용성, 보안성 등을 심사받은 후 'Final' 단계에 있는 ERC를 표준처럼 사용하고 있다. 'Final' 화 된 대표적인 ERC 표준은 이더리움 토큰 표준 (ERC-20), 스마트 컨트랙트 인터페이스 (ERC-165)[21], 이더리움 대체 불가 토큰 표준 (ERC-721)[22] 등이 있다. 이렇게 표준화된 ERC 표준안은 이더리움 블록체인 기반의 서비스에 적용해 사용하기도 하지만, 해당 표준이 가진 기술적 특징을 더 발전시키거나, 새로운 기술을 적용해 다양한 블록체인 서비스를 만들 수 있도록 기존 표준에 기능을 확장(Extension)하는 표준안 또한 존재하며, 대표적으로 토큰 상호작용 표준 (ERC-777)[23], 다중 토큰 표준 (ERC-1155)[24] 가 있다.

2.1.3. Digital Token (NFT, SBT)

NFT는 데이터 혹은 디지털 자산에 대한 소유권을 보장하고 가치를 부여할 수 있는 토큰을 의미하며, 이는 앞서 설명한 ERC-721, ERC-1155 표준 API를 사용해 발행할 수 있다. 각각의 NFT는 고유한 토큰 아이디(Token ID)를 가지게 되는데, 이러한 특징을 통해 대체 불가능한 토큰이라는 이름으로 불린다.

Table 1. Research Comparison and Our Approach

Reserch	Research Title	Summary and Threshold	SBT implement	ERC application	Final ERC application
[13]	Digital authentication system in metaverse using DID and SBT	The proposed wallet is authenticated by users through DID and SBT. However, there is no way to implement SBT.	X	X	X
[14]	Soulbound Token for Covid-19 Vaccination Certification	propose to implement the vaccination history of users through SBT, Although the introduction to the SBT standard is made, the actual SBT implementation issues SBT to the ERC721 standard without transfer-related features.	O	X	X
[15]	Patient-centric soulbound NFT framework for electronic health record	Propose SBT as a means to store and identify patient medical records. However, the concept of SBT is used in approaches to simply implement non-transmitting capabilities	O	X	X
[16]	SBTCERT: A SOULBOUND TOKEN CERTIFICATE VERIFICATION SYSTEM	Implement SBT through the ABT of ERC-4973, one of NFT's SBT Extensions, However, the ERC-4973 Interface is not yet finalized, so security and usability reviews have not been properly reviewed and has not been updated for a long time	O	O	X
Our Approach	**ERC-5192 based Locked SBT**		**O**	**O**	**O**

이러한 NFT를 기반으로 이더리움 재단의 설립자 비탈릭 부테린으로 부터 Web3.0 시대의 DeSoc을 구현하기 위한 수단으로 SBT(Soul-Bound Token)라는 개념이 제안되었다[12]. 대체 불가능한 특징은 NFT와 같지만, Web3.0 환경에서의 인증 수단, 유저의 ID로 사용하기 위해 타인에게 전송 불가능한 가장 큰 차별점이 존재한다. SBT의 개념을 바탕으로 사용자의 신원정보, 경험 기록과 같은 분야에 적용하고자 하는 방안을 제안했다.

NFT와는 다르게 공개된 지 얼마 되지 않은 기술인 SBT는 NFT의 ERC-721처럼 해당 기술을 대표하는 기술 표준이 존재하지 않고, 결국 현재로서는 단순히 전송 불가능한 특징을 구현하기 위해 NFT의 기능을 제한하는 형태로 사용되고 있다. 하지만, 개념상으로만 존재하고 기능이 제한된 SBT를 실제 서비스에 적용하기 위해서 NFT가 가진 대체 불가능한 특징을 기반으로 SBT를 구현하기 위한 ERC Extension이 공개되고 있다. 즉, NFT가 가진 특징은 제안된 SBT가 가져야 할 최소조건이다. 현재, Final 단계인 SBT 관련 ERC 표준은 3개[25, 26, 27] 정도가 있으며, Final 단계가 아니더라도 다양한 SBT 관련 ERC 표준이 현재 검토받고 있다[28, 29]. 해당 연구에서 살펴본 SBT ERC 표준은 아래 [표 2]에 나타냈다.

2.1.4. Web3.0

웹 3.0(Web3.0)은 인터넷의 새로운 형태로써, 기존 읽기 기능만 가능했던 웹 1.0 (Web1.0)에서 현재 대부분의 웹 서비스가 이루어지는 환경인 웹 2.0(Web2.0) 인터넷 기술로 발전했다. 이제 인터넷으로부터 데이터를 읽고 쓸 수 있게 되었고 이는 소셜 관계망 서비스(Social Network Service: SNS), 인터넷 게임 등 다양한 활동을 할 수 있게 되었다. 하지만 인터넷을 사용하는 사용자가 만들어 낸 데이터의 소유권이 확보되는 것이 아닌 인터넷의 데이터 운영 주체는 개인이 가지는 것이 아닌, 플랫폼이 독점하는 형태로 운영되고 있다. 즉, 사용자의 데이터를 통해 플랫폼은 끊임없이 성장하고 수익을 창출할 수 있지만 사용자는 인터넷 사용을 위해 플랫폼 이용료를 지불하고 데이터의 소유권을 확보하기 어려운 환경이 되었다.

Web3.0은 데이터의 운영을 플랫폼이 독점하는 것이 아닌, 개인이 데이터의 소유권을 확보할 수 있고 블록체인 기술을 통해서 데이터를 탈 독점적으로 관리해 보안을 강화할 수 있는 인터넷 생태계이다. 이러한 탈중앙 인터넷 생태계에 핵심인 데이터 주권을 확보하기 위해 자기 주권 신원 증명(Self-sovereign identifiers)과 같은 개념이 Worldwide Web Consortium(W3C)과 같은 팀으로부터 등장하여, Decentralized Identifier(DID)[30], 디지털 토큰과 같은 기술을 통해 웹상의 데이터의 주권을 확보한다. Web3.0은 아직은 태동기에 머물고 있어 모든 인터넷 생태계에 적용이 되지 않았지만, 블록체인, 메타버스 등이 적용된 새로운 형태의 Web3.0 서비스가 생겨나고 있다. 현재 대표적인 Web3.0 기반의 서비스들은 브레이브(Brave) 브라우저, 디센트럴랜드(Decentraland) 메타버스 플랫폼

등이 있다. Web2.0, Web3.0에 적용되는 인증의 특징 및 해당하는 인증 수단은 (Fig. 1)에 표현했다.

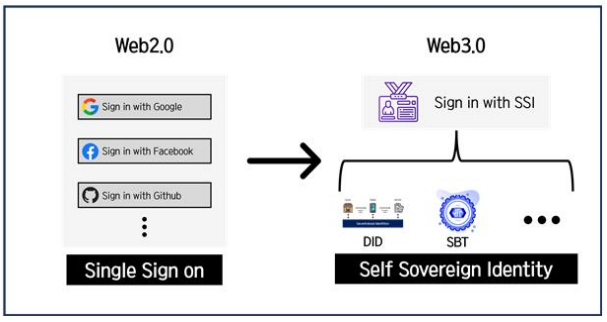

Fig. 1. Authentication type for each web version

2.2. Related Works

관련 연구는 SBT 토큰을 활용한 사용자 인증 분야에 적용하고자 하는 연구를 다룬다. 본 연구와 관련된 관련 연구는 <Table. 1>에서 다룬다.

K. Kim, and J. Ryu. 가 제안한 DID와 SBT를 활용한 메타버스 내에서의 디지털 인증 체계 연구[13]에서는 메타버스 상에서의 사용자를 나타내는 아바타의 신원 인증과 사용자 간의 연결을 위해 DID(Decentralized Identity)와 SBT 토큰 기술을 가질 수 있는 메타버스를 위한 블록체인 지갑을 제안했다. 메타버스에서 사용자의 데이터의 소유권 혹은 신원을 증명하기 위한 수단으로 DID와 SBT 기능을 가진 블록체인 지갑을 제안하고 해당 지갑을 통해 사용자를 인증하기 위한 신원 인증 체계를 제안한다. 그러나 해당 연구에서 다루는 SBT를 어떻게 구현하는지에 대한 정보를 확인할 수 없다.

Lunesu, Maria llaria. et al. 이 진행한 Soulbound Token for Covid-19 Vaccination Certification 연구[14]에서는 COVID-19의 백신 접종 이력을 증명할 수 있는 증명서를 SBT 토큰을 통해 발행하는 기법을 제안한다. SBT를 통해 발행된 접종 이력 데이터를 통해 사용자 개인정보를 노출하지 않고 접종 이력을 인증할 수 있는 특징이 있다. 증명서로 사용할 SBT 표준안 중 ERC-5114(Soulbound Badge)를 통해서 SBT 증명서를 발행할 수 있는 부분을 연구에서 다룬다. 하지만, 해당 연구에서는 ERC-5114 표준을 통해 증명서 SBT 토큰을 발행하는 것이 아닌, ERC-721 표준에서 전송기능을 제외하는 방법으로 증명서 SBT 토큰을 발행한다. 이는, SBT의 전송 불가능(Non-Transfer)한 특징만을 구현했다는 단점이 있다.

Tanwar, Namrta, and Jawahar Thakur가 진행한 연구[15]에서는 민감하게 다뤄야 할 환자들의 데이터를 보호하기 위해 SBT를 활용하는 프레임워크를 제안한다. 해당 연구에서는 기존 NFT는 거래를 가능하게 하는 전송할 수 있는(Transferable) 특징으로 인해 SBT를 통해 환자의 데이터를 토큰화하고 이를 중앙기관에서 관리하는 방안을 제시한다. 환자데이터를 관리하기 Soulbound 기능을 할 수 있는 NFT 스마트 컨트랙트를 기반의 프레임워크를 연구에서 제안하지만, SBT를 구현하는 데 있어 분석이 부족하다. 특히 SBT를 구현하기 위해서 어떠한 표준을 사용했는지, 기존의 NFT와 어떤 차이점을 통해서 SBT를 구현했는지에 대한 설명이 부족하며, 사용한 스마트 컨트랙트에는 전송 관련 함수를 모두 제외하는 방법을 통해 단순 전송 불가능성을 구현했는데, 이는 전송 함수를 사용하지 않더라도 전송 관련 이벤트(Event)를 사용하는 다른 기능들도 사용이 제한되는 한계점이 있다.

Tumati, Tarun Vihar이 진행한 SBTCERT: A SOULBOUND TOKEN CERTIFICATE VERIFICATION SYSTEM 연구[16]에서는 교육기관에 소속된 학생들의 학력 및 기타 데이터를 포함한 증명서를 SBT 토큰을 통해 발급하고 이를 학교 내의 기관에 사용하기 위한 접근 방안을 제안한다. 학생들의 학번, 이름 등의 데이터가 저장된 데이터베이스에서 일괄 발행(Batch Minting)을 통해 일괄적으로 학생들의 증명서를 토큰 형태로 발행하고, 인증이 완료된 학생들은 해당 토큰에 접근할 수 있는 권한을 얻는 방식으로 SBT 학생 증명서를 제안했다. 단순 전송 관련 기능을 제외해 SBT로 활용하는 것이 아닌, ERC를 활용해 NFT에 Soulbound 기능을 구현했다는 것이 특징이다. 해당 논문에서는 SBT를 구현하기 위해 계정 기반 토큰(Account Based Token: ABT)의 규정인 ERC-4973 [29]를 Extension으로 사용한다. ERC-4973 표준에서는 게임 워크래프트(Warcraft)에 나오는 게임 아이템을 캐릭터에게 잠그는 개념에서 착안해 계정에 토큰을 귀속시키기 위한 착용(equip), 착용 해제(unequip) 두 가지 상태를 토큰에 부여하는 방안을 제안하였고 이를 ABT라고 명시했다. 하지만, ERC-4973은 앞서 다룬 ERC 표준과는 다르게, 1년 이상 Review가 진행 중인 표준이며, 보안성, 사용성 등의 합의가 되지 않아 안정적인 SBT를 발행하는 데에는 한계가 있다.

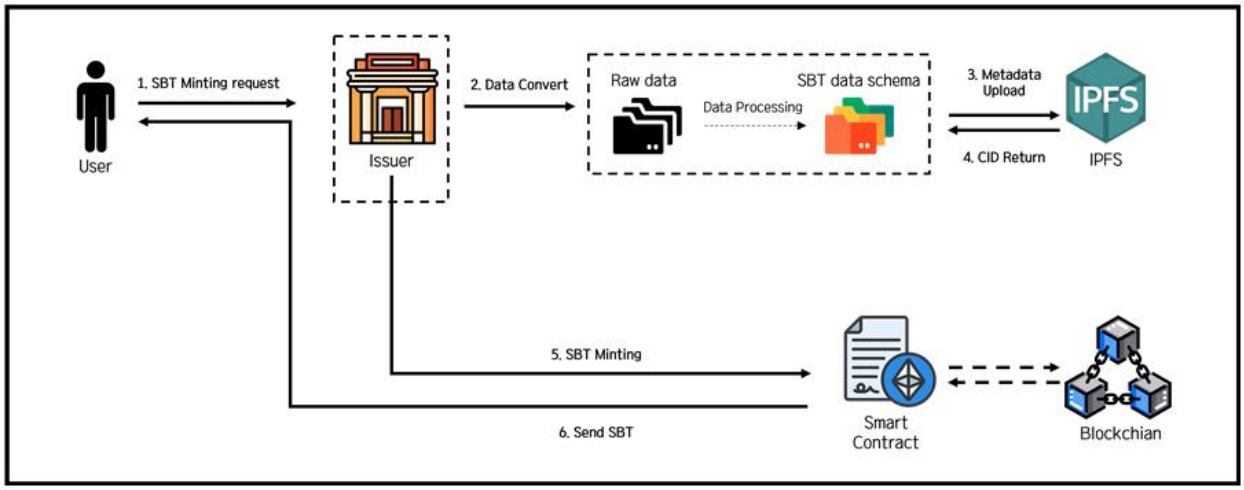

Fig. 2. Overview of SBT Based User Authentication Domain

III. The Proposed Scheme

본 장에서는 기존 연구들이 인증 수단으로 제안했던 전송 관련 함수 및 이벤트를 제거해 Soulbound의 개념이 적용되지 않거나, 표준을 따르지 않은 형태로 구현된 SBT 관련 연구 들의 한계점을 개선한 접근 방안을 제안한다. 이를 개선하기 위해 본 연구의 접근 방안에서는 Final 화된 ERC 표준의 기능을 활용하여 SBT를 구현하는 방법을 제안한다.

본 연구에서 제안하는 SBT는 NFT Extension인 ERC-5192 표준에서 제안한 기능들을 기반으로 구현이 된다. 연구에서 제안하는 기법을 통해서 발행한 SBT는 NFT의 기능을 그대로 사용할 수 있어 대체 불가능성을 확보하고 ERC-5192 표준을 통해 Soulbound 기능을 가져 전송 불가하며, 이러한 SBT 토큰을 블록체인에서 이벤트를 통해 확인할 수 있어 SBT를 식별할 수 있다.

SBT 사용자 인증의 경우 블록체인과 디지털 토큰을 활용한 인증 수단으로써 필요한 기능들을 구현하고, 탈중앙 데이터베이스(InterPlanetary File System IPFS)와 SBT를 활용하여 각 사용자는 자신의 데이터에 대한 주권을 확보할 수 있게 되면서, 탈중앙 환경에 적합한 인증 수단으로 사용자 인증을 진행할 수 있다. 또한 ERC-5192 표준의 기능들을 기반으로 인증 수단을 관리하는 데에 있어, 무단 소각, 중복된 SBT 발행 등과 같은 취약점을 예방할 수 있다. 제안하는 접근 방안을 통한 인증 도메인은(Fig. 2)와 같이 구성된다.

본 연구에서 제안하는 접근 방안의 구성은 (1) ERC-5192를 활용한 사용자 인증 SBT 토큰 설계, (2) 본 논문에서 설계한 SBT를 활용해 사용자 인증 시나리오를 제안 및 SBT의 구현을 검증하기 위한 테스트를 진행 두 가지로 분류할 수 있다. 접근 방안의 각 부분에 대한 설명은 3.1절과 3.2절에서 각각 설명한다.

3.1. User Authentication SBT Design

본 절에서는 이전에 진행된 SBT 토큰을 활용한 사용자 인증 연구에서 구현된 SBT 토큰을 보완하기 위해 ERC 표준의 기능을 활용해 SBT를 구현하는 방법을 설명한다.

ERC-5192 SBT 확장 표준을 활용하여 SBT에 Unlocked 상태를 설정하고 전송 불가능을 구현하는 접근 방안은 다음과 같은 절차를 따르며, 이를 (Fig. 3)를 통해 나타낼 수 있다.

1. SBT에 Locked, Unlocked 상태를 추가하기 위해 mapping을 통해 각 SBT 토큰의 boolean 자료형으로 매핑되는 _locked 상태 변수를 추가한다.
2. 발행한 SBT를 전송 불가능 기능을 추가하기 위해 스마트 컨트랙트를 통해 발행된 토큰의 식별자(tokenId)에 해당하는 _locked 상태 변수를 true 값을 대입해 전송 불가능한 특성을 부여한다.
3. SBT의 _locked 상태 변수를 통해 전송을 방지하기 위한 함수 변경자(modifier) IsTransferAllowed를 정의하고, 스마트 컨트랙트에서 사용된 계정 간 토큰 전송 함수(tranferFrom)관련 함수에 해당 modifier를 사용한다.
4. modifier를 통해 _locked 상태 변수를 확인하고 _locked가 true 값을 가지는 경우(true = locked) 해당 함수의 호출을 막고 트랜잭션을 revert 한다.

Table 2. User Authentication SBT Issuance & Management Processes

Process	Realted Function	Process Description
Deploy Contract	deploy	Deploy SBT smart contracts for user management to the Ethereum network.
Mint SBT	safeMint	Minting SBT to user
Check SBT State	checkSBTState	Check locked/unlocked status of SBT mapping to token URI
Lock SBT (Locked)	lock	Convert the state variable of the SBT to Locked and generates and event
Unlock SBT (Unlocked)	unlock	Convert the state variable of the SBT to Unocked and generates and event
Transfer SBT	safeTransferFrom	The User transfer SBT to other User
Burn SBT	burn	Burn the SBT of the user with Unlocked state variable to remove it from the contract

3.2. SBT User Authentication Scenario

본 절에서는 3.1 절에서 구현한 SBT를 적용해 탈중앙 사용자 인증의 전반적인 프로세스를 스마트 컨트랙트를 통해 구현한다. 해당 연구에서 제안하는 SBT 사용자 인증 시나리오는 1. SBT 발행을 위한 사용자 데이터 전처리 과정, 2. SBT 발행 및 관리 두 가지 부분으로 나눌 수 있다.

3.2.1. SBT Data Preprocessing

데이터 전처리 프로세스의 경우 SBT를 발행하기 이전의 과정들을 다루고 있다. 사용자의 데이터를 탈중앙화된 데이터베이스에 저장하기 위해 IPFS를 활용하여 데이터를 저장하기 위한 과정을 다룬다. IPFS에 데이터를 업로드하기 위해선 IPFS 데이터베이스에 연결하기 위한 정보인 APIKey, APIKeySecret, IPFS API Address를 사전 정의한다.

APIKey는 해당 IPFS의 식별자를 의미하고, APISecret은 APIKey를 사용하기 위해 필요한 비밀키와 같은 기능을 한다. IPFS API Address는 해당 APIKey가 가리키고 있는 IPFS 상의 주소를 의미한다.

이후, 시스템과 IPFS 간의 연결 설정이 완료되면, 시스템은 사용자의 데이터를 입력받아 업로드하기 위해 멤버리스트(memList)에 데이터를 추가한다. IPFS에 업로드하기 위해서 프론트엔드 상에서 데이터를 메타데이터(Metadata)화 한다. 업로드될 사용자 메타데이터는 자바스크립트 객체 표기법(JavaScript Object Notation: JSON) 형태로 변환하는 과정을 거치게 된다. SBT를 발행하는 데에 활용되는 메타데이터의 구성은 해당 논문[12]에서 설명한 바와 같이 속성(Property)은 환경에 적합하게 구성하면 된다. 메타데이터를 JSON 형태로 변환하는 과정을 마치면 해당 데이터를 연결된 IPFS 저장소에 업로드하고 Content Identifier(CID)를 반환받는다. CID는 IPFS에 저장된 데이터에 접근할 수 있는 유일 식별자이며, 이는 보안 해시 알고리즘 (Secure Hash Algorithm 256: SHA-256) 해시 함수를 통해 256 비트의 해시 문자열로 이루어져 있다. 해당 CID는 SBT 발행 프로세스에서 SBT를 식별하기 위한 토큰 통합 자원 식별자(tokenURI)로 사용된다.

3.2.2. SBT Issuance, Management process

SBT 발행 및 관리 프로세스는 다음과 같은 과정을 거친다. 1, 2. 사용자의 데이터를 업로드 하는 과정을 마친 후 반환받은 CID 값을 통해 본 논문에서 제안한 SBT를 스마트 컨트랙트를 통해 사용자에게 발행한다. 해당 프로세스에서 사용자가 보유 중인 SBT는 사용자 인증 수단으로써 작용하며, 3. 발행받는 즉시 SBT의 _locked 상태를 true로 설정해 전송 불가능한 SBT의 특징을 가지게 된다.

4. 인증 수단으로 사용하는 SBT는 일반적인 사용자가 임의로 발행하는 것을 방지하기 위하여 스마트 컨트랙트를 배포한 인증 기관(onlyOwner modifier)만이 SBT 발행 함수를 호출해 SBT 발행할 수 있다. 5. SBT는 tokenURI를 통해 식별할 수 있는 기능을 가지는데, 해시

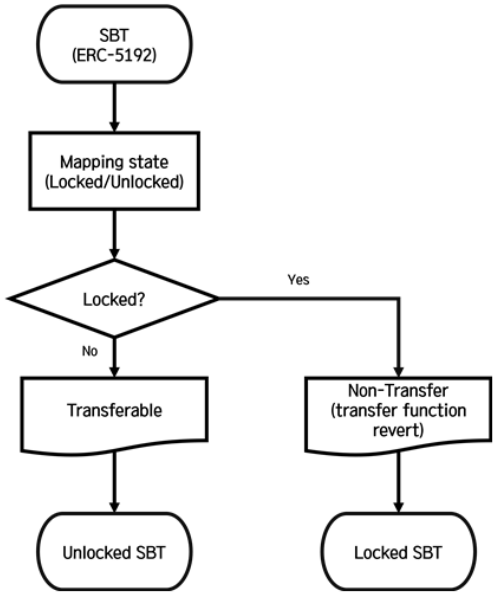

Fig. 3. Overview SBT Using ERC-5192

함수를 통해서 대체 불가능한 특징을 가지고 있어 tokenURI 당 하나의 SBT만을 식별할 수 있고 이는 고유한 값을 가져 중복되지 않으면서, 전송할 수 없다.

인증을 위한 SBT를 발행하고 SBT를 관리하는 프로세스는 (Fig. 4)와 같이 나타낼 수 있다.

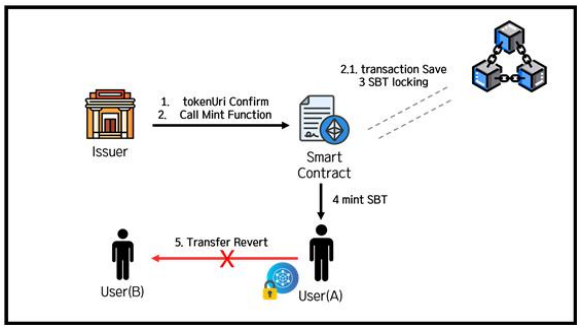

Fig. 4. Overview of SBT Features

SBT는 고유한 사용자 한 명의 Soul을 의미하며, 이를 기반으로 인증 수단으로 사용하기 위해 사용자당 하나의 SBT를 가질 수 있다.

사용기간이 지나거나 더 이상 권한이 없는 사용자의 인증 수단을 소멸시켜야 하는 상황에서는 Burn 함수를 통해 SBT를 소각할 수 있다. tokenURI에 해당하는 SBT의 _locked 상태 변수를 통해 아직 Locked 되어있고 사용중인 인증 SBT는 Burn 함수의 호출을 하게 되면 트랜잭션을 실패시킨다. 또한 소각을 위한 Burn 함수 호출은 SBT를 가지고 있는 사용자만이 호출할 수 있어 다른 사용자가 무단으로 타인의 SBT를 소각시킬 수 없다.

소유하고 있는 SBT 토큰에 대해서는 잠그고 해제할 수 있는 함수가 구현되어 있다. SBT 토큰의 잠금은 SBT 토큰 발행자만 함수를 호출함으로써, Unlocked 상태를 가진 토큰을 Locked 상태로 바꿈으로써, SBT의 기능을 확보할 수 있다. 반대로 SBT를 소유하고 있는 사용자는 토큰 해제 및 소각을 위해 Locked 상태의 토큰을 Unlocked 상태로 바꿔 전송할 수 있는 형태의 토큰을 소유하게 된다. 본 논문에서 사용자 인증 SBT를 활용하기 위한 스마트 컨트랙트를 통해서 인증 수단을 발행 및 관리하는 전반적인 프로세스는 <Table. 2>와 같이 나타낼 수 있다.

IV. Experiments

본 장에서는 논문에서 제안하는 접근 방안의 구현 및 실현 가능성을 알아보기 위해 접근 방안에서 제안하는 SBT 구현(1) 접근 방안에서 구현한 개선된 SBT의 안전성 및 사용성 및 안전성을 평가하기 위한 여러 가지 환경에서의 컨트랙트 함수 호출(2) 두 가지 실험을 진행한다. 실험 환경 및 사용된 장비 및 환경 스펙은 <Table. 3>에 나타냈다.

Table 3. Equipment Specification

Category	Specification
OS	Window 10 Pro
Processor	Intel(R) Core(TM) i7-12700F 2.10 GHz
Memory	32 GB
Graphics	NVIDIA GeForce RTX 3060
Ethereum Network	Goerli Test Network

4.1. Implementation

본 절에서는 특정 기관에서 사용자를 인증하기 위한 수단으로 SBT를 사용하기 위해 본 연구에서 제안한 접근 방안을 활용한 SBT의 구현을 4.1.1절에서 서술하고, 4.1.1절에서 구현된 SBT를 사용자 인증을 위해 발행 및 관리하는 시나리오를 구현하는 방법을 4.1.2절에서 서술한다.

4.1.1. SBT Token Implementation

SBT는 3.1장에서 서술한 ERC-5192를 기반으로 구현되었다. 일반적인 토큰 발행처럼 안전 발행 함수(safeMint)를 호출함으로써, 토큰을 발행한다. safeMint 함수는 컨트랙트 배포자(msg.sender)로부터 호출될 수 있도록 Ownerable.Sol 컨트랙트에 선언된 modifier 컨트랙트 배포자 확인(onlyOwner)을 safeMint 함수에서 사용한다. 이는 제안된 사용자 인증 시나리오에서 SBT 토큰을 발행하는 데에 있어, 일반적인 사용자들이 기관에 사용하고자 하는 SBT를 무단 발행하는 것을 막기 위함이다.

SBT를 받을 주소(to)와 토큰을 식별할 수 있는 식별자 tokenId(tokenUri)를 인자로 입력해 토큰을 발행한다. 발행한 토큰의 형태는 ERC-721의 표준을 기반으로 발행되며, ERC-5192 locked 상태 변수를 통해 to 주소에 잠긴 상태가 된다. 블록체인에서 SBT를 식별하기 위해서 safeMint 함수 내부에서 locked 이벤트를 발생시켜 해당 토큰의 상태를 확인할 수 있도록 하고, 이를 로그(log)로 출력한다.

해당 접근 방안을 통해서 발행된 SBT는 두 가지 특징을 가지고 있다. 1. safeMint 함수를 통해 사용자는 자신에게 귀속된 SBT 소유하게 되며, tokenId는 고유한 값을 가져 SBT는 중복될 수 없으며, 인증 수단으로 사용하기 위해서 사용자는 하나의 SBT만을 소유할 수 있다. 2. ERC-5192 표준을 사용함으로써, 스마트 컨트랙트에는 전송 관련 함

수들이 그대로 구현되어 있으며, Unlocked 된 토큰만 전송 함수를 호출할 수 있다. 사용한 SBT의 표준 및 Spec은 (Fig. 5)로 표현할 수 있다.

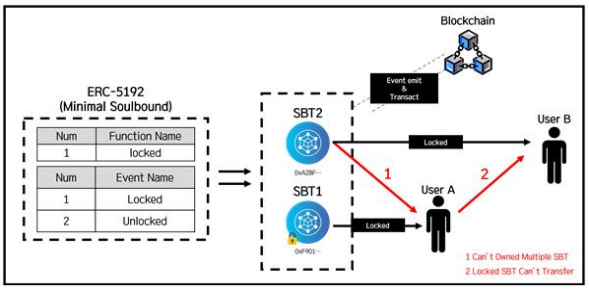

Fig. 5. Overview of Implemented SBT

4.1.2. SBT User Authentication Scenario Imp-lementation

본 절에서는 연구센터에 참여 사실이 확인된 연구원들의 사용자 인증 수단으로 SBT를 제공하고, 관리하는 가상의 시나리오를 설정한 후 해당 시나리오를 구현하는 방법을 서술한다. 시나리오를 구현한 시스템의 아키텍처는 다음 장의 (Fig. 6)과 같다.

구현하고자 하는 시나리오에서 SBT를 활용한 사용자 인증 스마트 컨트랙트는 9개의 주요 함수들로 동작한다. 각 함수는 SBT 토큰 발행(safeMint), 토큰 전송을 통한 사용자 인증 권한 이전(safeTransferFrom, transferFrom), SBT 토큰 잠금(lock), SBT 토큰 잠금 해제(unlock), SBT 토큰 소각(burn), SBT 토큰 잠금 상태 확인(checkSBTState), SBT 소유자 확인(ownerOf), 잠겨있는 SBT 토큰 확인(locked)을 나타낸다. 해당 함수들은 3.2.2절에 표시한 표 n에서 나타낸 프로세스상에서 작동할 수 있도록 스마트 컨트랙트를 설계했다. 구현 방법을 설명하기 이전 사용자 인증 스마트 컨트랙트는 해당 시나리오의 인증 기관인 연구센터로부터 deploy 되어있는 상황을 가정한다.

사용자 인증 SBT 토큰을 발행하기 전 탈중앙 데이터베이스 IPFS를 활용하여 데이터를 저장하기 위해 데이터를 전처리하는 과정은 (Fig. 7)에서 나타낸 다이어그램의 절차를 따른다.

센터의 참여 인력을 인증하기 위한 사용자 인증 SBT 토큰 발행은 safeMint 함수를 호출 함으로써 진행된다. safeMint 함수는 연구원에게 SBT를 전달하기 위한 함수를 호출하고, 해당 토큰을 SBT처럼 사용하기 위해 Locked 이벤트를 발생시킨다.

연구원은 보유하고 있는 SBT의 잠금 여부를 확인하기 위해서 checkSBTState 호출한다. checkSBTState는 사용자가 보유하고 있는 SBT 토큰이 자신에게 잠겨있는지, 잠금 해제되어 있는지 확인할 수 있고, 이는 Locked, Unlocked 이벤트를 발생(emit)함으로써 블록체인 네트워크에서 확인할 수 있다.

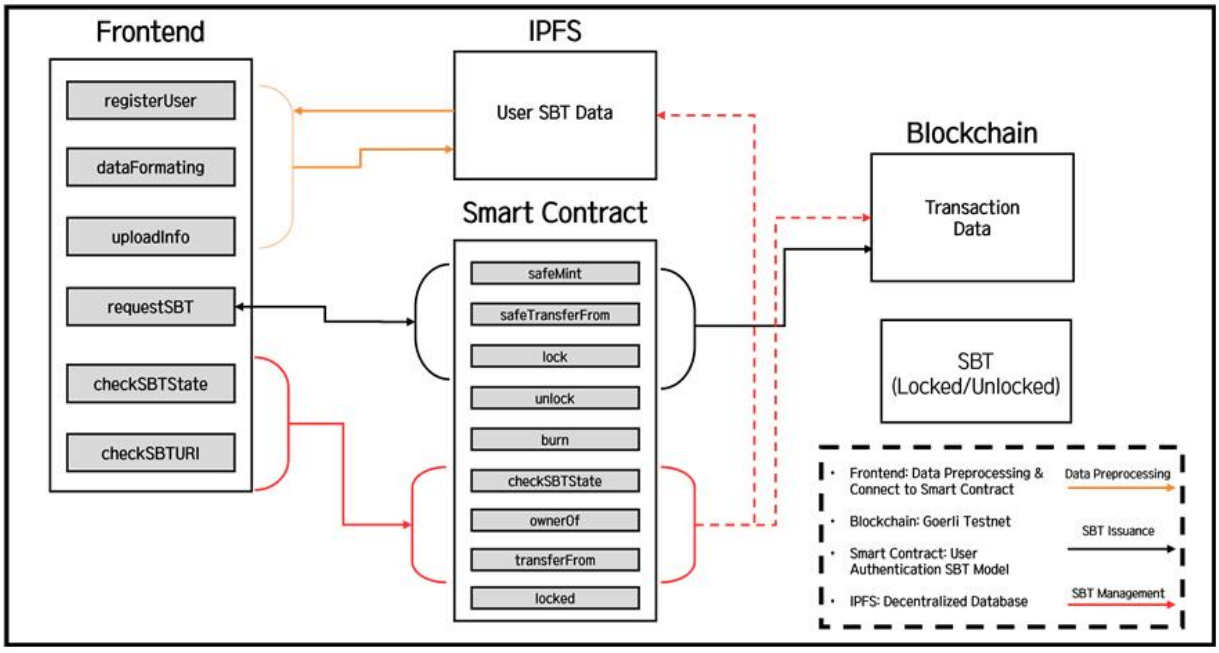

Fig. 6. Entire Process and System Architecture

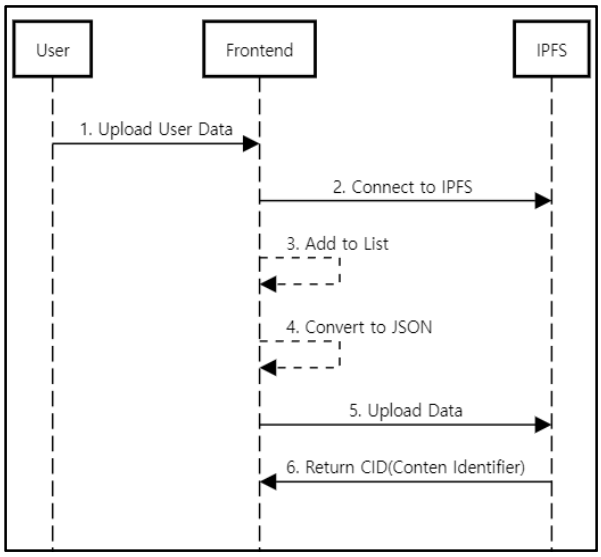

Fig. 7. Process Data Preprocessing

사용자가 소유하고 있는 SBT의 상태 변수 _locked의 값에 따라서 귀속되어있는 SBT 토큰처럼 사용할 수 있고, 일반적인 ERC-721 기반의 대체 불가능 NFT로 사용할 수 있다. 사용자는 자신의 토큰을 unlock 함수를 호출하여 자신의 SBT 토큰을 전송할 수 있는 형태로 전환할 수 있다. 이때 unlocked 이벤트를 발생시켜, 블록체인 네트워크에 해당 SBT 토큰의 상태 변수가 변했음을 기록한다. 반대로 Unlocked 상태 변수를 인증 수단으로 사용하고자 할 때 발행 기관으로부터 lock 함수를 통해 다시 인증용 SBT 토큰으로 사용할 수 있다.

unlocked 상태 변수를 가진 토큰은 더 이상 인증 수단이 필요 없어진 상황이거나, 만료되었을 때 burn 함수를 호출 함으로써 소유한 SBT 토큰을 소각시켜, 사용자 인증 수단으로 사용하는 SBT 토큰의 탈취 및 도용을 방지할 수 있다. 또한, 단순 자신의 SBT 토큰을 타인에게 양도해도 되는 예외 상황이나, 해당 스마트 컨트랙트를 사용하면서, NFT의 특징을 가진 토큰을 발행해 사용해야 하는 경우가 있을 때, unlocked 된 SBT 토큰을 safeTransferFrom, transferFrom 함수를 호출해 토큰을 받는 사람의 주소(from)에 해당하는 사용자에게 토큰을 전송할 수 있다. 이는 Transfer 함수를 제거해 SBT를 구현한 이전 연구들로부터 개선되었다고 할 수 있다.

SBT를 발행하고 잠금을 해제하여 인증 수단을 소멸시키는 과정은 (Fig. 8)로 표현할 수 있다.

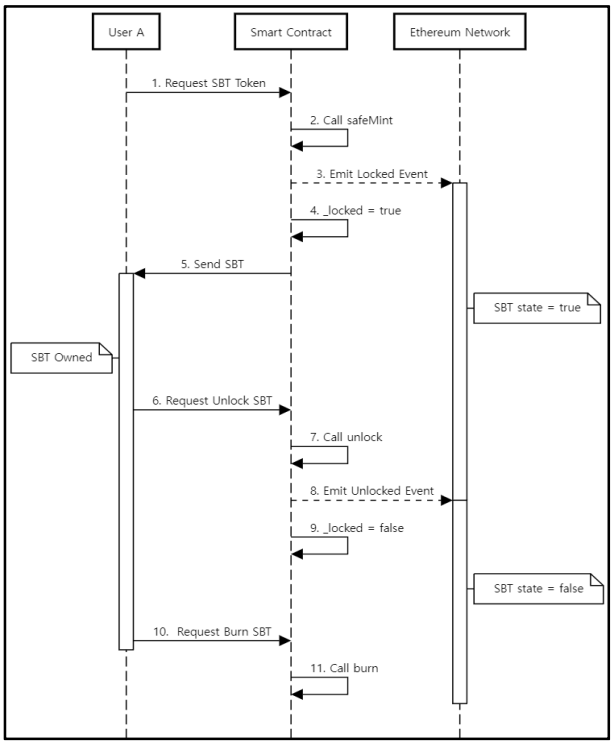

Fig. 8. Process SBT Issuance and Management

4.2. Implement Verification

본 절에서는 3.2절에서 제안한 시나리오를 구현하고 블록체인 네트워크에 배포된 스마트 컨트랙트의 함수 호출 시나리오를 충족시키는지 확인한다.

실험에 사용한 블록체인 네트워크는 이더리움의 테스트 네트워크 중 하나인 Goerli 테스트 네트워크를 사용했으며, Remix IDE를 통해 작성한 스마트 컨트랙트를 작성했다. 실험에는 세 개의 계정이 사용하여 제안한 시나리오에서 사용되는 함수의 정상 호출, 잘못된 인자 및 값을 대입해 호출하는 비정상 호출을 통한 방식으로 진행되며, <Table. 3>에서 제안한 프로세스를 기반으로 시나리오별 함수 호출 결과를 <Table. 4>에 표시했다.

Remix IDE를 통해 작성한 후 배포한 사용자 인증 SBT 스마트 컨트랙트의 프로세스에서 실제 발생할 수 있는 19개의 시나리오를 적용했을 때, 모든 시나리오에서 문제가 발생하지 않고 제대로 동작하는 것을 확인할 수 있다.

에러를 발생시키는 시나리오를 살펴보면, (3) ~ (5) 시나리오는 safeMint 함수를 통한 SBT 발행 시나리오로써, (3) 일반 사용자가 SBT를 발행하는 시나리오로, 해당 프로세스의 스마트 컨트랙트 배포자(Issuer) 이외의 SBT 발행은 불가하며, (4) SBT는 사용자를 나타내는 인증 수단으로써 같은 SBT가 여러 개 존재해서는 안 된다. 물론 SBT의 개념을 제시한 [12] 의 연구에서는 SBT의 형태는 다양하

고, 여러 개 존재해도 된다고 나와 있지만, 본 프로세스에서는 특정 사용자를 인증할 수 있는 수단으로 SBT를 사용하려는 의도로 설계했으며, 무분별한 인증 수단의 발행으로 인한 인증 수단의 효율성과 가스비 낭비 등의 이유로 하나만 존재할 수 있다. (5)는 (4)와 마찬가지로 한 사람이 가질 수 있는 인증 수단(Soul)은 하나뿐이므로, 2개 이상의 SBT를 소유하는 것은 불가능하다.

(7) 시나리오는 SBT의 Locked, Unlocked 상태 변수를 확인하기 위한 함수인 checkSBTState와 관련된 시나리오로써, SBT의 잠금 여부 확인은 소유자 본인만 할 수 있다.

(9), (10) 는 특정 사용자에게 잠겨있는 SBT의 상태 변수를 Unlocked로 바꾸기 위한 함수인 unlock과 관련된 시나리오로써, (9) 자신이 소유하지 않은 SBT의 잠금 여부를 변경할 수 없다. 이는, 타인이 자신의 SBT를 무단으로 잠금 해제하는 것을 방지하기 위함이며, 오직 소유자 본인만이 SBT를 잠금 해제시킬 수 있다. (10) 이미 Unlocked 상태 변수를 가진 SBT를 잠금 해제하지 못하게 함으로써, 불필요한 가스비 낭비를 줄일 수 있다.

(12), (13) 는 ERC-721 표준에 있는 전송(Transfer) 함수를 통해 특정 상황에서 잠금이 해제된 SBT 토큰을 전송하는 시나리오로써, (12) 전송하고자 하는 SBT 토큰이 특정 사용자에게 Locked 된 상태로 존재하면, 전송할 수 없다. 이는 ERC-5192 표준을 통해 전송 불가능성을 구현한 것이다. (13) 타인의 잠금 해제된 토큰을 무단으로 전송할 수 없다. 해당 시나리오는 ERC-721의 기능을 통해 무단 전송을 방지했다.

(15), (16) 는 Unlocked 된 SBT 토큰을 잠그는 lock 함수와 관련된 시나리오로써, (15) 자신이 소유한 Unlocked 된 SBT 토큰을 잠글 수 없다. 특정 사용자한테 Locked 된 SBT 토큰은 해당 사용자의 인증 수단(Soul)이며, SBT는 해당 프로세스에서 일반 사용자가 직접 발행할 수 없듯이, lock 함수를 통한 SBT의 잠금 기능도 일반 사용자에게서 제한한다. (16) (10) 과 마찬가지로 이미 Locked 된 SBT 토큰을 잠그기 위해 함수 호출을 통한 가스비를 사용할 필요가 없으므로 Locked 된 토큰에 대해서 lock 함수를 호출할 수 없다.

(18), (19) 는 기간이 만료되거나, 더 이상 불필요한 SBT 토큰을 소각할 때 호출하는 burn 함수와 관련된 시나리오로써, (18) SBT 토큰의 소각은 토큰을 소유하고 있는 사용자만이 할 수 있으며, 다른 계정의 사용자가 타인의 SBT 토큰을 소각할 수는 없다. 이는 일반적인 인증 수단을 자신이 직접 관리할 수 있는 기능을 구현한 것이다. (19) 상태 변수가 Locked인 SBT 토큰은 소각할 수 없다. 인증 수단의 만료와 같은 상황을 대비하기 위해 Unlocked 상태 변수를 활용하므로, SBT 토큰의 소각은 Unlocked 된 SBT 토큰만 가능하다.

Table 4. Function Call Scenarios

Num	Scenario	function name	call result
(1)	Deploy Smart Contract	deploy	O
(2)	Mint SBT	safeMint	O
(3)	Normal User Mint SBT	safeMint	X
(4)	Mint Duplicated SBT	safeMint	X
(5)	Mint SBT to users who owned SBT	safeMint	X
(6)	Checking SBT State Variable	checkSB Tstate	O
(7)	Checking Not Owned SBT State Variable	checkSB Tstate	X
(8)	Unlocking My SBT	unlock	O
(9)	Unlocking Others SBT	unlock	X
(10)	Unlocking Unlocked SBT	unlock	X
(11)	Transfer of Unlocked SBT	safeTran sferFrom	O
(12)	Transfer of Locked SBT	safeTran sferFrom	X
(13)	Transfer of Unlocked SBT form Others	safeTran sferFrom	X
(14)	Locking Unlocked SBT	lock	O
(15)	Locking Unlocked SBT Myself	lock	X
(16)	Locking Locked SBT	lock	X
(17)	Burn Expired SBT	burn	O
(18)	Burn SBT from Others	brun	X
(19)	Burn Locked SBT	burn	X

V. Conclusions

본 연구에서는 탈중앙 사용자 인증을 위한 SBT를 활용한 사용자 인증 프로세스를 제안하였다. 사용자 인증에 사용되는 SBT를 ERC-5192 표준을 활용하여 구현했으며, 구현한 SBT를 바탕으로 구현한 인증 프로세스에서 발생할 수 있는 시나리오를 통해 안전성을 확보했다. 본 연구의 기여를 살펴보면,

첫 번째, ERC-721 표준을 준수하는 SBT를 구현하였다. 해당 연구에서는 ERC-5192 표준을 통해 Unlocked 된 토큰은 전송을 가능하게 구현했고, 기존 ERC-721에 존재하는 safeTransferFrom을 통해 문제없이 전송되는 것을 확인할 수 있다.

두 번째, 이벤트를 발생시켜 블록체인 네트워크에서 SBT를 인식할 수 있게 하였다. 구현한 SBT 토큰의 잠금 여부를 확인하기 위해서 ERC-5192에서 제안된 Locked, Unlocked 이벤트를 확인할 수 있는 checkSBTState를

함수를 구현해, 해당 SBT의 상태 변수를 확인할 수 있고 이를 이벤트로 발생시킨다.

세 번째, ERC 표준을 통해 SBT를 구현했다. 첫 번째, 두 번째 기여는 ERC-5192 표준을 기반으로 기능을 확장해 구현할 수 있다. ERC에서 안정성 및 사용성 평가를 마친 Final 표준을 기반으로 특정 사용자한테 귀속될 수 있는 SBT를 구현했다.

네 번째, 구현한 SBT를 통해서 제안된 사용자 인증 시나리오를 만족시킬 수 있도록 구현했다. SBT를 활용한 인증 프로세스에서 발생할 수 있는 정상, 비정상 시나리오를 구현하여, SBT가 정상 작동되는 것을 확인했다. 호출 불가 시나리오도 모두 시나리오대로 작동하는 것을 확인할 수 있었다.

그러나, 본 논문에서는 사용자 데이터를 탈중앙 데이터베이스지만, 결국 오프체인에서 관리하는 특징을 가지고 있어, 완벽한 탈중앙성을 확보했다는 부분에 있어서는 한계가 존재한다. 또한, ERC 표준은 지속해서 새로운 제안이 등장하고 있어 새로운 기술의 적용이 매우 잦은 빈도로 일어난다. 본 연구에서 사용된 ERC-5192의 확장 표준이 나오게 되면, 본 연구에 이어서 향후 연구를 더 필요로 할 것으로 보인다.

ACKNOWLEDGEMENT

"This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program" (RS-2023-00259099)

“This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Graduate School of Metaverse Convergence support program(IITP-2023-RS-2022-00156318) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation)”

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2023. (RS-2023-00219237)

REFERENCES

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System”, Decentralized Business Review, 21260, October 2008. https://www.debr.io/article/21260.pdf.

[2] B. Vitalik, “A Next-Generation Smart Contract and Decentralized Application Platform”, White Paper, 3(37), 2-1, December 2014. https://nft2x.com/wp-content/uploads/2021/03/EthereumWP.pdf.

[3] E. Lee, “The Bored Ape Business Model: Decentralized Collaboration via Blockchain and NFTs”, Available at SSRN 3963881, November 2021. http://dx.doi.org/10.2139/ssrn.3963881.

[4] W. Kaidong, Y. Ma, G. Huang, X. Liu "A first look at blockchain-based decentralized applications." Software: Practice and Experience 51.10 (2021): 2033-2050.

[5] B. Kumar Mohanta, Debasish Jena, Soumyashree S. Panda, Srichandan Sobhanayak, "Blockchain technology: A survey on applications and security privacy Challenges", Internet of Things, Vol. 8, 2019, 100107, ISSN 2542-6605, https://doi.org/10.1016/j.iot.2019.100107.

[6] X. Jia, N. Hu, S. Yin, Y. Zhao, C. Zhang, & X. Cheng, "A2 Chain: A Blockchain-Based Decentralized Authentication Scheme for 5G-Enabled IoT", Mobile Information Systems, Vol.2020, Article ID 8889192, 19 pages, 2020. https://doi.org/10.1155/2020/8889192

[7] A. Petcu, B. Pahontu, M. Frunzete, D. A. Stoichescu, “A Secure and Decentralized Authentication Mechanism Based on Web 3.0 and Ethereum Blockchain Technology.” Appl. Sci. 2023, 13, 2231. https://doi.org/10.3390/app13042231

[8] M. Zhaofeng, M. Jialin, W. Jihui and S. Zhiguang, "Blockchain-Based Decentralized Authentication Modeling Scheme in Edge and IoT Environment," in IEEE Internet of Things Journal, Vol. 8, no. 4, pp. 2116-2123, 15 Feb.15, 2021, doi: 10.1109/JIOT.2020.3037733.

[9] A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, K. -K. R. Choo and M. Aledhari, "Decentralized Authentication of Distributed Patients in Hospital Networks Using Blockchain," in IEEE Journal of Biomedical and Health Informatics, Vol. 24, no. 8, pp. 2146-2156, Aug. 2020, doi: 10.1109/JBHI.2020.2969648.

[10] X. Zhao and Y. -W. Si, "NFTCert: NFT-Based Certificates With Online Payment Gateway," 2021 IEEE International Conference on Blockchain (Blockchain), Melbourne, Australia, 2021, pp. 538-543, doi: 10.1109/Blockchain53845.2021.00081.

[11] Salleras, Xavier, Sergi Rovira, and Vanesa Daza. "FORT: Right-proving and Attribute-blinding Self-sovereign Authentication." Mathematics 10.4 (2022): 617.

[12] Weyl, E. Glen, Puja Ohlhaver, and Vitalik Buterin. "Decentralized society: Finding web3's soul." Available at SSRN 4105763 (2022).

[13] K. Kim, and J. Ryu. "Digital authentication system in metaverse using DID and SBT" Proceedings of KIIT Conference. 2022.

[14] M. I. Lunesu, R. Tonelli, A. Pinna and S. Sansoni, "Soulbound

Token for Covid-19 Vaccination Certification," 2023 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Atlanta, GA, USA, 2023, pp. 243-248, doi: 10.1109/PerComWorkshops56833.2023.10150304.

[15] N. Tanwar, J. Thakur, "Patient-centric soulbound NFT framework for electronic health record (EHR)." J. Eng. Appl. Sci. 70, 33 (2023). https://doi.org/10.1186/s44147-023-00205-9

[16] Tumati, T. Vihar. "SBTCERT: A SOULBOUND TOKEN CERTIFICATE VERIFICATION SYSTEM." Diss. CALIFORNIA STATE UNIVERSITY, NORTHRIDGE, 2023.

[17] P. Dutta, T. Choi, S. Somani & R. Butala, "Blockchain technology in supply chain operations: Applications, challenges and research opportunities." Transportation research part e: Logistics and transportation review 142 (2020): 102067.

[18] Wang, Qin, et al. "Blockchain for the IoT and industrial IoT: A review." Internet of Things 10 (2020): 100081.

[19] Z. Cai, "Usage of Deep Learning and Blockchain in Compilation and Copyright Protection of Digital Music," in IEEE Access, Vol. 8, pp. 164144-164154, 2020, doi: 10.1109/ACCESS.2020.3021523.

[20] "EIP-1: EIP Purpose and Guidelines," Ethereum Improvement Proposals, no. 1, October 2015. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-1.

[21] "ERC-20: Token Standard," Ethereum Improvement Proposals, no. 20, November 2015. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-20.

[22] "ERC-165: Standard Interface Detection," Ethereum Improvement Proposals, no. 165, January 2018. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-165.

[23] "ERC-721: Non-Fungible Token Standard," Ethereum Improvement Proposals, no. 721, January 2018. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-721.

[24] "ERC-777: Token Standard," Ethereum Improvement Proposals, no. 777, November 2017. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-777.

[25] "ERC-1155: Multi Token Standard," Ethereum Improvement Proposals, no. 1155, June 2018. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-1155.

[26] "ERC-5192: Minimal Soulbound NFTs," Ethereum Improvement Proposals, no. 5192, July 2022. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-5192.

[27] "ERC-5484: Consensual Soulbound Tokens," Ethereum Improvement Proposals, no. 5484, August 2022. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-5484.

[28] "ERC-6239: Semantic Soulbound Tokens," Ethereum Improvement Proposals, no. 6239, December 2022. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-6239.

[29] "ERC-4973: Account-bound Tokens [DRAFT]," Ethereum Improvement Proposals, no. 4973, April 2022. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-4973.

[30] "ERC-5114: Soulbound Badge [DRAFT]," Ethereum Improvement Proposals, no. 5114, May 2022. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-5114.

[31] OpenZeppelin [Online], Available: https://github.com/OpenZeppelin/openzeppelin-contracts

Authors

Sung-Woo Cho is currently M.S degree strudent in Computer Science and Engineering from Sogang University. Sungwoo Cho's research interests are in Blockchain, Consensus Algorithm and Soulbound Token.

Jung-Won Seo received M.S degree in Computer Science and Engineering from Sogang University, Korea in March 2020, and is currently pursuing Ph.D degree. Jungwon Seo's research interests are in blockchain and Consensus Algorithm.

Professor Soo-Yong Park holds Ph.D in Information Technology from George Mason University, M.S. in Computer Science from the Florida State University, and B.S. at Sogang University.

Dr. Soo-Yong Park is currently Director of Blockchain Research Center at Sogang University sponsored by Korean Government and was a President & CEO of National IT Industry Promotion Agency (NIPA) from September, 2012 until November 14, 2014. Before joining NIPA, he was a computer science professor (March, 1998-September, 2012) and dean of Graduate School of Information and Technology (March, 2011-September, 2012) at Sogang University.