산업융합연구 제21권 제5호 pp. 133-142, 2023
ISSN 2635-8875 / e-ISSN 2672-0124
DOI : https://doi.org/10.22678/JIC.2023.21.5.133

대학 학력 검증을 위한 E-AV 모델 설계와 구현 방법에 관한 연구

박중오*
성결대학교 파이데이아학부 조교수

A Study on the Design and Implementation of E-AV Models for University Academic Qualification Verification

Jung-Oh Park*
Assistant Professor, Division of Paideia, Sungkyul University

요 약 최근 학력 위변조 문제는 단순히 교육기관에서 생성된 증명서 조작, 재학 중에 수료를 졸업으로 속여 기록하는 등 자동화된 관계 검증과 검증 자체를 신뢰하기 어려운 문제가 존재한다. 이는 대학 기관들의 학력 데이터베이스 공유 부재와 학력 추적이 어려운 독립된 시스템을 구축/운영하고 있기 때문이다. 본 연구는 대학 기관을 중심으로 학력 검증을 위한 E-AV 모델을 설계 및 구현한다. 기존 학력에 대한 연계 정보들을 요약하여 암호화된 데이터베이스로 저장하고, 기존 시스템의 호환성과 확장성을 고려하여 웹 표준 기술로 구현했다. 샘플 데이터 검증 결과 위변조에 안전성을 개선하고, 저장 공간과 실행성능의 준수한다는 것을 확인했다. 본 연구는 향후 국내 대학 교육기관 학력 관리 등 온라인 검증 서비스 개선에 이바지하고자 한다.

키워드 : 학력, 데이터 공유, 데이터 표준, 교육기관, 정보 검증

Abstract In recent years, the problem of academic credential falsification is not simply the manipulation of certificates generated by educational institutions, but also the difficulty of trusting automated relationship verification and verification itself, such as falsely recording completion as graduation while still in school. This is due to the lack of sharing of educational background databases among university institutions and the establishment/operation of independent systems that make it difficult to track educational backgrounds. This study designs and implements an E-AV model for academic credential verification centered on university institutions. It summarizes and stores the linked information on the existing academic background in an encrypted database and implements it with web standard technology considering the compatibility and scalability of the existing system. The results of sample data verification show that it improves safety against forgery and complies with storage space and execution performance. This study aims to contribute to the improvement of online verification services such as academic records management in domestic universities.

Key Words : Academic background, Sharing the information, Data standards, Educational institutions, Information verification

*Corresponding Author : Jung-Oh Park(pjo21@naver.com)
Received March 28, 2023
Revised April 24, 2023
Accepted May 20, 2023
Published May 28, 2023

1. 서론

2022년 전국 대학 수(일반/전문/교육/산업)는 336개 중 일반 대학 학과는 12,862개, 전문 대학은 6,343개 등 세부 학과 분류가 약 49,630개이다[1-2]. 최근 부정 취업, 입학, 승진 등 경력 및 학력 위변조가 사회적인 문제가 되고 있다. 형법 225조(공문서 등의 위조·변조)에 따르면 위변조에 대해 10년 이하의 징역을 받을 수 있다[3]. 대학의 독립적인 교육 정책과 수업 로드맵 등 대학 학력은 단일 증명서만으로는 재직/졸업 등 이외의 세부 학력(수업/출석 등)의 검증이 어렵다. 다양한 검증을 위해서는 대학 학력을 저장하는 표준 데이터베이스 모델이 요구된다. 즉, 학력의 세부 정보를 모두 살펴보려면 복잡한 행정절차와 사실 여부를 파악에 시간이 크게 지연될 수밖에 없다.

본 논문은 대학 기관을 대상으로 학력 위변조에 안전한 E-AV(Education-Authentication/Verification) 모델을 설계 및 구현한다. 본 논문의 구성은 다음과 같다. 2장 관련 연구는 학력 검증의 문제점과 모델 구현에 필요한 표준 웹 기술을 분석한다. 3장은 제안하는 E-AV의 요구사항 분석과 동작 과정을 설명한다. 4장은. 모델 검증을 위한 구현/성능 분석, 5장은 결론으로 마친다.

2. 관련 연구

2.1 기존 시스템의 학력 검증의 문제점

사용자의 이력을 정확하게 증명하기 위해서는 학력 증명 서류 확인뿐만 아니라 학적 문의, 동기생 확인, 전공 능력 등 학력에 기재된 모든 정보가 일치해야 신뢰성을 보장할 수 있다. 즉, 검증에 필요한 정보의 범위가 명확하지 않으면 학력 추적과 검증이 어려워질 수 있다.

2.1.1 부분 허위 이력

학력 관련 정보 10개 항목에서 9개가 정상이고, 1개가 일치하지 않는 경우 학력 위조 또는 허위 이력에 대한 여부를 판단이 애매모호해질 수 있다. 학력 허위 기재의 예로 정상적인 학교 출신이지만, 학력에 전공과 관련 없는 추가적인 내용들을 허위로 기재하는 경우이다. Table 1은 학력 검증에 필요한 일반적인 정보의 예를 나타낸다.

Table 1. Verification information example

List	Verification	Possibility of forgery
Graduation/transcript	Certificate	All possible
Contact department	Direct inquiry	Impossible
Acquaintance Verification	Photo/Talk	All possible
Relevance of detailed major	Professional Knowledge	Impossible
Graduation Work/Thesis	Direct inquiry	Some available
Credit completion/ non-regular, etc.	Direct inquiry	Impossible

2.1.2 위변조 가능성

학교 졸업 여부는 증명서와 학력 조회로 충분히 검증할 수 있다. 문제는 일반적으로 증명서 수준에서 학력을 검증할 수밖에 없다는 점이다. 최근 각종 공·사문서(박사학위 증명서, 대학 졸업증명서, 성적증명서, 어학성적서, 가족관계증명서 등)를 의뢰받아 위조해주는 업체의 90명 등이 불구속 입건되었다[4]. 온라인 증명서와 출력할 수 있는 문서 파일은 짧은 시간 안에 정교한 수준으로 위조할 수 있다. 학력 문제가 입사 및 채용에서 발견된다면 큰 문제가 안 되지만 퇴직 이후 시간이 지난 기관의 명예 실추, 위조에 대한 법적인 책임을 묻기가 어렵다. 오랜 시간이 지난 이후 학력 검증은 증명서와 학력 조회 외에 항목들은 검증이 거의 어렵다. 직접 사람을 통해 검증하는 일은 기억이 잘 안 나거나 허위 증명 가능성도 존재하기 때문에 신뢰성이 낮다.

관련 처벌 사례로 사문서위조 징역 8개월 집행유예 2년 등 실형받은 의사가 의사면허 취소 처리에 대해 면허 결격사유가 아니라고 법원에서 판정했다[5]. 벌금 수준 외에 강력한 처벌 정책이 존재하지 않는다. 형법 231조(사문서 등의 위조변조) 등은 최대 5년 이하 1천 ~ 3천만원 벌금형에 처하고 있다[6]. 즉, 문서 내용의 특정 정보만을 숨기거나 위변조하는 경우 처벌 수준이 비교적 약하다는 걸 알 수 있다. Table 2는 학력에서 세부 정보의 일부분(3번째)을 허위로 기재한 예를 나타낸다.

Table 2. Partial forgery example

Educational History	Information
University	Name, Department
Student	Name, Class number
Relevance of detailed major	Class information (false statements)
Graduation Work/Thesis	Topic, Team members, Roles
Credit completion	Detailed grades

2.1.3 연관분석의 어려움

학력에 전공과목과 상관없는 내용 또는 참여하지 않은 졸업작업/학위논문은 졸업자로서 졸업작품에 참여했는

지 학위논문이 실제 등록되었는지 정확한 검증이 어렵다. 즉, 결론적으로 학력 검증은 정확한 검증을 위해 관련된 표준 데이터 모델과 함께 정보에 대한 연관분석이 필요하다. 또한 학력을 구성하는 모든 정보가 위변조에 안전해야 한다.

2.2 대학 기관 시스템 및 기능 현황 분석

학력 검증은 기본적으로 정보가 잘 일치하는지 확인하는 작업이다. 데이터 입력/수정/조회/삭제 등은 자체 구축하고 있는 웹 환경에서 제어하는 것이 일반적이다. Table 3은 국내 10개 대학교(CWUR)[7]가 운영하는 홈페이지의 웹 프로그래밍 언어를 직접 확인한 결과이다.

Table 3. University development language(web)

University	Homepage	Language
SEOUL NATIONAL	https://www.snu.ac.kr	HTML, JQuery, JS, JSP
KAIST	https://www.kaist.ac.kr	HTML, JQuery, JS
Korea, KU	https://www.korea.ac.kr	HTML, JQuery, JS, JSP
Yonsei	https://www.yonsei.ac.kr	HTML, JQuery, JS
POSTECH	https://www.postech.ac.kr	HTML, JQuery, JS
Sungkyunkwan	https://www.skku.edu	HTML, JQuery, JS, JSP
HANYANG	https://www.hanyang.ac.kr/	HTML, JQuery, JS, JSP
UNIST	https://unist-kor.unist.ac.kr	HTML, JQuery, JS
Kyung Hee	https://www.khu.ac.kr	HTML, JQuery, JS, JSP
GIST	https://www.gist.ac.kr/kr	HTML, JQuery, JS, JSP

웹 표준 기술은 기본 HTML, JS(JavaScript)와 서버 내부 애플리케이션 처리를 위한 서버 프로그래밍 언어를 사용할 수 있다. 분석 결과 HTML, JQuery, JS를 기본으로 대부분 JAVA 기반 스프링(Spring) 프레임워크[8]를 적용했다. 주요 웹 개발 언어가 JAVA 언어임을 알 수 있다. 국내 홈페이지는 분야별로 전자정부 표준 프레임워크[9]의 사용 비중이 높다. 웹상에서의 대부분 입력 창이 HTML5 기반으로 JAVA와 JS를 구현하고 있다. Table 4는 웹 로그인 화면의 헤더(Header) 값을 참조하여, 직접 인증 기능을 확인한 결과를 나타낸다.

Table 4. University user authentication function(web)

University	Function	etc
SEOUL NATIONAL	Basic, Social login, Certificate	Naver, Kakao, etc.
KAIST	Basic, Simple authentication, Iam2(AWS)	Mobile app, SSO
Korea, KU	Basic, Mobile phone	PASS(Certificate), SMS
Yonsei	Basic, NAVER Certificate	Naver, SSO
POSTECH	Basic, Smart app	Patterns, OTP, etc., SSO
Sungkyunkwan	Basic	Kingo ID Login
HANYANG	Basic	SSO
UNIST	Basic	Facebook etc.
Kyung Hee	Basic, Certificate	KSignCASE, SSO
GIST	Basic, Certificate	AnySignForPC, SSO

기본(Basic) 공통으로 아이디/패스워드 로그인을 지원하고, 다음으로 인증서(Certificate)와 모바일 인증(Mobile)의 비중이 크다. 이외 자체 개발/외부 연동에 SSO(Single Sign-On)[10]를 다수 지원했다. 주요 로그인 접근 경로는 학교 메인 홈페이지와 학과 홈페이지 로그인으로 구분되며, 로그인 이후 내부 포털 서비스로 연계되는 형태이다.

표준 기술 현황 분석 결과 웹브라우저상에서 안전한 통신에 HTTPS(Hypertext Transfer Protocol Secure) 프로토콜[11]을 사용하고, 웹사이트 내부 접근에 사용자 인증이 필수이다. HTTPS 보안 프로토콜은 데이터의 입력/저장에 RSA[12], AES[13]등 암호화 알고리즘을 사용한다. 기존 대학의 시스템을 고려하면 학력 검증 모델의 기능을 구현하는데, 프론트는 JS 언어, 백엔드는 자바 웹 어플리케이션이 호환성과 확장성에 적절한 것으로 분석된다. 그러나 JS는 내부 웹 애플리케이션 보다 해킹에 대한 위험성이 존재한다[14]. 실제 프로젝트 진행에는 S/W에 대한 소스코드 압축(Compress) 및 난독화(Binary obfuscation) 등 보안 강화가 필요한 것으로 분석된다.

3. E-AV 모델 정의 및 동작

3.1 표준 데이터 모델 요구사항 분석

학력을 검증할 수 있는 공통 정보를 대상으로 필요한 기본 데이터 모델을 추출한다. 대학교 내 학력은 데이터

Table 5. Conceptual data model example

LIST	Identifier	Other
University	Name, Code	Address, Contact
Department	Name, Code	Contact
Class	Name, Code, Professor	Classroom, time
Student	Name, Resident Registration Number, Student ID	Address, Contact
Professor	Name, Resident Registration Number, ID	
Assistant	Name, Resident Registration Number, ID	
Attendance	Name, ID,	Tardy/Absent
Graduation work	Name, ID	File, Time
Paper	Name, Time	Email
Grade	Document identification number, Time	Details
Attending		
Graduated		

베이스 등 로컬, 서버에 분산되어 저장된다. 즉, 기존 시스템에 분산되어 저장된 학력을 새로 조합하여 연계/설계할 필요가 있다. Table 5는 검증에 필요한 학력의 개념적 데이터 모델의 예를 나타낸다.

일반적으로 대학 기관에서 저장 보관하는 학력은 '학과와 학생 등 공통 정보들이 다수 존재한다. 단일 및 그룹별로 고유 식별 정보(Identifier)를 추출할 수 있다. 학생의 경우 '이름, 주민등록번호, 학번' 직원은 '이름, 주민등록번호, 교번' 등으로 식별된다. 이외 기관 내 시스템마다 기타 정보(Other)들을 각자 다르게 보유할 수 있다. Table 6은 각 학력의 상·하위 방향으로 연계된 데이터의 예를 나타낸다.

Table 6. Reciprocal data linkage example

List	Relation	Input
University	–	–
Department	University	System
Class	Department	System
Student	Attendance	Student
Professor	Attendance	System
Assistant	Department	System
Attendance	Class	Student
Graduation work	Professor, Student	Student
Paper		
Grade	Student	System
Attending		
Graduated		

예로 성적의 경우 '학생', '출석', '수업', '학과', '학교' 순서로 연계된다. 즉, 분산된 학력 항목들은 각각 연계된 식별자를 1개 이상 가짐으로써, 학력 전체를 통합/연계된 모델로 구축할 수 있다. 특정 항목은 시스템(System) 내부가 아닌 사용자(Student)에 의해서 입력되기 때문에, 초기 기본(default) 설정과 함께 데이터 관리에 엄격한 보안 정책이 설정되어야 한다. Table 7은 기존 시스템에 필요한 학력에 대한 안전한 저장/제어 방법의 예를 나타낸다.

Table 7. Data protection technology example

Function	Technology	Policy
Basic Communication	HTTPS	Activation Check(must)
data access	Mobile OTP	2 channel
	RSA algorithm	Session
		2048 bit
Data storage (protection)	AES algorithm	Session Key
		Separate Storage
		256 bit
Data forgery/ falsification	DSA algorithm	Quarterly Integration
Data integrity	SHA algorithm	256 bit

적용 기술(Technology)은 웹 환경에서 쉽게 구현할 수 있는 기술을 고려하였다. HTTPS 보안 프로토콜(모든 웹 브라우저 지원)을 기반으로 데이터 접근에 가장 범용적인 RSA 알고리즘을 활용(Mobile OTP는 2채널 인증 용도)한다. 데이터 저장에 AES 알고리즘, 위/변조 방지를 위해 DSA 전자 서명 알고리즘, 무결성 검사에 SHA 알고리즘(해시)을 추가 활용한다. Table 8은 직접 입력되는 데이터 모델의 기타 정책 예를 나타낸다.

Table 8. Data protection policy example

Function	Policy
Input data	Character/Script filtering
Time related	Time Limit
Login	OS, IP, etc.
Database	Data type, Length, etc.

데이터 모델에서 정의하는 명확한 학력 데이터가 저장되어야 한다. 모든 데이터 입력에 특수문자 조합이나, JS가 포함된 문자열 등을 검사해야 한다. 시간제한은 출석의 경우 주차 별 수업 시간 내에 체크가 가능하고, 이후 수정할 수 있는 권한은 교수만이 가능하다. 모든 입력/수정에 대한 시간 정보를 생성하고, 이외 접속 환경(OS:Operating System)이나 IP(Internet Protocol) 주소를 검증할 수 있다. Table 9는 암호화 및 서명의 키 분리 예를 나타낸다.

Table 9. Key usage(separation) example

Input	Function	Separation
System	Access	Professor, Private Key (RSA)
	Access	Assistant, Private key(RSA)
	Edit/Save	Server, Session Key(AES)
Student	Access	Student, Private Key(RSA)
	Edit/Save	Server, Session Key(AES)

사용자와 서버 사이 키의 사용이 구분된다. 기본 RSA는 인증/접근, AES는 학력을 수정/저장한다. 접근과 수정/저장의 주체에 따라 개인키와 세션 키를 사용에 차이가 있다. 예로 교수 또는 조교의 학력을 수정할 수 있지만 접근하는 개인키가 다르다. 다수 키는 실제 처리 과정에서 시스템 성능에 악영향을 끼칠 수 있다. 검증에 필요한 서명은 학년 마지막에 최종 서명을 추가하여 데이터베이스에 저장한다. 이후 조회 권한(읽기)과 외부 접근/조회에 대한 URL 페이지(링크)를 공개한다.

3.2 E-AV 데이터 모델 구조와 동작 과정

Fig. 1은 E-AV 모델의 전체 구조와 역할을 나타낸다. 전체 4계층은 앞서 요구사항 분석 결과, 필수 기능들을 JS 소스 코드 단위로 분리(프로젝트 내 클래스)한다. 하위 기능들은 메서드(함수)로 구현되었다.

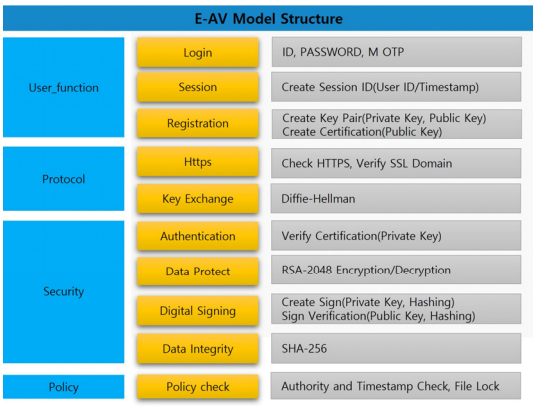

Fig. 1. Proposed E-AV model structure

① 기능(User_Function) : 사용자 로그인, 세션, 등록 등 기본적인 웹 확장 모듈로서 기능을 포함한다. 초기 등록과정에서 사용자의 개인키를 생성하고, 공개키를 교환해야 한다.

② 프로토콜(Protocol) : HTTPS 프로토콜의 검사와 함께 강제 수행하며, 암호 프로토콜을 수행하기 위한 키 교환 프로토콜을 포함한다.

③ 보안(Security) : 사용자 인증과 함께, 데이터의 암/복호화를 수행한다. 서명 생성/검증과 데이터 무결성 체크 과정을 포함한다.

④ 정책(Policy) : 주요 시간 설정, 파일 권한 설정 등 보안 정책을 포함한다.

Fig. 2는 E-AV 모델의 기능별 전체 동작 과정을 나타낸다.

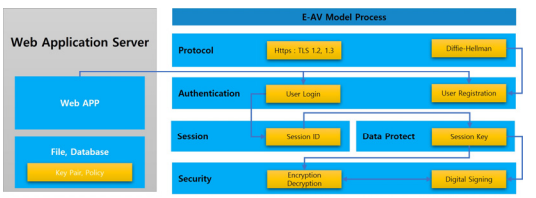

Fig. 2. E-AV model execution process

① 프로토콜(Protocol) : 기본 웹 브라우저에서 통신되는 모든 데이터는 Https 프로토콜이 강제되어야 한다. JS 스크립트 상에서 URL과 포트 번호를 직접 검사하고, 웹 서버는 Https 리디렉션(redirect)을 강제 수행한다. Https도 SSL 취약점을 통한 해킹 공격이 가능하므로, 브라우저와 서버 사이에서 지원하는 SSL-TLS의 버전을 검사(1.2 버전 이상)한다.

② 인증(Authentication) : 시스템 사용자는 로그인 이전에 반드시 등록해야 한다. RSA 알고리즘(2048)을 기반으로 두 소수에서 n = p * q, n을 구하고, $\phi(n) = (p - 1) * (q - 1)$를 구한다. $1 < e < \phi(n)$와 서로소 조건을 만족하는 e를 구하여 $(e * d) \bmod \phi(n) = 1$를 만족하는 개인키(n, d)를 생성한다. 개인키와 공개키(n, e)는 사용자의 로그인 인증에서 활용된다. 웹 서버 SSL 인증서는 VeriSign, Thawte, Geo Trust 등 공식기관에서 받은 인증서를 사용해야 안전하다.

③ 세션(Session) : 세션을 저장하는 저장소는 표준 웹 스토리지의(Web Storage)의 세션 저장소를 사용한다. 사용자가 서버에 정상 로그인하는 경우 시간 정보를 포함하는 고유 세션 ID(임의값)를 발급한다. XSS 등 취약점을 보완하기 위해, 저장되는 데이터의 크기와 문자열 저장방식을 제한한다. 데이터 암호화/복호화를 위한 세션 키를 생성해야 한다. AES-256 알고리즘 기반 패스워드 m(로그인에 입력된)으로부터 공유 대칭 키(K)를 생성한 후 새로운 세션 키(세션 ID, 공개키(n, e))) S를 생성한다. 세션 키는 현재 접속한 사용자의 검증하고, 전자 서명(Singing) 과정에서 활용된다.

④ 보안(Security) : 사용자 등록 이후 생성된 키 쌍을 통해 암호화 및 복호화를 수행한다. RSA 알고리즘 기반으로 공개키(n, e)로 암호화($C = (M \hat{} e) \bmod n$)하고, 개인키(n, d)로 복호화(검증)하여 사용자 인증($M = (C \hat{} d) \bmod n$)을 수행한다. 세션 키(S)는 내부 데이터를 암호화하고, 데이터의 추가/수정/삭제 등에 전자 서명을 생성한다. 세션 키(S)를 공개키(n, e)로 암호화하여 서명에 사용한다. 사용자의 개인키(n, d)로 세션 키(S)를 얻을 수 있다. 이외 전자 서명에 포함되는 무결성 검사용 해시값은 SHA 해시 알고리즘을 활용하여 생성한다.

⑤ 기타 – 보안 정책 : 초기 생성된 키 쌍은 회원가입 시에 생성된 후 보관 기간은 1개월, 3개월마다 패스워드와 함께 키 쌍을 자동 갱신한다. 무분별한 키 생성을 방지하기 위해 회원가입 요청 횟수를 제한하고, 가입 승인은 반드시 관리자를 통해 진행한다. 이외 세션 키는 현재 로그인 세션이 종료되면 사용할 수 없다. 이는 세션의 유효 시간을 지정하는 방식으로 구현한다.

데이터 모델에서 정의하는 학력 이외의 정보들을 새 데이터로 추가해야 하는 경우, 입력 값에 대한 JS 식별자인 ID 속성을 모델 내부 항목에 자동 추가하는 기능을 구현한다. 새로 추가된 항목의 경우 최소 1개 이상의 연관된 식별 정보를 입력해야 한다.

4. 구현 분석

4.1 개발환경 및 학력(샘플 데이터) 확인

Table 10은 웹 환경에서 세부 기능을 구현한 개발환경을 나타낸다.

Table 10. Development environment

–	S/W and Tech
OS	Linux 16.04 Kernel
Web Server	Apache 2
Programing Language	HTML5, JS, JAVA
Database	Mysql 5.7
External Library	OPENSSL, JSEncrypt Cipher Class(javax.)

개발 범위는 연구의 핵심인 학력(샘플 데이터)의 연계 검증 부분을 구현하고, 안전성을 테스트한다. 리눅스 16.04 커널에서 80 포트에서 동작하는 임시 웹 서버(Apache) 환경을 구축하였다. 내부 확장 모듈에는 RSA 기반 암호화/복호화를 위한 openssl, JSEncrypt 라이브러리, 자바 언어 내부 Cipher 클래스를 활용했다.

Fig. 3, 4는 openssl로 생성한 서버 RootCA와 클라이언트의 인증서를 나타낸다.

관련 연구 분석 결과, 안전하다고 알려진 서명을 위한 RSA 2,048비트와 무결성을 위한 SHA256 해시 알고리즘을 적용한다. 검증 전 선행 단계로써 초기 사용자 등록단계의 인증서를 생성한다. RootCA는 하위 사용자의 인증기관으로써 최상위 인증서가 된다. 1:1 관계의 인증서를 서버 내부에서 openssl 명령어를 통해 직접 생성했다. 클라이언트 인증서는 서버의 RootCA의 하위 체인 인증서이다. 즉, 최상위 기관의 구성에 따라 인증서 관리 체계가 달라질 수 있다. 예로 학교(1):학과(N): 학생(N)의 관계로 인증서 체인을 구성할 수 있다.

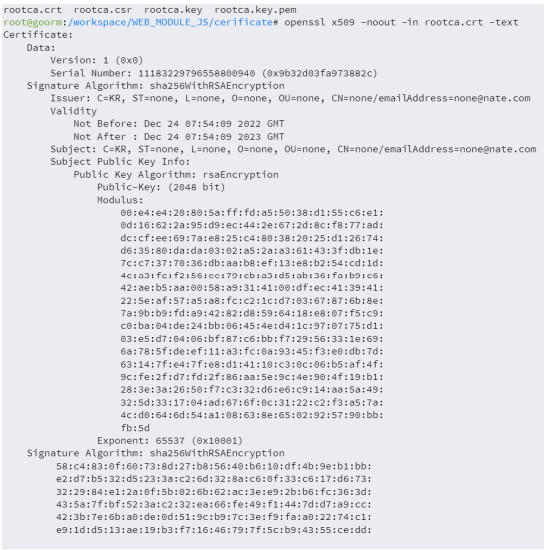

Fig. 3. Server – rootCA cerificate(crt)

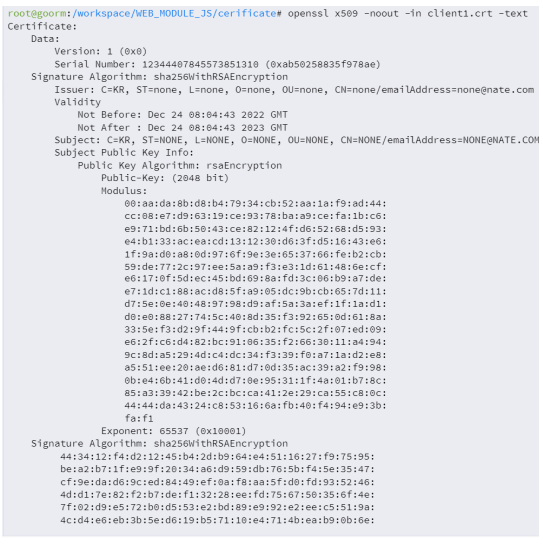

Fig. 4. Client – chained cerificate(crt)

Table 11은 E-AV 모델의 내부 데이터베이스 테이블 구조를 나타낸다. 구현 단계에서 학력 검증을 위해 사용자와 서버의 1:1 관계를 학력을 입력하여 샘플 데이터를

준비했다. 본 연구의 학력 검증 과정에서 필요한 필수 테이블로 구성되었다. 각 필드는 정수 타입과 가변형 varchar() 문자열 타입으로 초기화되었으며, 각 테이블에 관계를 연결하여 표현했다.

Table 11. E-AV database – table

TABLE NAME	COLUMN NAME	RELATION
EDU	UNI, GRAD	–
MEMBER	USER_ID, NAME, NUM	EDU
LECTURE	NUM, NAME, ROOM, TIME	MEMBER
ATTENDANCE	NUM, CHECK,. ROOM, TIME	LECTURE
SESSION	USER_ID, S_ID, S_KEY, TIME	MEMBER
DATA	ID, MEMBER, LECTURE, ATANDENCE	MEMBER, LECTURE, ATTANDENCE

Fig. 5는 데이터베이스에 입력한 GUEST1 사용자의 학력(임시)과 암호문 관련 키와 파일을 웹 페이지로 출력한 결과이다.

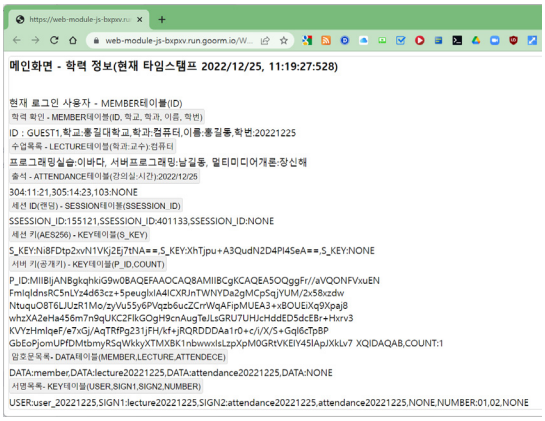

Fig. 5. Webpage print – sample data

수업 3개 중 2개 수업까지 출석한 상태이다. 수업 예로 '프로그래밍실습', '서버프로그래밍', '멀티미디어개론'은 학기 시작과 함께 시스템 내부에서 자동 입력되거나 사용자의 수강 신청에 따라서 입력된다. 시스템과 사용자 각각 다른 세션과 서명을 생성한다. 로그인 수행과 동시에 세션 ID와 함께 세션 키를 생성한다. 즉, 학력에 대한 암호화하는 세션 키의 경우 지속하여 갱신된다.

수업과 연계된 출석(attendance)의 예로 출석 체크 유무에 따라 서명 목록을 생성한다. 3번째 수업의 경우 출석이 존재하지 않는 상태이다. 세션 ID 및 서명 목록 그리고 암호문이 NONE 출력된다. 서명의 경우 서버 공개키 정책에 따라 갱신 주기가 달라질 수 있다. 현재 공개키는 아직 갱신된 적이 없으므로, COUNT 부분은 1로 설정되었다.

4.2 학력 검증

Fig. 6은 E-AV 모델의 입력된 사용자의 학력(샘플 데이터)을 검증하는 방법을 나타낸다.

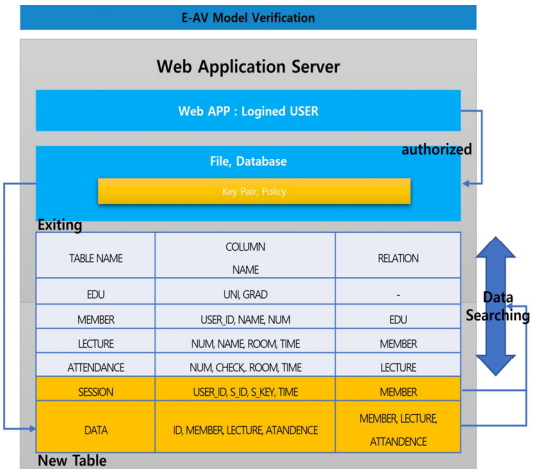

Fig. 6. E-AV model – verification

① 권한 인가(authorized) : 사용자는 웹브라우저상에서 교내 웹사이트에 로그인 인증 후 내부 데이터베이스에 대한 접근 권한을 부여받는다. 이후 학력 조회(일반적으로 시간 범위)에 해당하는 학력의 Session 테이블과 DATA 테이블에 접근하게 된다.

② 테이블(Exiting/New) : Exiting 영역은 기존 대학 내에 보유하고 있는 분산된 학력 데이터베이스를 의미한다. New 영역은 E-AV 모델에서 지속하여 저장한 학력을 의미한다. 기존 시스템은 학력을 새로운 데이터베이스로 생성하지 않고, 통합할 수 있다. 그러나, 이 부분은 보안에 문제가 생길 수 있기 때문에, 실제 데이터베이스는 두 영역을 분리하는 방식이 적절하다.

③ 검색(Data Search) : New 테이블 영역에서 참조하고 있는 다수의 Exiting 테이블들을 검색하는 과정이다. 현재 로그인된 사용자의 고유 식별자, 세션 값 확인 후 조회 범위에 해당하는 암호문을 찾아 복호화하고 내용을 출력한다. 모든 데이터를 대상으

로 복호화하는 경우 웹 서버 성능에 영향을 끼칠 수 있다. 학년 단위로 정보를 조회하도록 제한할 수 있다. 또는 서명(공개키) 부분의 학력 검증 페이지를 분리하여 구현할 수 있다.

Fig. 7, 8은 GUEST1 사용자의 학력 검증 결과를 웹 페이지로 출력한 결과이다.

Fig. 7. Webpage – verification

Fig. 8. Webpage – verification

GUEST1 사용자에 대하여 '서버프로그래밍' 수업에 대한 학력을 조회하고, 암호화된 학력 부분을 복호화하여 출력했다. 기본적으로 수업, 출석, 암호화 서명에 대한 유무와 보안 키를 확인했다. 이후 조회 결과 각 관련 세부정보와 연계 정보를 출력한다. 이오 서버 내부에서 데이터 제어에 암호 알고리즘을 사용하기 때문에, 전체 조회의 경우 성능에 저하가 발생할 수 있다.

Table 12는 4년제 기준 학사의 암호문과 서명의 파일 개수와 크기를 나타낸다.

Table 12. E-AV database – file and size

Number of files		1 year (1 semester), more than 150
Size	Ciphertext	At least 100 bytes or more
	Sign	256 bytes fixed

일반 학사학위가 130~150학점 사이며 재학 중에 3학점 수업 기준 약 40~50개 이하 수업이 진행된다. 파일 개수는 한 학년(2개 학기, 주차:15주)에 약 150개이다. 4년 동안 파일 개수를 예측하면 최소 600개 이상이 된다. 본 연구 E-AV 모델의 암호문은 이름, 고유 ID, 시간 등 100byte 이하의 작은 문자열에 대한 암호문과 고정 크기의 256byte의 서명을 생성한다. 사용자의 수업 질문, 게시판 업로드, 수업 평가 등 서술 입력 등의 경우 암호문의 크기가 약간 증가할 수 있다.

4.3 성능 평가

사용자 1인 기준 파일의 전체 크기(600개인 경우)는 암호문이 100byte × 600개로 약 60KB, 256byte 서명은 약 125KB다. 국내 대학생 수(대학 재학생 수 통계)가 가장 많은 인원이 약 2만 8천 명이다[15]. 3만 명을 예로 암호문과 서명을 합한 185KB × 30,000명의 필요한 서버 저장 공간은 4학년 동안 전체 약 5,550Mbyte이다. 구현 및 운영 단계에서 기능이 추가되어 파일 개수가 10배(6,000개)로 증가해도 약 55.5GB로 이다. 즉, 서버에 저장소 공간에 크게 영향을 끼치지 않는다. 파일 개수가 증가하는 경우 서버 성능에 영향을 주는 큰 요인은 처리(암/복호화) 연산이다. Table 13은 JSEncrypt에서 파일 개수에 따른 암호화/복호화 속도(ms : millisecond)를 비교한 결과(최소/평균/최대) 이다.

Table 13. E-AV database – performance(ms)

Count(Change)	Session(all)	SIGN(1)
600	402/775/1120	401/514/634
1200	619/653/860	651/752/809
2400	1070/1130/2540	826/923/881
4800	1098/2050/2210	1626/1940/2280
6000	2056/2560/5210	2236/2664/2960

세션 구현에 사용된 AES 암호는 기존 웹 환경에서 범용적으로 활용되는 표준 알고리즘이다. 본 연구는 세션 구현의 경우 기존 AES 암호를 사용하고, 서명(SIGN) 검증의 경우 보안이 강화된 RSA 암호를 추가 사용했다. 기존 시스템과 차이점은 서명 검증에 추가 지연이 요구된다는 점이다.

curl 명령어를 통해 확인 결과, 단일 웹 페이지의 요청에서 응답 속도까지 최소 150(ms), 최대 180(ms), 평균 341(ms)초이다. 사용자 체감 속도는 평균 0.2초이다. 세션 키의 경우 AES256 인코딩/디코딩 함수 수행 결과 평

균 775ms에서 2560ms로 나타났다. 4년 이내 학력 전체를 단순 조회하는 경우 암/복호화로 인해 약 2~3초 시간이 지연됐다. 시간 분석 결과 최소부터 최대까지 시간이 불규칙하게 분포되었는데, 다른 세션 키(세션 키 갱신)를 사용하기 때문이다.

서명의 경우 RSA2048 기준으로 인코딩/디코딩 함수 수행 결과 평균 514ms에서 2664ms로 나타났다. 조회 이외 검증 기능으로 약 3초 정도 시간이 추가 지연됐다. 서명은 정상 사용자 확인 용도로 마지막 저장된 최종 서명 값을 대상으로 단일 검증을 수행했다. 앞서 AES256과 다른 부분은 RSA 기반 키 쌍의 경우 설정된 시간 동안 지속되기 때문에 같은 키로 암/복호화를 수행하게 된다. 즉, 복호화에 같은 키를 사용하기 때문에 서명 검증 횟수가 증가해도 성능의 오버헤드가 증가하지 않음을 확인했다.

5. 결론

기존 대학 기관에서 운영하는 시스템에서 데이터베이스 질의(Query)를 수정 후 출력 기능의 구현은 어렵지 않다. 문제는 단순 증명서뿐만 아니라 시스템 내부에 저장되는 모든 정보는 위변조에 안전하지 않다. 본 연구의 E-AV 모델은 기존 표준 데이터 모델의 부재를 해결하기 위해 알려진 학력 관련 데이터베이스를 새로 생성하였다. 또한 학력의 안전한 저장을 위해 기존 Https 등과 다른 독립적인 채널에서 보안 프로토콜을 수행하여 학력 조회/검증에 대한 관계 검증의 신뢰성을 제공했다. 또한 기존 시스템 환경을 고려하여 설계했기 때문에, 추가 확장 모듈로써 쉽게 적용할 수 있다.

RSA 알고리즘은 HTTPS, 공인인증서 등 다양한 분야에서 알려진 대표 기술이다. E-AV 데이터 모델을 도입하는 경우, 자체 서버에서 발급된 키 쌍과 인증서는 안전하게 생성/저장/관리 되어야 한다. 구현 방법에 따라서 대학 기관, 학과, 수업 수준으로 인증서 키 체인을 추가하여 관리할 수 있다. 본 연구에서 제안한 E-AV 모델은 새로운 학기가 시작되는 시점에서 도입/적용하는 경우 새로운 저장 데이터 로그에 대한 리포트 작성과 이를 승인하는 행정절차가 추가 구현될 수 있다. 핵심은 내부 암호화 기술을 통해 '누가 언제 학력을 재수정했는가?' 모든 과정을 신뢰 검증 가능하다는데에 있다. 향후 연구로는 대학 시스템 간에 P2P(Peer to Peer) 네트워크를 기반으로 학력 블록체인을 생성하여 공유/적용하는 연구로 발전시킬 계획이다.

REFERENCES

[1] Statistics Korea. (2022). Number of Universities: Number of Universities by Region, Retrieved from https://kosis.kr/

[2] KEDI. (2022). National University Department Status, Retrieved from https://www.kess.kedi.re.kr/

[3] Korea Ministry of Government Legislation. (2021). criminal law. Article 225 (Forgery or Alteration of Public Documents, etc.), Retrieved from https://www.law.go.kr/

[4] Choi, Y. L. (2022). "Forgery of public and private documents is rampant in employment, admission, and promotion"… Arrest of 90 people, Retrieved from https://www.hani.co.kr/

[5] Lee, J. H. (2022). Doctor sentenced to prison for forgery of private documents… "Illegal Disposition of License Cancellation", Retrieved from https://www.doctorsnews.co.kr/

[6] Korea Ministry of Government Legislation. (2021). criminal law. Article 231 (Forgery or Alteration of Private Documents, etc.), Retrieved from https://www.law.go.kr/

[7] CWUR. World University Rankings 2022-23, Retrieved from https://cwur.org/

[8] SPRING. (2022). Spring Framework, Retrieved from https://spring.io/

[9] Choi, J. W., Koh, S. J. (2022). Enhancement of e-Government Standard Framework for Cloud Transformation in Public Sector Information System. Proceedings of Symposium of the Korean Institute of communications and Information Sciences, 606-608.

[10] Alaca, F., & Oorschot, P. C. V. (2020). Comparative analysis and framework evaluating web single sign-on systems. ACM Computing Surveys (CSUR), 53(5), 1-34.

[11] Alwazzeh, M., Karaman, S., & Shamma, M. N. (2020). Man in The Middle Attacks Against SSL/TLS: Mitigation and Defeat. Journal of Cyber Security and Mobility, 449-468.

[12] R. L. Rivest, A. Shamir, and L. Adleman. 1978. A

method for obtaining digital signatures and public-key cryptosystems. Commun. ACM 21, 2 (Feb. 1978), 120–126.
DOI : 10.1145/359340.359342

[13] Heron, S. (2009). Advanced encryption standard (AES). Network Security, 2009(12), 8-12.

[14] Van Ginkel, N., De Groef, W., Massacci, F., & Piessens, F. (2019). A server-side JavaScript security architecture for secure integration of third-party libraries. Security and Communication Networks, 2019.

[15] Ministry of Interior And Safety - Data Portal. (2022). Ministry of Education, Major Status by Department by University nationwide, Retrieved from https://www.data.go.kr/

박 중 오(Park, Jung Oh) [정회원]

• 2000년 7월 : 성결대학교 컴퓨터공학과 졸업
• 2003년 3월 : 명지대학교 전자계산교육 석사
• 2011년 8월 : 숭실대학교 컴퓨터공학 박사
• 2016년 3월~현재 : 성결대학교 조교수
• 관심분야 : PKI, Network security, Cryptography
• E-Mail : pjo21@naver.com