情報保護學會論文誌
第14券 第6號, 2004. 12

PMI기반의 RBAC를 이용한 NEIS의 DB 보안 구현*

유 두 규†‡, 문 봉 근, 전 문 석
숭실대학교

An Implementation of NEIS'DB Security Using RBAC based on PMI

Du-Gyu Ryoo†‡, Bong-Keun Moon, Moon-Seog Jun
Soongsil university

요 약

공개키기반구조(PKI)는 강한 인증을 제공한다. 새로운 인증기술로서 권한기반구조(PMI)는 사용자의 속성 정보를 제공한다. PMI의 중요한 기능은 인증이 이루어진 이후에 보다 상세화된 권한과 역할을 부여하는 것이다. 즉, 전자서명에 의하여 사용자를 인증하고 권한을 부여한다. 본 논문은 DB 보안을 위해 PMI 기반의 역할기반접근제어(RBAC)를 제안하였다. RBAC는 보안 관리에 대하여 보이지 않는 유연성을 제공한다. 기존의 교육행정정보시스템(NEIS)은 PKI 인증 방식을 사용하여 속성에 따른 접근 인증을 제공하지 못하였다. 또한, 평문 데이터의 전송으로 정보 인권의 사회적 문제가 발생하였다. 제안된 시스템(RENIS)은 속성인증서(AC)를 이용한 RBAC 인증 기능을 가지며, 보안 관리자에서 사용자의 AC에 저장된 역할과 역할 DB의 역할인증서를 검증하여 접근제어를 허가하고 전자서명에 의한 DB 암호화를 제안한다.

ABSTRACT

Public Key Infrastructure(PKI) provides a strong authentication. Privilege Management Infrastructure(PMI) as a new technology can provide user's attribute information. The main function of PMI is to give more specified authority and role to user. To authenticate user and role, we have used digital signature. Role Based Access Control(RBAC) is implemented by digital signature. RBAC provides some flexibility for security management. NEIS(National Education Information System) can not always provide satisfied quality of security management. The main idea of the proposed RNEIS(Roll Based NEIS) is that user's role is stored in AC, access control decisions are driven by authenticaion policy and role. Security manager enables user to refer to the role stored in user's AC, admits access control and suggests DB encryption by digital signature.

Keywords : *Authentication, Digital Signature, Attribute Certificate, Role Based Access Control*

Ⅰ. 서 론

최근 비인가자에 의한 전산 원장 부당 변경 및 조회, 비밀번호 변경등 고객 정보 유출에 의한 예금 부당 인출등 고도의 전문 기술을 이용한 보안사고가 증가하고 있어, 국가적인 차원에서 인터넷 기반의 전자문서 관리 및 데이터베이스, 인터넷 뱅킹과 같은 인터넷 비니지니스에 관한 취약점을 보완하기 위한 기술적인 노력이 요구되고 있다.[1] 최근 교육행정정보시스템(NEIS)에 대한 정보 인권의 문제는 개인 정보에 대한 문제를교육 현장에까지 그 문제를 확산 시키는 계기가 되었다.

접근제어의 목표는 비인가자 또는 통신 시스템

접수일 : 2004년 8월 4일 ; 채택일 : 2004년 12월 6일
* 본 연구는 숭실대학교 교내연구비 지원으로 이루어짐.
† 주저자, ‡ 교신전자 : bima@dreamwiz.com

의 위협으로부터 응용프로그램 및 시스템을 보호하는 것이다. 기존의 접근제어 방식에서 많은 응용 서비스가 실행중일때 항상 보안 관리가 만족스러운 것은 아니다. 다양한 인터넷 응용 서비스를 만족시키기 위해서 새로운 접근제어 방식이 요구되고 있다.

최근 역할기반접근제어(RBAC, Role Based Access Control) 방법에 대한 연구가 많이 이루어지고 있다. 기본적 개념은 개별적인 사용자보다 임무, 직책에 따라 권한 또는 역할이 주어지는 접근제어이다. 사용자들은 서로 다른 권한에 따라 정보 시스템내의 행위가 주어진다. RBAC는 사용자와 그룹이 사용하는 전통적인 접근법을 통하여 보안 관리에 대하여 보이지 않는 유연성을 제공한다.[2-6]

공개키기반구조(PKI, Public Key Infrastructure)[7-9]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 필요한 시스템에서는 적용하기 곤란한 점이 존재한다.

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 명세화되어 있다. 공개키인증서가 인증서 소유자의 공개키를 보증해주는 것처럼 속성인증서도 속성인증서 소유자의 속성들에 대하여 속성기관이 보증한다. 그러나 PMI가 아직까지 널리 적용되고 있지 않다.[9-10]

본 논문에서 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 공개키인증서의 사용자 인증과 전자서명의 기능을 확장하고 AC를 이용하여 응용 서비스(DB)에 접근할 수 있는 권한과 자격을 부여하여 DBMS 관리자에 의한 사용자의 비밀정보가 노출되는 것을 방지하였다. 또한 역할 인증을 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 성능 평가와 결과를, 5장에서 본 논문의 결론을 맺는다.

Ⅱ. 관련연구

2.1 PMI의 구조

X.509 인증서의 확장 필드를 이용하면 기존의 공개키기반구조 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 재 발행해야 한다. 예를 들어, 반년에 한번 인사 이동이 있는 기업이나 조직의 경우를 보면 반년 마다 인증서의 재발행이 필요하게 된다. 인증서폐지목록(CRL)도 커져버린다. 공개키인증서의 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 사원인지 확인하는 것은 인사부지만 각 사원에 사내의 어느 정보를 이용할 수 있는지는 각 정보의 관리 책임자가 결정하는 것이다. 때문에 인증기관과 권한 부여자는 분리할 필요가 있다.

셋째, 공개키인증서에 자격 등의 속성 정보를 넣으면, 공적으로 이용하기가 어려워진다. 예를 들어 통신을 할 때에 본인인 것을 확인하기 위해서 공개키인증서를 통신 상대에게 보내는 경우 공개키인증서에 속성 정보가 기재되어 있으면 전할 필요가 없는 속성 정보까지 상대에게 전한다.[11]

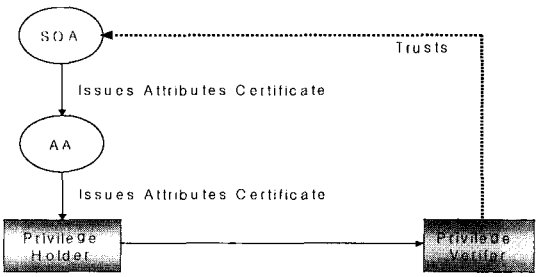


그림 1. PMI의 구조

따라서 이러한 문제들을 해결하기 위해서는 사용자의 고유식별정보를 인증하는 공개키기반구조 이외에 권한, 역할등의 속성에 대한 관계를 보증하는 별도의 인증 구조가 필요하다. 권한관리기반 구조는 이와 같이 권한 관련 자원과 소유자간의 관계를 인증기관이 인증하고 유지하는 구조를 말한다. 그림 1은 PMI 구조를 나타낸다.

공개키 인증서는 인증기관(CA, Certification Authority)에서 사용자에 대한 신원을 확인후 발급하는 반면, 속성인증서는 속성기관(AA, Atribute Authority)에서 발급한다.

그림 1과 같이 루트 인증기관의 역할을 PMI에서는 SOA(Source of Authority)가 수행하며 인증기관의 역할을 속성기관이 한다. 사용자의 신원을 확인하기 위해서 공개키인증서를 검증하고 사용자의 권한이나 역할을 확인하기 위해서 속성인증서를 검증하면 된다. 이러한 검증 과정에서 권한검증자(Privilege Verifier)는 속성인증서와 공개키인증서를 연결하여 사용자가 정당한 권한을 가지고 있는지 판별하게 된다. 사용자는 여러 속성기관으로부터 다수의 속성인증서를 가질 수 있다.

2.2 PMI 모델

ITU_T X.509 표준[12]에서는 속성인증서를 정보보호 메커니즘으로 활용할 수 있도록 일반모델, 제어모델, 위임모델, 역할모델을 제시하고 있다. 여기서는 본 논문에 적용된 역할모델에 관하여 분석한다.

2.2.1 역할모델

역할모델은 사용자 개인에게 간접적으로 권한을 할당할 수 있는 기능을 제공해 준다. 이 모델

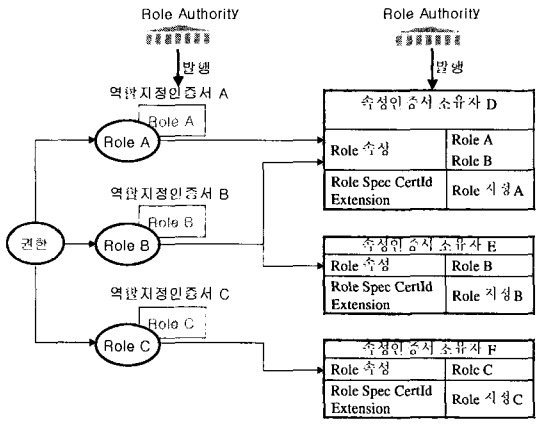


그림 2. 속성인증서와 역할지정인증서

에서는 개인들에게 직접 권한을 주지 않고 단지 속성인증서의 Role 속성 필드에 사용자에게 필요한 역할을 할당한다. Role속성을 기술한 속성인증서는 역할할당인증서(role assignment certificate) 또는 역할지정인증서(role specification certificate)라 한다. 역할지정인증서는 공개키인증서로는 제공할 수 없으며 속성인증서를 이용해야 한다.

이 역할지정인증서는 권한(역할)부여기관(role authority)이 발행한다.

그림 2는 속성인증서 소유자와 역할과의 관계를 나타낸다. 속성인증서 소유자 D는 역할지정인증서 A의 역할 Role A와 역할지정인증서 B의 Role B 2가지 역할을 Role 속성 필드에 가지고 있다. 이 모델의 특징은 RBAC에서 큰 장점을 가진다.

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 접근통제 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 데이

터 관리 및 보호, 다단계 보호(Multilevel protection), 접근제한 등을 들 수 있다.[13]

데이터베이스에 대한 부당 접근 및 변경은 내부 관리자에 의하여 이루어지는 경우와 외부 네트워크를 통한 두가지 경우가 있으나 어떤 경우든 중요 정보가 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 내부 관리자에 의하여 정보가 누출되는 경우 심대한 영향을 초래하게 된다.

데이터베이스 암호화의 경우 데이터베이스 자체 내에 제공하는 암호함수를 이용할 수 있지만 이것은 성능면에서 비효율적이고, 키 관리 측면에서 노출이 쉽다. 즉 데이터베이스 암호화시 키 관리 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

2.4.1 NEIS 프로토콜 분석

그림 3은 NEIS의 프로토콜 흐름도를 나타내며 그림 3을 기반으로 설명하면 동작 순서는 다음과 같다. NEIS의 인증 메시지는 교사용 컴퓨터로부터 NEIS 서버 접근을 위한 요청 메시지를 전자서명과 함께 보내면 교육행정정보서버에서 인증서버를 통해 인증서 검색을 요청한다. 인증서 검색 요청에 의해 전송된 인증서를 통해 교사용 컴퓨터로부터 온 전자서명을 검증하면 사용자는 인증이 되고 권한을 획득하게 된다. 그러면 교육행정정보서버는 요청 메시지에 의하여 데이터베이스의 데이터를 웹상의 Viewer 프로프로그램을 통하여 전송한다.

중·고등학교에서는 학기초 학년말, 중간 기말

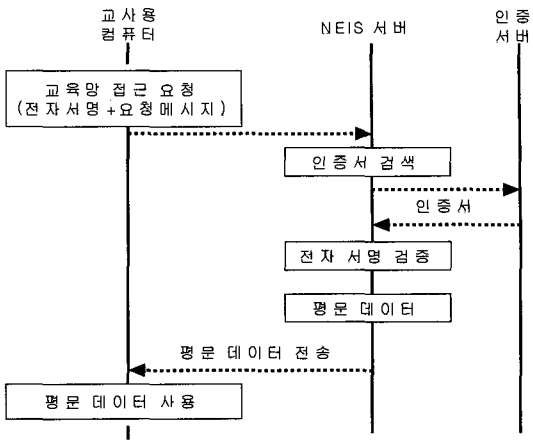


그림 3. NEIS의 프로토콜 흐름도

고사 등 교무학사 및 성적관련 처리를 위해 특정한 기간에 업무 프로세스의 증가와 집중 현상이 뚜렷하다. 기존의 교육행정정보시스템은 각 교육청 단위의 서버에 집중되는 구조로 구성되어 있어 데이터베이스의 집중화로 데이터의 안전성이 취약하다. 또한 서버가 한곳으로 집중되어 있는 관계로 많은 서비스 요청의 경우 네트워크 트래픽의 병목 현상을 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 인증에 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 접근통제

와 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 교육행정정보시스템의 신뢰성에 따른 논란으로 많은 사회적 갈등을 초래하였다.

2.4.2 기존 NEIS 시스템의 구성

그림 4는 기존 교육행정정보시스템의 구조를 나타내고 있다.

그림 4를 기반으로 설명하면 동작 과정은 다음과 같다.

① 사용자는 단위학교의 관리자에게 인증서를 신청한다.
② 단위학교의 관리자는 교육청의 인증서 등록관리자에게 인증서 발급 신청서를 제출한다.
③ 등록관리자는 인증정보를 확인 하고 교육인적자원부의 인증서 관리자에게 인증서 발급 신청서를 제출한다.
④ 교육인적자원부의 인증서 관리자는 인증서 발급 신청서를 인증기관(한국전산원)에 제출한다.
⑤ 인증기관은 인증서 등록정보를 교육인적자원부의 인증서 관리담당자에게 전달한다.

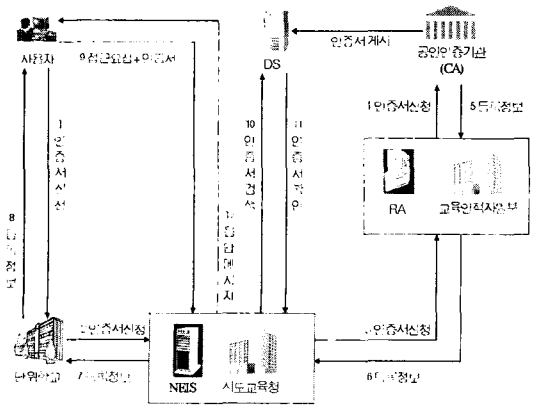


그림 4. 기존 NEIS 시스템의 구성

⑥ 교육인적자원부의 인증서 관리자는 교육청의 등록관리자에게 등록정보를 전달한다.
⑦ 교육청의 등록정보 담당자는 인증서 등록정보를 단위학교의 인증서 관련 담당자에게 전달한다.
⑧ 단위학교의 관리자는 등록정보를 사용자에게 전달한다.
⑨ 인증서 등록정보를 접수한 사용자는 등록정보를 이용하여 인증기관에 온라인으로 인증서 발급을 신청하고 인증서발급이 이루어지면 NEIS에 접근요청 메시지를 보낼 수 있다. 동시에 인증기관은 디렉토리서버(DS)에 인증서를 게시한다.
⑩ 접근요청 메시지를 수신한 NEIS 서버는 인증기관의 디렉토리 서버에서 인증서를 검색한다.
⑪ 인증서의 유효성이 확인되면 NEIS 서버는 사용자의 접근을 허가한다.
⑫ 사용자는 NEIS 서버로부터 응답 메시지를 받는다.

Ⅲ. 제안된 교육행정정보시스템(Role Based Access Control NEIS)

2장에서 기술한 바와 같이 기존 교육행정정보시스템의 문제인 DB의 무결성, 기밀성, 사용자의 속성에 따른 접근제어 시스템을 구현하기 위해서 보안관리자를 설계하고 보안관리자의 보안관리 모듈에서 사용자의 서비스 정보를 감시하도록 하여 기존 시스템이 가지고 있는 보안 취약성을 해결한다. 본 논문에서는 RBAC에 의한 DB 접근통제를 적용하여 제안된 시스템의 명칭을 RNEIS (Role Based Access Control NEIS)로 한다.

3.1 제안된 RNEIS 시스템

제안된 RNEIS 시스템의 구성은 그림 5와 같으며 시스템의 동작은 다음과 같다.

① 사용자는 공개키, 개인키 쌍을 생성하여 인증기관에 공개키에 대한 인증서를 신청한다.
② 인증기관은 사용자의 등록정보를 조회한 후

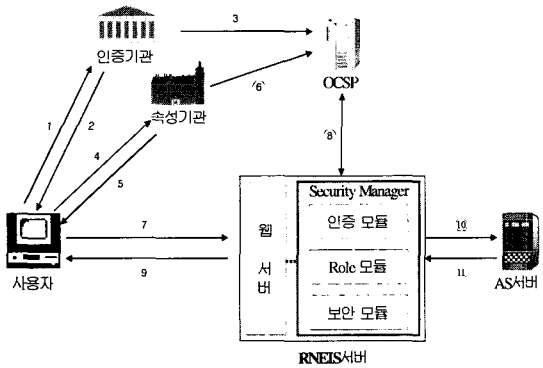


그림 5. 제안된 RNEIS 시스템의 구성도

공개키 소유자의 공개키에 대하여 인증기관의 개인키로 서명하여 발급한다.

③ 인증기관은 공개키인증서를 OCSP 서버에 게시한다.

④ 사용자는 역할지정인증서를 이용하여 속성기관에 역할에 대한 속성인증서를 신청한다.

⑤ 속성기관은 역할지정 인증서를 확인하여 사용자에게 속성인증서를 발급한다.

⑥ 속성기관은 속성인증서를 OCSP 서버에 게시한다.

⑦ 사용자는 공개키인증서+속성인증서와 서비스 요청 메시지를 인증서버에 전송한다.

⑧ 인증서버의 보안관리자는 공개키인증서와 속성인증서의 유효성을 OCSP를 통하여 확인한다.

⑨ 인증모듈과 Role 모듈에서 사용자 인증과 역할인증을 수행하여 정보 자원에 대한 접근권한을 인증하며 세션키를 사용자에게 전송한다.

⑩ RNEIS 서버의 보안모듈은 사용자의 서비스 요청 메시지에 대하여 메시지 인증을 수행하고 서비스 요청 메시지를 어플리케이션 서버(AS)에 전송한다.

⑪ 어플리케이션 서버는 서비스요청 메시지를 수행한다.

3.2 등록관리와 인증서 발급

신원 확인은 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 개인 식별번호(PIN)를 OFF-LINE으로 발급 받아 단위학교의 사용자에게 문서로 전달한다. 사용자는 개인 식별번호를 이용하여 인증기관에 ON-LINE으로 인증서를 신청하여 발급받는다. 이로써 비인가자에 의한 인증서 불법도용 및 개인정보 유출을 방지하여 시스템의 신뢰성을 증가 시킨다.

3.3 전자서명에 의한 사용자 인증과 역할 인증

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.

제안하는 시스템의 전자서명에 의한 사용자 인증과 역할인증은 그림 6과 같다. 인증서와 속성인증서 기반의 인증 및 전자서명, 암호화 기술을 적용하면 개방된 웹 프로그램에서 발생할 수 있는 데이터의 노출, 위·변조, 신원확인 문제, 법적효력 문제 등을 해결하여 교육행정정보시스템의 안전성

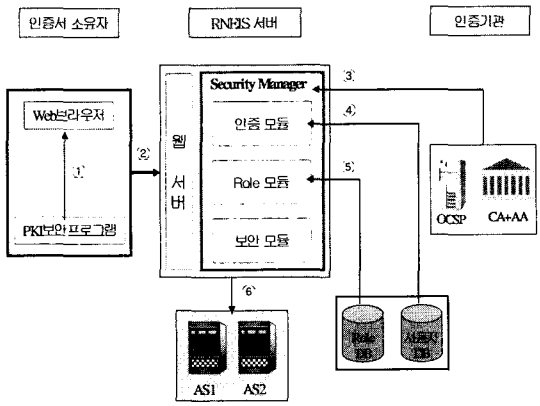


그림 6. 전자서명에 의한 사용자 인증과 역할 인증

과 신뢰성이 확보되도록 할 수 있다. 사용자 신원 확인과 인증서 발급 후 접속인증은 그림 6을 기반으로 설명하면 다음과 같다.

① 사용자가 RNEIS 서버에 접속하기 위해서 사용자의 ID, 패스워드를 입력하면, PKI 보안프로그램은 공개키인증서와 속성인증서를 요구한다. 사용자의 공개키인증서를 이용하여 암호화된 로그인 정보(사용자 인증+권한인증값)가 생성된다. 로그인 정보는 사용자의 ID, 사용자의 공개키인증서, 속성인증서, 사용자의 전자서명을 포함하며, 웹 브라우저에 전달된다.

② 웹 브라우저는 암호화 데이터+전자서명의 정보와 PKC+AC를 인터넷을 통하여 RNEIS 서버(웹서버+보안관리자)로 전송한다.

③ RNEIS 서버의 보안관리자에서 사용자의 인증을 위하여 접근제어와 전자서명을 검증한다. 이의 검증을 위하여 OCSP 서버의 인증서폐지목록을 검색하여 공개키인증서와 속성인증서의 유효성을 검증한다. 검증 결과를 인증모듈과 Role 모듈로 전달한다.

④ 인증모듈은 사용자 DB와 인증서를 비교하여 사용자를 인증한다. 인증을 수행 후 처리 결과를 보안관리자에 반환한다.

⑤ Role 모듈은 보안관리자로부터 사용자 인증 결과를 받은 후 속성 인증서와 Role DB로부터 사용자의 역할을 검증하고 사용자에게 응용프로그램에 대한 접근 권한이 있는지 확인하여 보안관리자에게 결과를 반환한다.

⑥ 보안관리자는 접근권한이 확인되면 보안모듈을 통하여 메시지에 대한 복호화 및 메시지 인증 절차를 수행한 후 어플리케이션 서버에게 서비스 메시지를 전송한다.

3.4 RBAC에 의한 접근제어와 DB 암호화

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

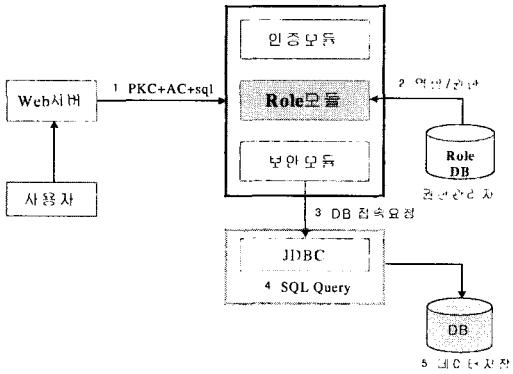


그림 7. RBAC에 의한 접근 제어와 DB 암호화

이외의 사용자가 정보의 변경 및 조회하는 것을 방지할 필요가 있는 업무에 적용한다. 또한 본 논문에서는 데이터 중에서 교무학사, 성적관련정보 등 학교단위에서 관리해야 하는 DB 데이터에 한해서 암호화를 적용한다. RBAC에 의한 접근제어와 DB 암호화 구조는 그림 7과 같으며 동작순서는 다음과 같다.

① RSA+SHA1 전자서명 알고리즘을 적용하며 PKC+AC+sql 메시지를 보안관리자에게 전송

② 보안관리자는 PKC와 AC를 이용하여 사용자 인증, 역할을 인증하며 Role 모듈은 Role DB을 조회하여 해당하는 권한을 부여한다.

③ 권한이 부여된 후 보안모듈은 sql 메시지를 복호화한 다음 메시지 인증 절차를 수행하고 메시지가 인증이 되면 sql 쿼리를 DB에 전송한다.

④ DB에서 sql 쿼리가 수행된다.

⑤ 처리 결과가 DB에 저장된다.

3.5 RNEIS 시스템의 수행 프로토콜

3.5.1 구성요소

RNEIS 구성요소는 그림 8과 같다.

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

시 인증이 필요한 주체

② 속성인증서 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) : RNEIS 시스템 사용자와 서버에게 인증서를 발급해 주거나 관리 기능을 제공해주는 신뢰기관이다.

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

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

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

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

3.5.2 표기

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(RNEIS 서버)의 ID

KR_v : 검증서버의 개인키

KU_v : 검증서버의 공개키

KR_{au} : 인증기관 CA의 개인키

KU_{au} : 인증기관 CA의 공개키

E_{KRau} : 인증기관의 개인키로 암호화, 전자서명

D_{KUau} : 인증기관의 공개키로 복호화

C_A : 사용자 A의 공개키 인증서

C_V : 검증서버의 공개키 인증서

T_i : 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

Ro_i : 속성기관에 의해서 인증된 역할 속성

S_i : 서버에 대한 서비스 요청 메시지

K_s : Role Secure Session Key

$H(SQL)$: SQL 메시지를 일방향 해쉬

n : Replay Attack 방지를 위한 비표

$E_{Ks}(D)$: 세션키 K_s로 비밀키(대칭키)암호화

|| : 연접

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 : $C_A \| KU_{au}$

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

③ A : $D_{KUau}[C_A]$

$= D_{KUau}[E_{KRau}(T_A \| ID_A \| KU_a) \| H(T_A \| ID_A \| KU_A)]$

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

2) RNEIS 서버(VS)의 공개키 인증

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

⑤ CA -> VS : $C_V \| KU_{au}$

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

⑥ VS : $D_{KUau}[C_V]$

$= D_{KUau}[E_{KRau}[(T_V \| ID_V \| KU_v)] \| H(T_V \| ID_V \| KU_v)]$

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

3) RNEIS 서버(VS)가 A의 공개키 인증서 검증

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

⑧ VS : $D_{KUau}[C_A] = D_{KUau}[E_{KRau}(T_A \| ID_A \| KU_a)]$

사용자 A는 인증기관에 대하여 공개키 인증서 발급을 요구하여 신청한다. 사용자 A의 경우 인증기관은 다음 형태의 인증서를 발급한다. KR_{au}

은 인증기관에 의해 사용되어 지는 개인키이다. 사용자 A는 서비스를 이용할 때 인증서를 전달하여 주며, 서버 VS는 사용자의 인증서를 읽어 다음과 같이 인증서를 확인한다. 사용자는 자신의 식별자 ID_A와 공개키 KU_a를 서버 VS에게 전송한다. 타임스탬프 T_A는 인증서의 현재성이 정당함을 확인한다.[15]

인증서를 수신한 서버 VS는 인증서를 복호하기 위하여 인증기관의 공개키 KU_{au}를 사용한다.

$$D_{KUau}[C_A] = D_{KUau}[E_{KRau}(T_A||ID_A||KU_a)] = (T_A||ID_A||KU_a)$$

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

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

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

인증서를 수신한 서버 VS는 인증서를 복호화하기 위하여 인증기관의 공개키 KU_{au}를 사용한다. 인증서는 인증기관의 공개키를 사용하여야 만이 읽을 수 있기 때문에, 서버 VS에 대한 인증서가 인증기관으로부터온 것임을 확인할 수 있다.

$$D_{KUau}[C_V] = D_{KUau}[E_{KRau}(T_V||ID_V||KU_v)] = (T_V||ID_V||KU_v)$$

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

3.5.4 권한 인증

1) 속성 인증서 신청

① A -> AA : $(ID_A||KU_a||C_A||Ro_i)||E_{KRa}[H(ID_A||KU_a||C_A||Ro_i)]$

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

2) 속성 권한 부여와 인증

② AA -> A : $R_A||KU_{aa}$

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

3) 속성 인증서 복호화

③ A : $D_{KUaa}[R_a]$

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

$= (T_r||ID_A||KU_a||Ro_i)$

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.[17] 즉 속성 인증서는 소유자의 공개키 인증서의 시리얼 번호와 발행자가 기술되고 속성인증서를 검증할 때는 공개키 인증서와 조합해서 이용한다.

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

여기서 Ro_i는 속성인증서 소유자의 권한 및 역할이 되며 응용프로그램 서버 AS 들의 프로그램 사용권한이나 역할이 된다. 타임스탬프 T_r은 속성인증서의 현재성이 정당함을 증명한다.

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

속성 인증서를 수신한 서버 VS는 속성 인증서를 복호하기 위하여 속성기관의 공개키 KU_{aa}를 사용한다. 속성 인증서는 속성기관의 공개키를 사용하여 읽을 수 있기 때문에 이것은 사용자 ID_A의 권한 및 역할을 속성기관이 보증하게 된다.

3.5.5 RNEIS 서버(VS)와 응용프로그램서버(AS) 대한 접근 요청

1) 공개키인증서+속성인증서를 이용한 접근

① A -> VS : $(ID_A||KU_a||R_A||C_A||S_i)||E_{KRa}$

$[H(ID_A||KU_a||R_A||C_A||S_i)]$

② VS -〉 A :

$M_s=(ID_A||S_i||ID_V||C_V)||E_{KR_V}(K_s)||E_{KR_s}[H(K_s)]$

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

③ A : 메시지 수신

$M_s=(ID_A||S_i||ID_V||C_V)||E_{KR_V}(K_s)||E_{KR_V}[H(K_s)]$

$D_{KU_V}[E_{KR_V}(K_s)]$로 복호화, 세션키 공유

④ A -〉 VS -〉 DB :

$[(ID_A||KU_A||S_i||C_A||R_a)||E_{KR_a}(K_s||H(SQL)||n||SQL)]$

서비스 신청자 A는 식별자 ID_A와 서비스 요청 메시지 S_i 공개키 인증서 C_A와 속성인증서 R_A로 이루어진 요청메시지 $(ID_A||KU_a||R_A||C_A||S_i)$를 사용자의 개인키 KR_a로 서명 및 암호화하고 $E_{KR_a}[H(ID_A||KU_a||R_A||C_A||S_i)]$와 함께 서버 VS로 전송한다.

RNEIS 서버 VS는 사용자의 공개키인증서와 속성인증서를 검증하고 사용자의 공개키를 이용하여 사용자 인증은 물론 권한과 역할을 검증한다. 공개키인증서와 속성인증서를 검증한 RNEIS 서버 VS는 사용자에게 응용프로그램의 암호화에 사용할 Role Secure Session Key K_s를 VS의 개인키 KR_v로 서명하고 암호화 한 다음 VS의 공개키 인증서 $C_V=E_{KR_{ca}}[(T_V||ID_V||KU_v)||H(T_V||ID_V||KU_v)]$와 메시지 M_s를 함께 전송한다.

단 서버 VS로부터의 메시지 M_s는

$$M_s=(ID_A||S_i||ID_V||C_V)||E_{KR_V}(K_s)||E_{KR_s}[H(K_s)]$$

이다.

사용자는 서버 VS로부터 받은 메시지 M_s를 서버 VS의 공개키 KU_v를 이용하여 복호화한다음 SQL 메시지를 일방향 해쉬함수를 이용하여 $H(SQL)$의 메시지 인증코드를 생성한다. 그리고 SQL문과 Replay Attack을 방지하기 위한 Nonce를 함께 전송한다.

이때 전송메시지는 M_s

$$M_s=(ID_A||S_i||C_A||R_a)||E_{KR_a}[K_s||H(SQL)||n||SQL)]$$

3.5.6 응용프로그램(AS) 접근 및 파일 암호화

① A -〉 VS -〉 DB :

$M_s=(ID_A||S_i||C_A||R_a)||E_{KR_a}[K_s||H(SQL)||n||SQL)]$

$D_{KU_a}[E_{KR_a}(K_s)]$ 세션키 복호화, 해쉬를 통한 SQL 문 인증, SQL Injection 방지

② DB : SQL Query 실행

③ DB -〉 VS -〉 A :

$M_D=(ID_A||S_i||C_V)||E_{K_S}(D)||E_{KR_V}[H(D)||n]$

④ A : $D_{KU_V}[E_{KR_V}[H(D)||(n)]]$ 데이터 복호화

접근제어의 정책에 따라 서버 VS에서 사용자 인증과 권한 및 역할에 대한 검증후에는 사용자는 RNEIS 시스템의 DB 서버에 접근할 수 있게 된다. DB 서버에 접근하게 될 때는 사용자의 속성 인증서에서 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 부여 할 수 있다.

사용자로부터 다음과 같은 메시지

$$M_s=(ID_A||S_i||C_A||R_a)||E_{KR_a}[K_s||H(SQL)||n||SQL)]$$

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 DB 서버에 전달하게 된다.

DB의 SQL Query 실행이후에 서버는 DB에 저장된 데이터 파일을 암호화하고 해쉬 함수를 이용하여 Data에 대한 해쉬값 $H(D)$를 구하고 서버의 개인키 KR_v로 서명하여 전달한다.

$$M_D=(ID_A||S_i||C_V)||E_{K_S}(D)||E_{KR_V}[H(D)||n]$$

가 전송되는 데이터로서 데이터 파일에 대한 파일 암호화는 이미 서버와 사용자 사이에 공유하고 있는 Role Secure Session Key 인 K_s를 이용하여 대칭키 암호 방식으로 실행한다. DB에 저장되

어 있는 파일의 크기가 큰 경우는 파일 암호화시 암호학적 연산에 많은 시간이 걸린다. 따라서 데이터 파일의 암호화에는 암호화 속도가 빠른 대칭키 암호 방식을 사용하고 인증서, ID, 세션키, 메시지 인증코드등의 암호화 또는 서명에는 공개키 암호를 사용하여 사용자의 인증, 전자서명, 기밀성, 무결성등을 확보한다.

서버 VS로부터 데이터 M_D를 수신한 사용자는 서버의 공개키 KU_t를 이용하여 *Nonce* n과 메시지 인증값 $H(D)$를 복호화하고 세션키 K_s를 이용하여 암호화된 데이터 파일을 복호화한다. 또한 데이터 파일을 해쉬하여 메시지 인증값과 비교하여 신뢰된 메시지인지를 확인한다.

Ⅳ. 제안된 RNEIS 시스템의 구현과 분석

시스템 개발은 CPU 700MHz의 Windows 2000 서버운영체제에서 개발언어는 java jdk 1.4와 tomcat 4.1서버, DBMS는 mySQL4.0을 사용하여 구현하였다.

4.1 웹 기반의 RNEIS 시스템 구현

그림 9는 RNEIS의 관리자로 접속한 경우의 화면으로 관리자가 가질 수 있는 메뉴로 구성되어 있다. 관리자는 사용자의 속성인증서를 조회하거나 발급하는 권한과 역할을 가지고 있다. 또한 사용자의 역할지정과 취소 등의 권한을 소유하고 있다.

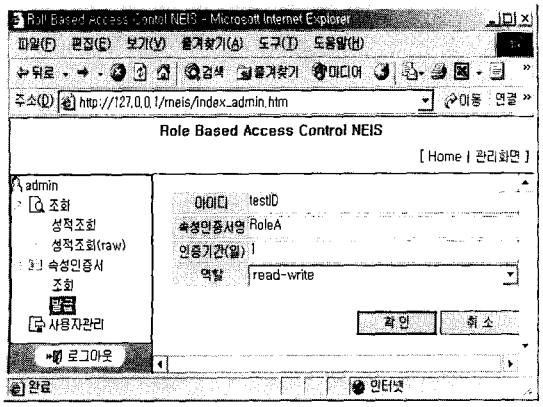


그림 9. RNEIS 시스템 구현 화면

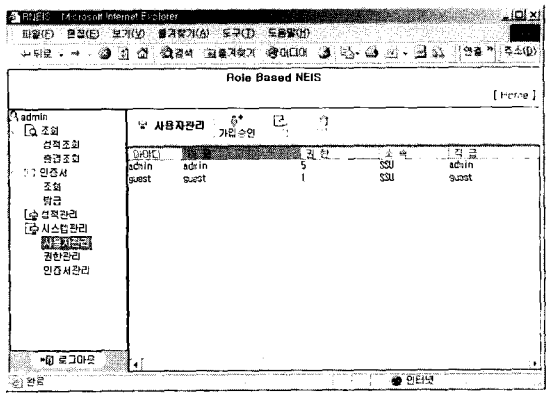


그림 10. RNEIS시스템의 권한관리 화면

그림 10은 사용자의 권한등에 대하여 관리자가 부여할 수 있는 권한 레벨을 보여준다. 권한은 사용자의 권한과 역할에 따라 각기 다른 권한을 허가할 수 있다. 즉 사용자의 직책이나 임무등에 따라 사용자가 소유할 수 있는 역할이 변화하며 해당 역할에 따라 인증서의 검증이 이루어진다.

그림 11은 DB에 접속하여 암호하기 전의 컬럼 내용을 나타낸다. 사용자는 보안이 필요한 데이터에 대하여 암호화한 데이터를 저장할 수 있다. 사용자의 개인키로 전자서명하여 암호화한다.

그림 12는 DB를 암호화한 후의 컬럼의 내용을 보여준다. 컬럼의 내용이 사용자의 개인키로 암호화 되어 있으며 사용자의 공개키는 공개되어 있으므로 해당 공개키를 조회할 수 있는 권한이 있는 다른 사용자들은 파일의 내용을 조회하여 볼 수 있다. 즉 속성인증서의 Role 속성을 이용하여 같은

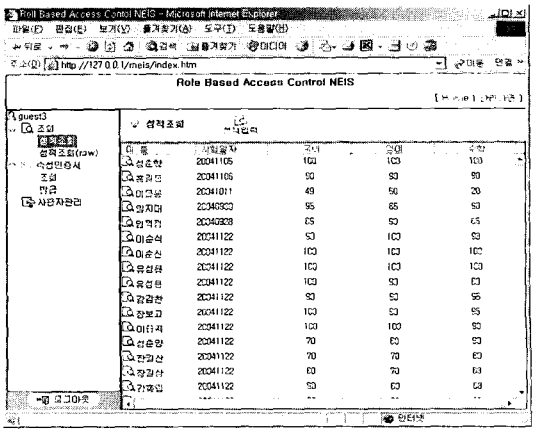


그림 11. DB를 암호화 하기전의 컬럼 내용

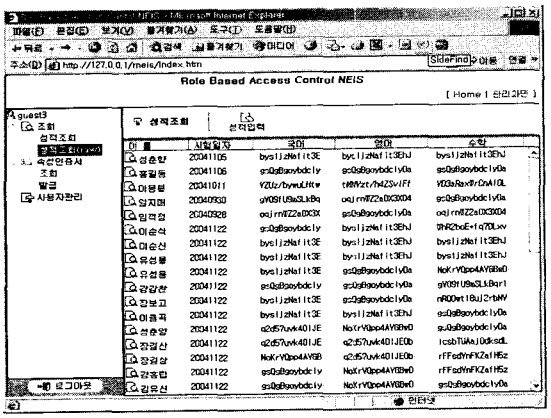


그림 12. DB 암호화 이후의 컬럼 내용

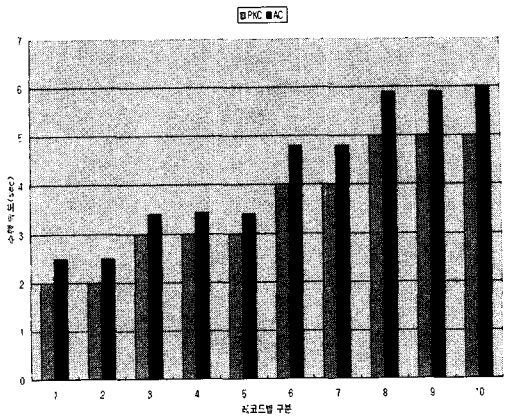


그림 13. DB데이터 암(복)호화 수행 속도

속성 그룹에 속한 사용자들은 공개키를 서로 찾을 수 있도록 하여 파일의 복호화가 가능하다.

4.2 수행시간 비교

RNEIS 시스템의 암(복)호화 수행 속도 분석을 위하여 Visual Studio 6.0의 Active Control Test Container의 Log 분석 기능을 이용하여 웹 브라우저에서 처리되는 과정이 소요되는 시간을 측정하였다. 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 보안모듈의 도입으로 불가피한 경우라고 할 수 있다.

표 1. DB 데이터의 암복호화 분석

record	PKC사용	AC사용
100	2.1[sec]	2.5[sec]
200	2.1[sec]	2.5[sec]
300	3.0[sec]	3.4[sec]
400	3.0[sec]	3.4[sec]
500	3.0[sec]	3.4[sec]
600	4.3[sec]	4.8[sec]
700	4.3[sec]	4.8[sec]
800	5.4[sec]	5.9[sec]
900	5.4[sec]	5.9[sec]
1000	5.4[sec]	6.0[sec]

4.3 DB 암호화에 따른 보안평가 분석

제안한 RNEIS는 기존 NEIS가 가지고 있는 문제를 최대한 해결할 수 있도록 개선하였다.

① 단위 학교의 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 존

재하였으나 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.

〈표 2〉는 제안 시스템(RENIS)과 기존 시스템(NEIS)의 프로토콜의 비교 분석 결과를 나타낸다.

V. 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 역할필드를 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 속성인증서에서 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. DB 데이터의 암호화는 DB를 생성

표 2. 프로토콜 분석

구분	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 재발급시 프로세스 증가
부인방지	• 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
검증 서버	• 단위학교의 응용서버나 서비스에 대한 권한만을 저장하여 소량의 프로세스 발생	• 교육청 단위의 응용서버와 서비스에 대한 권한을 저장하여 대량 프로세스를 발생
네트 워크 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

하는 사용자의 개인키를 이용하여 암호화함으로써 데이터의 저장과 생성은 데이터의 소유자만이 가능함을 보였다. 복호화에는 공개키를 입수할 수 있으므로 데이터를 암호화한 사용자 뿐만이 아니라 권한이 있는 관리자의 경우도 공개키를 입수할 수 있는 경우는 복호화가 가능하도록 하여 암호된 데이터가 관리됨을 보였다. 또한 일방향 해쉬함수를 이용하여 SQL문의 메시지 인증기능을 구현하여 DBMS의 보안을 강화하였다. 사용자와 서버간의 데이터파일 전송은 대칭키암호화 및 전자서명을 동시에 수행한 후 교환함으로써 파일 암호화의 속도가 지연되는 것을 방지함은 물론, 메시지에 대한 부인방지 기능과 데이터의 기밀성을 보장하였다. 제안 시스템의 분석에서 기술한 바와 같이 암호화의 과정이 추가되는 과정에서 수행시간의 지연이 발생하였다. 향후 이 부분에 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] Rabi S. Sandhu, Edward J. Coyne, "Role-Based Access Control Models", IEEE Computer, pp. 38-47, February 1996.

[4] Ravi S. Sandhu, "Rational for the RBAC96 Family of Access Control Models", In Proceedings of 1st ACM Workshop on Role-based Access control, ACM, Article No. 9, 1996.

[5] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein and Charls E. Younman, "Role-based access control model", IEEE Computer, Volume 29, pp. 38-47, February 1996.

[6] John Barkly, "Comparing Simple Role Based Access Control Models and Access Control Lists", In Proc.of ACM RBAC 97, pp. 127-132, 1997.

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

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

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

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

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

[12] ITU_T Recommendation X.509, "Public-Key And Attribute Certificate Frameworks", ISO/IEC 9594-8, May 2001.

[13] 문봉근, 홍성식, 유황빈, "RSA 방식을 이용한 데이터베이스 암호화 구현", 통신정보보호학회 논문지, 제3권 제2호, pp53-62. December 1993.

[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.

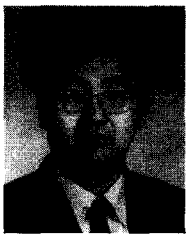
[15] Adams, et al, "Internet X.509 Public Key Infrastructure Time-Stamp Protocol", IETF RFC3161, August 2001

[16] S. Farrell, R. Housley, "An Internet Attribute Certificate Profile for Authorization", IETF RFC 3281 April 2002.

〈著者紹介〉

유 두 규(Du-Gyu Ryoo) 정회원
1984년 2월 : 숭실대학교 전기공학과 학사
2001년 2월 : 숭실대학교 컴퓨터교육과 석사
2001년 3월~현재 : 숭실대학교 대학원 컴퓨터학과 박사과정
1984년 3월~현재 : 세명컴퓨터고등학교 인터넷영상 부장
〈관심분야〉 네트워크 보안, 정보보안, DB보안, DRM, 암호학
e-mail : bima@dreamwiz.com

문 봉 근(Bong-Keun Moon) 정회원
1988년 2월 : 수원대학교 전자계산학과 학사
1993년 8월 : 광운대학교 전자계산학과 석사
2001년 3월~현재 : 숭실대학교 대학원 컴퓨터학과 박사과정
1988년 11월~1993년 6월 : 한신공영(주) 전산실
1993년 6월~1998년 3월 : 한라정보시스템(주) 마이스터 IS팀
〈관심분야〉 네트워크, 침입탐지시스템, 정보보안
e-mail : mbk2000@chol.com



전 문 석(Moon-Seog Jun) 정회원
1980년 2월 : 숭실대학교 전자계산학과 학사
1986년 2월 : University of Maryland 전산과 석사
1989년 2월 : University of Maryland 전산과 박사
1989년 : Morgen State University 전산수학과 조교수
1989~1991년 : New Mexico State University 부설 Physical Science Lab. 책임연구원
1991년~현재 : 숭실대학교 정보과학대학 정교수
〈관심분야〉 네트워크 보안, 컴퓨터 알고리즘, 암호학
e-mail : mjun@computing.ssu.ac.kr