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.2018.28.4.987

HyperCerts : 개인정보를 고려한 OTP 기반 디지털 졸업장 블록체인 시스템*

정 승 욱†‡
건양대학교

HyperCerts : Privacy-Enhanced OTP-Based Educational Certificate Blockchian System*

Seung Wook Jung†‡
Konyang University

요 약

블록체인의 tamper-free 특성으로 많은 응용들이 개발되고 있다. MIT Media Lab 등은 기존 학력 증명의 진본 여부를 확인하는 절차의 복잡한 문제를 해결하기 위해서 블록체인 기반 디지털 졸업장 시스템을 개발하였다. 기존의 연구는 public blockchain 기반으로 원칙적으로 누구나 디지털 졸업장을 발급자가 될 수 있으나 이를 해결하기 위한 방법을 명확히 제시하고 있지 않다. 기존의 학력 증명 블록체인 시스템은 블록체인의 무결성을 활용하지만 개인정보가 다수 들어 있는 졸업장의 기밀성 문제를 해결하지 못하고 있다.

본 논문에서는 최초로 private blockchain 기반으로 HyperCerts라 명명된 디지털 졸업장 시스템을 제안한다. Private blockchain 기반이므로 신원이 확인된 신뢰할 수 있는 자만이 디지털 졸업장을 발급할 수 있으며 practical byzantine fault tolerance를 합의 알고리즘으로 이용하여 작은 컴퓨팅 파워를 필요로 하며 합의에 따른 지연이 매우 적은 장점이 있다. 디지털 졸업장은 민감한 개인정보를 포함한다. 따라서 디지털 졸업장의 privacy는 보장되어야 한다. HyperCerts는 디지털 졸업장의 hash값만 분산 원장에 저장하므로 악의적 노드의 참여로 인한 개인정보 유출 문제를 원천적으로 차단한다. 또한 디지털 졸업장은 암호화되고 OTP와 함께 제공되어 복호화 횟수 제한 등으로 디지털졸업장이 유출되었을 때 무분별한 복호화를 방지하도록 하였다.

ABSTRACT

Blockchain has tamper-free, so many applications are developing to leverage tamper-free features of blockchain. MIT Media Labs proposed BlockCerts, educational certificate blockchain System, to solve problems of legacy certificate verifications. Existing educational certificate blockchain Systems are based on public blockchain such as bitcoin, Ethereum, so any entity can participate educational institute in principal. Moreover, the exisitng educational certricate blockchain system utilizes the integrity of blockchain, but the confidentiality of the educational certificate is not provided.

This paper propose a digital certificate system based on private blockchain, name HyperCerts. Therefore, only trusted entity can participate in the private blockchain network, Hyperledger, as the issuer of digital certificate. Furthermore, the

Received(07. 09. 2018), Modified(08. 03. 2018), Accepted(08. 06. 2018)
* 본 연구는 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구입니다.(NRF-2017R1D1A3B03036404),
* 이 논문은 2018년도 정부(교육부)의 재원으로 한구연구재단의 대학중점연구소지원사업으로 수행된 연구임(NRF-2018R1A6A1A03025542).

† 주저자. swjung@konyang.ac.kr
‡ 교신저자, swjung@konyang.ac.kr(Corresponding author)

practical byzantine fault tolerance is used as consensus algorithm, HyperCerts reduce dramatically the latency of issuing digital certificate and required computing power. HyperCerts stores the hash value of digital certificate into the ledger, so breach of personal information by malicious entity in the private blockchain is protected.

Keywords: Blockchain, Digital Educational Certificate, privacy, OTP

I. 서 론

"성적표·졸업 증명서까지 … 뭐든지 만들어드립니다."[1] 기사 뿐 아니라 구글에서 졸업장 위조라고 검색하면 수많은 위조 관련 전자 우편 주소 등을 확인할 수 있다. 신정아 학력 위조 사건처럼 실제로 학력 위조는 발생하고 있으며 구인을 해야 하는 회사의 입장에서 학력 증명서의 진본확인은 반드시 필요한 절차이다. 하지만 진본 확인 절차는 시간이 걸리고 복잡하다[1].

신뢰기계라고 불리는 블록체인은 tamper-free 특성으로 문서 위조의 문제를 해결할 수 있다. 하지만 블록체인은 transparency 특성으로 blockchain network에 참여한 개체(entity)들은 누구나 분산원장을 볼 수 있어 개인정보 보호가 필요하다[2]. 졸업장/성적증명서도 민감한 개인정보를 포함하므로 비인가자가 접근할 수 없도록 구현해야 한다.

기존의 졸업장 blockchain system[1][3]은 public blockchain 기반이다. [1]은 ethereum 기반이며 [3]은 bitcoin기반이다. public blockchain network 특성상 누구나 public blockchain network에 참여하여 (개인키와 공개키를 생성해서) 디지털 졸업장을 발급할 수 있으며 [1][3]은 디지털 졸업장 발급자에 대한 신원 증명에 관해서 명확하게 설명하고 있지 않다.

또 기존 시스템[1]의 문제점은 졸업장 자체(암호화든 아니든)를 저장하고 있어 block의 크기가 커서 시간이 지날수록 저장 용량이 급격히 증가하여 운용상의 문제가 있다.

본 논문은 private blockchain인 hyperledger fabric 1.0[4]을 기반으로 디지털 졸업장/성적증명서(이하 디지털인증서) 시스템을 개발하였다. 본 시스템에서는 개인정보 노출을 방지하기 위해서 졸업장의 해쉬값을 분산 원장에 저장하고 졸업장은 암호화하여 off-blockchain storage에 저장한다(confidentiality). 또한 한번 발급한 디지털 인증서은 디지털 문서이므로 무한 복제될 수 있으며 인증서의 소유자가 원치 않는 제3자에게 전달되어 개인정보가 노출될 수 있다. 이에 본 논문에서는 one-time pad (OTP)를 이용하여 발급된 디지털 인증서의 검증은 1회만 할 수 있도록 하였다 (one-time).

본 논문의 contribution은 1) 최초로 private blockchain 기반으로 디지털인증서 시스템을 구현하였다. 2) blockkchain이 무결성을 제공해 주고 있으나 transparency 특성으로 기밀성을 보장해 주지 못하는 문제를 hash-and-OTP 방식으로 blockchain 상에서 기밀성을 제공하는 방식을 제안하는데 있다.

본 논문의 구성은 2장에서 관련 연구를 소개하며 3장에서 블록체인 기반 디지털졸업장 시스템을 소개한다. 4장에서는 hash-and-OTP 방식을 설명한다. 5장에서 블록체인 기반 디지털졸업장의 보안 및 추가 논의를 한다. 마지막으로, 6장에서 결언을 맺는다.

II. 관련 연구

2.1 기존의 디지털 졸업장 시스템

블록체인 기반 디지털 졸업장 시스템은 최초로 MIT Media Lab에서 기존 디지털 졸업장 시스템의 신뢰와 보안 문제를 해결하기 위해서 blockcerts라는 시스템을 소개하였다[3]. 전체적인 시스템 구성은 간단하다. 디지털 졸업장 발급자가 인증서를 서명하고 그 해쉬값을 bitcoin 분산원장에 저장한다[1]. Blockcerts는 디지털인증서의 무결성을 해결하지만 bitcoin network을 이용하므로 써 PoW(Proof of Work)에 의해서 많은 컴퓨팅 파워를 필요로 하며, 블록체인에 등록되는데 걸리는 지연이 발생한다. 사용자가 디지털 졸업장을 wallet에 저장해야 하며 wallet 분실 시 문제점이 발생할 수 있다. 또한, 해당 학생에 대한 블록체인에 등록된 과거이력을 모두 검색하는데 많은 시간이 소요된다[1].

blockcerts의 검색 시간 소요를 단축시키기 위해서 [1]은 ethereum 기반으로 효율적인 검색이 가

1) 일요신문 2015.02.09

능한 블록체인기반 졸업장 시스템을 제안하였다. 하지만 개인정보를 공개키 기반으로 암호화하므로 써 암호화 및 복호화의 비효율성을 가지고 있으며, 디지털졸업장 자체를 분산원장에 저장함으로써 시간이 지날수록 필요한 저장 공간이 매우 커지는 문제를 가지고 있다. blockcerts와 마찬가지로 public blockchain기반으로 누구나 발급자가 될 수 있는 문제점을 가지고 있으며 이를 해결하기 위해서 복잡한 관리 절차를 가져야 한다.

본 논문에서는 신원이 확인된 발급자가 디지털 인증서를 발급할 수 있도록 private blockchain기반으로 디지털 인증서 시스템을 구축하였다. 또한, private blockchain은 신원이 인증된 신뢰할 수 있는 참여자만 참여하므로 Practical Byzantine Fault Tolerance 등 컴퓨팅 파워를 많이 필요로 하지 않는 합의 알고리즘을 쓸 수 있다. 따라서 기존의 시스템이 PoW등을 사용하여 많은 컴퓨팅 파워를 필요로 하였으나 본 논문에서는 이와 달리 많은 컴퓨팅 파워를 필요로 하지 않는다. 마지막으로 사용자가 디지털인증서를 보관해야 하는 불편함을 제거하였으며 디지털인증서는 대칭키 기반으로 암호화하여 암호화 효율을 높였으며 디지털인증서의 해쉬만 저장하므로 저장 용량에 대한 문제도 제거하였다.

2.2 Hyperledger

Hyperledger[4]는 2015년 12월 설립하고 2016년 1월에 정식 발촉한 Linux foundation 오픈 소스 프로젝트로 private blockchain 구현을 목표로 하고 있다. Hyperledger는 현재 IBM, 엑센추어, JP 모건등 100여개 기업이 참여하고 있다.

Hyperledger Fabric은 Hyperledger의 core 부분으로 2017년 7월에 버전 1.0의 정식 버전이 발표되었다. Hyperledger의 구조는 Fig.1에 나와 있는 것처럼 크게 Membership 서비스, Application, Peer (Endorser, Committer), Chaincode, Ordering 서비스 등으로 나눌 수 있다. Membership 서비스는 신원이 확인된 참여자만 참여할 수 있도록 하는 CA(Certificate Authority) 기능을 포함하여 신원이 확인된 참여자에게 X.509 인증서를 발급해 주고 X.509인증서를 이용하여 신원을 확인한다. Application은 다양한 응용이 될 수 있으며 Application은 Hyperledger SDK를 통하여 Hyperledger

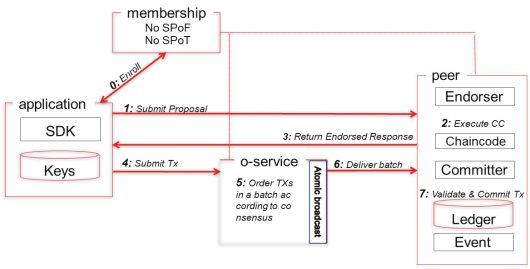

Fig. 1. Hyperledger Fabric 1.0 Architecture

network과 통신한다. 본 논문에서는 application이 디지털인증서 서비스이다. Application이 chaincode를 통하여 블록체인에 데이터를 저장하거나 읽어들일 것이다. Chaincode도 application-specific하게 구현해야 한다. Application이 chaincode를 호출하기 위해서 proposal을 보내면 endorser peer가 proposal이 형식에 맞는지 chaincode가 정확히 동작하는지 확인한다. 그 결과를 application에 돌려보내면 사용자는 트랜젝션을 ordering service에 제출한다. Ordering 서비스는 분산 원장에 들어갈 내용을 순서에 맞게 정리하여 committer peer에게 보낸다. Committer peer는 제출받은 트랜잭션들을 분산원장과 world state DB에 기록한다. Hyperledger는 분산원장 뿐 아니라 최종 상태를 world state DB라는 key/value DB에 저장한다.

III. HyperCerts

3.1 시스템 구성

HyperCerts의 전체 구성은 그림 2와 같다. 시스템 구성은 크게 HyperCerts Network(빨간색 선안 : 디지털 인증서 등록, 발급, 검증을 하는 HyperCerts application server 및 각 학교의 Peer 및 Orderer), 학교 등록 시스템(University), 디지털인증서 수령인(Student), 디지털인증서 소비자(Employer)로 구성된다.

Hyperledger network은 학교로 구성된다. 구성된다. 각 Peer(학교)들이 World state DB와 분산 원장을 관리하고 Orderer가 합의(consensus)하여 Committer Peer들이 최종 상태를 분산 원장과 World State DB에 저장하도록 한다.

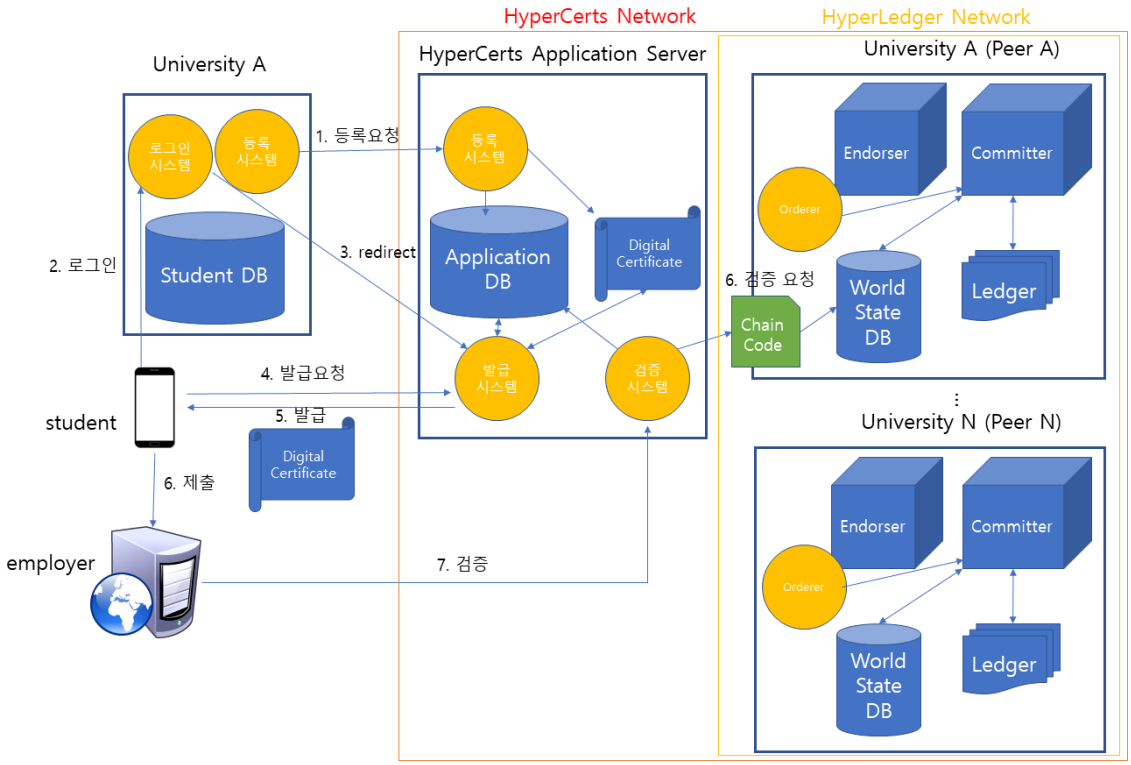

Fig. 2. Overall System Architecture

World State DB는 CouchDB를 이용하며 Key/Value를 저장한다. 본 논문에서 Key값은 다음과 같다.

CouchDB_Key_Value := h(학교|학과|학번|이름|생년월일|졸업년도|학위)

CouchDB_Key_Value 값은 학교, 학과, 학번, 이름, 생년월일, 졸업년도, 학위로 유일하게 식별할 수 있는 값을 이용하고 있으며 value값은 다음과 같다.

value := h(Cert.Binary)

여기서 h는 암호학적으로 안전한 일방향 hash 함수이며 Cert.binary는 PDF파일과 같이 졸업장 binary file이다. value값은 또한 분산 원장에 저장되어 변경이 불가능해진다.

HyperCerts Application DB는 student table과 OTP table로 구성된다.

student_table := primary_key|학교|학과|학번|이름|영문이름|생년월일|Email Address|졸업년도|학위|AES-Key|File_path(ENC_CERT)

OTP table := OTP|primary_key

여기서 File_path는 해당 파일의 경로를 URL로 return해주는 함수이며 ENC_CERT는 암호화된 디지털인증서이다. OTP(One-time password)는 학생이 졸업장을 발급할 때마다 random하게 생성되어 student_table의 저장된 해당 학생의 primary_key 값과 같이 OTP table에 저장된다.

또한 HyperCerts Application File system에 학생의 암호화된 디지털 인증서는 다음과 같이 저장한다.

$$ENC\_CERT := AES\_256_{AES\text{-}Key}(DEC\_CERT)$$

여기서 AES 256bits로 암호화하고 키는 student_table에 저장된 AES-Key를 사용하였다.

또 원본 디지털인증서 파일은 JSON형식으로, 다음과 같다.

DEC_CERT := {“CouchDB_Key”: Couch_DB_Key_Value, “Cert”:Cert.Binary}

학교 시스템은 학생 정보가 들어 있는 DB와 학생 로그인 시스템을 가지고 있으며 최초에 학생 DB에 기초하여 Cert.Binary을 만들어서 HyperCerts Application의 등록 시스템에 Cert.Binary와 학생 정보(학교|학과|학번|이름|영문이름|생년월일|Email Address|졸업연도|학위)를 등록한다.

각 학교가 HyperCerts Application에 최초로 등록할 때(enrollment) 각 학교의 인증서와 schoolID를 등록한다.

학생은 별도의 프로그램 없이 Web 브라우저로 발급시스템을 통하여 졸업장 발급을 요청을 한다. 발급 App에 접근하기 위해서 학교 시스템에서 로그인하고 로그인 성공 결과와 학생 정보를 HyperCerts Application Server의 발급 App에 해당 정보를 전송한다. 발급 App은 학생의 요청에 따라 학생의 학교|학번|이름|생년월일|졸업년도|학위 등을 기반으로 student_table에서 File_path(ENC_CERT)를 확인하여 ENC_CERT를 읽어 들이고 해당 졸업장에 랜덤하게 생성된 OTP를 append하여 Cert.hcrt파일을 다음과 같은 형식으로 만들어 학생에게 제공한다.

Cert.hcrt : = ENC_CERT | OTP

학생은 발급받은 Cert.hcrt 파일을 디지털인증서 소비자(employer, 이하 고용주)에 제출한다. 여기서 이렇게 디지털 인증서를 학생에게 제공되는 이유는 많은 회사가 졸업장 등을 파일로 인터넷상으로 업로드 하여 받기 때문이다. 그렇지 않으면 학생이 디지털인증서를 우리 시스템에서 복호화하여 프린트 하여 제출할 수 있을 것이다. 고용주는 HyperCerts Application Server의 검증 App에 접근하여 Cert.hcrt파일을 제출한다. 검증 App은 Cert.hcrt파일에서 OTP value를 추출하여 OTP table에서 OTP value를 가지고 primary_key를 조회한다. 만약 OTP가 없으면 이미 사용된 파일로써 error를 return한다. OTP value에 값이 있으면 획득한 primary_key값으로 student_table에서 AES-Key를 추출한다. 추출된 AES-Key를 가지고 ENC_CERT를 복호화하여 DEC_CERT를 획득한다. DEC_CERT에서 CouchDB_Key_Value와 Cert.Binary를 획득하여 chaincode에 두 값을 전달한다. Chaincode는 Cert.Binary를 hash하여 CouchDB_Key_Value에 대한 world state DB에 저장된 hash값을 비교한다. 만약 같으면 OTP table에서 해당 OTP와 primary key를 삭제하여 두 번 다시 복호화하지 못하도록 한다. 진본이 확인되면 복호화된 디지털인증서를 고용주에게 보여준다.

여기서는 분산 원장에 저장해야 하는 개인정보 파일을 암호화하여 application DB에 저장하고 파일의 해쉬값을 분산 원장에 저장한다. 제공된 파일의 무한대로 복호화되는 것을 막기 위해서 OTP를 사용하였다. 본 논문에서는 hash값을 분산 원장에 저장하고 OTP를 이용하여 개인정보가 들어있는 디지털 졸업장을 한번만 획득할 수 있도록 하였다. 복호화되어 사용자에게 보여 진 상태에서 저장 및 무한 복제되는 것을 막는 것은 이 논문의 범위를 벗어나며 다양한 일반적인 방법으로 무한 복제를 막을 수 있을 것이다. 예를 들어, 복호화된 파일을 보여주는 응용프로그램에 다운로드 기능이 없든지 또는 DRM등의 방법을 적용할 수 있을 것이다.

3.2 디지털 졸업장 시스템 흐름

3.2.1 디지털 졸업장 등록

1. 각 학교의 졸업증명서 등록시스템은 학생 정보(학교|학과|학번|이름|영문이름|생년월일|Email Address|졸업년도|학위)를 가지고 있다. 성적증명서도 유사하게 학생 정보가 들어가 있을 것이다. DB에서 학생 정보를 읽어서 gofpdf PDF generation library[2]와 같은 PDF generator library를 이용하여 디지털 인증서 원본(Cert.binary)을 만든다.

2. 만들어진 PDF파일과 학생 정보를 JSON 형식으로 HyperCerts Application Server의 등록 App에 등록 요청을 한다. 이때 보안을 위해서

2) https://github.com/jung-kurt/gofpdf

TLS 1.2 기반의 secure channel로 통신한다. 또한 발신자의 신원 및 부인 방지를 위해서 서명을 붙인다. 서명 형식은 [5]을 따른다.

```
registerJSON := {"binary":Cert.Binary,
"name":"홍길동",
"schoolID": "1234", "school":"konyang",
"department":"cybersecurity",
"schoolID":"15010137",
"englishName":"Alice",
"birthDay":"2000-12-06",
"emailAddress":"hong@naver.com",
"dayOfGraduate":"2018-02-22",
"degree":"bachelor"}

SignedJSON := {
        "payload": XJyZW5jeSI6...gICB9",
        "protected": "eTU2IgoICAgfQ==",
        "header": {
            "signature":
            "DtEhU3ljbEg8L38VWM6-Xx-"
        }
    }
```

payload는 registerJSON의 base64URL encoding된 정보이며 protected는 서명 알고리즘의 base64URL encoding format이다. siganture는 school ID에 해당하는 private key로 payload를 서명한 base64URL이다. public key는 학교가 HyperCerts에 등록할 때 생성되어 저장된 인증서에 들어 있다.

3. HyperCerts Application Server는 우선 SchoolID에 해당하는 공인인증서를 찾아서 서명이 정확한지 확인한다.

4. 서명이 확인되면 base64URL로 된 payload를 JSON형태로 만들고 졸업장 binary (Cert.Binary)를 추출하고 hCert.binary := h(Cert.Binary)를 생성한다. 또한 CouchDB_Key_Value := h(학교|학과|이름|생년월일|졸업년도|학위)를 만든다. hCert.binary를 value으로 CouchDB_Key_Value를 key로 하여 chain code의 invoke 함수를 Hyperledger SDK를 통하여 호출한다. 호출된 결과로 transaction proposal을 endorser peer에 제출하게 된다. Endorser peer는 proposal 제출자의 서명, 메시지 format, 정책 등을 확인하여 이상이 없으면 proposal result에 200 OK를 보내게 되고 이를 받은 HyperCerts Application의 등록 App은 orderer에 트랜젝션을 제출한다. orderer들이 합의를 하고 나면 committer peer에 제출하고 committer peer는 최종적으로 분산원장과 world state DB에 CouchDB_Key/hCert.binary를 Key/Value로 저장한다.

5. Key/Value로 저장이 성공하면 DEC_CERT := {"Couch_Key":Couch_Key_Value, "Cert": Cert.Binary}를 생성한다.

6. Random한 AES-Key를 생성하고 이를 이용하여 생성된 DEC_CERT을 암호화한다.

7. ENC_CERT를 file server에 저장하고 File_path(ENC_CERT)를 확인하다. HyperCerts Application은 student table에 primary_key|학교|학과|학번|이름|영문이름|생년월일|Email Address|졸업년도|학위|AES-Key|File_path(ENC_CERT)를 저장한다.

3.2.2 디지털 졸업장 발급

1. 학생은 학교에 발급 요청을 하면 학교는 login 화면을 보여 준다. 학생이 로그인하여 성공하면 HyperCerts Application의 발급 App으로 학교|학번|이름|생년월일 정보와 함께 HyperCerts Application의 등록 App으로 redirect한다.

2. 학생은 학위와 졸업년도를 선택하고 발급 요청을 한다.

3. HyperCerts Application의 발급 App은 학교|학번|이름|생년월일|졸업년도|학위 정보를 가지고 student DB에 query하여 File_path (ENC_CERT)를 획득한다.

4. HyperCerts는 랜덤하게 OTP를 생성하고 생성된 OTP_Value와 student_table의 primary_key를 OTP table에 기록한다.

5. HyperCerts Application의 발급 App은 최종적으로 Cert.hcrt := ECN_CERT|OTP를 만들고 학생에게 발급한다.

6. 학생은 Cert.hcrt 파일을 다운로드하고 고용주에게 Cert.hcrt파일을 제출한다.

3.2.3 디지털 졸업장 검증

1. 고용주는 HyperCerts Application의 검증 App에 접속하여 Cert.hcrt 파일을 제출한다.
2. HyperCerts Application의 검증 App은 Cert.hcrt에서 OTP를 획득하고 OTP table에 query하여 primary_key를 획득한다. 만약 null이 return되면 해당 OTP가 없으므로 error를 return한다.
3. 획득된 primary_key를 이용하여 student_table에서 AES-key를 획득하고 Cert.hcrt에서 ENC_CERT를 복호화한다.
4. 복호화된 결과 DEC_CERT을 획득하게 되고 DEC_CERT에서 CouchDB_key와 Cert.Binary를 획득한다.
5. HyperCerts는 CouchDB_key와 Cert.Binary를 ChainCode의 query로 전송한다.
6. ChainCode는 CouchDB_Key_Value를 이용하여 저장된 hash값을 획득하고 Cert_Binary를 hash한 결과와 비교한다.
7. 비교 결과 두 hash가 동일하면 검증 성공 결과와 Cert.Binary를 응용프로그램을 통하여 보여준다. 또한, OTP table에서 해당 OTP_Value와 primary key를 삭제하여 오직 한번만 복호화하는 것을 보장한다.
8. 고용주는 졸업장의 진본 여부를 확인한다.

3.2.4 디지털 졸업장 구현

Hyperledger는 Node.js와 Java 두 가지 버전의 SDK를 제공하고 있다. 이 SDK를 application이 호출하여 Hyperledger Network과 통신한다. 현재 Java는 SDK에 대한 문서화가 충분하지 않고 참고할 응용프로그램도 충분하지 않다. 따라서 본 논문에서는 Node.js를 이용하여 application을 구현하였다.

Fig. 3이 HyperCert 메인 화면이다. 우선 각 대학교에서 Fig. 4의 화면을 통해서 디지털 졸업장

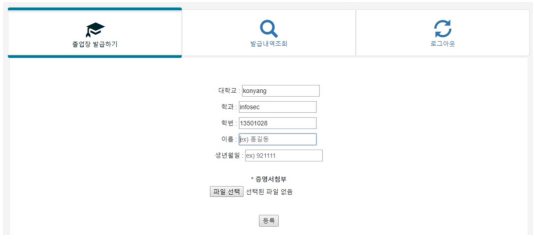

Fig. 3. Digital Certificate Registration Screen Shot

Fig. 4. HyperCerts Main Screen

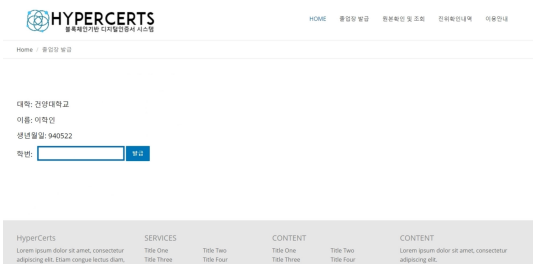

Fig. 5. Digital Certificate Issues

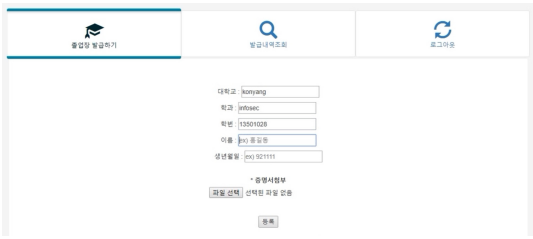

Fig. 6. Digital Certificate Register Screen Shot

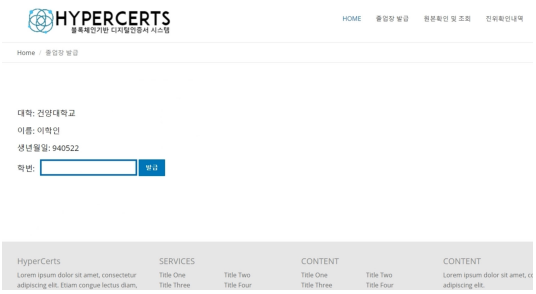

Fig. 7. Digital Certificate Issue Screen Shot

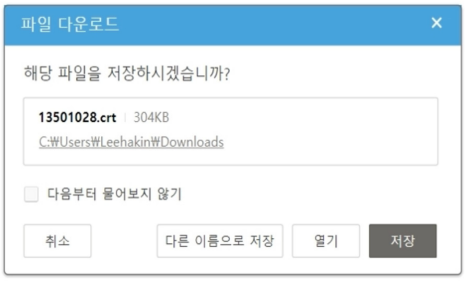

Fig. 8. Digital Certificate Download Screen Shot

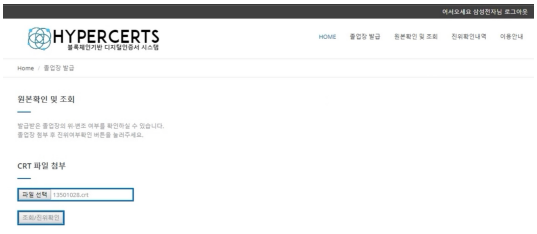

Fig. 9. Digital Certificate Verification Screen Shot

을 등록해야 한다.

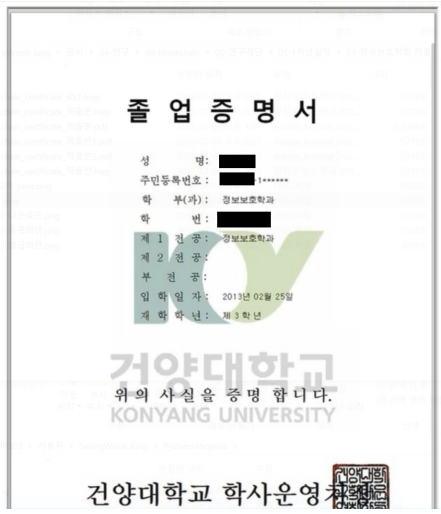

Fig. 10. The Verification Result

IV. Hash-and-OTP

본 논문에서는 blockchain 상에 기밀성을 요구하는 자료를 저장할 때 기밀성을 제공하기 위해 hash-and-OTP 방식을 제안한다.

hash-and-OTP 방식은 3장에서 기술한 것과 같이 디지털 졸업장과 같이 기밀성을 필요로 하는 문서는 분산 원장에서는 hash값만 저장한다. 그리고 원본 파일은 암호화하여 off-blockchain storage에 저장한다. 해당 문서를 필요로 하는 사람에게 발급할 때는 OTP를 생성하고 OTP table에 해당 OTP와 필요한 정보(복호화 키 등)을 저장하고 OTP와 암호화된 파일을 합쳐서 발급한다. 발급된 파일을 가지고 무결성을 검증하거나 복호화하고자 할 때 파일을 블록체인 시스템에 제출하면 OTP를 검색하여 복호화 키를 획득하고 OTP table에 해당 row를 삭제하여 다시 복호화하는 것을 방지하고 복호화하여 무결성을 검증하고 복호화된 파일을 제출자에게 제공한다. 여기서 OTP table에 횟수 필드를 추가하면 해당 횟수만큼 해당 문서를 검증하고 복호화 할 수 있을 것이다. 또는 OTP table에 검증 IP 주소 필드나 IP 주소 대역을 넣어서 최초 검증자의 IP 주소나 특정 IP 주소 대역만 반복하여 검증하고 복호화할 수 있도록 할 수 있다. 또는 해당 OTP table에 기간 필드를 넣어 특정 기간만 복호화할 수 있도록 할 수 있다. 또는 횟수와 검증 IP, 특정 기간 등을 복합적으로 사용하여 적절하게 시스템을 구성하고 접근

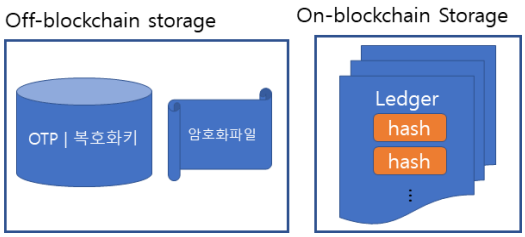

Fig. 11. Hash-and-OTP Storage Structure

통제를 할 수 있도록 하였다.

Hash-and-OTP 방식은 디지털졸업장 뿐 아니라 기밀성을 필요로하는 blockchain 응용에서 다양하게 이용할 수 있을 것이다.

V. 보안 및 논의

5.1 보안

블록체인 기반 디지털 인증서 시스템은 블록체인의 tamper-free 특성을 이용하여 디지털 인증서의 무결성과 진본 여부를 확인하였다. 여기서 디지털 인증서 원본을 분산 원장에 저장하기보다는 디지털 인증서의 해쉬값을 분산 원장에 저장함으로써 개인정보를 보호하고 분산 원장의 저장 공간을 절약하였다. 따라서, 악의적 노드(peer)들이 블록체인 네트워크에 참여하여 분산 원장을 확인하여도 개인정보를 획득할 수 없다.

디지털인증서는 암호화하여 HyperCerts 시스템에 저장한다. 암호화된 디지털인증서와 OTP를 결합하여 하나의 파일로 학생에게 제공된다. 이것은 오직 처음에 제출받은 미래 고용주만이 한번 복호화할 수 있는 특성을 부여한다. 또한 4장에서 설명한 것처럼 정해진 횟수만큼 복호화하거나 최초 복호화를 요청한 컴퓨터의 IP 또는 IP 대역을 기록하여 해당 IP나 IP 대역은 반복적으로 복호화할 수도 있을 것이다. 또는 특정 기간에만 복호화할 수 있도록 할 수 있다. 외부로 개인정보를 전달할 때 암호화하는 것은 키를 가진 사람만 복호화할 수 있도록 하는데 예를 들어 개인정보보호법에서는 보안서버를 이용하여 개인정보를 송수신하게 하여 정당한(즉 키를 가진) 통신 상대방만 이를 복호화하고 도청자는 키가 없어서 복호화할 수 없도록 하고 있다. 즉 암호화를 통해 접근제어를 하고 있다. 개인정보를 수신하여 복호화하는 것은 개인정보 주체인 경우 개인정보 주체가 복호화된 정보를 다른 제3자에게 제공하는 것은 개인정보 주체의 권리이므로 개인정보 주체에 대한 접근제어나 통제는 필요하지 않다. 하지만 디지털 졸업장의 경우는 개인정보 주체가 복호화하는 것이 아니라 미래의 고용주일 것이다. 이 문제는 제3자에게 제공된 개인정보 또는 정보의 기밀성를 유지하는 문제로 귀결된다. 이는 동의 받지 않은 제3자 제공제한 등 법제도상으로 해결할 수 있다. 이에 더해 본 논문에서는 개인정보보호법상의 안전성 확보조치에서 접근 권한 최소화의 원칙에 따라 접근권한이 있는 사람을 최소화하기 위해서 최초 디지털 인증서를 받은 사람이 한 번 복호화하거나 해당 IP 또는 해당 IP 대역만 반복 복호화할 수 있도록 접근 권한을 최소화하도록 시스템 상으로 구현하고자 하였다. 또 다른 측면에서 보면 본 논문은 복호화 횟수 제한, 복호화 IP 또는 IP 대역, 복호화 기간 등으로 접근제한을 시스템 상으로 구현하여 디지털인증서가 유출된 경우 무분별하게 복호화되는 것을 막도록 하고자 하였다. 단 복호화된 파일은 안전하게 보호된다는 가정을 하였다. 복호화된 파일의 download되어 다수의 제3자에게 제공되는 것은 Viewer 응용프로그램이 다운로드 기능을 제공하지 않거나 DRM을 적용하든 많은 현실적인 방법이 있을 것이다.

블록체인 기반 디지털 인증서 시스템은 기존 [1][3]과 달리 private blockchain인 hyperledger기반으로 구현하여 인증된 신뢰할 수 있는 개체만 블록체인 네트워크에 참여할 수 있도록 하였다. public blockchain에 누구나 발급자로 참여할 수 있는 문제를 효과적으로 해결하였다. 다만 발급자의 신원은 오프라인으로 확인할 필요가 있을 것이다.

[1]의 경우 디지털 졸업장을 공개키 암호화를 이용하여 암호화하고 블록체인에 저장하였다. 공개키 암호화는 키 전달 문제를 효과적으로 해결할 수 있으나 큰 문서를 암복호화하는 데는 비효율적이다. 본 시스템에서 기밀성을 위하여 키를 Application 서버에 저장만 하고 전달하지 않는다. 따라서 키 전달 문제가 없고 암복호화의 효율성을 높이기 위해서 대칭키 방식으로 암호화하여 기밀성을 제공하고 있다. 또한 [1]의 경우 공개키로 암호화하여 누구나 접근할 수 있는 분산원장에 저장하고 있다. 이 경우 해당 공개키의 보안 강도가 30년간 안전하다면 30년 후에는 해당 개인키를 획득한 사람은 누구나 접근할 수 있는 분산원장에서 공개키로 암호화된 졸업장을 획득하고

Fig. 12. Digital Certification Verification Monitoring

복호화할 수 있을 것이다. 본 논문은 이와 달리 분산원장에는 복호화할 수 없는 일방향 hash값만 저장하여 hash값으로부터 원본 문서를 획득하지 못하도록 하고 있다.

본 논문에서 사용하고 있는 Practical Byzantine Fault Tolerance 합의 알고리즘의 경우 3/4이상의 orderer가 찬성하면 해당 내용을 분산원장에 기록을 하도록 하고 있다. 따라서, 3/4 이상의orderer가 공모를 하면 당연히 잘못된 내용을 분산원장에 기록할 수 있을 것이다. 3/4 이상의 orderer가 내부 공모를 한다면 해당 시스템은 tamper-free 기능을 상실할 것이다.

인증은 Hyperledger의 Membership 서버스를 이용하였다. 즉 모든 개체는 인증서를 가지고 인증된 경우만 Hyperledger Network에 접근할 수 있다. 이를 위해서 개체는 enrollment를 해야 하면 디지털 졸업장 블록체인 시스템 운영자는 offline으로 개체의 신원을 확인할 수 도 있다. 또한 hyperledger의 분산원장에 대한 모든 접근을 개체의 id와 함께 transaction log라 불리는 로그에 기록되므로 향후 문제가 발생하였을 때 추적성을 가지고 있으며 HyperCert에서 개인별로 자신의 디지털 인증서의 복호화 기록을 Fig. 10과 같이 개인정보 주체가 확인할 수 있도록 하였다.

5.2 기존 시스템과의 비교

Table 1은 기존 방식과 제안 방식의 차이점을 표로 정리하였다.

5.3 논의

누가 블록체인 네트워크에 참여하고 관리할 지에 대해서 [2]는 신뢰할 수 있는 제3자가 블록체인을 관리할 것을 주장하였다. [6]에서 아직 학교가 블록

Table 1. Comparision between Existing Systems and Proposed System

	BlockCert	ECBC[1]	제안시스템
blockchain framework	bitcoin	Ethereum	Hyperledger
issuer authenticity	public blockchain, so any one can issue the certificate	public blockchain, so any one can issue the certificate	private blockchain, so identified entity can issue the certificate
integrity of certificate	supported	supported	supported
consensus algorithm	PoW which requires a lot computing power	PoW which requires a lot computing power	PBFT which does not require a lot computing power
storage	saving the hash of certificate	encrypted certificate so it requires a lot of storage	saving the hash of certificate
privacy	no control of submitted certificate	only who has the private key, but it is not clear who has the private key	control over the submitted certificate with hash-and-OTP

체인 네트워크를 관리해야 하는지 신뢰할 수 있는 제3자가 관리하는 것이 맞는지 아직 명확하지 않다고 적시하고 있다. 본 논문에서는 HyperCerts off-blockchain storage (student_table, OTP_table, ENC_Cert file)를 신뢰할 수 있는 제3자가 관리하고 on-blockchain storage(분산원장 및 World State DB)는 학교에서 관리하도록 하였다. 즉 off-blockchain storage와

on-blockchain 스토리지를 관리하는 주체를 분리하여 보안을 향상 시켰다.

본 논문에서는 디지털 인증서 발급과정에서 학생이 학교에서 로그인하고 로그인 결과를 HyperCerts 시스템으로 redirect하였다. 이를 좀 더 효율적으로 관리하기 위해서 OAuth와 같은 인증/인가방법을 이용할 수 있을 것이다. 본 논문에서 간단한 방법을 사용하였지만 향후 많은 blockchain application이 OAuth와 결합되어야 할 것으로 예상된다. 이는 향후 주제로 남겨 놓는다.

VI. 결 론

본 논문은 private blockchain인 hyperledger기반의 디지털인증서 시스템, HyperCerts를 제안하였다. HyperCerts는 최초의 private blockchain 기반 디지털인증서 시스템이다. private blockchain을 이용함으로써 신뢰할 수 있는 인증된 참여자만 디지털 인증서를 발급할 수 있도록 하였다. 또한, 디지털인증서는 개인정보이므로 개인정보를 보호 및 분산 원장 저장 용량의 효율적 사용을 위하여 디지털인증서의 해쉬값을 분산 원장에 저장하고 디지털 인증서가 유출되었을 때 부분별한 복호화를 방지하기 위해서 OTP를 결합한 hash-and-OTP 방식을 제안하였다.

References

[1] Yuqin Xu, Shangli Zhao, Lanju Kong, and Qingzhong Li, "ECBC: A High Performance Educational Certificate Blockchain with Efficient Query," International Colloquium on Theoretical Aspects of Computing 2017, LNCS 10580, pp. 288-304, Sept. 2017.

[2] G. Zyskind, O. Nathan and A. Pentland, "Decentralizing Privacy: Using Blockchain to Protect Personal Data," Proceedings of the 2015 IEEE Symposium on Security and Privacy, pp. 180-184, May 2015.

[3] MIT Media Lab, Digital Certificates Project, http://certificates.media.mit.edu, Aug. 2018.

[4] Hyperledger Fabric, https://hyperledger-fabric.readthedocs.io/en/release, Aug. 2018.

[5] M. Jones, J. Bradley, and N. Sakimura, "JSON signature," RFC 7515, May 2015.

[6] John Rooksby and Kristiyan Dimitrov, "Trustless Education? A Blockchain System for University Grades," June 2017.

〈저자소개〉

정 승 욱 (Seung Wook Jung) 정회원
1998년 2월: 숭실대학교 전자공학과 졸업
2000년 2월: 숭실대학교 전자공학과 석사
2005년 12월: University of Siegen, 전자정보공학박사
〈관심분야〉 개인정보, 암호학, 네트워크 보안, IoT보안, 블록체인