PKC와 AC를 이용한 데이터베이스 보안 및 효율적인 서비스 제공 연구

안민호*, 송오영*, 박세현*,

*중앙대학교, 전자전기공학부

A Study of Database Security and Efficient Service with Public Key Certificate and Attribute Certificate

Min ho Ahn*, Oh young Song*, Se hyun Park*

*Chung-Ang University, School of Electrical & Electronic Engineering

요 약

본 논문에서는 기존 데이터베이스의 보안적인 취약점에 대해서 알아보고 보안적인 취약점을 해결할 수 있는 방법으로써 Public Key Certificate와 Attribute Certificate를 이용한 서비스 모델을 제시한다. 즉 Public Key Certificate를 이용해서 인증 강도를 높이고 Attribute Certificate를 이용해서 데이터베이스를 사용하는 사용자들에게 Role 기반 권한을 제공해서 사용자들이 데이터베이스를 사용할 수 있는 권한을 손쉽게 세분화 할 수 있는 방법을 제안한다. 또한 공개키 기반 암호화를 사용해서 사용자가 특정 자료를 암호화해서 데이터베이스에 저장함으로써 비도덕적인 데이터베이스 관리자나 혹은 데이터베이스 시스템 내부의 침입자에 의해서 사용자의 데이터가 유출되는 것을 방지하는 방법을 제안한다.

I. 서론

현재 인터넷의 발전은 눈부시게 이뤄지고 있다. 통신망의 발전, 전송 기술의 발전 그리고 통신 하드웨어의 발전 등이 이러한 결과를 낳고 있다. 이로 인한 인터넷 사용자의 수도 폭발적으로 늘어나고 있고 인터넷을 활용하는 업무의 종류도 좀 더 전문화가 되고 있다. 즉 인터넷 환경이 발전함에 따라서 효율적인 업무를 추구하기 위해서 인터넷 망을 사용하는 기업이 늘고 있고 일반 사용자를 대상으로 하는 e-business가 활성화되고 있다. 인터넷의 특성상 앞으로 많은 기술들이 인터넷 활성화를 가속화할 것이고 수많은 사용자들이 실생활에 더욱 밀접하게 인터넷 망을 이용하게 될 것이다. 지금까지 인터넷은 그 특유의 개방성 때문에 많은 새로운 기술들이 인터넷에 적용될 수 있었지만 이러한 개방성은 인터넷 망을 통과하는 많은 데이터들에게도 적용이 됨으로써 보안적인 문제점들이 대두되게 되었고 이러한 보안적인 취약함을 해결하기 위해서 많은 기술들이 나오게 되었다. 이러한 기술들을 사용한다면 인터넷 망에 흘러가는 데이터를 높은 강도로써 암호화를 할 수 있다. 또한 인터넷 망을 통과하는 모든 정보들은 효율적인 사용을 위해서 데이터베이스 시스템을 사용한다. 결국 다양한 정보를 저장하는 데이터베이스 내부의 보안도 중요하다는 것을 알 수 있다. 그러나 기존의 데이터베이스는 인증 절차부터 보안적 취약점을 드러낸다. 기존의 데이터베이스는 아이디/패스워드 기반 인증과 IP based access control을 하기 때문에 발전해 가는 공격 방법들에 의해서 쉽게 공격당할 수 있다. 그러므로 이 논문에서는 여러 사용자들이 접근할 수 있는 데이터베이스를 X.509 공개키 기반 인증서와 속성 인증서를 사용해서 안전하게 관리할 수 있는 모델에 대해서 제시한다.

II PKI와 PMI

PKI(Public Key Infrastructure)와 PMI(Privilege Management Infrastructure)는 이미 많은 논문에서 다루었던 내용이므로 이 논문에서는 자세하게 설명하지 않을 것이다. 기본적으로 PKI는 X509 Public Key Certificate를 사용함으로써 통신 대상자를 인증하고 검증하는 시스템이고 PMI는 Attribute Certificate를 사용해서 Public Key Certificate로 인증된 인증 대상자와 인증 대상자가 가지고 있는 권한 관계를 인증하는 시스템이다. 즉 PKI는 사용자 인증을 주목적으로 하고 PMI는 사용자의 권한 확인을 주목적으로 한다.

III 기존 Database의 보안적 취약점

우선 인증 절차의 취약점을 꼽을 수 있다. 기존 데이터베이스는 사용자 인증을 아이디와 패스워드 기반으로 진행하고 사용자의 IP 기반 접근 제어를 하기 때문에 다음과 같은 Attack 방법들에 취약하다.

- 패스워드에 대한 Bruteforce Attack
- Packet Sniffing에 의한 ID/PW 유출
- IP Spoofing
- Replay Attack
- 데이터베이스 내부의 위험

앞서 언급했던 것처럼 공개된 인터넷 망에서 아이디와 패스워드 기반 인증을 사용하고 있기 때문에 아이디와 패스워드에 대한 Sniffing이 있을 수 있다. 또한 사람이 기억할 수 있고 조합할 수 있는 패스워드의 길이에는 한계가 있기 때문에 패스워드에 대해서 Bruteforce Attack이 가능하다. 또 하나의 취약점은 클라이언트의 접근 시 클라이언트 IP 기반 접근 제어를 하기 때문에 공격 방법 중 IP Spoofing 기법을 사용한다면 어렵지 않게 Database 내부 자료를 도용할 수 있다. 그리고 아이디와 패스워드를 암호화한다고 할지라도 암호화된 데이터 자체를 다시 사용하는 재전송 공격에 노출될 위험이 있다.

또한 IP based access control은 지금처럼 유동적인 인터넷 사용자를 처리하는데는 한계가 있다. 그리고 Database의 자료들도 실제로는 컴퓨터 내부 저장장치의 지정된 장소에 file로써 저장이 되기 때문에 Database file을 보안하는 것도 시급하다. 지금은 데이터베이스에 사용자를 추가할 때 데이터베이스 관리자가 수동으로 사용자의 권한과 접근할 수 있는 데이터베이스와 테이블을 추가하는데 좀더 유동적인 권한 관리를 한다면 데이터베이스 관리를 좀 더 손쉽게 할 수 있을 것이다.

마지막으로 기존의 데이터베이스 관리자는 강력한 권한이 있어서 다른 사용자의 내부 자료도 마음만 먹으면 언제든지 획득할 수 있었다. 이러한 문제점은 데이터베이스 사용자의 사생활 침해나 다른 중요한 손실을 가져 올 수 있고 데이터베이스 시스템 내부의 침입자로부터는 안전하지 못하다는 취약점이 있다.

IV. Secure Database Service 모델

본 논문에서는 위와 같은 보안적 취약점을 해결하기 위한 모델로서 그림1과 같은 모델을 제안한다.

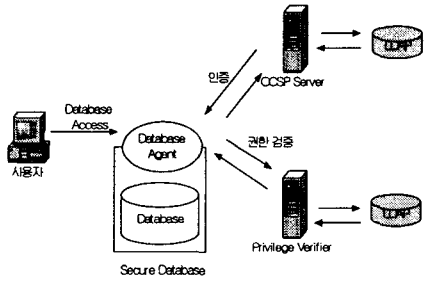

그림 1 Secure Database Model

각 구성 모델을 설명하기에 앞서 사용자의 데이터베이스 접근시 전체적인 절차에 대해서 설명한다. 전체적인 절차에 대한 것이 아래의 그림 2이다.

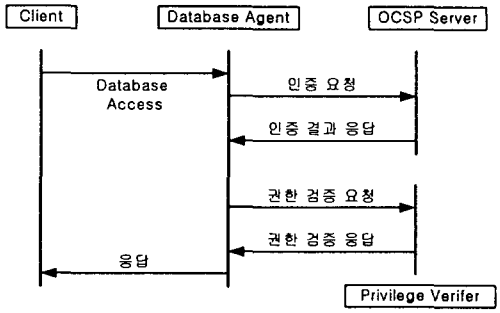

그림 2 Database Access 절차

그림 2를 보면 Client는 Database Agent에 Database Access Message를 전송한다. Database Access Message는 Client에서 사용자의 Public Key Certificate와 Attribute Certificate를 입력으로 받아 만든 Message로써 Database Agent가 OCSP server와 Privilege verifier에게 요청을 할 수 있는 데이터들을 포함하고 있다.Database Agent가 Database Access Message를 받게 되면 내부적으로 OCSP request를 만들어서 OCSP server에게 보내고 응답을 받는다. 이 검증 결과가 올바르다면 권한 검증 요청을 하고 만일 올바르지 않다면 Database에 접근할 권한이 없는 사용자로써 취급하고 그에 따른 응답을 보내게 된다.

이제 Secure Database Model의 구성 요소 중 새로운 Module에 대해서 설명하겠다.

1) Client

Client는 기본적으로 Database Agent와 메시지를 주고받아 Database Access를 인증 받기 위한 사용자 Module이다. Client는 사용자로부터 Public Key Certificate와 Attribute Certificate에 대한 정보를 입력받아 메시지를 구성한다. 또한 Access 허락을 받고 데이터베이스에 자신의 데이터를 읽거나 저장할 때 암호화 과정을 수행 할 수 있다. 즉 Client는 암호화를 할 대상에 대해서 암호화 강도를 달리해 랜덤 대칭키를 생성한다. 데이터를 암호화/복호화를 하는데 비대칭키를 사용하면 많은 대칭키에 비해 많은 시간이 소요되기 때문에 암호화를 하기 위해서 대칭키를 사용한다. 대칭키를 생성해서 데이터를 암호화를 하고 데이터와 같은 장소에 대칭키를 사용자의 공개키로 암호화를 해서 저장한다. 대칭키를 공개키로 암호화하는 것은 대칭키의 키 길이가 자료에 비해 월등히 작은 크기를 가지고 있고 후에 대칭키를 획득하기 위해서는 자신만이 가지고 있는 자신의 개인키로 복호화를 해서 획득한다.

2) Database Agent

이 모듈은 Client와 통신을 해서 사용자의 인증과 권한 검증을 수행하는 모듈이다. 통신을 할 때 Server Side SSL을 사용해서 외부로부터 제 3자가 통신 내용을 알 수 없게 한다. 이 모듈을 둠으로서 기존 Database Module을 크게 변경하지 않고 인증과 권한 검증을 할 수 있다. 또한 실시간으로 Database에 접속해서 사용하는 사용자의 권한 정보를 Database Module에게 알려줌으로써 사용자 권한에 따른 유동적인 자료 제공을 할 수 있다.

3) Database Module

기존에 사용하는 Database Module로써 Database Agent와 내부 통신을 하면서 사용자에게 자료를 제공한다.

V. 취약점 보안 및 성능 향상

기존 데이터베이스 모델과의 차이점을 나타내면 다음 표1과 같다.

1) Server Side SSL

클라이언트가 Database Agent에 접근하려고 하면 Database Agent의 인증서를 Database Agent로부터 받게 되고 인증서에서 Database Agent의 공개키를 얻는다. 그리고 클라이언트는 Session Key를 생성해서 Agent의 공개키로 암호화해서 Agent에게 전달하게 된다. 그렇게 함으로써 Session Key를 확보할 수 있고 데이터를 전송할 때 Session Key로 암호화를 함으로써 악의적인 제 3자로부터 Agent와 정당한 사용자의 traffic을 보호할 수 있다.

2) Authentication

기존 데이터베이스의 인증은 아이디/패스워드 기반의 인증이었다. 이러한 인증에 따른 문제점은 앞에서 이야기를 했던 것처럼 전사적 공격(Brute Force Attack)이나 재전송 공격(Replay Attack)으로 쉽게 공격당할 수 있다는 것이다. 그러므로 그림 1에서 제안한 모델에서는 인증에 아이디/패스워드를 사용하지 않고 공개키 기반 인증서를 사용한다.

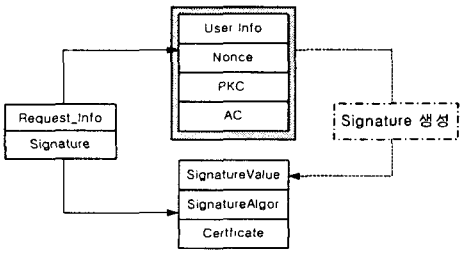

그림 3 Request Message

	기존 모델	제안 모델
인증	ID/PW 기반	PKC 기반
권한	상용자 ID 기준	AC/ID 기반
데이터 암호화	지원 안함	공개키 기반 데이터암호화지원
통신의 암호화	지원 안함	SSL

표 1 제안 모델의 성능향상

우선 클라이언트는 사용자의 인증서와 개인키를 입력받아서 그림 3과 같은 정보를 가진 Message를 생성한다. 이 메시지를 사용하면 패스워드를 사용하지 않으므로 제 3자가 정당한 사용자의 개인키를 훔쳐내지 않는 이상 메시지를 위조하기는 불가능하다. 즉 패스워드 방식의 전사적 공격을 막을 수 있다. 또한 암호화된 같은 메시지를 사용하는 재전송 공격도 메시지 사용마다 달라지는 Nonce를 추가함으로써 막을 수 있다. 인증서를 사용해서 인증을 하면 좋은 점은 Man in the middle Attack도 불가능하다는 것이다. 사용자는 검증서버에 Database Agent의 인증서를 검사해서 다른 자가 검증서버인 것처럼 속이는 것을 방지 할 수 있다.

3) Authorization

그림 3의 메시지를 보면 Public Key Certificate Key Certificate와 Attribute Certificate에 대한 정보를 입력받아 메와 더불어 Attribute Certificate를 전송하는 것을 볼 수 있다. Attribute Certificate는 속성 인증서로써 인증서의 대상이 특정 권한이 있는지를 인증하는 것이다. 이 인증서에 대해 Privilege Verifier에게 검증 요청을 함으로써 권한 검증이 완료된다. 특정 사용자가 인증과 권한 검증을 완료하게 되면 사용자에게 응답을 보내게 된고 Database Agent는 내부 저장소에 인증된 사용자의 기본정보와 권한정보를 저장한다. Database Module이 사용자에게 자료를 제공할 때 Database Agent와 공유하는 내부 저장소의 정보를 바탕으로 권한에 따른 정보를 제공한다. 권한 인증서를 사용함으로써 관리자는 내부의 자료를 자주 수정할 필요가 없어진다. 단지 Database Agent에서 발행하는 권한에 대한 것만 정의해주면 된다. 또한 정책에 따라서 Role-based Attribute Certificate를 사용한다면 데이터베이스 내부의 ACL을 가지고 좀더 세분화된 서비스 제공이 가능하고 Role과 ACL 사이에 대응만 이뤄지면 됨으로 데이터베이스를 관리하는 측면에서 더욱 효율적이다. 예를 들어 3개의 Role 즉 Role A, Role B, Role C에 대해서 사용자의 Attribute Certificate를 발급한다고 가정한다. 사용자에게 발급되는 Attribute Certificate는 이 세 개의 Role 중에서 하나의 Role을 담아서 발급되게 된다. 또한 Database agent는 각 Role에 대한 정책을 가지고 있다. 또한 Access Control List를 사용함으로써 각 Role에서 좀더

세분화해서 데이터베이스 자료 제공을 할 수 있다. 기존의 모델에서 이러한 기능을 사용하기 위해서는 데이터베이스 관리자가 수동으로 사용자의 권한에 관한 내용을 설정해줘야 했다. 이러한 방법은 앞으로 더욱 늘어나는 데이터와 데이터베이스 사용자들을 고려할 때 비효율적인 방법이다.

4) 암호화

사용자가 중요한 특정 파일을 저장할 때 클라이언트를 사용해서 암호화를 할 수 있어야 한다. 왜냐하면 외부적인 보안적인 취약점들을 강화시켰을 지라도 내부적인 보안적 취약점은 강화되지 않기 때문이다. 즉 비도덕적인 관리자가 사용자의 자료를 도용할 수도 있고 혹은 관리자가 작업하는 회사의 동료나 침입자가 관리자의 관리 정보를 얻을 수 있으면 데이터베이스 내부의 자료는 심각한 위험에 노출 될 수 있다. 암호화 절차는 특정 자료를 사용자의 공개키로 암호화를 해서 저장을 한다. 이렇게 암호화된 자료는 공개키와 대응되는 개인키로만 복호화가 되지 않기 때문에 안전하다. 자료를 암호화함으로써 약간의 Overhead가 더 해 질 수가 있으나 자료의 중요도에 따라서는 약간의 Overhead는 감소해야 한다.

VI. 결론

인터넷 통신 기술이 발전함에 따라서 인터넷 사용자들이 급증했고 많은 기업들도 전산화가 이뤄지고 있다. 이러한 가운데에 사용자들의 막대한 자료들이 모이게 되고 좀더 안전하고 효율적인 자료 관리가 필요하게 되었기 때문에 기존 데이터베이스의 보안적인 취약점에 대해서 개선하고 좀더 효율적인 관리를 할 수 있는 데이터베이스 모델에 제안했다. 실제로 구현한 모델은 아니지만 데이터베이스 프로그램을 만드는 회사에서 이러한 모델을 이용한다면 보안적인 면에서 높은 성능향상을 초래할 수 있을 것이다.

참고문헌

[1] S.M. Bellovin and AT&T Bell Laboratories Murray Hill, Security Problems in the TCP/IP Protocol Suite, Computer Communications Review, 1989

[2] Ahmed, Q.N. and Vrbsky, S.V, Maintaining security in firm real-time database systems, Computer Security Applications Conference, 1998

[3] ITUT public-key and attribute certifica te frameworks, X509, 2000

[4] R. Housley, W. Ford, W. Polk, and D. Solo, Internet X.509 Public Key Infrastructure Certificate and CRL Profile, RFC 2459, 1999

[5] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP, RFC 2560, 1999

[6] R. Housley, W. Ford, W. Polk, and D. Solo, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC 3280, 2002