데이터베이스에서 개인정보보호를 위한 정책기반 쿼리 변환기 설계 및 구현

김미영*, 이영록*, 이형효**, 김용민***
*전남대학교 정보보호협동과정
**원광대학교 정보전자상거래학부
***전남대학교 문화컨텐츠학부
e-mail: moodol@lsrc.jnu.ac.kr

Policy-based Query Translator Design and Implementation for the Privacy Protection in Database

MiYeong Kim*, YoungLok Lee*, HyungHyo Lee**, BongNam Noh*
*Interdisciplnary Program of Information Security,
Chonnam National University
**Div. of Information and Electronic Commerce, WonKwang University
***Div. of Culture Contents, Chonnam National University

요 약

인터넷으로 대표되는 정보통신망 및 컴퓨터를 이용한 개인 정보 수집과 활용이 일반화됨에 따라 수집된 개인정보의 불법적인 접근 유출 사례가 증가하고 있다. 현재의 개인정보 이용 환경은 데이터 접근 시 사용자의 질의 내용과 그에 대한 결과가 그대로 노출되어 사용자의 프라이버시를 침해하는 문제를 안고 있다. 본 논문에서는 데이터베이스에서 개인정보보호를 위해 접근제어 정책 기반 쿼리 처리 시스템인 보안 게이트웨이를 설계하고 구현한다. 이 시스템은 클라이언트가 TDS 프로토콜을 이용하여 DBMS에 접근해 정보를 요청할 때 보안 정책을 반영함으로써 단순한 차단은 물론 변환된 쿼리 응답을 한다. 본 시스템은 불법적인 접근에 대한 제어는 물론이고, 정당한 인증자의 실수나 고의적인 개인정보 유출로 인한 경제적, 사회적 손실을 방지할 수 있다. 또한 주민등록번호 등 보안 대상 정보를 제외한 기타 정보에 대한 접근을 허용함으로써 데이터베이스 가용성을 보장한다.

1. 서론

인터넷을 이용한 개인정보 수집과 활용이 일반화됨에 따라 개인 정보의 불법적인 접근 및 유출 사례가 점점 증가하고 있다. 개인정보는 생존하는 개인을 식별하거나 식별할 수 있는 일체의 정보로 당해 정보만으로는 개인을 식별할 수 없을지라도 다른 정보와 용이하게 결합하여 개인을 식별할 수 있는 모든 정보를 말한다. 따라서 성명, 주민등록번호, 개인에 관한 부호, 문자, 음성, 음향 및 영상 등을 포함한다[1].

각각의 개별 정보는 큰 가치를 발휘하기 힘들지만, 산재한 정보들을 한군데 모아 체계적으로 관리한 정보는 큰 가치를 지닌다. 현재 개인정보 이용 환경은 데이터 접근 시 사용자의 질의 내용과 그에 대한 결과가 그대로 노출되어 사용자 프라이버시가 침해되고 있다. 이에 따라 개인정보보호의 필요성과 개인정보보호 기술의 필요성이 대두된다.

대량의 데이터를 체계적으로 저장, 관리하면서 인증, 인가된 이용자에 대해서만 데이터 접근을 허용하는 데이터베이스 시스템은 더욱 보안이 필요하나, 데이터베이스에 저장된 정보를 적절히 보호하지 못해 유출시킴으로써 발생하는 금전적 피해가 심각하다.

데이터베이스 보안은 정보 보호 관리에 있어서 비 인가된 변경, 파괴, 정보 누출을 발생시키는 사건으로부터 보호하기 위한 방법을 말한다[2].

기존의 데이터베이스 보안은 주로 단순한 데이터베이스관리시스템에 대한 접근 제어나 통제를 위한 인증 위주의 정보보호였다. 개인정보보호를 고려한 데이터베이스 모델인 히포크라테스 데이터베이스나 LDHD 방식은 프라이버시 보호라는 개념을 데이터와 함께 저장한다고 하나 레코드나 셀 단위로 제어하기 때문에 데이터 하나하나에 정책을 부여하므로 용량이 커지고 추가 변경이 어렵다. 또한 레코드 단위의 접근 제어로 가상 사설 데이터베이스에서 사용자의 계정별로 특정 테이블에 대한 결과가 달라지는 쿼리를 변형하는 방법으로 개인정보를 보호한다. 그러나 가상 사설 데이터베이스 상에서 레코드 단위로 처리되므로 제공자의 의도대로 개인정보를 보호하기 어렵다.

본 논문에서 제안한 시스템은 클라이언트가 TDS 프로토콜을 이용하여 DBMS에 접근할 때 게이트웨이 상에서 쿼리를 추출하고 파싱하는 과정을 거친다. 파싱된 쿼리는 개인정보보호 정책을 적용해 허가와 치환 또는 거부인 접근 통제를 결정한다. 이로써 클라이언트가 DBMS에 요청하고 응답을 받을 때 보안 정책을 위반한 SQL문을 재작성하여 비인가 사용자에 대한

접근을 차단한다.

논문의 구성은 다음과 같다. 2장의 관련 연구에서는 데이터베이스 관리기술과 개인정보보호 데이터베이스 구현에 대해 알아본다. 3장과 4장에서는 시스템을 설계, 구현하고 개인정보보호 정책을 적용한다. 5장에서 결론을 맺는다.

2. 관련연구

가. 데이터베이스 관리 기술 관련 연구

히포크라테스 데이터베이스는 기존의 관계 데이터베이스 관리 시스템 상에서 프라이버시-정책과 같은 메타데이터를 이용하여 프라이버시 보호 기능을 제공하는 데이터베이스이다. 히포크라테스 데이터베이스에서는 데이터를 수집할 때 데이터 제공자의 데이터 제공여부를 나타내는 프라이버시-선호와 데이터베이스 시스템의 프라이버시-정책이 모두 일치하는 경우에만 데이터를 수집한다. 따라서 데이터 수집에 한계가 있다. 또한 기본적으로 레코드 단위의 접근 제어 방식이기 때문에 셀 단위 접근 제어보다 세밀하지 않다[3].

LDHD (Limiting Disclosure in Hipocratic Database)는 히포크라테스 데이터베이스의 단점을 보완하기 위해 제안된 것으로 셀 단위의 접근제어 방식을 제공하는 새로운 형태의 데이터베이스 보안 모델이다 . 프라이버시를 보호하기 위해서는 사용자 질의가 선호 사용자-목적 테이블을 이용하도록 수정해야 한다. 이를 위해 LDHD에서는 선호 사용자-목적 테이블에서 데이터의 제공을 허용하면 사용자에게 실제 데이터를 제공하고 그렇지 않으면 “NULL ”을 반환하는 case문을 이용한 질의 수정 알고리즘을 사용한다. 하지만 LDHD 방식을 실제 데이터베이스 환경에 적용할 경우첫째 데이터 제공시 데이터베이스를 이용하는 사용자 수만큼 제공자의 선호를 기입해야 한다. 두 번째는 저장하는 메타데이터의 크기가 크다는 것이고, 세 번째는 새로운 데이터 사용자가 추가되는 경우 변경되어야 하는 메타데이터의 크기가 크다는 단점이 있다[4].

데이터베이스에 저장된 정보 주체들이 프라이버시를 보장하면서 SQL 쿼리 역시 가능하게 하는 방법으로 데이터 자체를 암호화하지 않고 치환하는 방법을 이용한다. 이는 정보와 정보 주체 간 연관을 짓지 않으므로 비연결성을 보장하면서도 동시에 산술 연산 및 속성들 간의 교차 연산을 제외한 모든 종류의 SQL 쿼리를 가능하게 한다[5].

또한 레코드 단위의 접근 제어로 사용자 계정별로 특정 테이블에 대한 결과가 달라지는 가상 사설 데이터베이스를 생성하고 지정된 정책에 의해 SQL 문장이 변형되는 방법이 있다. 각각 다른 사용자가 같은 질의를 하지만 정책의 적용을 받아 사용자 별로 다른 값의 SQL 변형 응답을 받는다[6].

나. 개인정보보호 데이터베이스 제품

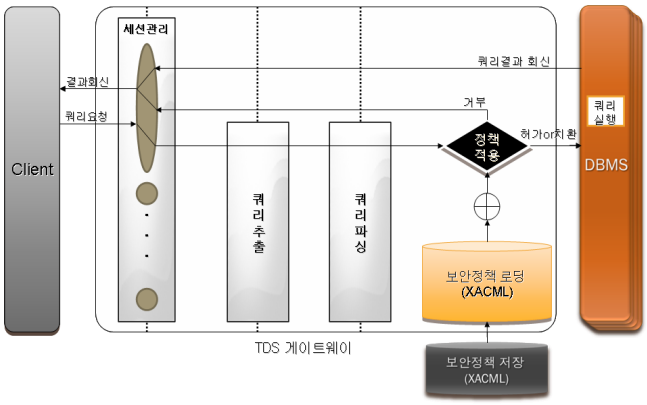
DB-i (SOMANSA, http://www.somansa.com)[7]는 8가지 서로 다른 종류 DBMS들을 통합해 보안 관리하는 특징을 지니고 있다. DB 접근 권한이 있는 쿼리 툴 이용자의 권한 오용까지 차단하기 위해 주민번호, 계좌번호 등 개인 식별번호에 마스킹 처리를 하는 고객 정보보호 쿼리 툴 쿼리 마스크를 보유하고 있다.

SQLCanvas (R2WARE, http://www.r2ware.com)[8]은 DB 보안게이트웨이는 사용자 신원 및 역할 정보를 활용한 보안정책 기반 데이터베이스 접근통제 서비스를 제공하는 제품이다. SQL Canvas 기능으로는 데이터베이스 보안 관리자에 의해 설정된 사용자 신원(ID, 그룹, 역할 등) 기반 접근통제, 보안정책에 기반 한 데이터베이스에 대한 fine-grained 접근통제 등이 있다

지금까지의 관련 연구와 제품들은 모두 개인정보보호 기능을 적용시키고 있지만, 실제로 외부의 불법적인 접근이 아닌 내부자의 정당한 권리를 가진 개발자나 관리자가 다양한 데이터베이스에 불법 접근이 가능하기 때문에 개인정보유출과 침해가 빈번히 일어나고 있다. 강력한 개인정보 보호를 위한 정책을 적용하고 불법 접근을 했더라도 개인 정보 유출을 최소화 할 수 있는 방법이 필요하다.

3. 시스템 설계

클라이언트가 DBMS로 쿼리를 요청하여 응답 받는 주요 구성 모듈의 흐름은 그림1과 같다.



(그림 1) 보안 정책기반 쿼리 변환 처리 설계도

로그인한 클라이언트는 DBMS에게 쿼리를 요청하는 패킷을 보낸다. 이 패킷은 스레드를 이용한 세션 관리를 거쳐 패킷 중 필요한 SQL쿼리를 추출하고 쿼리를 파싱하여 정책을 적용하기 위한 보안 정책을 거친다. 클라이언트는 허가 되거나 적절하게 치환되어 DBMS로부터 회신되는 결과를 보게 된다.

가. 세션 관리

DBMS는 클라이언트와 접속이 제한적이기 때문에 서버의 리소스 관리적 측면에서 세션 관리가 필요하다. 클라이언트가 DBMS에 쿼리를 요청하면 지속적인 연결을 유

지하는 세션을 맺게 되는데, 클라이언트에 대한 응답 속도가 향상되고 분산되는 로드 밸런스 기능으로 서버에 대한 부하가 감소한다.

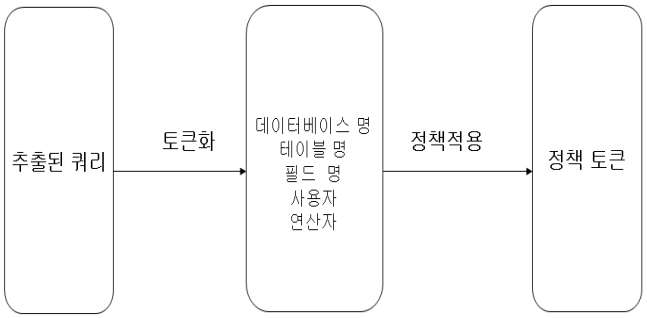
나. 쿼리 추출

스레드를 거쳐 TDS 게이트웨이에 들어오는 전체 패킷 중 쿼리를 읽어 들여 버퍼의 정해진 위치에 따라 필요한 쿼리를 따로 추출한다. 패킷을 읽어 소켓에 대한 처리가 있는지 확인 후 모아진 패킷을 버퍼에 넣고 조합한다. 조합할 패킷이 없다면 바로 패킷을 읽어 들여 헤더 길이와 테일 길이 그리고 버퍼의 sql 길이를 구분하여 해당 쿼리를 추출한다. 이런 각각의 쿼리가 서로 연결되어 있는 문장이여서 패킷 조합이 필요한지 여부를 알기 위해 TDS 프로토콜 분석이 필요하다.

TDS (Tabular Data Stream) protocol (http://www.freetds.org)[9,10]은 데이블 형식 데이터 스트림으로 TCP/IP와 같은 프로토콜 내에 캡슐화 되어 내장되는 형식을 가진 predefine된 메시지라고 할 수 있다. TCP/IP 혹은 명명된 파이프(Named Pipe)와 같은 방법으로 SQL 서버에 연결했을 때, 쿼리문 등의 처리가 이 TDS 형식으로 전달된다. 서버에는 listening port와 비슷한 개념인 TDS 종점(Endpoint)가 생성되고, 서버/클라이언트의 Net-Library를 통해 서로 통신을 한다.

다. 쿼리 파싱

추출한 쿼리는 빈공백 구분자를 기준으로 데이터베이스 명, 테이블 명, 필드 명, 사용자, 연산자로 토큰화 시킨 후 정책 적용을 하기 위한 정책 토큰으로 파싱한다.



(그림 2) 쿼리 파싱

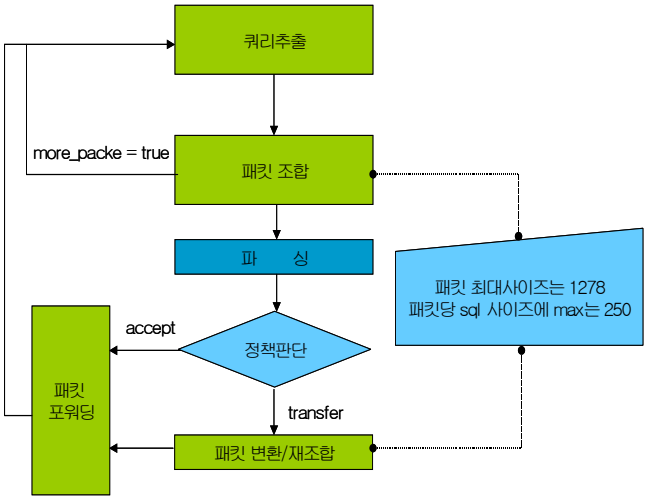
라. 보안 정책 기술

파싱된 쿼리는 개인정보보호를 위한 정책이 기술된 보안 정책에 의해 변환된다. 정책은 정책을 적용 될 대상인 타겟과 정책 행위인 액션으로 구성된다. policy_target은 데이터베이스명, 테이블명, 필드명을 대상으로 사용자와 그룹, 연산자, 권한 모드, 변환규칙의 policy_action을 갖는다. 변환 규칙에 의해 변환된 쿼리는 재조합 되어 패킷 포워딩 된다.

(그림 3) 보안 정책 논리 구조

4. 시스템 구현

그림 4는 쿼리 추출에서 정책 판단으로 쿼리 패킷이 변환되고 재조합하여 포워딩되는 과정이다.



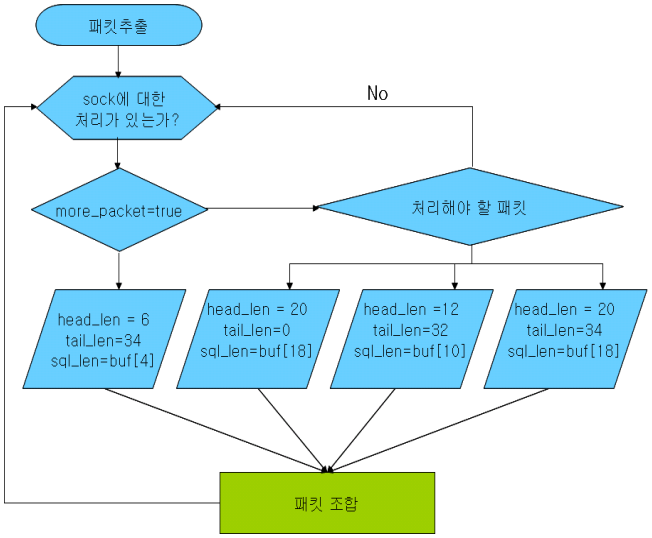
(그림 4) 시스템 구현 순서도

가. 스레드를 이용한 세션 관리

세션을 관리하기 위한 방법으로 스레드를 이용한다. 스레드는 운영 체제에서 프로세스를 사용하여 실행 중인 다양한 응용 프로그램을 구분하고 프로세서 시간을 할당하는 기본 단위로 두 개 이상의 스레드가 해당 프로세스 내에서 코드를 실행할 수 있다. 두 개 이상의 스레드를 사용하는 것은 클라이언트에 대한 응답성을 향상시키고 동시에 작업을 완료시키기 위해 필요한 데이터를 처리할 수 있는 강력한 기술이다. 데이터베이스 관리시스템에 쿼리를 요청할 수 있는 수가 50으로 한정되어 있다면 이 수가 넘지 않게 조절하는 기능을 세션 관리가 담당한다.

나. 쿼리 추출

클라이언트의 소켓 패킷을 받아 읽어 들여 클라이언트 소켓과 최대 버퍼 부분을 따로 추출한다. 이것을 data_len와 sql_len, query 부분으로 나눈다. 패킷을 읽어 소켓에 대한 처리가 있는지 확인한 후 처리할 패킷이 있다면 more_packet이 있는지 확인한다. 있다면 더 패킷을 버퍼에 넣고 조합하고 없다면 바로 패킷을 읽어 들여 헤더 길이와 테일 길이 그리고 버퍼의 sql 길이를 구분하여 해당 쿼리를 추출한다.



(그림 5) 쿼리 추출

다. 쿼리 파싱

추출된 쿼리를 빈 공백을 구분자로 하여 정책을 적용할 수 있는 토큰 상태로 변환한다. 여기에서 얻어내는 쿼리 내용은 데이터베이스 명, 테이블 명, 필드 명, 사용자, 연산자들이다.

라. 접근제어 보안 정책 적용

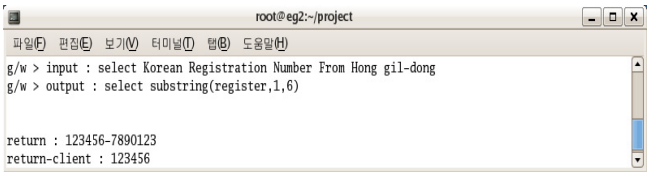
데이터베이스에 접근 제어 보안 정책을 적용하기 위한 설정으로 db_name, table_name, field_name의 policy_target과 user_group, operation, access_mode, translate의 policy_action을 설정한다. 실제 데이터베이스에 보안 정책을 적용해 보면 시용자의 access_mode와 translate 부분에서 개인정보보호를 위해 정책이 설정된 것을 볼 수 있다. 개인 정보를 요구하는 질의문 요청이 오면 주민등록번호가 총 13자리 중 앞의 6자리만 제공되고 뒤의 7자리는 변환되어 나오지 않을 것이다.

마. 개인정보보호 정책 적용

클라이언트는 개인정보를 요구하는 질의문을 데이터베이스관리시스템에 요청했다. 이 클라이언트는 이미 정당한 접근 권한을 가진 사용자로 적법한 인증 절차를 거쳤다. 클라이언트가 홍길동의 주민등록번호를 요구하는 질의에 대한 응답은 주민등록 번호의 앞자리 6자이다.

(그림 6)클라이언트 요청과 응답

이는 다음 그림 7과 같이 개인정보보호 보안 정책이 6자리만 보여주도록 설정되어 있었기 때문에 게이트웨이를 거쳐 데이터베이스관리시스템에 요청 시 주민등록번호 중 6자리만 요구되었기 때문이다.



(그림 7) 게이트웨이 요청과 응답

5. 결론

기존의 데이터베이스 보안 모델들은 정당한 사용자가 인증된 상태로 민감한 개인정보를 요구하면 접근과 유출이 가능하다는 문제가 있다. 본 논문에서 클라이언트가 DBMS에 접속하여 쿼리를 요청할 때 클라이언트와 DBMS 사이에 위치한 TDS 프로토콜을 이용한 보안 게이트웨이를 만들었다. 이 게이트웨이는 클라이언트가 요청한 패킷 중 필요한 쿼리를 추출하고, 파싱한 후 개인정보보호 정책을 적용한다. 설정된 보안정책에 따라 접근을 허용할 것인지, 질의문을 재작성하여 부분차단으로 변환할 것인지, 거부할 것인지를 판단한 후 클라이언트에게 다시 응답함으로써 일반적인 DBMS보다 강력한 보안으로 개인정보 유출을 막을 수 있다. 향후 좀 더 다양한 데이터베이스 환경에서 서로 다른 형태의 쿼리 변환이 시도되는 연구가 이루어져야 할 것이며, 여러 쿼리를 요청하여 개인 정보를 추론하는 질의를 제어 할 수 있는 쿼리 변환기 개발도 필요하다.

참고문헌

[1] 김연수, “개인정보에 기반 한 마케팅 활동과 프라이버시 문제”, 2001.

[2] 이호균, “데이터베이스 보안 기술 동향” ITRIND 주간 기술동향 권호 1266, 2006.10

[3] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu “Hippocratic Databases”, In Proc. International Conference on Very Large Data Bases, 2002.

[4] K. LeFevre, R. Agrawal, V. Ercegovac R. Ramakrishnan, Y. Xu, and D. DeWitt, “Limiting Disclosure in Hippocratic Databases” VLDB 2004, pp 108-119.

[5] 박현아, 임종인, 이동훈, “프라이버시를 보장하는 SQL 쿼리 가능한 데이터베이스” 2006.

[6] 정성우, “DB 보안 중요성 및 수행 방안” 2007.

[7] 소만사, http://www.somansa.com

[8] 알투웨어, http://www.r2ware.com

[9] TDS 프로토콜 http://www.freetds.org/

[10] http://blogs.msdn.com/sql_protocols/default.aspx