NEIS를 위한 PMI 기반의 RBAC 인증과 DB 보안 구현

유 두 규[†] · 문 봉 근[††] · 전 문 석[†††]

요 약

기존의 교육행정정보시스템(NEIS)은 보안 관리에 많은 문제점이 있다. 사용자 인증에서 PKI 인증만을 사용하여 권한에 따른 접근제어를 실현하지 못하고 있으며, 중앙 집중식 DBMS 운용과 평문화된 데이터의 사용으로 NEIS의 해킹 가능성을 증가시키고 있다. 따라서 데이터의 안전한 관리와 권한 인증을 위하여 새로운 교육행정정보시스템(RNEIS)을 제안한다. 첫 번째, 권한기반관리구조(PMI)의 속성인증서(AC)에 의한 접근 권한과 역할에 따른 사용자 인증 및 접근제어(RBAC)를 제시한다. 두 번째, DB의 해킹 방지를 위하여 전자서명에 의한 DB암호화를 설계 구현하였다. 세 번째, 일방향 해쉬함수에 의한 SQL 쿼리 위변조 방지 및 안전한 데이터 전송 방안을 제시한다.

Implementation of RBAC Certification & DB Security Based on PMI for NEIS

Du-Gyu Ryoo[†] · Bong-Keun Moon[††] · Moon-Seog Jun[†††]

ABSTRACT

The established NEIS has a lot of problems in the management of security. It does not realize access control in following authority because it only uses PKI certification in user certification and the use of central concentration DBMS and plain text are increased hacking possibility in NEIS. So, This paper suggests a new NEIS for the secure management of data and authority certification. First, we suggest the approached authority in AC of PMI and user certification in following the role, RBAC. Second, we realize DB encryption plan by digital signature for the purpose of preventing DB hacking. Third, we suggest SQL counterfeit prevention by one-way hash function and safe data transmission performed DB encryption by digital signature.

키워드 : 인증(Authentication), 전자서명(Digiatal Signatire), 속성 인증서(Attribute Certificate), 권한기반접근제어(Role Based Access Control), 교육행정정보시스템(National Education Information System)

1. 서 론

최근 비인가자에 의한 전산원장 부당 변경 및 조회, 비밀번호 변경, 고객 정보 유출에 의한 예금 부당 인출등 고도의 전문기술을 이용한 보안사고가 증가하고 있어, 국가적인 차원에서 인터넷 기반의 전자문서 관리 및 데이터베이스, 인터넷 뱅킹과 같은 인터넷 비니지니스에 관한 취약점을 보완하기 위한 기술적인 보완이 요구되고 있다[1]. 최근 교육행정정보시스템(NEIS)에 대한 정보 인권의 문제는 개인정보에 대한 문제를 교육 현장에까지 그 문제를 확산시키는 계기가 되었다.

접근제어의 목표는 비인가자 또는 통신시스템의 위협으로부터 응용프로그램 및 시스템을 보호하는 것이다. 기존의 접근제어 방식에서 많은 응용 서비스가 실행 중일 때 항상 보안관리가 만족스러운 것은 아니다. 다양한 인터넷 응용서비스를 만족시키기 위해서 새로운 접근제어 방식이 요구되고 있다.

최근 역할기반 접근제어(RBAC : Role Based Access Control) 방법에 대한 연구가 많이 이루어지고 있다. 기본적 개념은 개별적인 사용자보다 권한 또는 역할이 주어지는 접근권한이다. 사용자들은 서로 다른 권한에 따라 정보 시스템 내의 행위가 주어진다. RBAC는 사용자와 그룹이 사용하는 전통적인 접근법을 통하여 보안관리에 대하여 보이지 않는 유연성을 제공한다[2].

공개키 기반 구조(PKI : Public Key Infrastructure)에서 사용되는 공개키인증서(PKC : Public Key Certificate)는 사용자의 신원 확인을 위한 인증에 사용되며 사용자의 신원을 보증하는 수단으로 가장 효율적인 것으로 평가받고 있다. 그러나 공개키 인증서의 경우는 DBMS와 같이 다단계 권한이 필

※ 본 연구는 숭실대학교 교내연구비 지원으로 이루어짐.
† 정 회 원 : 숭실대학교 대학원 컴퓨터학과
†† 준 회 원 : 숭실대학교 대학원 컴퓨터학과
††† 종신회원 : 숭실대학교 정보과학대학 교수
논문접수 : 2004년 8월 5일, 심사완료 : 2004년 9월 1일

요한 시스템에서는 적용하기 곤란한 점이 존재한다[3-5].

접근제어를 위한 새로운 기술로 권한관리기반구조(PMI : Privilege Management Infrastructure)가 사용되어진다. PMI의 중요한 기능은 인증이 이루어진 이후에 보다 강화된 권한을 부여하는 것이다. PMI의 데이터 구조는 X.509 속성인증서(AC : Attribute Certificate)에 명세화 되어 있다. 공개키인증서가 인증서소유자의 공개키를 보증 해주는 것처럼 속성인증서도 속성인증서 소유자의 속성들에 대하여 속성기관이 보증한다. 그러나 PMI가 아직까지 널리 적용되고 있지 않다[6-7].

본 논문에서는 PKI, PMI의 구조와 RBAC 모델을 적용하여 AC에 저장되어 있는 권한정책과 역할에 의하여 접근제어가 이루어지도록 하는 것을 주요 개념으로 한다. 또한 X.509 PKCs와 X.509 ACs를 연결하여 인증이 이루어진 후 권한에 따른 서비스 이용이 이루어지도록 했다. 즉 공개키 인증서의 사용자인증과 전자서명의 기능을 확장하고 AC를 이용하여 응용서비스(DB)에 접근할 수 있는 권한과 자격을 부여하여 DBMS 관리자에 의한 사용자의 비밀정보가 노출되는 것을 방지하였다. 또한 Role Authority에 의하여 권한과 역할이 인증되는 다단계 및 그룹 속성을 이용하여 DB에 접근하고 데이터를 암호화 할 수 있는 프로토콜을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 적용된 기술의 동향과 구조를 살펴보고 3장에서 본 논문에서 제안한 교육행정정보시스템의 스키마에 대하여 설명하였다. 그리고 4장에서 구현을 통한 암호학적 성능평가와 결과를, 5장에서 본 논문의 결론을 맺는다.

2. 관련 연구

2.1 PMI의 구조

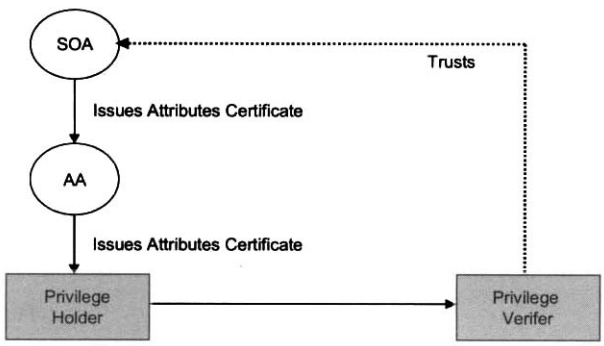
X.509 인증서의 확장 필드를 이용하면 기존의 공개키 기반 구조 시스템의 큰 변경 없이 인증, 권한을 부여하는 절차가 간소화 된다. 그러나 공개키인증서를 사용하게 되면 다음과 같은 문제가 발생한다.

첫째, 인증서 소유자의 자격과 권한이 변할 때 공개키 인증서를 폐지해서 재발행해야 한다. 예를 들어 반년에 한 번 인사이동이 있는 기업이나 조직의 경우를 보면 반 년 마다 인증서의 재발행이 필요하게 된다. 인증서 폐지 목록(CRL)도 커진다. 공개키 인증서의 유효기간은 통상 1년 이상이다.

둘째, 인증기관과 등록기관은 인증서 소유자가 갖고 있는 권한에 대해서는 관계하지 않는다. 권한을 관리하는 것은 인증기관과 별도의 부서이며 조직이다. 예를 들어 기업에서의 이용을 생각하면 사원인지를 확인하는 것은 인사부지만 각 사원에게 사내의 어느 정보를 이용할 수 있는지는 각 정보의 관리 책임자가 결정하는 것이다. 때문에 인증기관과 권한 부여자는 분리할 필요가 있다.

셋째, 공개키 인증서에 자격 등의 속성 정보를 넣으면 공적으로 이용하기가 어려워진다. 예를 들어 통신을 할 때에 본인인 것을 확인하기 위해서 공개키 인증서를 통신 상대에게 보내는 경우 공개키 인증서에 속성 정보가 기재되어 있으면 전할 필요가 없는 속성 정보까지 상대에게 전한다[8].

따라서 이러한 문제들을 해결하기 위해서는 사용자의 고유식별정보를 인증하는 공개키기반구조 이외에 권한, 역할 등의 속성에 대한 관계를 보증하는 별도의 인증구조가 필요하다. 권한관리 기반구조는 이와 같이 권한관련 자원과 소유자간의 관계를 인증기간이 인증하고 유지하는 구조를 말한다. (그림 1)은 PMI 구조를 나타낸다.



(그림 1) PMI 구조

공개키 인증서는 인증기관(CA : Certification Authority)에서 사용자에 대한 신원을 확인후 발급하는 반면, 속성인증서는 속성기관(AA : Attribute Authority)에서 발급한다. 사용자의 신원을 확인하기 위해서는 공개키인증서를 검증하고 사용자의 권한이나 역할을 확인하기 위해서는 속성인증서를 검증하면 된다. 이러한 검증과정에서 권한 검증자는 속성인증서와 공개키인증서를 연결하여 사용자가 정당한 권한을 가지고 있는지 판별하게 된다. 사용자는 여러 속성기관으로부터 다수의 속성 인증서를 가질 수 있다.

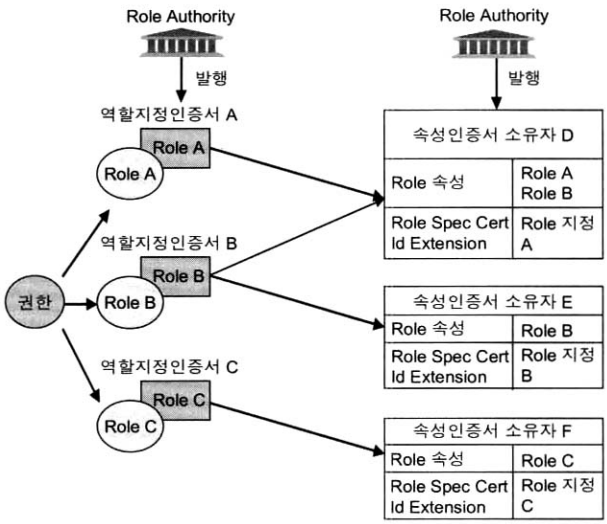
2.2 PMI 모델

ITU_T X.509 표준에서는 속성 인증서를 정보보호 메커니즘으로 활용할 수 있도록 일반 모델(General Model), 제어 모델(Control Model), 위임 모델(Delegation Model), 역할 모델(Roles Model)을 제시하고 있다. 여기서는 본 논문에 적용된 역할 모델에 관하여 분석한다.

2.2.1 역할 모델

역할(Roles) 모델은 사용자 개인에게 간접적으로 권한을 할당할 수 있는 기능을 제공해 준다. 이 모델에서는 개인들에게 직접 권한을 주지 않고 단지 속성 인증서의 Role 속성에 사용자에게 필요한 역할을 할당한다. Role 속성을 기술한 속성 인증서는 역할할당 인증서(role assignment certificate) 또는 역할지정 인증서(role specification certificate)라 한다.

역할지정 인증서는 공개키 인증서로는 제공할 수 없으며 속성인증서를 이용해야 한다[8].



(그림 2) 속성인증서와 역할지정인증서

이 역할지정 인증서는 권한(역할)부여기관(Role Authority)이 발행한다. (그림 2)는 속성인증서 소유자와 역할과의 관계를 나타낸다. 속성인증서 소유자 D는 역할지정인증서 A의 역할 Role A 와 역할지정인증서 B의 2가지 역할을 Role 속성 필드에 가지고 있다. 이 모델의 특징은 Role Based Access Control에서 큰 장점을 가진다. 교육행정정보 시스템에서 Role 속성은 학년 학급별 교과성적은 교과담당교사, 학생의 인적사항은 담임교사, 학생의 건강에 관한 사항은 담임교사, 보건교사 등으로 권한과 역할이 부여 될 수 있다. 또한 학생이나 학부모의 경우는 학교생활기록부의 조회만 가능한 역할이 주어질 수 있다. 여기서 다른 학생의 데이터는 볼 수 없도록 제한을 한다. 기존의 시스템은 인증프로세스의 문제로 학부모와 학생에게는 교육행정정보시스템의 사용이 제한되어 있다. 속성인증서는 위에서 기술한 바와 같이 기존시스템의 문제를 해결할 수 있다.

2.3 DBMS의 접근제어와 암호화

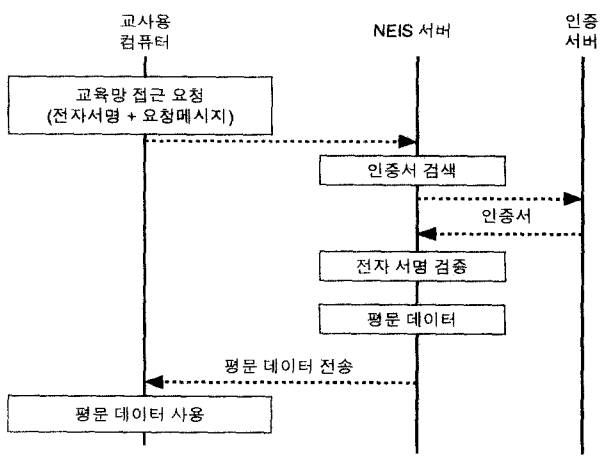
데이터베이스의 보안 침해 요소를 살펴보면 침해 공격, 유추 문제, 집합 문제 등을 들 수 있다. 이러한 보안 침해 요소로부터 데이터베이스를 보호하기 위하여 여러 가지 보안 요구사항이 고려되어 왔는데 기본적인 방법으로 시스템 감사, 사용자인증, 정당한 사용자의 데이터 접근 통제 등이 있다. 논리적 일관성 유지를 위한 요구사항으로는 데이터 무결성 유지, 데이터 연산의 무결성 유지 등이 있으며, 다양한 데이터베이스 응용들을 위한 강력한 보안 요구사항으로는 추론 방지, 기밀 데이터 관리 및 보호, 다단계 보호(Multilevel protection), 접근제한(Confinement) 등을 들 수 있다[9].

데이터베이스에 대한 부당접근 및 변경은 내부 관리자에 의하여 이루어지는 경우와 외부 네트워크를 통한 두 가지 경우가 있으나 어떤 경우든 중요정보가 누출되었을 때 심각한 문제를 야기하게 된다. 기존의 DBMS 시스템에 의한 데이터의 관리는 DBMS 엔진 자체적으로 이루어지는 권한 관리에 의하여 데이터에 대한 접근제어를 제한하였으나 이 또한 내부관리자에 의하여 정보가 누출되는 경우 심대한 영향을 초래하게 된다.

데이터베이스 암호화의 경우 데이터베이스 자체 내에 제공하는 암호함수를 이용할 수 있지만 이것은 성능 면에서 비효율적이고, 키 관리 측면에서 노출이 쉽다. 즉 데이터베이스 암호화시 키관리 부분은 데이터베이스 운영자들이 관여하지 않고 키를 자동적으로 생성하고 분배해 줄 수 있는 중앙 집중적인 키 관리 프로토콜이 존재한다면 가장 이상적이지만, 현실적으로 합당한 키 관리 솔루션이 상용화 되어 사용되지 못하고 있는 실정이다. 따라서 키 생성은 사용자 스스로 생성하고 키 관리는 제3의 기관을 통해서 키의 관리 및 사용자에 대한 인증을 수행하게 하여 이러한 사용자만이 데이터베이스에 접근하고 암호화 하게 된다면 위에서 제기한 문제를 해결할 수 있다.

2.4 기존 교육행정정보 시스템 분석

인증메시지는 교사용 컴퓨터로부터 교육망 접근을 위한 요청메시지를 전자서명과 함께 보내면 교육정보망서버에서 인증 서버를 통해 인증서 검색을 요청한다. 인증서 검색요청에 의해 전송된 인증서를 통해 교사용 컴퓨터로부터 온 전자서명을 검증하면 본 사용자는 인증이 되고 권한을 획득하게 된다. 그러면 교육정보망 서버는 요청 메시지에 의하여 데이터베이스의 데이터를 웹상의 Viewer 프로그램을 통하여 전송한다.



(그림 3) NEIS 인증 흐름도

중고등학교에서는 학기 초 학년 말, 중간 기말고사 등 학사행정 및 성적관련 처리를 위해 특정한 기간에 업무 프로세스의 증가와 집중현상이 뚜렷하다.

기존의 교육행정정보시스템은 각 교육청단위의 서버에 집중되는 구조로 구성되어 있어 데이터베이스의 집중화로 데이터의 안전성이 취약하다. 또한 서버가 한곳으로 집중되어 있는 관계로 많은 서비스 요청의 경우 네트워크 트래픽의 병목현상을 초래하고 있다.

또한 단위학교에서 관리해야 하는 학생들의 각종 정보가 교육청의 서버에 저장되는 구조로 인하여 일부 교육단체에서 주장하는 바와 같이 국가에 의한 정보 독점의 가능성이 존재하기도 한다. 또한 데이터가 한곳에 집중되어 있는 관계로 자료의 분산처리를 통한 데이터의 안전한 관리 측면에서도 불리하다.

교육행정 업무의 특성상 서비스를 이용하는 대상은 교사, 행정요원, 학생, 학부모, 해당학교의 졸업생 등이 서비스를 이용하게 된다. 이는 사용자 인증의 관점에서 보면 사용자 인증의 프로세스가 많이 발생함을 예측할 수 있다.

현재 사용자 인증의 방식에서는 PKI 기반의 인증에만 의존하여 교사 및 행정요원에게만 사용자 인증을 하고 있다. 대부분의 사용자가 될 수 있는 학생, 학부모, 졸업생 등에 대한 사용자 인증을 위해서 이들 사용자에 대한 등록기관인 교육청단위의 등록업무와 교육인적자원부의 인증서 발급 업무가 폭증하게 될 것이다.

그러므로 안전한 사용자 인증방식과 서비스 이용자들의 특성을 충분히 반영하여 DB 접근통제와 실행권한을 부여하여야 하는 것이 시급한 과제가 된다. 특히 외부의 학부모 또는 졸업생에 대한 사용자 인증이 완전하다고 해도 이들에 대한 서비스 제공에는 일정 부분에만 국한되어야 한다.

DB 데이터를 암호화 없이 평문으로 사용자의 컴퓨터에 전송하는 방식을 채택한 관계로 해킹에 의한 데이터의 노출 위험이 증가하였고, 데이터의 기밀성, 무결성이 보장되지 않고 있다. 이로 인해 교육행정정보서비스의 신뢰성에 따른 논란으로 많은 사회적 갈등을 초래하였다.

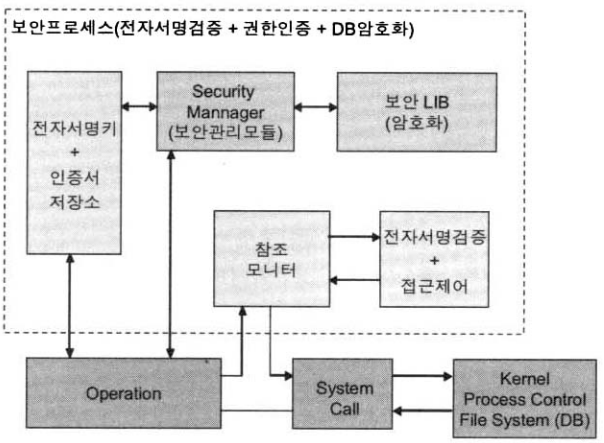
3. 제안된 교육행정정보 시스템(RNEIS : Role Based Access Contorl NEIS)

3.1 보안 관리 모듈(정보흐름 감시모듈)

기존의 교육행정정보시스템은 사용자의 인증을 위하여 인증서를 검색하고 검증이 확인되면 사용자를 인증하는 방식으로 실행되고 있다. 제안된 시스템의 보안관리에서 참조모니터는 안전한 운영체계에서 프로세스(주체)와 파일(객체)의 정보 흐름을 감시하는 보안모듈이다.

참조모니터를 일반 운영체계에 구현하기 위한 방법 중 가장 널리 이용되는 방법은 정보흐름 통로가 되는 시스템 콜을 감시하는 것이다. 시스템 콜 후킹(System Call Hooking) 기술을 이용하면 쉽게 시스템 콜을 감시할 수 있다. 시스템 콜은 일반적으로 시스템 콜 테이블이라고 하는 배열 구조체로 이루어져 있다. 시스템 콜 번호는 배열 구조체에 저장되어 있는 시스템 콜 함수 포인터를 가리키는 번지이다.

시스템 콜 후킹은 시스템 콜 테이블에다 기존의 시스템 콜을 대치하는 새로운 시스템 콜 함수 포인터를 입력함으로 이루어진다. 대치되는 시스템 콜에는 전자서명과 보안정책이 적용된다. (그림 4)는 보안모듈의 구조를 나타낸 것이다. 참조모니터는 운영체계 커널과 독립적으로 동작할 수 있도록 모듈 형태로 구현되는 것이 최적이다. 모듈로 작성하면 이식성과 효율성이 높아진다. 참조모니터는 주체와 객체의 접근권한을 정의한 데이터베이스를 참조함으로 보안정책을 수행한다. 후킹으로 가로챈 시스템 콜은 객체에 대한 주체의 접근권한을 데이터베이스에 정의된 보안정책을 참고하여 판단한다[10-13].



(그림 4) 보안관리 모듈

3.2 등록관리와 인증서 발급

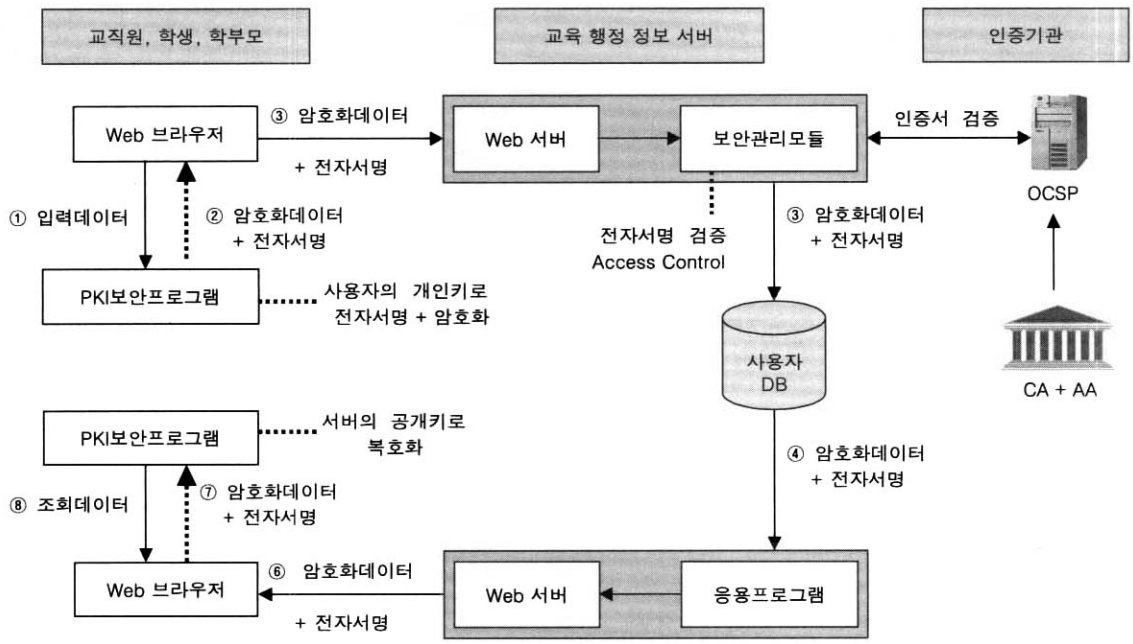
신원확인은 단위학교에서 교육행정서비스 사용자의 신원을 OFF-LINE으로 확인하고 인증서가 필요한 실제 사용자에 대한 공인인증서를 신청한다. 등록관리 업무를 담당하는 기관은 상급기관인 교육청으로 하며 등록관리 서버(RA)를 통하여 등록관리 업무를 수행한다.

교육청은 사용자의 신분을 확인하고 사용자 정보를 인증기관에 제공하며 공인인증기관의 신원확인 서비스를 이용하여 사용자의 개인식별(DN) 정보와 인가코드(PIN)를 OFF-LINE으로 발급 받아 단위학교의 사용자에게 문서로 전달한다. 사용자는 개인식별 정보와 인가코드를 이용하여 인증기관에 ON-LINE으로 인증서를 신청하여 발급받는다.

이로써 비인가자에 의한 인증서 불법도용 및 개인정보 유출을 방지하여 시스템의 신뢰성을 증가시킨다.

3.3 전자서명과 사용자 인증

① X.509 기반 강한 인증 구현
② 사용자 : 전자서명 생성 및 전송
③ 교육행정정보서버 : 전자서명 검사



(그림 5) 전자서명과 사용자 인증

인증서의 비밀 암호화용 개인키와 Time Stamp를 이용하여 전자서명 생성 후 서버로 전송한다.

교육행정정보시스템은 교육인적자원부 및 16개 시·도교육청, 관련 부처 간 연계가 필요한 부분과 민원인과의 연계 두 가지 부분이 존재함으로 공인인증기관기반(NPKI)의 인증 인프라를 사용한다. 기 구축된 공인인증기관의 인증 인프라를 수용하고, 등록관리 시스템인 RA 서버를 교육청 단위에, 단위학교에 교육행정정보 서버를 설치하여 사용자에게 편리한 인증서 등록환경을 제공한다.

전자서명을 사용한 강한 인증(Strong Authentication)을 적용한다. 강한 인증 값 생성시 Time Stamp를 적용하여 이전의 접속 인증 값 재사용을 방지한다. PKI는 정보 시스템에 안전성을 부여하고 통신 시스템의 신뢰성을 높이기 위한 기반구조로, 네트워크 상에 연결된 각 사용자 및 메시지에 대한 인증기능을 부여하기 위해서 공개키 방식을 이용한 인증용 기반구조로서 사용자의 공개키에 대하여 인증기관의 개인키로 서명하며 공개키에 대칭되는 개인키가 있음을 인증기관이 보증하는 구조이다.

인증서 기반의 인증 및 전자서명, 암호화 기술을 적용하면 개방된 웹 프로그램에서 발생할 수 있는 데이터의 노출, 위·변조, 신원확인 문제, 법적효력 문제 등을 해결하여 교육행정정보시스템의 안전성과 신뢰성이 확보되도록 할 수 있다. 제안 교육행정정보시스템의 보안서비스 구현형태는 (그림 5)와 같다.

- 인증서 발급 후 접속인증은

① 사용자의 전자서명 인증서를 이용하여 암호화된 로그인정보(강한 인증값)를 생성하여 웹 서버로 전송. 로그인정보는 사용자의 ID, 사용자의 인증서, 사용자의 전자서명, Time Stamp를 포함

② 사용자로부터 전송받은 로그인 정보는 보안관리모듈에서 사용자 전자서명 검사, 인증서의 유효성을 검사하고 검사결과를 응용 프로그램으로 전달

③ 응용 프로그램은 사용자 등록 DB에 사용자등록 여부를 확인한 후 Time Stamp, 접속권한을 검사한 후 접속권한을 응용프로그램에 설정, 사용자 DB에 사용자의 인증서가 등록되어 있지 않으면 인증서를 사용자 DB에 등록

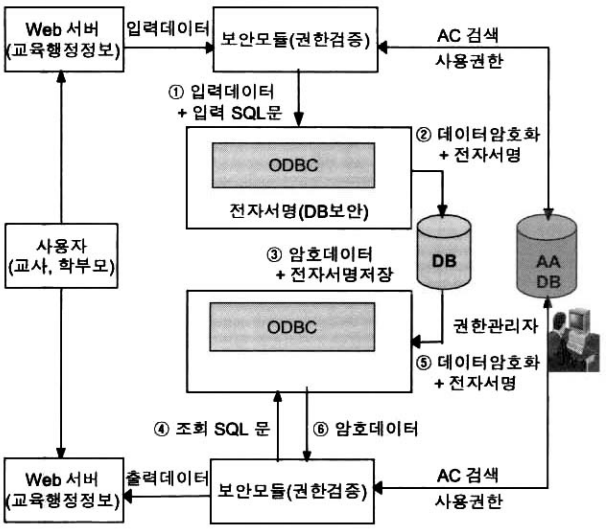
3.4 데이터 보안 및 암호화

개방된 인터넷 환경에서 정보보호를 위한 웹 서비스 보안, 접근권한이 있는 주체만이 정보의 변경 및 조회를 할 수 있는 DB 보안을 적용 통합적인 관점에서 응용 프로그램 보안을 구성한다.

① RSA With SHA1 전자서명 알고리즘 적용

② 전자서명과 암호화를 위한 개인키와 공개키를 각각 생성, 운영

③ 개인키는 비밀키 알고리즘을 적용한 패스워드 기반 암호화 방식으로 암호화하여 개인PC에 저장 활용한다. DB 데이터에 접근 도중 접근권한이 있는 사용자 이외의 사용자가 정보의 변경 및 조회하는 것을 방지할 필요가 있는 업무에 적용한다. DB접근제어와 암호화 구조는 (그림 6)과 같다.



(그림 6) DB 접근제어와 암호화

① DB에 데이터를 입력 시 응용 프로그램에서 입력구문과 데이터를 입력하면 보안 적용규칙에 따라 암호화 및 전자서명을 수행하여 이 값을 DB에 저장한다.

② 데이터를 조회 시 응용 프로그램에서 조회 문장을 입력하면 보안모듈에서 DB에 저장된 암호화 및 전자서명을 읽어 복호화 및 전자서명 검사를 수행한 후 복호화된 데이터를 반환한다.

3.5 RNEIS 시스템의 수행 프로토콜

3.5.1 구성요소

① 인증서 소유자(User, Certification Subject) : 교육행정정보시스템을 사용하는 사용자(교사, 학생, 학부모)로서 시스템 사용 시 인증이 필요한 주체

② 속성 인증서 소유자(User, Attribute Certification Holder) : 인증기관으로부터 인증서를 발급받은 교육행정정보시스템을 사용하는 사용자(교사, 학생, 학부모)로서 시스템 사용 시 권한 인증과 역할이 필요한 접근 주체

③ 인증기관(CA, Certification Authority) : 교육행정정보시스템 사용자와 서버에게 인증서를 발급해 주거나 관리 기능을 제공해 주는 신뢰기관이다.

④ 속성 기관(AA, Attribute Authority) : 교육행정정보시스템 사용자에게 권한과 역할을 부여하는 기관으로 속성인증서를 발급과 폐지 기능을 제공해 주는 기관이다.

⑤ OCSP서버(Online Certificate Service Protocol) : 인증서와 속성인증서 상태를 온라인으로 제공하는 서버[14]

⑥ 교육행정정보서버(VS, Verification Server) : 인증서 기반의 로그온이나 전자서명을 전송할 때 서비스 사용자의 인증서와 속성인증서를 검증하는 서버로서 보안모듈의 접근정책에 따라 응용프로그램에 대한 접근을 허가한다.

⑦ DB 서버(AS, Application Server) : 교육행정정보에 관리를 위한 DBMS 서버 또는 응용프로그램을 제공하는 서버

3.5.2 표 기

ID_A : 사용자 A의 ID

KR_a : 사용자 A의 개인키(Private Key)

KU_a : 사용자 A의 공개키(Public Key)

E_{KRa} : 사용자 A의 개인키로 암호화, 전자서명

ID_V : 검증서버(교육행정정보서버 VS)의 ID

KR_v : 검증서버의 개인키

KU_v : 검증서버의 공개키

KR_{auth} : 인증기관 CA의 개인키

KU_{auth} : 인증기관 CA의 공개키

E_{KRauth} : 인증기관의 개인키로 암호화, 전자서명

D_{KUauth} : 인증기관의 공개키로 복호화

C_A : 사용자 A의 공개키 인증서

C_V : 검증서버의 공개키 인증서

T_A : 인증서의 유효성을 나타내는 타임스탬프

KR_{aa} : 속성기관의 개인키

KU_{aa} : 속성기관의 공개키

E_{KRaa} : 속성기관의 개인키로 암호화, 전자서명

D_{KUaa} : 속성기관의 공개키로 복호화

R_A : 사용자 A의 속성 인증서

T_r : 속성인증서의 유효성을 나타내는 타임스탬프

$Role_{id}$: 속성기관에 의해서 인증된 역할 속성

$Service_{id}$: 서버에 대한 서비스 요청 메시지

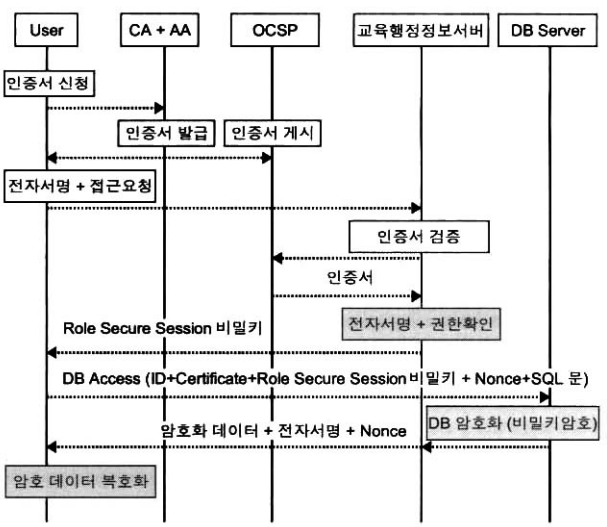
K_S : Role Secure Session Key

$H(SQL)$: SQL 메시지를 일방향 해쉬

$Nonce$: Replay Attack 방지를 위한 비표

$E_{Ks}(Data)$: 세션키 K_S로 비밀키(대칭키)암호화

|| : 연접



(그림 7) RNEIS 시스템의 동작 수행도

3.5.3 사용자 인증

1) 사용자 A의 공개키 인증

① A → CA : $(ID_A||KU_a)||E_{KRa}[H(ID_A||KU_a)]$

② CA → A : $C_A||KU_{auth}$

단, $C_A = E_{KRauth}[(T_A||ID_A||KU_a)||H(T_A||ID_A||KU_a)]$

③ A : $D_{KUauth}[C_A]$

$= D_{KUauth}[E_{KRauth}(T_A||ID_A||KU_a)||H(T_A||ID_A||KU_A)]$

$= (T_A||ID_A||KU_a)$

2) 교육행정정보서버 VS의 공개키 인증

④ VS → CA : $(ID_V||KU_v) = E_{KRv}[H(ID_V||KU_v)]$

⑤ CA → VS : $C_V||KU_{auth}$

단, $C_V = E_{KRauth}[(T_V||ID_V||KU_v)||H(T_V||ID_V||KU_v)]$

⑥ VS : $D_{KUauth}[C_V]$

$= D_{KUauth}[E_{KRauth}[(T_V||ID_V||KU_v)]||H(T_V||ID_V||KU_v)]$

$= (T_V||ID_V||KU_v)$

3) 교육행정정보서버 VS가 사용자 A의 공개키 인증서 검증

⑦ A → VS : $(ID_A||KU_a||C_A)||E_{KRa}[H(ID_A||KU_A||C_A)]$

⑧ VS : $D_{KUauth}[C_A]$

$= D_{KUauth}[E_{KRauth}(T_A||ID_A||KU_a)]$

사용자 A는 인증기관 CA에게 공개키 인증서 발급을 요청한다. 사용자 A의 경우 기관은 다음 형태의 인증서를 발급한다. KR_{auth}은 인증기관에 의해 사용되어지는 개인키이다. 사용자 A는 서비스를 이용할 때 인증서를 전달하여 주며, 교육행정정보서버(VS)는 사용자의 인증서를 읽어 다음과 같이 인증서를 확인한다. 사용자는 자신의 이름 ID_A와 공개키 KU_a를 VS에게 전송한다. 타임스탬프 T_A는 인증서의 현재성이 정당함을 확인한다[15].

인증서를 수신한 VS는 인증서를 복호화하기 위하여 인증기관의 공개키 KU_{auth}를 사용한다.

$$D_{KUauth}[C_A] = D_{KUauth}[E_{KRauth}(T_A||ID_A||KU_a)] = (T_A||ID_A||KU_a).$$

인증서는 인증기관의 공개키를 사용하여야 만이 읽을 수 있기 때문에, 이것은 사용자 ID_A의 인증서 $C_A = E_{KRauth}[T_A||ID_A||KU_a]$가 인증기관이 보증하는 것임을 확인할 수 있다.

VS의 경우도 사용자와 같은 방법으로 인증기관으로부터 서버용 인증서를 발급받는다. 서버의 인증서 형태는 다음과 같다.

$$C_V = E_{KRauth}[(T_V||ID_V||KU_v)||H(T_V||ID_V||KU_v)]$$

인증서를 수신한 VS는 인증서를 복호하기 위하여 인증기관의 공개키 KU_{auth}를 사용한다. 인증서는 기관의 공개키를 사용하여야 만이 읽을 수 있기 때문에, VS에 대한 인증서가 인증기관으로부터 온 것임을 확인할 수 있다.

$$D_{KUauth}[C_V] = D_{KUauth}[E_{KRauth}(T_V||ID_V||KU_v)] = (T_V||ID_V||KU_v)$$

위의 경우에 사용자 A 및 서버 VS의 공개키인증서는 각각 사용자와 서버만이 그에 해당하는 개인키를 가지고 있음을 인증기관이 증명하고 있으므로 강력한 사용자 인증이 된다.

3.5.4 권한 인증

1) 속성 인증서 신청

① A → AA : $(ID_A||KU_a||C_A||Role_{id})||E_{KRa}$

$[H(ID_A||KU_a||C_A||Role_{id})]$

단, $C_A = E_{KRauth}[(T_A||ID_A||KU_a)||H(T_A||ID_A||KU_a)]$

2) 속성 권한 부여와 인증

② AA → A : $R_A||KU_{aa}$

$R_A = E_{KRaa}[(T_r||ID_A||KU_a||Role_{id})]||H$

$(T_r||ID_A||KU_a||Role_{id})]$

3) 속성 인증서 복호화

③ A : $D_{KUaa}[R_a]$

$= D_{KUaa}[E_{KRaa}(T_r||ID_A||KU_a||Role_{id})||H$

$(T_r||ID_A||KU_a||Role_{id})]$

$= (T_r||ID_A||KU_a||Role_{id})$

속성기관 AA는 인증기관 CA로부터 인증서를 발급받은 사용자에 한해서 속성인증서 소유자의 권한과 역할 및 응용프로그램의 서비스 종류에 따라 속성기관의 개인키 KR_{aa}로 서명하여 발행한다. 속성 인증서는 소유자의 공개키 인증서의 시리얼 번호 등으로 공개키 인증서와 연결되어 되어 있다. 즉 속성 인증서는 소유자의 공개키 인증서의 시리얼 번호와 발행자가 기술되고 속성인증서를 검증할 때는 공개키 인증서와 조합해서 이용한다.

$$R_A = E_{KRaa}[T_r||ID_A||KU_a||Role_{id}]$$

여기서 $Role_{id}$는 속성인증서 소유자의 권한 및 역할이 되며 서버들의 응용프로그램의 사용권한이나 역할이 된다. 타임스탬프 T_r은 속성인증서의 현재성이 정당함을 증명한다.

$$D_{KUaa}[R_A] = D_{KUaa}[E_{KRaa}(T_r||ID_A||KU_a||Role_{id})] = (T_r||ID_A||KU_a||Role_{id})$$

속성 인증서를 수신한 서버는 속성 인증서를 복호하기 위

하여 속성기관의 공개키 KU_{aa}를 사용한다. 속성 인증서는 속성기관의 공개키를 사용하여 읽을 수 있기 때문에 이것은 사용자 A의 권한 및 역할을 속성기관이 보증하게 된다.

3.5.5 서버에 대한 접근 요청 절차

4) 응용프로그램

① A → VS : $(ID_A \| KU_a \| R_A \| C_A \| Role_{id} \| Service_{id}) \| E_{KRa}$
$[H(ID_A \| KU_a \| R_A \| C_A \| Role_{id} \| Service_{id})]$

② VS → A : $M_s = (ID_A \| Role_{id} \| Service_{id} \| ID_V \| C_V) \|$
$E_{KRv}(K_s) \| E_{KRv}[H(K_s)]$

단, $C_V = E_{KRauth}[(T_V \| ID_V \| KU_v) \| H(T_V \| ID_V \| KU_v)]$

③ A : 메시지 수신

$M_s = (ID_A \| Role_{id} \| Service_{id} \| ID_V \| C_V) \| E_{KRv}(K_s) \| E_{KRv}[H(K_s)]$

$D_{KUv}[E_{KRv}(K_s)]$로 복호화, 세션키 공유

④ A → VS : $M_{sql} = [(ID_A \| KU_A \| Role_{id} \| Service_{id} \|$
$C_A \| R_a) \| E_{KRa}(K_s \| H(SQL) \| Nonce \| SQL)]$

서비스 신청자 A는 사용자의 이름 ID_A와 서비스 요청 메시지 $Service_{id}$공개키 인증서 C_A와 속성인증서 R_A로 이루어진 요청메시지 $(ID_A \| KU_a \| R_A \| C_A \| Role_{id} \| Service_{id})$를 사용자의 개인키 KR_a로 서명 및 암호화하고 $E_{KRa}[H(ID_A \| KU_a \| R_A \| C_A \| Role_{id} \| Service_{id})]$와 함께 서버(VS)로 전송한다.

서버는 사용자의 공개키 인증서와 속성인증서를 검증하고 사용자의 공개키를 이용하여 사용자 인증은 물론 권한과 역할을 확인한다. 인증서와 속성인증서를 검증한 서버는 사용자에게 응용프로그램의 암호화에 사용할 Role Secure Session Key K_S를 서버의 개인키 KR_v로 서명하고 암호화한 다음 서버의 공개키 인증서 $C_V = E_{KRauth}[(T_V \| ID_V \| KU_v) \| H(T_V \| ID_V \| KU_v)]$와 메시지 M_s를 함께 전송한다.

단 서버로부터의 메시지 $M_s = (ID_A \| Role_{id} \| Service_{id} \| C_V) \| E_{KRv}(K_s) \| E_{KRv}[H(K_s)]$이다.

사용자는 서버로부터 받은 메시지 M_s를 서버의 공개키 KU_v를 이용하여 복호화한 다음 SQL 메시지를 일방향 해쉬함수를 이용하여 $H(SQL)$의 메시지 인증코드를 생성한다. 그리고 SQL 문과 Replay Attack을 방지하기 위한 Nonce를 함께 전송한다.

이 때 전송메시지는 M_{sql}이다.

$$M_{sql} = (ID_A \| Role_{id} \| Service_{id} \| C_A \| R_a) \| E_{KRa}[K_S \| H(SQL) \| Nonce \| SQL)]$$

3.5.6 서버 응용프로그램 접근 및 파일 암호화

① A → VS : $M_{sql} = (ID_A \| Role_{id} \| Service_{id} \| C_A \| R_a) \| E_{KRa}[K_S \| H(SQL) \| Nonce \| SQL)]$

② A → DB : SQL Query 실행

③ DB → VS → A : $M_{Data} = (ID_A \| Role_{id} \| Service_{id} \| C_V) \| E_{Ks}(Data) \| E_{KRv}[H(Data) \| (Nonce)]$

④ A : $D_{KUv}[E_{KRv}[H(Data) \| (Nonce)]]$ 데이터 복호화

접근제어의 정책에 따라 서버의 응용프로그램의 프로세스를 사용하게 되면 사용자는 웹을 통하여 교육행정정보시스템의 DB에 접근할 수 있게 된다. DB에 접근하게 될 때는 사용자의 속성인증서에서 명세화되어 있는 권한과 역할에 따라 DB의 사용이 허가되고 해당 프로세스를 실행할 수 있다. 교사의 경우는 DB에 접근하여 성적 관련 자료를 Insert, Delete, Select, Append 할 수 있다. 학생이나 학부모인 경우는 DB의 조회 기능에 대해서만 권한을 부여할 수 있다.

사용자로부터 다음과 같은 메시지 $M_{sql} = (ID_A \| Role_{id} \| Service_{id} \| C_A \| R_a) \| E_{KRa}[K_S \| H(SQL) \| Nonce \| SQL)]$를 수신한 서버는 사용자의 공개키를 이용하여 세션키와 Nonce를 복호화하고 SQL 문의 인증을 위하여 평문으로 전송된 SQL Query를 해쉬하여 메시지인증코드로 전송된 $H(SQL)$와 비교하여 신뢰된 메시지인 경우 SQL Query를 DB에 전달하게 된다.

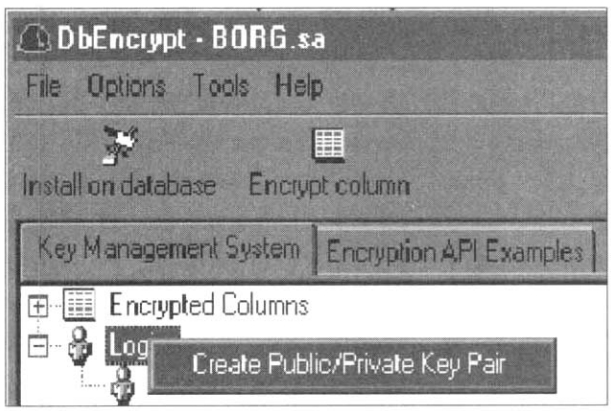
DB의 SQL Query 실행 이후에 서버는 DB에 저장된 데이터 파일을 암호화하고 해쉬함수를 이용하여 Data에 대한 해쉬값 $H(Data)$를 구하고 서버의 개인키 KR_v로 서명하여 전달한다.

$M_{Data} = (ID_A \| Role_{id} \| Service_{id} \| C_V) \| E_{Ks}(Data) \| E_{KRv}[H(Data) \| (Nonce)]$가 전송되는 데이터로서 데이터에 대한 암호화는 이미 서버와 사용자 사이에 공유하고 있는 Role Secure Session Key인 K_S를 이용하여 비밀키 암호 방식으로 실행한다. DB에 저장되는 파일의 크기가 큰 경우는 파일 암호화 시 암호학적 연산에 많은 시간이 걸린다. 따라서 데이터의 암호화에는 암호화 속도가 빠른 비밀키 암호 방식을 사용하고 인증서, ID, 세션키, 메시지 인증코드 등의 암호화 또는 서명에는 공개키 암호를 사용하여 사용자의 인증, 전자서명, 기밀성, 무결성 등을 확보한다.

서버로부터 데이터 M_{Data}를 수신한 사용자는 서버의 공개키 KU_v를 이용하여 $Nonce$와 메시지 인증값 $H(Data)$를 복호화하고 세션키 K_S를 이용하여 암호화된 데이터를 복호화한다. 또한 데이터를 해쉬하여 메시지 인증 값과 비교하여 신뢰된 메시지인지를 확인한다.

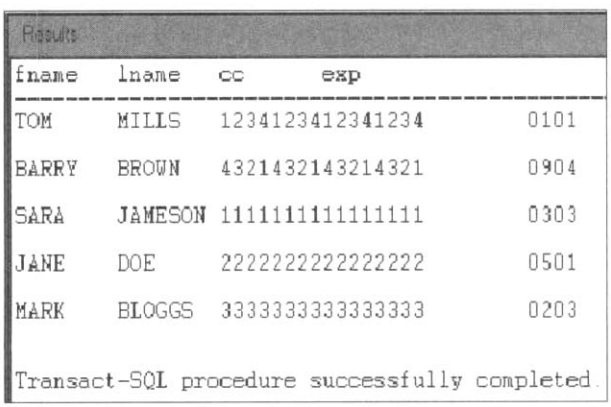
3.6 DB 암호화 구현

① (그림 8)은 개인키와 공개키를 생성을 나타낸다. DBMS에 대한 접근허가를 받게 되면 (그림 8) 같이 개인키와 공개키를 생성하는 과정이 실행된다. 사용자는 자신의 개인키를 이용하여 DB 데이터에 대한 전자서명을 수행할 수 있는 키를 가진다. 여기서 사용하는 개인키와 공개키는 속성인증서에 저장된 키를 불러서 사용한다.



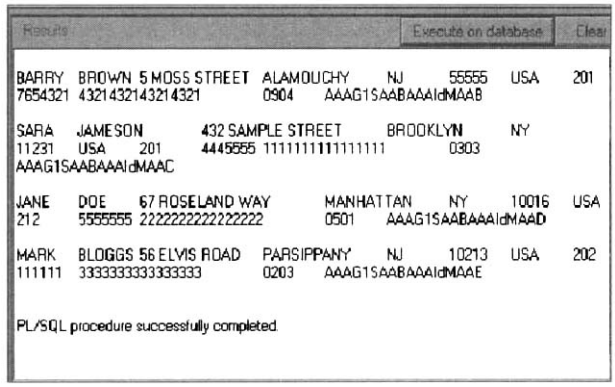
(그림 8) 개인키와 공개키 쌍 생성화면

② (그림 9)는 DB에 접속하여 암호하기 전의 컬럼 내용을 나타낸다. 평문의 데이터가 그대로 보인다.



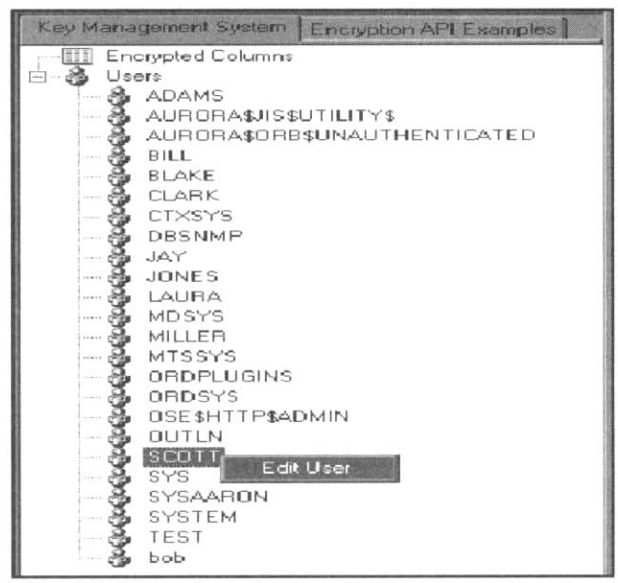
(그림 9) DB를 암호화 하기 전의 컬럼 내용

③ (그림 10)은 DB를 암호화한 후의 컬럼의 내용을 보여준다. 컬럼의 내용이 사용자의 개인키로 암호화 되어 있어 사용자의 공개키를 공개키 저장소에서 확인할 수 있으며 해당 공개키를 가진 다른 사용자들은 파일의 내용을 조회하여 볼 수 있다. 즉 속성인증서의 role 속성을 이용하여 같은 속성 그룹에 속한 사용자들은 공개키를 서로 찾을 수 있도록 하여 파일의 복호화가 가능하다.



(그림 10) DB 암호화 이후의 컬럼 내용

④ (그림 11)은 공개키 저장트리 화면으로 같은 role 속성이 같은 그룹이 공유하는 공개키 저장소를 보여준다. 같은 업무를 수행하는 사용자들은 DB 파일에 접근할 때 다른 사용자에 의하여 전자서명된 파일의 내용은 작성자의 공개키를 얻어서 파일의 내용을 볼 수 있다.



(그림 11) 공개키 저장 트리 화면

4. 제안시스템 분석

4.1 수행시간 비교

DB 암호화에 따른 프로그램의 수행 시간을 비교하여 제안 시스템의 성능을 분석하였다. 제안시스템의 적용결과 암호화 적용 이후 약간의 처리속도의 저하가 있음을 알수 있다. 그러나 암호화에 따른 응답속도의 저하는 새로운 보안 모듈의 도입으로 불가피한 경우라고 할수 있다. 중요한 고객의 정보나 컨텐츠의 해킹에 따른 손실을 감안하면 미미한 정도의 성능저하라고 볼수 있다.

〈표 1〉 DB암호화에 따른 프로그램 수행시간 비교

구 분	암호화 적용전	암호화 적용후	적용 필드
ORACLE	2.3[sec]	2.5[sec]	사용자성, 사용자 이름, 신용카드번호
msSQL	2.4[sec]	2.6[sec]	사용자성, 사용자 이름, 신용카드번호
mySQL	2.6[sec]	2.7[sec]	사용자성, 사용자 이름, 신용카드번호

4.2 DB암호화에 따른 보안평가 분석

① 단위학교의 학생 정보를 교육청에서 집중 관리하여 발생하는 문제는 단위 학교별로 교육행정정보서버를 설치하여 일부 교육단체가 주장하는 정보인권의 문제를 해결하였으며 분산 관리에 의한 데이터의 안전성이 증가되었다.

② 분산 환경에 의한 서버 설치로 네트워크 트래픽의 발생량이 감소하게 되었으며 서비스 응답 속도를 증가시켰다.

③ 속성인증서에 의한 역할기반 접근제어로 정당한 사용자만이 시스템에 접근할 수 있도록 하였으며, 비인가자 뿐만 아니라 시스템 관리자에 의한 DB 불법 침입을 방지하였다.

④ 속성인증서의 role 속성 필드를 사용하여 인증서 발급 프로세스를 감소시켰다. 기존의 방식에서는 PKI 인증 기능만을 사용하여 교사의 전보나 학생의 전입학과 같은 업무 프로세스가 발생할 때 인증서를 폐지하여 인증서폐지목록이 증가하는 경향이 많았다. 제안된 시스템에서는 속성인증서를 이용하여 인증서가 유효할 경우 폐지하지 않는다.

⑤ 기존 시스템의 경우 웹 환경에서 서비스 요청 메시지 또는 응답 메시지를 평문으로 전송하여 해킹에 의한 정보의 누출 위험이 존재하였으나 제안된 시스템은 SQL Query를 서비스 요청자의 전자서명으로 암호하고 해쉬된 SQL Query와 함께 전송하여 해킹에 의한 메시지 변조가 일어나지 않도록 하였다.

⑥ 모든 메시지 전송에는 전자서명 프로토콜을 이용하여 메시지에 대한 부인방지 기능과, 데이터 무결성, 데이터 기밀성을 보증한다.

⑦ 전자서명 인증에 사용하는 해쉬함수는 응답속도가 빠른 SHA1을 사용하여 전자서명의 무결성을 확보하였다.

⑧ 파일암호화에 사용하는 비밀키 알고리즘은 대용량 데이터의 경우에도 암호화 속도가 빠른 RC4 알고리즘을 사용하였다.

다음 <표 2>는 제안 시스템(RNEIS)과 기존 시스템(NEIS)을 비교 분석했다.

〈표 2〉 프로토콜 비교 분석

구 분	제안 시스템(RNEIS)	기존 시스템(NEIS)
인증	◦ 전자서명에 의한 PKI 인증	◦ 전자서명에 의한 PKI 인증
접근제어	◦ 전자서명에 의한 PMI 인증 ◦ 속성인증서의 Role 속성 필드 사용 ◦ RBAC에 의한 접근제어로 시스템 관리자 및 비인가자 제어 ◦ 속성인증서의 사용으로 인증서는 폐지되지 않으므로 인증서 유효함	◦ PKI 인증 ◦ 공개키 인증서의 확장 필드 사용 ◦ 시스템 관리자의 접근 제어 불가 ◦ 권한 및 역할 변경시 인증서폐지목록이 증가하고 재발급 시 프로세스 증가
부인 방지	◦ 전자 서명에 의한 부인 방지	◦ 없음
데이터 무결성	◦ 전자 서명과 해쉬에 의한 데이터 암호화로 데이터 무결성 확보	◦ 없음
데이터 기밀성	◦ File 암호화(비밀키 암호)에 의한 Secure DBMS 구현	◦ 없음
데이터 안정성	◦ 서버에 분산 저장하여 위험 분산	◦ 서버에 집중 저장하여 위험 증가
SQL Injection	◦ SQL 암호화에 의한 안전한 SQL Query 전송으로 해킹으로부터 보호	◦ 평문 전송으로 해킹에 취약
데이터 전송	◦ 암호화된 데이터 전송	◦ 평문 데이터 전송
검증서버	◦ 단위학교의 응용서버나 서비스에 대한 권한만을 저장하여 소량의 프로세스 발생	◦ 교육청 단위의 응용서버와 서비스에 대한 권한을 저장하여 대량 프로세스를 발생
네트워크 트래픽	◦ 소량의 패킷 발생으로 신속한 서비스 가능	◦ 대량의 패킷 발생으로 서비스 지연
비 용	◦ 단위 학교별 서버 설치로 장비 비용 증가	◦ 교육청 단위의 설치로 비용 감소

5. 결 론

본 논문에서는 권한과 역할에 따라 사용자의 인증을 강화하기 위하여 공인인증시스템을 변경하지 않고 속성인증서의 역할지정인증서를 이용하여 기존의 교육행정정보시스템에 적용된 방법보다 세분화 된 사용자 인증 방법을 제안하였다. 구체적 인증방법으로 공개키인증서를 발급받은 사용자에 대하여 속성인증서를 발행하고 속성인증서의 확장 필드를 이용하여 사용자의 권한과 역할을 지정함으로써 공개키인증서의 인증서폐지목록(CRL)을 검색하지 않고 사용자의 인증을 신속하게 처리할 수 있음을 보였다.

교육행정정보 시스템에서는 한 사람의 교사에게 여러 가지의 권한과 역할이 주어진다. 즉 담임교사, 교과담당, 학교내 업무 등 동일인에게 복수의 권한과 역할을 부여하여 교육행정업무가 이루어진다. 이와 같은 경우 속성인증서에 제공하는 복수의 Role 속성은 이와 같은 문제가 해결됨을 보였다. 즉 기존의 시스템에서 복수의 역할을 주기 위해서는 공개키인증서의 확장필드를 이용해야 하므로 인증서의 유효성을 확인하는 프로세스를 증가시킨다.

또한 모든 메시지 전송은 전자서명을 이용하여 교환함으로써 메지시에 대한 부인방지 기능과 데이터의 기밀성을 보장하였다. 또한 일방향 해쉬함수를 이용하여 SQL문의 메시지 인증기능 및 DB 암호화를 구현하여 DBMS의 보안을 강화하였다.

제안 시스템의 분석에서 기술한 바와 같이 암호화의 과정이 추가되는 과정에서 수행시간의 지연이 발생하였다. 향후 이부분에 대한 보완 연구가 진행되어야 할 것이다.

참 고 문 헌

[1] 박지숙, "고객정보보호를 위한 DB 암호화 구현 사례", 삼성SDS, IT ERVIEW, 2003.

[2] D. W. Chadwick, A. Otenko, E. Ball, "Implementing Role Based Access Controls Using X.509 Attribute Certificates," IEEE Internet Computing, March-April 2003, pp. 62-69.

[3] R. Housely, W. Ford, W. Polk, D. Solo, "Internet X.509 Public Key Infraxtructure Certificate and CRL Profile," IETF RFC2459, January, 1999.

[4] C. Adams, S. Farrell, "Internet X.509 Public Key Infrastructure Certificate Protocols," IETF RFC2510, March, 1999.

[5] M. Myers, C. Adams, D. Solo, D. Kemp, "Internet X.509 Certificate Request Message Format," IETF RFC2511, March, 1999.

[6] 이덕규, 이임영, "PMI를 이용한 확장 권한위임에 관한 연구", 정보처리학회 춘계학술발표논문집, 제9권 제1호, pp.947-950, March, 2002.

[7] 이승훈, 송주석, "PMI 인증서 검증 위임 및 검증 프로토콜", 정보보호학회논문지, 제13권 제1호, pp.59-67, February 2003.

[8] 전문석, 유두규, 문주영, 문봉근, 엄기원, 고명선, 강정호, "정보이론 및 PKI", 미래컴, October, 2003.

[9] 문봉근, 홍성식, 유황빈, "RSA 방식을 이용한 데이터베이스 암호화 구현", 통신정보보호학회논문지, 제3권 제2호, pp.53-62. December, 1993.

[10] 장승주, 이정배, "Linux 운영체제 동적 모듈 개념을 이용한 보안 파일 시스템 모듈 설계", 정보처리학회논문지C, 제10-C권 제7호, pp.929-936, December, 2003.

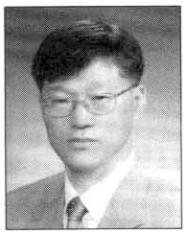
[11] 김익수, 김명호, "관리자 인증 강화를 위한 추가적인 패스워드를 가지는 보안커널모듈 설계 및 구현", 정보처리학회논문지C, 제10-C권 제6호, pp.675-682, November, 2003.

[12] 이남훈, 유신근, 심영철, "Windows 2000기반의 파일보호 시스템 설계 및 구현", 정보처리학회논문지C, 제8-C권 제6호, pp.741-756, December, 2001.

[13] 임재덕, 유준석, 김정녀, "유닉스 시스템에서 다양한 접근 제어 정책을 이용한 커널 수준의 자동 암호화 기법", 정보처리학회논문지C, 제10-C권 제4호, pp.387-396, October, 2003.

[14] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol," IETF RFC2560, June, 1999.

[15] Adams, et al., "Internet X.509 Public Key Infrastructure Time-Stamp Protocol," IETF RFC3161, August, 2001.



유 두 규

e-mail : bima@dreamwiz.com

1984년 숭실대학교 전기공학과 학사

2001년 숭실대학교 컴퓨터교육과 석사

1984년~현재 세명컴퓨터고등학교 인터넷 영상 부장

2001년~현재 숭실대학교 대학원 컴퓨터학과 박사과정

관심분야 : 네트워크 보안, 정보보안, DB보안, DRM, 암호학

문 봉 근

e-mail : mbk2000@chol.com
1988년 수원대학교 전자계산학과 학사
1993년 광운대학교 전자계산학과 석사
1988년~1993년 한신공영(주) 전산실
1993년~1998년 한라정보시스템(주) 마이스터 IS팀
2001년~현재 숭실대학교 대학원 컴퓨터학과 박사과정
관심분야 : 네트워크, 침입탐지시스템, 정보보안

전 문 석

e-mail : mjun@computing.ssu.ac.kr
1980년 숭실대학교 전자계산학과 학사
1986년 University of Maryland 전산과 석사
1989년 University of Maryland 전산과 박사
1989년 Morgen State University 전산수학과 조교수
1989년~1991년 New Mexico State University 부설 Physical Science Lab. 책임연구원
1991년~현재 숭실대학교 정보과학대학 정교수
관심분야 : 네트워크 보안, 컴퓨터 알고리즘, 병렬처리, VLSI 설계, 암호학