Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.4, Aug. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.4.1021

블록체인과 바이오메트릭 인증 기술을 이용한 고객 정보 관리 시스템의 개선 방안*

나 윤 석,[1†] 조 상 래,[2] 김 수 형[1,2‡]
[1]과학기술연합대학원대학교, [2]한국전자통신연구원

Improving Customer Information Management System by Using Blockchain and Biometric Authentication*

YunSeok Na,[1†] Sangrae Cho,[2] Soo-hyung Kim[1,2‡]
[1]UST-ETRI, [2]ETRI

요 약

현대는 모든 곳에서 컴퓨터를 활용할 수 있는 유비쿼터스 환경이 형성되고 있다. 이런 환경의 변화에 맞추어 회사들은 IT를 통해 더 나은 서비스를 고객에게 제공하기 위해 고객의 정보를 관리할 수 있는 시스템과 데이터베이스를 개발하고 관리한다. 대부분의 회사들이 사용하는 시스템은 서버에 고객의 정보를 넣고 관리하는 방식이다. 본 논문에서는 이런 시스템의 문제점을 보안성과 편의성 관점에서 도출하고 이를 개선하는 해법을 블록체인 기술과 바이오메트릭 인증을 사용해서 제안한다.

ABSTRACT

Nowadays, the ubiquitous environment that can utilize the computer everywhere is being formed. As the environment changes, services develop and manage systems and databases that can manage customer information to provide better services to customers through Information Technology. The system that most services maintain is a way of putting and managing customer information on the server. In this paper, we first find the problem in terms of security and convenience. After that, we propose a solution that improves the problem through blockchain technology and biometric authentication.

Keywords: Blockchain, Biometrics, Privacy, Personal Information Management

I. 서 론

현대는 모든 곳에서 컴퓨터를 활용할 수 있는 유비쿼터스 환경이 형성되고 있다. 이런 환경의 변화에 맞추어 회사들은 IT를 통해 더 나은 서비스를 고객에게 제공하기 위해 고객의 정보를 관리할 수 있는 시스템과 데이터베이스를 개발하고 관리한다. 회사는 개인 정보 관리 시스템을 이용해서 효율적으로 고객의 정보를 관리하고 활용한다. 고객은 회원 정보에 기반을 둔 맞춤 서비스나 실적에 따른 마일리지를 제공받을 수 있다.

그러나 기존의 개인 정보 관리 시스템은 몇 가지 문제점이 존재한다. 예를 들어 서비스들은 개인 정보를 공유하지 않기 때문에 고객은 이용하는 서비스마다 반드시 등록 과정을 거쳐야 한다. 이는 매번 같은 등록 과정을 고객에게 진행하도록 하므로 사용자의 편의성을 감소시킨다.

Received(07. 10. 2018), Modified(08. 08. 2018), Accepted(08. 08. 2018)
* 이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.2016-0-00097, 비대면 본인확인을 위한 바이오 공개키 기반 구조 기술 개발)
† 주저자, neil123@etri.re.kr
‡ 교신저자, lifewsky@etri.re.kr(Corresponding author)

이런 반복 등록은 서비스의 보안 수준도 저하시킨다. 보안이 취약해지는 이유는 같은 비밀번호를 사용하는 사용자들의 비율이 높기 때문이다. 미국의 CSID 사에 따르면 61%의 사람들은 같은 비밀번호를 여러 웹사이트에서 사용한다[1]. 그러므로 보안이 가장 취약한 웹사이트의 비밀번호가 유출될 경우 같은 고객의 타사 계정이 위험에 노출된 가능성이 높아진다. 개인 정보 관리 시스템의 이런 단점은 고객과 서비스의 보안성과 편의성에 문제를 발생시킨다. 따라서 개인 정보 관리 시스템의 보안성과 편의성을 개선할 필요성이 존재한다.

본 논문을 통해서 기여하고자 하는 바는 다음과 같다. 기존 개인 정보 관리 시스템의 문제점을 보안성과 편의성의 관점에서 도출한다. 그 후 도출된 문제점을 개선하기 위해서 관계형 데이터베이스를 활용한 해법을 만들고 어떤 문제점이 있는지 확인한다. 그 후 해당 문제를 블록체인과 바이오메트릭 인증을 통해서 개선하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 관련 연구들과 배경지식을 정리한다. 3장에서는 기존 시스템의 문제점을 알아보고 블록체인과 바이오메트릭 인증을 도입하는 이유를 설명한다. 4장에서는 개선된 시스템의 구조를 제안한다. 5장에서는 개선된 시스템의 평가를 한다. 6장에서 결론을 맺는다.

II. 기존 연구 및 배경 지식

2.1 바이오메트릭 정보 관리에 관한 연구

2.1.1 바이오메트릭 템플릿

바이오메트릭 인증을 사용하는 시스템은 사용자의 바이오 정보를 암호화해서 저장한다. 암호화된 바이오 정보의 데이터 구조를 바이오메트릭 템플릿이라고 한다. 바이오메트릭 템플릿의 중요한 조건 중 하나는 템플릿이 유출되어도 원래 신체 정보를 추론하거나 복호화 할 수 없어야 한다. 바이오 정보는 평생 변경이 불가능하기 때문이다[2]. 예를 들어 지문 정보가 유출되면 젤라틴 지문을 이용한 스푸핑 공격이 가능하다[3].

바이오메트릭 템플릿의 종류는 크게 4가지가 있다. 1) Salting, 2) Noninvertible Transform, 3) Key-binding biometric cryptosystem, 4) Key-generating biometric cryptosystem[4].

먼저, Salting은 바이오메트릭 정보를 키를 사용해서 암호화 하는 방식이다. 이 방법은 입력 받은 바이오메트릭 정보를 양자화 시킨 뒤 키를 통해 암호화한 후 데이터베이스에 저장한다. 인증의 경우 바이오메트릭 쿼리를 양자화 시킨 후 키를 통해 암호화 한 후 저장되어 있는 정보와 일치하는지를 확인해서 인증을 진행하게 된다[4].

Noninvertible Transform은 비가역적인 연산 또는 변환을 바이오메트릭 정보에 적용시킨 후 해당 정보를 저장하는 방식이다. 역연산이 불가능하므로 템플릿이 유출되어도 원래 정보의 추론이 불가능하다. 인증은 바이오메트릭 쿼리에 비가역적인 변환과 연산을 실행한 후 템플릿과 일치하는지 비교해서 진행하게 된다[4].

Key-binding biometric cryptosystem은 키와 바이오메트릭 정보를 결합해서 연산한 헬퍼 데이터(템플릿)를 데이터베이스에 저장한다. 이 헬퍼데이터와 바이오메트릭 쿼리를 연산하면 키를 연산할 수 있다. 인증은 키가 일치하는지 여부를 확인하는 방식으로 진행 된다[4].

마지막으로 Key-generating biometric cryptosystem은 바이오메트릭 정보를 연산한 헬퍼 데이터(템플릿)를 데이터베이스에 저장한다. 이 헬퍼 데이터와 바이오메트릭 쿼리를 연산하면 키를 연산할 수 있다. 인증은 키가 일치하는지 여부를 확인하는 방식으로 진행 된다[4].

네 가지 방식 중 Key-generating biometric cryptosystem의 경우 엔트로피와 안정성의 트레이드오프 관계가 있다[4]. 이는 보안성(엔트로피)과 편의성(안정성)을 동시에 유지해야 하는 본 논문의 목표에 부합하지 않는 템플릿의 형태이다.

바이오메트릭 템플릿은 복호화를 거치지 않고 입력으로 들어오는 바이오메트릭 쿼리와 연산을 진행하여 True, False 같은 결과나 비밀정보(키)를 반환한다.

공격자가 바이오메트릭 쿼리를 재전송 공격(Replay Attack)을 하는 경우가 있다. 이런 재전송 공격(Replay Attack)에 강인한 바이오메트릭 인증 시스템을 만들기 위해서 일회용 템플릿에 관한 연구가 나오게 되었다. 바이오메트릭 일회용 템플릿에 관한 표준으로는 TTAK.KO-12.0098[5]가 있다. 그리고 가장 많이 사용 되는 템플릿 방식인 Fuzzy Vault에만 적용 되는 일회용 템플릿 기법들이 있다[6][7]. Fuzzy Vault용 일회용 기법들은

기존 템플릿에 있는 특이점 정보를 회전 변환 또는 평행 이동을 시킨 후 거짓 정보를 더해서 원본 템플릿을 변형하는 형식을 사용한다. 템플릿과 쿼리에 변형함수를 모두 적용시키는 방식으로 일회용 템플릿을 만들 수 있다. One-Time Fuzzy Vault의 경우에는 기존 Fuzzy Vault의 약점이던 상관관계 공격에 저항하는 템플릿을 만들어 내는 장점이 있다.

2.1.2 한국은행과 금융결제원의 저장 모델

한국은행은 바이오메트릭 정보를 금융사와 분산 관리 서버(금융결제원)에 인증이 불가능한 형태로 분리해서 보관하는 방식을 표준으로 채택하고 있다. 이는 금융사나 분산 관리 서버 중 한 곳이 해킹 되더라도 부분 정보만을 유출하므로 기존의 방식보다 안전한 방법이다. 인증 시에는 분산 관리 서버와 금융사에 있는 바이오메트릭 정보를 합쳐서 인증을 진행하게 된다[8].

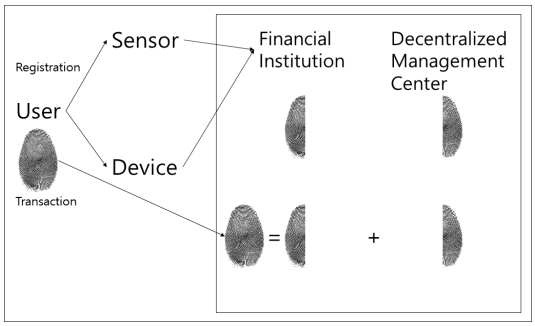

Fig. 1. The Bank of Korea Standard

2.1.3 FIDO

FIDO는 간단하고 안전한 인증을 목표로 하는 확장 가능한 개방 플랫폼이다. FIDO는 두 가지 프로토콜을 제공한다.

첫 번째는 UAF(Universal Authenticaion Framework)로 비밀번호 없이 인증하는 프로토콜이다. 사용자는 디바이스를 온라인 서비스에 등록하고 인증 방식을 선택한다. 등록 이후에는 선택한 방식을 통해서 서비스에 인증을 할 수 있다.

두 번째는 U2F(Universal Second Factor)로 비밀번호가 있는 서비스에 보안을 강화하기 위해 두 번째 인증 요소를 추가하는 방식이다. 두 프로토콜 모두 온라인에서 인증 요청이 들어오면 로컬 디바이스에서 인증을 먼저 진행하는 형식을 갖추고 있다. FIDO의 장점은 개인키를 서버에 저장하지 않는 방식이기 때문에 서버가 해킹 당해도 사용자의 개인키에는 영향이 없으며 로컬 디바이스에서 사용하는 인증 방식의 확장성이 있다는 점이다[9, 10].

Fig. 2. FIDO Registration

Fig. 3. FIDO Login

2.2 블록체인에 개인 정보를 저장하는 방법에 관한 연구

Guy Zyskind 등은 블록체인과 분산 해시 테이블을 활용해서 블록체인에 개인 정보를 저장하고 관리하는 방법을 제안했다[11]. 사용자는 개인 정보를 저장하고 액세스 리스트(ACL)를 블록체인에 저장한다. 액세스 리스트(ACL)는 블록체인에 저장되므로 무결성을 보장할 수 있다. 사용자의 개인 정보는 분산 해시 테이블(DHT)에 저장된다. 정보의 연산은 Secure Multi-party Computation[12]을 사용하게 된다. 이 방법의 장점은 데이터의 저장과 쿼리에 필요한 연산이 효율적인 점이다. 단점은 블록체인에 저장되어 있는 데이터에 연산이 필요할 경우 연산이 비효율적이라는 점이다.

Masayuki Fukumitsu 등은 P2P 네트워크에 개인 정보를 안전하게 저장하는 방법[13]을 블록체인과 Secret Sharing[14]을 활용해서 제안했다. 이 방법은 개인 정보와 더불어 메타 데이터도 숨기므로 공격자는 목표로 하는 사용자의 정보를 찾을 수 없다.

이외에도 이더리움[15]의 스마트 컨트랙을 활용해서 의료 정보를 관리하는 MedRec 등이 있다[16].

III. 기존 시스템의 문제점

3.1 등록 과정과 문제점

고객은 다음과 같은 과정을 거쳐서 등록을 진행한다.

1) 서비스가 고객에게 개인 정보와 인증에 사용할 비밀정보를 요구한다.
2) 고객은 개인 정보와 비밀정보를 제공한다.
3) 서비스는 고객의 데이터가 유효하면 가입을 승인하고 결과를 통보한다.

이 같은 등록 방식의 단점은 다음과 같다. 고객은 이용하게 될 서비스마다 같은 등록 절차를 반복 한다. 이러한 반복은 고객의 편의성을 감소시킨다. 예를 들어 A 서비스에 등록한 고객은 A 서비스는 이용 가능하지만 등록하지 않은 타 서비스는 이용이 불가하다. 이 단점을 보완하는 방법은 서비스 간 공용 개인 정보 저장 공간을 만드는 것이다.

먼저 공용 저장소를 관계형 데이터베이스(MySQL, MariaDB, Oracle 등)를 이용해서 구현이 가능하다면 데이터베이스를 사용하는 것이 좋다. 기존의 데이터베이스 프로그램은 오랜 기간 최적화와 디버깅이 이루어진 검증된 소프트웨어이기 때문이다[17]. 여러 서비스가 한 개의 단일 데이터베이스를 사용하면 반복적인 등록 과정이 필요 없다. 한 번만 서비스에 등록하면 타 서비스도 이용이 가능하다. 하지만 서비스 간 신뢰가 없는 상태이기 때문에 공용 저장소를 만들기 힘들다. 그 이유는 중앙 서버 관리 주체(intermediary)의 권한이 강하기 때문이다. 또한 이런 구조의 시스템은 공용 데이터베이스 시스템이 단일 장애점(single point of failure)이 된다. 다시 말해서 기존 데이터베이스 시스템을 사용하는 해법의 문제점은 서버를 관리하는 주체와 시스템의 안정성에 관한 문제이다. 기존의 데이터베이스 시스템은 반드시 신뢰를 받는 객체가 있어야 한다. 하지만 신뢰를 받는 객체를 두는 방식은 서비스 간 신뢰가 없는 모델에서 사용하기 어렵다. 그러므로 다수의 서비스들이 같은 권한을 가진 탈금융 중개화 해법(disintermediation solution)이 필요하다.

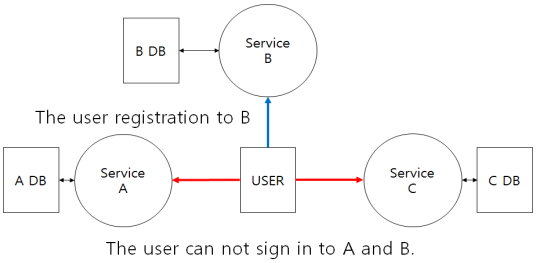

Fig. 4. Traditional Model

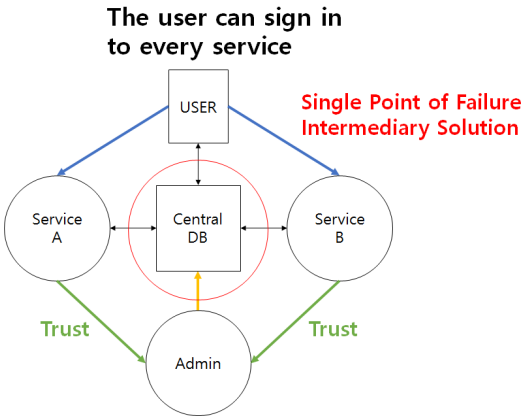

Fig. 5. Solutions using relational databases

블록체인은 처음부터 탈금융 중개화(disintermediation solution)를 염두에 두고 설계되었다[18]. 기존에는 관리자 간 신뢰가 없을 때 신뢰할 수 있는 객체가 반드시 하나 이상 필요했지만 블록체인을 사용하게 된다면 대상이 아니라 블록체인 시스템만을 신뢰하면 된다. 그러므로 그전까지 신뢰할 수 있는 객체 없이 서비스가 불가능했던 서비스도 블록체인을 사용하면 구현이 가능하다.

블록체인을 사용하면 기존 관계형 데이터베이스를 활용한 해법에서 발생하는 세 가지 문제가 모두 해결된다. 먼저 서비스들의 신뢰의 주체가 객체가 아니라 시스템이 된다. 두 번째로 중앙에서 데이터를 관리하는 주체가 있는 것이 아니라 합의 알고리즘을 통해 트랜잭션을 처리하게 된다. 마지막으로 각자 블록체인을 소유하고 관리하므로 몇 개의 노드가 서비스 불가 상태일 때도 전체 시스템을 동작 시킬 수 있다.

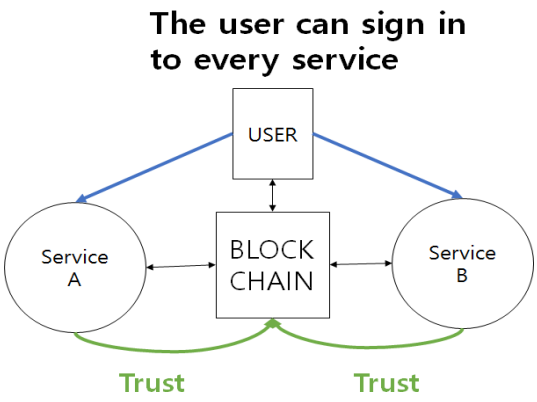

Fig. 6. Solutions using Blockchain

3.2 로그인 과정의 문제점

온라인 로그인의 경우 아이디와 비밀번호를 사용해서 로그인을 진행하게 된다. 이와 같은 온라인 로그인 방식의 단점은 비밀번호 관리다. 모든 서비스마다 같은 비밀번호를 설정하면 보안이 가장 취약한 서비스의 비밀번호가 유출될 경우 타사 계정 비밀번호의 유출 가능성을 높인다. 반대로 서비스마다 다른 비밀번호를 설정하면 사용자의 비밀번호 관리가 어렵다.

오프라인 로그인의 경우 전화번호, 신분증, 회원카드 등을 이용해서 인증을 진행한다. 이 경우에도 단점이 있다. 먼저 타인의 전화번호를 이용해서 적립금 등을 무단으로 사용하는 경우가 있다. 두 번째로 신분증, 회원카드의 경우 도용과 분실의 가능성이 존재한다.

온라인과 오프라인 로그인에서 이와 같은 단점이 발생하는 원인은 지식이나 소유 기반 인증에서 강력한 본인 확인이 어렵기 때문이다. 그러므로 높은 수준의 본인 확인이 가능한 인증 방법을 사용하면 단점을 개선할 수 있다. 그중 한 가지 방법이 바이오메트릭 인증이다. 신체정보는 위조의 가능성이 적고 모든 사람이 항상 소유하고 있다. 또한 엔트로피가 일반적인 지식 기반 인증보다 높으며 사람이 지닌 신체 정보는 구분이 가능할 정도로 다른 경우가 대부분이므로 인증에 사용하기 적합하다. 그러므로 바이오메트릭 인증은 소유 기반 인증의 단점인 분실이나 도용 가능성이 없고 지식 기반 인증의 단점인 낮은 엔트로피와 비밀번호 관리 문제를 보완한다. 본 논문에서는 로그인 과정에서의 문제점을 바이오메트릭 인증을 활용하여 개선한다[2].

3.3 적립금 시스템의 문제점

각 서비스가 적립금을 제공하는 이유는 고객이 다시 서비스를 이용하게 만들기 위해서이다. 그러므로 다른 서비스와 적립금을 공유하는 것은 해당 서비스 입장에서는 바람직하지 못하다. 하지만 고객 입장에서는 개별 서비스들의 적립금을 일일이 관리하거나 기억하기 어렵고 또한 공용 적립금을 사용할 수 있으면 편의성이 증가한다. 이런 공용 적립금 시스템을 위해서는 타 서비스와의 협력 또는 고객의 동의가 필요하다. 공용 적립금 서비스에 대한 케이스를 타 서비스와의 협력이 되어 있는 경우와 고객의 동의만 있는 시나리오를 나누어서 생각하면 다음과 같다.

타 서비스와의 적립금 관련 협력이 되어있는 상태라면 등록 과정에서 살펴 본 해법인 관계형 데이터베이스를 사용한 해법을 사용하면 이 문제를 해결할 수 있다. 다만 이 경우에는 데이터베이스 관리 주체에 대한 신뢰가 반드시 필요하다. 이런 신뢰가 없을 경우에는 등록 과정과 마찬가지로 블록체인을 사용하는 해법을 생각할 수 있다.

그 다음 시나리오는 고객이 공용 적립금 시스템을 원하나 업체들끼리 협력이 되어 있지 않은 경우이다. 이 경우에는 신뢰 받는 주체가 있어야 하는 관계형 데이터베이스는 사용하기 어렵다. 업체들은 신뢰가 없는 상태에서도 블록체인 시스템의 무결성 및 투명성을 신뢰할 수 있으므로 고객이 원한다면 블록체인을 활용하는 공용 저장소를 활용하면 된다[17].

두 시나리오에서 모두 업체 간의 신뢰가 없는 경우에는 블록체인을 사용한 적립금 시스템을 사용하는 것이 더 유용하다.

IV. Proposed Solution

4.1 요구 사항

3장에서 도출한 문제점을 통해서 개선된 시스템의 요구 사항을 다음과 같이 정리했다.

1) 시스템에서 사용하는 데이터 구조는 다중 객체의 쓰기를 지원한다.
2) 온라인과 오프라인 상에서 모두 사용 가능해야 한다.
3) 객체 간 통신에서 중간자 공격(Man-in-the-middle attack)과 재사용 공격(Replay Attack)을 방지한다.
4) ID와 크리덴셜(Credential)을 통해 로그인을 진행한다.
5) 시스템이 해킹 당해도 크리덴셜(Credential)은 침해당하지 않는다.

4.2 시스템 개요

비트코인과 유사한 구조를 사용한다. 헤더에 필요한 정보는 전 블록의 해시 값, 논스, 해시 트리의 루트 값이 필요하다[18].

4.2.1 트랜잭션

기존의 비트코인과 다른 점은 트랜잭션의 구조이다. 기존 비트코인 트랜잭션은 Version, Input Counter, Inputs, Output Counter, Outputs, Locktime 6개의 필드가 있다[19]. 이를 User ID, Service ID, Point Information, Type of Transaction, Error Code, Biometric Data, Locktime, Genesis Pointer, Prev Pointer의 9개 필드로 변경해서 사용한다.

1) **UserID**: 사용자의 ID
2) **ServiceID**: 서비스의 ID
3) **Point**: 적립금 액수
4) **Type**: 0은 가입, 1은 로그인, 2는 적립금 관련 거래를 의미함
5) **Biometric Data**: 바이오메트릭 템플릿이 저장 됨
6) **Locktime**: 거래가 일어난 시각의 타임스탬프
7) **Genesis Pointer**: 사용자가 가입한 블록을 가리키는 포인터이다. 사용자의 바이오메트릭 템플릿이 저장되어 있다.
8) **Prev Pointer**: 사용자의 마지막 거래 내역을 가리키는 포인터이다. 적립금 연산을 위해서 사용된다.

Table 1. Transaction Structure

Field	Variable Name	Size
User ID	uid	8 Bytes
Service ID	sid	8 Bytes
Point	p	8 Bytes
Type	type	1 Bytes
Biometric Data	bio	Variable
Locktime	time	4 Bytes
Genesis Pointer	gen	Varaible
Previous Pointer	prev	Variable

4.2.2 인증서와 키쌍

시스템은 중간자 공격과 재사용 공격을 방지하기 위해서 TLS/SSL을 통신에 사용한다. 그러므로 시스템은 서버 인증서가 필요하다. 그리고 블록체인 트랜잭션에 서명할 개인키와 공개키도 설정한다.

4.2.3 합의 알고리즘

기존의 비트코인과 동일하다. Nakamoto Consensus라 불리는 알고리즘을 사용한다.

1) 멤풀이라 불리는 저장소에 트랜잭션을 저장한다.
2) 트랜잭션을 모아서 블록을 생성한다.
3) 해시 값이 특정 값보다 작아지는 논스를 찾는다. 이 과정을 채굴 또는 Proof of Work라 한다.
4) 채굴이 끝나면 블록을 브로드캐스트 한다.

4.3 함수와 변수명

1) **find(id)**: 입력으로 들어오는 User ID의 마지막 트랜잭션을 반환한다.
2) **match(query, template)**: 바이오메트릭 쿼리와 템플릿을 연산하여 결과를 출력한다. 쿼리와 템플릿의 바이오메트릭이 동일하면 True를 반환한다.
3) **sign(m, SK)**: 메시지 m을 개인키 SK를 활용해서 디지털 서명하는 알고리즘
4) **mempool**: 유효성이 검증되었으나 아직 블록체인에 올라가지 않은 트랜잭션이 모이는 저장 장소이다.
5) $\mathbf{SK_{sid}}$: 서비스 ID가 sid인 서비스의 개인키

4.4 알고리즘

4.4.1 트랜잭션을 생성하는 알고리즘

gen(t,userID,point,type,bio): 매개변수를 통해서 트랜잭션을 생성하는 함수이다. t는 트랜잭션을 발생시킨 사용자의 마지막 트랜잭션을 의미한다. userID, point, type, bio는 각각 사용자 ID, 포인트 차감액, 트랜잭션의 종류, 바이오메트릭 정보를 입력받는 매개변수이다.

Algorithm1 for generate transaction

```
procedure gen(t,userID,point,type,bio):
    t'←∅
    t'.uid←userID
    t'.sid←sid
    t'type←type
    t'.locktime←getTime()
    if type=0
        t'.p←0
```

```
        t'.bio←bio
        t'.gen←NIL
        t'.prev←NIL
    elseif type=1
        t'.p←t.p
        t'.bio←NIL
        t'.gen←t.gen
        t'.prev←t
    else
        t'.p←t.p+point
        t'.bio←NIL
        t'.gen←t.gen
        t'.prev←t
    end if
    s=sign(t',SK_sid)
    return t'||s
end procedure
```

4.4.2 가입

가입을 원하는 ID의 중복을 먼저 체크한다. ID가 중복되면 False를 반환하고 그렇지 않으면 True를 반환하면서 트랜잭션을 생성한 후 멤풀에 트랜잭션을 전송한다. 매개변수 id는 사용자가 입력한 ID를, bio는 바이오메트릭 쿼리를 의미한다.

Algorithm2 for sign up

```
procedure signUp(id, bio)
    t, t'←∅
    t←find(id)
    if t≠NIL
        return False
    else
        t'←gen(t,id,0,0,bio)
        mempool←mempool ∪ t'
        return True
    end if
end procedure
```

4.4.3 로그인

로그인에 사용하는 함수이다. id는 입력받은 ID를 bio는 입력받은 바이오메트릭 쿼리를 의미한다.

1) ID가 존재하는지 체크한다.
2) 바이오메트릭 연산을 통해서 본인 확인을 한다.
3) 이상이 없으면 로그인 기록 트랜잭션을 생성한 후 멤풀에 트랜잭션을 전송하고 True를 반환한다.

Algorithm3 for sign in

```
procedure signIn(id, bio)
    t, t'←∅
    t←find(id)
    if t=NIL
        return False
    end if
    if match(bio,t->gen.bio)=False
        return False
    else
        t'←gen(t,id,0,1,bio)
        mempool←mempool ∪ t'
        return True
    end if
end procedure
```

4.4.4 포인트 적립 및 사용

적립금을 차감할 때 사용하는 함수이다. id는 사용자가 입력한 ID, bio는 바이오메트릭 쿼리, point는 포인트의 차감액을 뜻한다. point가 음수이면 포인트를 사용하는 것이고 point가 양수이면 포인트를 적립하는 것이다.

1) ID가 존재하는지 체크한다.
2) 포인트 사용의 유효성을 체크한다.
3) 바이오메트릭 연산을 통해서 본인 확인을 한다.
4) 이상이 없으면 포인트 이용 트랜잭션을 생성한 후 멤풀에 트랜잭션을 전송하고 True를 반환한다.

Algorithm4 for mileage point usage

```
procedure point(id, bio, point)
    t, t'←∅
    t←find(id)
    if t=NIL
        return False
    end if
    if point+t.p<0
        return False
    end if
    if match(bio,t.bio)=False
        return False
    end if
    t'←gen(t,id,point,2,bio)
    mempool←mempool ∪ t'
    return True
end procedure
```

V. 평 가

5.1 재전송 공격(replay attack) 방지

먼저 이 시스템은 통신에 TLS/SSL을 사용한다. TLS/SSL은 난수가 들어가는 단계가 존재하므로 원천적으로 재전송 공격이 불가능하다[20]. 특정 거래를 한 번 더 실행하기 위해서 재전송 공격을 하게 될 경우 통신에 필요한 난수가 틀리게 되므로 블록체인에 있는 노드들은 재전송 공격이 일어나게 됨을 알 수 있다.

두 번째로 만약 TLS/SSL 단계를 넘어서 재전송 공격이 일어나게 되는 경우를 살펴보면 과거에 일어났던 트랜격션과 똑같은 트랜잭션이 멤풀에 들어오게 된다. 이 경우 해당 트랜잭션의 타임스탬프가 너무 오래된 경우 해당 트랜잭션은 자동으로 폐기 된다[19]. 두 번째로 이미 처리된 트랜잭션인 것을 노드들이 인지하게 되는 경우 합의 알고리즘 상에서 합의를 하지 않는 방식으로 재전송 공격을 방지할 수 있다.

5.2 중간자 공격 방지

이 시스템의 중간자 공격은 서비스와 고객 사이에서 거래 관련 데이터를 가로채거나 수정하는 일을 의미한다. 이 시스템은 SSL/TLS를 사용하므로 CA가 서명한 인증서를 사용하게 된다[20]. 그러므로 고객이나 서비스는 사용하는 인증서가 제대로 서명했는지 확인을 하면 중간자 공격이 발생 여부를 명확히 확인할 수 있다. 또한 인증서를 제대로 확인하지 않아서 중간자가 트랜잭션을 마음대로 수정하거나 발생시켜도 트랜잭션이 의심스러운 경우 해당 트랜잭션에 대한 합의를 노드(서비스)들이 거부할 수 있으므로 중간자 공격의 가능성이 줄어든다.

5.3 바이오메트릭 템플릿 유출

바이오메트릭 정보가 블록체인에 저장 되게 되면 다른 노드들도 해당 바이오메트릭 정보에 접근할 수 있다. 바이오메트릭 정보는 유출 되면 변경이 힘든 특성이 있기 때문에 유출을 방지하거나 유출 되어도 원래 정보를 알 수 없게 암호화 시켜야 한다. 이런 바이오 메트릭 정보를 암호화 시켜서 저장하는 방식을 바이오메트릭 템플릿이라고 한다. 바이오메트릭 템플릿은 유출되어도 원래 신체 정보를 복호화 할 수 없도록 암호화되어 있으므로 안전하다. 이는 바이오메트릭 템플릿이 가져야 하는 보안 조건으로 템플릿에서 원래 신체 정보를 계산하는 것은 불가능하다[3]. 정리하면 바이오메트릭 정보를 암호화 시키지 않은 상태로 블록체인에 저장하는 것은 위험하므로 반드시 템플릿화 시켜서 저장되어야 한다. 템플릿은 또한 암호화 된 상태에서 바이오메트릭 정보를 처리할 수 있는 방법을 제공한다.

예를 들어 Key-binding biometric cryptosystem의 하나인 Fuzzy Commitment 방식[21]은 비밀정보를 오류정정부호(Error Correcting Code)로 인코딩한 값의 해시 값과 인코딩한 값에 바이오메트릭 정보를 XOR 한 값을 데이터베이스에 저장한다. 그러므로 비밀정보를 정확히 복호화하기 위해서는 바이오메트릭 쿼리가 오류정정부호의 보정 범위 내에 있어야 한다. 만약 바이오메트릭 쿼리가 오류정정부호의 보정 범위 밖에 있으면 다른 키가 연산되어서 나오게 된다. 이와 유사한 원리로 다른 바이오메트릭 템플릿을 적용한 시스템이 해킹 당해도 크리덴셜(Credential)은 침해당하지 않는다.

5.4 효율성, 저장량, 계산량

본 논문에서는 개인 정보 관리 시스템의 전반적인 개요를 다루었기 때문에 효율성, 저장량, 계산량에 관한 내용은 본 논문의 범위에서 벗어난다.

다만 트랜잭션의 크기를 128 바이트로 생각하고 일반적인 비트코인의 구조를 그대로 사용한다고 가정하면 처리량은 다음과 같다. 비트코인은 대략 1MB 크기의 블록 한 개를 10분에 처리한다[19]. 1MB를 128바이트로 나누게 되면 $2^{13}(=2^{20}\div 2^{7})$이므로 8096개의 트랜잭션을 10분에 처리할 수 있다. 그러므로 초당 약 13개의 트랜잭션을 처리할 수 있다.

그러나 비트코인의 난이도와 시간을 반드시 따를 필요는 없고 난이도 조절을 통해서 블록 생성 시간을 더 짧게 조절한다면 초당 13개보다 많은 트랜잭션을 처리할 수 있고 결과도 더 빠르게 전송받을 수 있다.

VI. 결 론

고객 정보 관리 시스템은 많은 곳에서 활용되고 있지만 인증, 등록, 적립금 활용 등에서 보안성과 편

의성에 문제점이 존재한다. 이 문제는 소유와 지식 기반 인증의 단점[2]과 개인 정보 저장을 위한 공용 저장소를 만들기 어렵기 때문에 발생한다. 본 논문에서는 관계형 데이터베이스를 활용하는 해법을 먼저 제안하였다. 하지만 관계형 데이터베이스를 활용한 해법은 단일 장애점(single point of failure) 문제를 가지고 있으며 해당 데이터베이스를 관리하는 객체가 신뢰의 대상이 되는 단점이 있었다. 그래서 본 논문에서는 공용 저장소를 신뢰하는 객체 없이 만들 수 있는 블록체인[18]을 활용하여 해당 문제를 개선하였다. 그리고 소유와 지식 기반 인증의 단점은 바이오메트릭 인증을 통해 개선하였다. 바이오메트릭 인증은 변경 불가능한 정보인 생체 정보를 이용하므로 유출에 민감하다. 이 문제는 바이오메트릭 정보를 그대로 저장하는 것이 아닌 템플릿 형태로 변형해서 저장하는 방식으로 해결하였다.

References

[1] CSID, "Consumer Survey: Password Habits" https://www.csid.com/wp-content/uploads/2012/09/CS_PasswordSurvey_FullReport_FINAL.pdf, Sep. 2012

[2] D. Maltoni, D. Maio, A.K. Jain, and, S. Prabhakar, Handbook of fingerprint recognition, 2nd Ed., Springer Science & Business Media, Apr. 2009.

[3] A.K. Jain, K. Nandakumar, and A. Nagar, "Biometric template security," EURASIP Journal on Advances in Signal Processing, Jan. 2008

[4] C. Roberts, "Biometric attack vectors and defences." Computers & Security, vol. 26.1, pp. 14-25, Feb. 2007

[5] TTA, "Biometric Authentication Framework based on One-Time Template", TTAK.KO-12.0098, Dec 2008

[6] Woo Yong Choi, Yongwha Chung, Jin-Won Park and Dowon Hong, "Fingerprint Template Protection Using One-Time Fuzzy Vault," KSII Transactions on Internet and Information Systems, vol. 5, no. 11, pp. 2221-2234, Nov 2011.

[7] Daesung Moon, Sungju Lee, Seunghwan Jung, Yongwha Chung, Miae Park, and Okyeon Yi "Fingerprint template protection using fuzzy vault." International Conference on Computational Science and Its Applications, LNCS 4707, pp. 1141-1151, 2007.

[8] Bank Of Korea, "Seminar on promotion of biometric authentication in financial sector" https://www.bok.or.kr/viewer/skin/doc.html?fn=FILE_201803300815507562.pdf&rs=/webview/result/P0000559/201507, Jul 2015

[9] ETRI, "Passwordless Authentication Technology-FIDO", Electronics and Telecommnucations Trends, vol. 29. no. 4, pp. 101-109, Aug 2014.

[10] FIDO Alliance, "FIDO UAF Architectural Overview 1.0" , https://fidoalliance.org/specs/fido-uaf-v1.0-ps-20141208/fido-uaf-overview-v1.0-ps-20141208.html, Dec 2014.

[11] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," IEEE Security and Privacy Workshops (SPW), pp. 180-184, May. 2015.

[12] A.C. Yao, "Protocols for secure computations." Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on. IEEE, Nov 1982.

[13] M. Fukumitsu, S. Hasegawa, J. Iwazaki, M. Sakai, D. Takahashi "A Proposal of a Secure P2P-type Storage Scheme by using the Secret Sharing and the Blockchain," IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), pp. 803-810, Mar. 2017.

[14] A. Shamir, "How to share a secret." Communications of the ACM vol. 22. no. 11 pp. 612-613, Nov 1979.

[15] V. Buterin, "A next-generation smart

contract and decentralized application platform." white paper 2014.
〔16〕 A. Azaria, A. Ekblaw, T. Vieira, A. Lippman "Medrec: Using blockchain for medical data access and permission management." Open and Big Data (OBD), International Conference on. IEEE, Aug. 2016.
〔17〕 K. Wüst and A. Gervais. "Do you need a Blockchain?." IACR Cryptology ePrint Archive 2017-375, Apr. 2017.
〔18〕 N. Satoshi, "Bitcoin: A peer-to-peer electronic cash system.", 2008.
〔19〕 A.M. Antonopoulos, Mastering Bitcoin: unlocking digital cryptocurrencies, 1st Ed., O'Reilly Media, Inc, Dec. 2014.
〔20〕 W. Stallings, Cryptography and network security: principles and practices, 4th Ed., Prentice Hall, Nov. 2005.
〔21〕 A. Juels and M. Wattenberg. "A fuzzy commitment scheme," CCS '99 Proceedings of the 6th ACM conference on Computer and communications security, pp. 28-36, Nov. 1999

〈저자소개〉

나 윤 석 (YunSeok Na) 학생회원
2015년 2월: 한동대학교 전산전자공학부 졸업
2017년 3월~현재: 과학기술연합대학원대학교 석사과정
2017년 3월~현재: 한국전자통신연구원 정보보호연구본부 UST 연수생
〈관심분야〉 정보보호, 블록체인, 계산이론

조 상 래 (Sangrae Cho) 정회원
1996년: Imperial College London, Computing 학사
1997년: Royal Holloway, University of London, Information Security 석사
1997년~1999년 LG 종합기술원 연구원
1999년~한국전자통신연구원 책임연구원
〈관심분야〉 인증, ID 관리, 바이오 인증

김 수 형 (Soo-hyung Kim) 정회원
1996년 2월: 연세대학교 컴퓨터과학과 졸업
1998년 8월: 연세대학교 컴퓨터과학과 석사
2016년 2월: KAIST 전산학 박사
2000년 11월: 한국정보통신연구원
2000년 12월~현재: 한국전자통신연구원 정보보호연구본부 기술총괄
〈관심분야〉 ID관리, 바이오인증, 핀테크 보안, 모바일 보안, 개인정보보호