http://dx.doi.org/10.5392/JKCA.2014.14.06.001

데이터베이스 암호화 솔루션 구현 및 도입을 위한 기술적 아키텍처

Technical Architecture for Implementation and Adoption of Database Encryption Solution

이병엽*, 임종태**, 유재수**
배재대학교 전자상거래학과*, 충북대학교 정보통신공학과**

Byoung-Yup Lee(bylee@pcu.ac.kr)*, Jongtae Lim(jtlim@chungbuk.ac.kr)**,
Jaesoo Yoo(yjs@chungbuk.ac.kr)**

요약

온라인 비즈니스의 활성화와 인터넷 모바일 기기의 발달을 통해, 불특정 다수의 민감한 사용자 데이터가 쉽게 노출 되고 있고, 개방된 비즈니스 환경에서 최근 민감한 개인정보의 유출 이슈가 자주 언급되면서 그 중요도 측면에선 데이터베이스의 보안기술의 도입은 기업의 최우선 과제가 되고 있다. 2011년 정부에서도 정보통신망법 상의 개인정보 보호 강화조치를 법률로 제정 이를 다양한 산업 군에 적용하고 있다. 기업은 개인정보의 보호를 위해 다양한 방안들을 마련해 이러한 규제를 준수하며 내부에 관리중인 개인정보에 대해 보안을 강화하기 위해 빠르게 보안 솔루션을 도입하고 있다. 이에 수많은 민감한 개인의 데이터들이 저장되어 사용되고 있는 데이터베이스 측면에서 이러한 규제를 준수하는 동시에 효과적으로 데이터 보안 기술을 확보하기 위한 방안을 보안 아키텍처의 기술적인 측면에서의 구분과 보안 솔루션 도입 시 고려가 되어야 하는 아키텍처와 기능적인 부분들에 대해 본 논문은 제시하였다.

■ 중심어 : | 데이터베이스 보안 | 보안 아키텍처 | 보안 체크리스트 | 개인정보 보호 |

Abstract

Through the development of internet mobile devices and online business activation, sensitive data of unspecified user is being easily exposed. In such an open business environment, the outflow of sensitive personal information has often been remarked on recently for which adoption of encryption solution for database became top priority in terms of importance. In 2011, government also legislated for the protection of personal information as an information network law, and is now applying the law to a variety of industries. Firms began to comply with these regulations by establishing various measures for protection of personal information and are now quickly introducing encryption solution to reinforce security of personal information they are managing. In this paper, I present architecture and technological parts that should be considered when introducing security solution.

■ keyword : | Database Security | Security Architecture | Security Checklist | Protect privacy Information |

* 본 연구는 농림수산식품부 (생명, 첨단, 수출, 식품, 수산)기술개발사업, 교육부와 한국연구재단의 지역혁신인력양성사업 (no. 2013H1B8A2032298) 및 미래창조과학부 및 정보통신산업진흥원의 대학IT연구센터육성 지원사업/IT융합고급인력과정 지원사업의 연구결과로 수행되었음(NIPA-2014-H0301-14-1022)

접수일자 : 2014년 04월 29일
수정일자 : 2014년 06월 02일
심사완료일 : 2014년 06월 05일
교신저자 : 유재수, e-mail : yjs@chungbuk.ac.kr

1. 기업 내 데이터베이스 보안 이슈

1.1 정보 보호 이슈

2006년부터 현재까지 개인정보의 유출사고가 빈번하게 발생되고 있다. 그 대표적인 사례로 H 캐피탈사의 고객정보 유출사고, N사의 전산장애 사고로 2011년 금융회사 IT 보안강화 종합 대책이 금융 위원회를 통하여 보도 자료를 각 금융기간에 전달되었다 보도 자료의 내용을 살펴보면, IT 보안에 대한 CEO의 역할 및 책임부여, 정보보호 최고책임자 지정의무화, IT 보안인력 및 IT 보안 투자 확대, 금융회사 IT보안 사고에 대한 제재 수준강화, 금융회사 IT부분 실태평가 확대, 침해행위 처벌 및 보고체계강화, 해킹 피해 최소화를 위한 시스템 개선, 망분리 등 접속경로 통제 강화, 시스템 계정관리 강화 등 다양한 각도에서 정보보호의 이슈가 제기되고, 강화 되고 있는 실정이다.

이에 기업은 IT 보안강화 대책 발표 후 IT 예산, 보안 인프라 구축이 잇따르고 있다.

현재 국내 보안시장은 정보통신망 이용촉진 및 정보보호 등에 관한 법률(현행), 개인정보 보호법(2011년 9월 30일 시행)관련 법률의 Compliance가 개별 기업의 필요를 유발 하므로, 시장 활성화의 강력한 견인 역할을 하고 있다. 더불어 국가 정보원 보안 적합성 검사 즉 국가용 암호제품 지정제도 및 국내외 CC 인증제품 지정제도를 통해 암호화의 기업 이슈에 대응을 하고 있다. 위의 언급된 법적인 제제가 가해진 근본적인 원은 은 최근 들어 빈번하게 발생하고 있는 수많은 데이터 보안 사고들의 핵심은 민감한 개인정보의 유출에 있다는 것을 알 수 있다. 이러한 정보들이 본인의 동의 없이 무단으로 사용되어 불법적인 거래 및 스팸메일에 이용되어 금전적인 혹은 정신적인 피해를 끼치고 있으며 나아가 사회적인 문제까지도 야기하고 있다[1]. 개인정보 유출 문제는 비단 국내만의 문제는 아니며, 전 세계적으로 관심이 집중되는 이슈 중 하나라 할 수 있다. [그림 1]과 같이 미국의 오픈 시큐리티 파운데이션(Open Security Foundation)에 따르면 유출된 개인정보가 08년 86,311,058개에서 09년 218,756,349개로 전년대비 153%로 상당히 증가 하였다. 유출된 개인정보는 상당히 증가한 것과 달리 공개적으로 보고된 개인정보 유출 사건 건수는 08년 717건에서 09년 436건으로 줄어들었다. 이는 개인정보 유출 사건의 피해 규모가 대형화되고 있다는 것을 의미 한다[9][10]. 따라서 이러한 개인정보의 유출이 개인 에게만 피해를 발생하는 것 뿐 만은 아니다. 현재 미션 크리티컬한 비즈니스를 영위하고 있는 수많은 온라인, 오프라인 기업들에게 기업 이미지 혹은 브랜드 이미지라는 것은 이미 단순한 이름 그 이상의 것을 제공하고 있다.

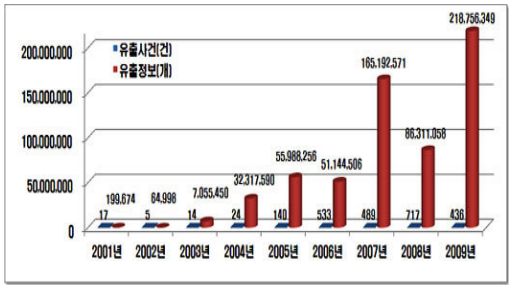

그림 1. 개인정보 유출 및 레코드 현황

최근 한국 인터넷진흥원에서 발표한 2012 국내 지식정보 보안 산업 실태 조사에 따르면 정보 보안 산업은 크게 세 가지로 분류가 될 수 있다. 정보보안 산업의 제품시장은 네트워크 보안, 시스템 보안, 콘텐츠/정보유출방지 보안, 암호/인증 등으로 구분되고, 서비스 시장은 보안 컨설팅, 유지보수, 보안관제, 인증 서비스군 으로 분류가 가능하고, 물리보안 산업은 DVR, 카메라, IP 영상장치, 주변장비, 접근 통제, 알람/모니터 등으로 구분되어 진다. 이러한 구분에 따라 국내 보안 시장의 현황과 향후 전망은 [그림 2]과 같다. 특히 데이터베이스 보안, 즉 접근 통제 시장은 2011년 매출 587억 원에서 735억 원으로 25.2%나 증가 했다. 개인정보보호법 적용을 받는 350만의 모든 기업들에게 개인정보 접근 통제 의무화가 시행됨에 따라 엄청난 성장이 예상되는 이 시장은 웨어벨리, 피엔피시큐어, 소만사, 신시웨이 등의 업종별로 시장을 점유하고 있다. 성장 일색의 보안 시장에서도 가장 높은 성장률을 보인 데이터베이스 암호화 시장의 2012년 매출액은 2011년 221억에서 425억 원으로 92.3%나 증가 했다.

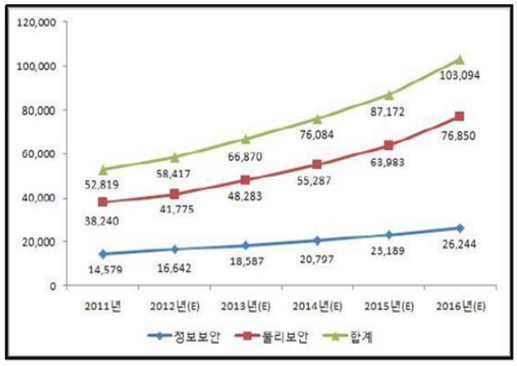

그림 2. 2012 국내 지식정보보안 산업 실태 조사(한국인터넷진흥원)

데이터베이스 암호화 시장은 보안 분야에서 초기 시장에 속하고 미 도입 기업들이 상당히 남아 있기 때문에 향후에도 성장 기회가 매우 높은 시장이다. 문제는 시장 성장이 확실한 분양이기 때문에 관련 업체들이 난립이 이어지고 있으며, 이로 인한 치열한 시장 경쟁은 자칫 기술적인 문제점을 통한 또 다른 기술적인 문제점이 야기 될 수 있다.

또한 최근 논문의 추세에 따르면 시스템 구축, 운영 과정에서의 소프트웨어의 보안 약점을 확인할 수 있다. 입력데이터 검증 및 표현, 보안기능, 시간 및 상태, 에러처리, 코드오류, 캡슐화, API오용 등으로 보안약점이 발생 한다고 알려져 있다[11].

이에 본 논문은 데이터베이스 암호화에 적용되는 다양한 기술들을 고찰 하고, 최적의 데이터베이스 암호화를 위한 기술적인 아키텍처를 제안하고자 한다.

2. 보안 구현 유형과 규제 준수

2.1 정보통신망법 상 개인정보보호 강화조치

인터넷, 모바일 등의 다양한 오픈 플랫폼의 급격한 확산으로 인해 수많은 어플리케이션들이 불특정 다수의 사용자에게 개방되어 있고 사용자 인증을 위해 개인정보를 필요로 함에 따라 대부분의 기업들이 개인 정보를 자체 관리함으로서 개인정보 보호를 강화하기 위한 조치가 국내에서 방송통신위원회 주도로 대통령령으로 시행하게 되었으며 2009년 적용 대상 업체의 확대로 인해 국내 대부분의 기업에서는 이를 반드시 준수해야 하며 위반 시 벌금 및 처벌을 받게 된다[6].

정보통신망법 상 강화된 개인정보보호 기준에 맞는 기술적 관리적인 보호조치를 구분해 각 항목별로 살펴보면 [표 1]과 같다[7].

표 1. 개인정보의 기술적 관리적 보호조치 기준

구분	항	내용
제3조		개인정보관리계획 수립 및 이행(관리책임자 및 취급자의 지정과 교육
제4조 접근 통제	1	최소 인원에만 접근 권한 부여
	2,3	접근권한 변경 관리 및 부여/변경/말소 내역 기록(최소 5년 보관)
	4	외부에서 정보통신망으로 접속 시 안전한 인증 수단의 적용
	5	접속권한 제한을 통한 불법적인 접근 및 침해사고 방지
	6,7	패스워드 작성 규칙 수립 이행
	8	처리 시스템 취급자PC설정(인터넷,P2P, 공유설정)
제5조 접속기록 위, 변조방지	1,2	접속기록(처리일시, 내역 등)저장 및 월 1회 확인/감독 및 최소 보존기간 규정
	3	접속기록 별도 저장 및 백업 고관(위변조 방지)
제6조 암호화	1	본인 인증정보(패스워드, 생체정보) 양방향 암호화
	2,3	주민번호, 신용카드번호, 및 계좌번호의 암호화 저장 및 인증정보 송/수신 시 암호화
	4	개인정보 PC 저장시 암호화
제7조 악성 프로그램	1	백신 S/W설치(악성프로그램 점검/치료)
	2	백신 S/W 월 1회 갱신/점검(최신 업데이트)
제8조 출력 및 복사시 보호조치	1	출력 시(인쇄, 화면, 파일생성 등) 용도 특성 및 항목 최소화
	2	인쇄/이동매체 복사 시 기록/사전승인(재복사포함)
	3	명칭 및 일련번호 표시
	4	2항 위법여부 확인 및 유출시 법적 책임 주지

3. 데이터베이스 보안 기술 유형

전통적인 외부 보안 솔루션들은 소위 경계(perimeter) 보안의 관점에 입각해 있다. 즉, 일정한 형태의 방화벽으로 특정 경계/네트워크를 둘러쌈으로써 외부 자에 의한 침입을 막고자 하는 것이다. 하지만 수년에 걸쳐서 정보 보호의 초점은 외부 보안에서 내부 보안으로 옮겨져 왔다. 구체적으로는 다음과 같은 요인들이 작용한 결과이다. 첫째, 보안에 대한 각종 통계 자료들이 축적되며 보안 사고의 최소 80% 이상이 세간의 인식과는 달리 외부자의 침입이 아닌 내부자의 소행이

었다는 사실이 입증되었다. 피해액의 경우에도 외부 침입이 아닌 내부 보안 사고가 훨씬 커다란 규모를 보여주고 있다. 둘째, 각종 compliance 또한 내부 보안에 주안점을 두고 있다. 대표적인 compliance인 SOX는 2001년 적발된 미국 Enron 사의 회계 부정행위로부터 제정되었으며, 동 사건 또한 외부 침입이 아닌 내부자의 행위로부터 비롯된 것이다. 내부 보안은 외부 보안과는 그 접근 방식이 다르다. 내부 보안은 이미 해당 경계/네트워크 내에 존재하는 내부자가 권한을 오/남용하는 것을 방지하는 것을 의미하기 때문이다. 이에 따라 보안 솔루션 또한 기존의 외부 보안 솔루션과는 다른 방식으로 구현되어야 한다. 내부 보안의 핵심은 데이터베이스 보안이다. 데이터베이스는 전사적인 정보 infrastructure의 핵심으로서 기업이나 조직의 주요 정보들을 모두 관리하기 때문이다. 아래 [그림 3]은 이상의 논의를 종합한 IT 보안의 지형도이다.

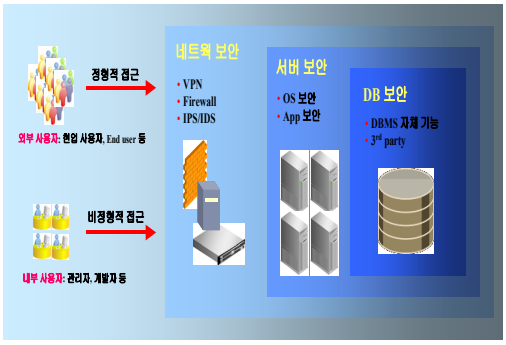

그림 3. IT 보안의 지형도

[그림 3]과 같이 데이터베이스 보안은 네트워크 보안이나 서버 보안과 같은 외부 보안 솔루션의 안쪽에서 정보 보호를 위한 최종적인 방어선으로서의 역할을 수행해야 한다. 특히 내부 보안이라는 관점으로 볼 때, Web 어플리케이션 등을 통하여 정형적인 접근만을 하는 외부 사용자보다는 SQL*Plus 등의 ad-hoc tool을 이용하여 비정형적인 접근을 수행하는 내부 사용자가 보안의 주된 관건이 된다. 본 논문은 데이터베이스가 제공하는 자체적인 보안 솔루션과 국내의 각종 데이터베이스 보안 솔루션들을 비교하고자 한다. 하지만 구체적인 내용으로 들어가기 전에 한 가지 근본적인 질문을 던져볼 필요가 있다. "과연 내부 보안을 외부 솔루션의 형태로 구현하는 것이 가능할까?" 국내외의 보안 솔루션의 유형은 데이터베이스에 내장되어 있는 솔루션과, 어플리케이션의 형태로의 보안 솔루션으로 크게 나뉘어 볼 수 있다. 이런 관점에서 본 논문은 먼저 아키텍처적인 관점에서 양자(내부, 외부)를 비교하고자 한다.

3.1 데이터베이스 보안 솔루션 기술 구분 현황

다양한 데이터베이스, 다양한 외부 보안 솔루션들이 존재 하지만, 데이터베이스보안이 다루는 분야는 궁극적으로 크게 다음과 같이 나누어 볼 수 있다.

표 2. 데이터베이스 보안 기술 구분

데이터베이스 보안 기능 구분	기능 설명
계정관리 및 인증	데이터베이스 계정관리 및 암호 기반 인증
권한 및 Role	전통적인 DAC & RBAC
감사	감사기능. 효율적인 감사를 위한 Fine Grained Auditing기능 포함
Virtual Private Database	테이블의 행단위의 접근 제어
Encryption API	데이터 암호화 API
Enterprise User Security	데이터베이스 계정이 아닌 실 사용자의 관리 기능
Strong Authentication	표준 인증 서비스와의 연동
Encryption	Network Encryption: 클라이언트와 데이터베이스 서버간의 모든 네트워크 통신 패킷을 암호화 Transparent Data Encryption: 사용자 SQL에 투명한 데이터 자동 암호화/복호화
보안등급관리	테이블 또는 테이블스페이스의 행 단위 접근 제어 및 보안 등급 관리
Database 실행권한	단일 데이터베이스 안에서 업무별로 독립적인 보호 영역을 구축, 또한, 데이터베이스의 어떤 명령에 대해서도 조건에 따라 실행 권한을 제어할 수 있게 해주는 내부 통제 기능
감사권한	다양한 Database 또는 감사 Tool에 분산되어 있는 감사 정보 통합 관리

첫째 인증 및 사용자 관리기능이다. 사용자의 신원은 모든 데이터베이스 보안 기능의 초석이라고 할 수 있다. 두 번째 접근제어는 사용자의 데이터베이스 접근은 최소 권한의 원칙에 따라 정책적으로 통제/관리 되어야 한다. 세 번째, 감사 및 모니터링은 사전 통제와는 별도

로 주요 데이터베이스사용 기록은 책임 소재의 확인을 위해 남겨 두어야 한다. 한편 지속적인 모니터링을 통해 보안 사고를 적시에 예방 할 수 있어야 한다, 넷째 암호화 부분이다. 데이터베이스데이터는 데이터베이스의 제어로부터 벗어나는 상태에 놓일 수 있다. 네트워크상에서 전송되는 데이터 또는 백업매체에 저장된 데이터 등이 그러한 경우이며, 그 어떤 경우에도 중요 데이터의 유출은 방지되어야 한다[4]. 암호화는 이에 대한 방안이 반드시 되어야 한다. 위의 열거한 네 가지데이터베이스의 보안 기술 구현을 정리하면 [표 2]와 같다. [표 2] 에 나타난 바와 같이 데이터베이스를 통한 기본 보안 기능은 최소한의 보안 정책 및 관리를 위한 기술적인 기능들이다.

3.2 보안 솔루션 아키택처 비교

보안 솔루션의 구분은 크게 두 가지의 기준에 의해 아키텍처가 구분되어 질 수 있다. 첫째, 기능에 따른 분류 이다. 기능에 따른 분류는 크게 데이터베이스의 접근 제어 방식과 암호화 방식으로 분류될 수 있다. 물론 인증이나 감사/모니터링 기능이 무시되는 것은 아니며, 접근 제어와 암호화 모두를 제공하여야 한다. 둘째, [표 2]와 같은 아키텍처에 따른 분류 이다.

표 3. 보안 아키텍처에 구분

분류		설명
별도보안 서버 방식	Sniffing 방식	네트워크 포트 mirroring또는 TAP 장비를 이용하여 네트워크 packet sniffing을 수행하고, 해당 packet을 보안 서버가 조사하여 감사/모니터링을 수행하는 방식. 하지만 이 자체로는 접근 제어는 불가능
	Gateway 방식	보안 서버가 데이터베이스서버에 대한 보안 gateway 역할을 담당하여 감사/모니터링은 물론 접근 제어까지 구현하는 방식
보안 Agent 방식		데이터베이스서버에 설치된 S/W agent를 통해 보안 기능이 구현되는 방식

구체적인 아키텍처상의 기능 비교에 들어가지 전에 먼저 [표 3]에서 제시된 바와 같이 아키택처를 파악하고, 아키텍처 상의 주요 기능들을 미리 짚어 볼 필요가 잇다. 아키텍처 특성은 보안 기능에 대한 평가를 위한 기본 전제이며, 동시에 아키텍처의 비교만으로도 기능 비교의 가장 핵심적인 부분들은 판단이 되었다고 사료된다.

아키텍처 체크포인트들은 크게 다섯까지로 나누어 볼 수 있다. 첫째, 접근 통제의 완전성이다. Database는 모든 접속 방법을 빠트리지 않고 다 커버할 수 있는가? Database의 접속은 크게 TCP/IP접속과 BEQ 접속으로 나누어 볼 수 있는데, 이 두 가지 모두를 통제할 수 있는가? 이다. 둘째 데이터베이스 접근 제어의 완전성이다. 데이터베이스에 들어오는 모든 요청에 대해 빠짐없는 권한 체크가 가능한가? 셋째, 통합성의 측면에서 다수의 데이터베이스에 대한 통합적 보안 관리가 가능한가? 넷째 성능적인 측면에서 보안 기능으로 인한 데이터베이스의 성능의 penalty가 있다면 어느 정도인가? 다섯째, 가용성의 측면이다. 제공되는 보안 기능은 얼마나 안정적인가? 보안 솔루션의 장애는 데이터베이스 서비스의 가용성에 대해 어떤 영향을 미치는가?

위의 다섯까지의 체크 포인트들은 외부 솔루션 형태의 솔루션을 평가하기 위한 항목이라고 할 수 있고, 내부에 기능이 탑재되어 있는 보안 솔루션에서는 그만큼의 의미를 갖는 것은 아니다. 따라서 접속 통제 및 접근 제어는 완전하며, 최상의 성능 및 가용성을 구현 할 수 있는 데이터베이스의 암호화 솔루션을 통해, 데이터베이스 서비스의 무결성 및 고 가용성 그리고 최상의 데이터 암호화를 구현 할 수 있는 아키텍처가 구현 되어야 한다.

3.3 보안 솔루션 아키텍처 구현 방법

[그림 4]와 같은 Gateway방식은 네트워크상에서 데이터베이스서버 앞단에 설치가 되어 데이터베이스서버에 대한 보안 Gateway역할을 수행하게 된다. 이러한 구성에서 데이터베이스서버에는 아무런 설정 변경이 필요 없게 되는데 이는 확실히 agent 방식에 비해 gateway 방식이 갖는 장점이기도 하다. 이 경우 클라이언트는 어떻게 데이터베이스가 아닌 Gateway를 바라보게 되는데, 이를 위해 Gateway는 사용자의 PC에 설치되는 NAT(Network Address Translation)클라이언

트 모듈을 제공한다. 이를 통해 클라이언트의 SQL*Net에서는 데이터베이스서버가 아닌 Gateway로 routing된다. 반면 gateway방식은 오직 TCP/IP 접속만을 관리 할 수 있어 BEQ 접속의 경우 전혀 대책이 없고, TCP/IP 접속의 경우에도 우회 접속을 원천 차단 할 수 있는 시스템적인 방법은 없다. 통상적인 실 환경에서는 데이터베이스 서버가 다양한 다른 서버들과 다수의 네트워크를 통해 연결되는 것이 보통이기 때문에 [그림 4]와 같은 tight한 구성은 현실적으로 쉬운 일은 아니다. 이 경우 NAT 모듈을 설치하지 않는 클라이언트가 Gateway를 우회하여 바로 데이터베이스에 접속할 수 있는 경우가 발생할 수 있다. 이 의미는 접속 통제가 완전하지 못하여 보안기능에 결함이 있음을 의미한다.

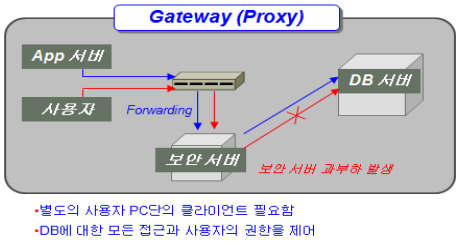

그림 4. Gateway 방식

[그림 5]는 sniffing 방식의 구현이다. Sniffing 방식은 Port를 Mirroring 할 수 있는 장치 또는 Tap 장비가 필요한 방신이다. 별도의 보안 서버를 통하여 문제가 발생하는 요소들에 대해서 1차적으로 필터링을 하는 기능을 가지고 있지만, 실질적인 SQL 레벨에서의 보안 부분들의 취약점을 가지고 있다.

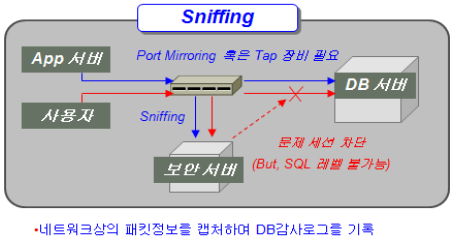

그림 5. Sniffing 방식

[그림 6]은 Hybrid 방식의 구현이다. Hybrid 방식의 전제는 역시 외부 사용자와 내부 사용자의 구분이다. Web 어플리케이션 접근과 같은 정형적 접근은 바로 데이터베이스 서버로 routing된다. 다만 중간에 있는 TAP 장비 등을 통해 sniffing되어 서버에 logging이 될 뿐인데, 이는 Hybrid 방식을 이전의 sniffing 부분들의 보안 기능이라고 볼 수 있다. 한편 Middleware외의 클라이언트가 데이터베이스 에 접근하는 경우에는 사용되는 클라이언트 어플리케이션의 종류에 따라 다르게 취급된다. 그것이 만일 현업 프로그램 같은 정형 접근이라면 바로 데이터베이스서버로 routing되고, 중간에 sniffing의 대상이 된다. 반면에 SQL*Plus와 같은 비정형 접근이라면 데이터베이스서버가 아닌 Gateway서버로 routing되며, 이때 Gateway서버는 보안 Gateway로서의 역할을 수행하게 되는 아키텍처이다. 한편 기업이나 조직이 사용 할 수 있는 비정형 데이터베이스접근 tool을 단 한 가지로 제한할 수 있다면 보안 기능이 강화될 가능성이 있다.

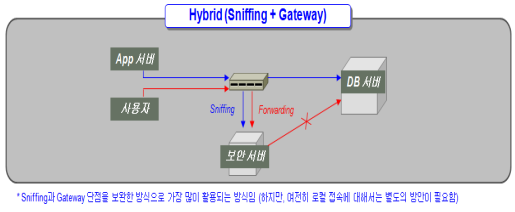

그림 6. Hybrid 방식

3.4 데이터베이스 암호화 구현 아키텍처

대부분의 기업들은 개인정보의 보호를 위해 다양한 보안 솔루션들을 채택해 적용해 오고 있다. 이러한 솔루션들은 해당 기업의 보안 우선순위에 의해 선택적으로 사용되고 있으며 이를 데이터 암호화, 접근제어, 감사와 같은 3가지 유형으로 나누어 보고 이에 속하는 솔루션을 분류해 보면 다음 [그림 7]과 같다. Plug in 방식을 살펴보면 암호화 솔루션 초기모델로 채택된 방식이고 데이터베이스 내에 트리거 테이블을 생성하여 데이터베이스부하의 가중을 유발 하고, 응용 프로그램 변경은 상대적으로 허용할 수 있는 아키텍처 이다. 두 번째

는 API 암호화 방식이다. API 방식은 Plug-in 방식의 대안으로 발전되어온 모델이며, Plug-in 방식이 가지고 있었던 트리거 테이블의 생성의 데이터베이스부하의 부분을 개선 한 아키텍처이다. 하지만 Batch성 업무에는 여전히 성능적인 문제를 유발 하고 있으며, 초기 암호화에 장시간의 문제가 걸린다는 단점이 존재 하고 있다. 또한 API방식을 채택함으로서 Web 어플리케이션의 수정을 요하게 되므로 인한 Source의 수정부분이 불가피한 방식임을 알 수 있다. 다양한 Package성 솔루션들은 source의 수정을 허락하지 않고 있기 때문에 범용적인 솔루션 아키텍처라고 하기에는 다소 어려움이 존재 하는 모델이다.

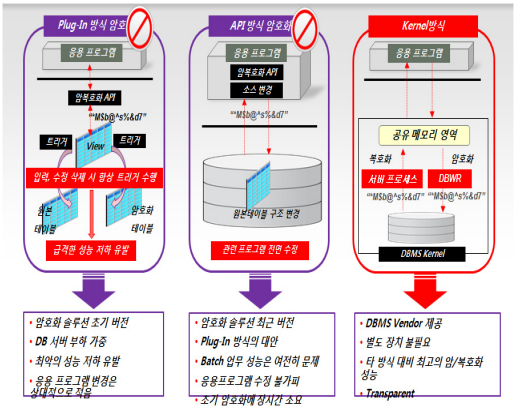

그림 7. 데이터베이스 암호화 구현 방식

세 번째는 데이터베이스의 커널 단에서 암호화, 복호화를 하는 모델이다. 데이터베이스 커널에서 공유 메모리 영역을 통해 처리되는 방식으로 현존하는 가장 합리적인 데이터베이스 암호화 방식이라고 할 수 있다[5][8]. 다만 단점은 공유 메모리 영역에서의 처리 과정에서 메모리 영역의 휘발성 데이터 처리 과정에서 데이터의 유실이 발생 할 수 있는 단점을 가지고 있다.

그렇다면 이러한 솔루션들은 어떻게 데이터베이스 내 데이터에 대한 암호화 및 접근제어를 구현하고 있는지 기술 구조를 살펴보면 [그림 8] 과 같다. 국내 암호화 솔루션들은 원천 기술을 보유하고 있는 데이터베이스 벤더에서 제공하는 공개된 기술을 사용하여 데이터 암호화를 수행하고 있다.

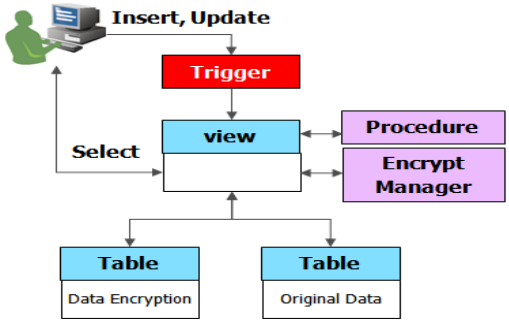

그림 8. 국내 암호화 솔루션 구현방법

일반적으로 공개된 기술을 이용하면 원천적인 데이터베이스 내부 커널을 이용하지 않고도 암호화 기능을 구현할 수 있다는 장점이 있다. 하지만 그에 따른 성능부하 및 데이터 변경이라는 위험이 있고 이를 감수하고 도입한 사례가 있으며 대표적으로 국내 솔루션 도입에 앞장섰던 다수의 공공기관에 적용되어 있다.

접근제어 솔루션들 역시 [그림 9] 와 같이 크게 다를 바가 없다. 이 역시 데이터베이스 내부에서 제어할 수 없기에 중간에 게이트웨이나 네트워크 패킷의 스니핑, 에이전트 방식을 채택함으로써 우회 접근에 대한 위험 및 복잡한 업무에 적용할 수 없는 단점을 보유하게 된다.

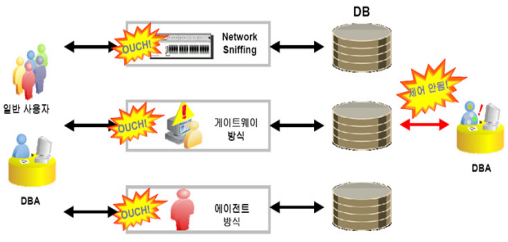

그림 9. 접근제어 솔루션의 리스크

3.5 데이터베이스 벤더의 데이터 암호화

데이터 암호화란 데이터 보안확보 유형 중 가장 일반화되어 있는 방식으로 기업 내 저장되어 관리되고 있는 데이터를 인증 받은 암호화 알고리즘을 통해 암호화하여 허가받지 않은 사용자에 의한 데이터 미디어(Disk) 및 백업본이 유출되더라고 이를 이용해 민감한 개인정보를 도용하지 못하도록 할 뿐만 아니라 네트워크를 통해 전송되는 데이터 패킷까지도 암호화하여 스니핑과

같은 해킹 기술을 이용하더라도 데이터를 안전하게 관리할 수 있도록 한다 하지만, 이러한 데이터 암호화를 구현 시 고려해야 되는 사항은 어떠한 것들이 있는지 이에 대한 고찰이 필요하다.

첫째, 어플리케이션의 수정이 필요한지의 여부다. 대부분의 경우 데이터를 암호화하는 시스템들은 지금까지 잘 사용해 오던 시스템일 경우가 많으며, 이러한 경우에 잘 사용해오던 수많은 어플리케이션 코드의 수정이 동반되어야 한다면 보안확보에 소요되는 경비가 지나치게 많이 소모될 뿐만 아니라 너무 번거롭기까지 하게 된다. 따라서 보안의 도입 시 어플리케이션의 도입 여부는 반드시 점검해 보아야 할 사항이다.

둘째, 성능을 유지할 수 있느냐의 여부다. 데이터를 암호화한다는 것은 특정 알고리즘에 의해 해당 데이터를 변환하고 이를 필요시 다시 원래의 데이터로 복호화해야 하기 때문에 이에 소모되는 리소스 (CPU 파워와 같은)가 더 많이 소비되기 때문에 시스템 성능의 하락을 수반하는 경우가 많다. 이를 얼마나 최소화할 수 있는 것이 데이터 암호화의 또 다른 관건이며 O사 과 같은 데이터베이스 벤더들은 이러한 암/복호화를 커널 내부에서 수행케 함으로서 성능 하락을 최소화 할 수 있는 메커니즘을 제공한다[3].

3.6 데이터베이스 보안구현

본 논문은 현존 하는 다양한 방식의 암호화 아키텍처 및 조금 더 세부적인 데이터베이스 암호화의 방식에 대한 고찰을 하였고, 이에 따른 데이터베이스 보안을 구현하기 위한 체크리스트 및 표준 아키텍처를 제언하고자 한다. 데이터베이스의 보안을 구현하기 위해서는 [표 4] 에서와 같이 다양한 부분들의 검증이 필요하다 또한 데이터베이스의 성능적인 측면에서도 반드시 테스트가 진행 되어야 한다. 이는 디스크의 성능 개선의 측면에서 살펴보면 암호화, 복호화에 따른 디스크 I/O가 발생하기 때문에 디스크 I/O성능이 개선되면, 선형적인 응답 속도를 향상시킬 수 있다. CPU의 처리속도 측면에서도 블록 싸이퍼 암호화 알고리즘의 특성상 암호화 성능을 개선할 수 있는 방법으로 CPU의 클록 스피드 개선이 필수이고, OLTP와 같이 병렬 처리가 불가능한 작업의 응답속도 개선에 매우 효과적인 개선을 할 수 있다. 또한 병렬처리를 통한 개선방법은 디스크 IO를 많이 사용하는 배치 작업의 경우, 병렬처리를 통해 CPU 개수가 증가에 따라 선형적인 성능향상을 구현해야 하고, 배치 작업의 경우에도 암호화에 따른 비용 보정은 디스크 I/O속도 및 CPU클록 스피드 개선이 동일하게 적용 되어야 한다.

표 4. 데이터베이스 보안구현 체크리스트

검증 영역	검증구분	검증항목	검증방법
1) 보안	보안관리	● 암호화알고리즘 안정성	기능테스트
	암호화 키 관리	● Master Key관리, 키 관리의 안전성	기능테스트
	솔루션운영 방안	● DB암호화 솔루션 프로세스 control, DB암호화와 키 재생성은	시연
2) Application	DB Object	● 암호화 테이블이 파티션이고 파티션 키에 암호화 항목이 포함되어 있는 경우 사용 가능 여부	SQL수행
	SQL Pattern	● 쿼리 툴 사용 시 사용자 권한에 의함 암/복호화 여부	SQL수행
		● SQL상의 연산자 (=,<,>,Between, Like, IN)사용	SQL수행
		● 각종 SQL 함수(Subset, Length, Decode, Case, Trim, Replace등) 사용 여부와 WHERE절 사용 시 인덱스 사용 여부	SQL수행
		● 암호화 항목이 포함되어 있는 결합 인덱스의 색인 검색 가능 여부	SQL수행
		● Outer Join/Sub Query, Order By사용가능 여부	SQL수행
3) 아키텍쳐	아키텍처	● Ukey 아키텍처(TmaxSoft ProFrame-C)적용 가능 및 영향도	프로그램
	Application운영	● 솔루션 자체 Error Message처리 방식적용 여부(자체 디버깅 기능 및 에러 메시지 처리가능 여부	프로그램
		● 데이터 관련 운영 Tool(ETL, SQL Loader, Connect Direct)지원여부	
4) 인프라	적용 환경관리	● 암호화 적용된 테이블 컬럼 및 환경의 변경 가능 여부(Alter, drop, reorg 등)	SQL수행
	Data Migration	● 초기 Data Migration 소요시간 및 테이블/인덱스 증가 Size	초기 이행
	성능	● 암호화 적용 후 OLTP 서비스 응답시간 변화율	성능테스트
		● 암호화 적용 후 DB 응답시간 변화율	성능테스트
		● 암호화 적용 후 Batch 업무 처리시간 변화율	성능테스트
		● 암호화 적용 후 시스템 사용량 증가율	성능테스트
		● On-Line SQL 발생 I/O 변화율	성능테스트

[표 5]는 앞서 기술적인 비교에 대한 각 데이터베이스의 암호화 방식의 기능별, 안정성, 확장성 및 특징에 대한 부분들을 제시하였다[12].

표 5. 데이터베이스 보안 구현 아키텍처 비교

구분	Sniffing 방식	Server Agent 방식		Gateway 방식 (In-line, Proxy 구성 가능)	TTP Gateway 방식 (Trusted Transparent Proxy로 In-Line, Proxy 구성가능)
		BEQ Agent	Node-Safer		
보안기능	보안통제가 사실상 불가능	강력한 보안 기능 제공	우회접속차단만 가능	강력한 보안 기능 제공	강력한 보안 기능 제공
안정성	Agent 가 설치되지 않는 방식, 데이터베이스 서버에 영향 없이 안정적 운영 가능	•Agent 설치로 인한 데이터베이스 서버 성능에 영향을 줄 수 있음 •Agent 장애로 인한 대책 미비 •서비스 재시작 필요	•서비스 재시작 없음 •서버 재시작 없음 •Software TAP방식 •장애 없음	•Agent 가 설치되지 않는 방식, 데이터베이스서버에 영향 없이 안정적 운영 가능 •Gateway 구성에 따른 장애 대응 방안 필요(이중화 or Bypass) •SSH 통제 및 모니터링 가능 •암호화 대체 기능 제공 (Data Masking)	•Agent 가 설치되지 않는 방식, 데이터베이스서버에 영향 없이 안정적 운영 가능 •SSH 통제 및 모니터링 가능 •암호화 대체 기능 제공 (Data Masking)
확장성	확장시 각각의 세그먼트 마다 H/W 연결 필요	•서버마다 Agent 설치 필요		•별도의 H/W 나 Agent 추가 없이 확장 가능	•별도의 H/W 나 Agent 추가 없이 확장 가능
특징	모니터링만 할 경우 권장	소규모 적용시 가격 유리	Gateway방식에서 우회경로차단 및 서버 접근제어 용으로 활용	보안성 및 확장성이 뛰어나며, 이중화 구성 시 가장 좋은 구성 제안이 가능	•안정성 : Sniffing 방식과 동일 •보안성 : Gateway방식과 동일 •성능 : Gateway 방식과 동일 •최적의 구성 방식

4. 데이터베이스 보안의 향후 과제 및 결론

1, 2장에서 제시된 데이터베이스의 보안 기능 및 아키텍처에서 살펴본 것과 같이 인증 및 사용자 관리는 데이터베이스 보안의 가장 기본이라고 할 수 있다. 사용자의 신원은 접근 제어 및 감사, 모니터링의 가장 기초적인 근거가 되기 때문이다. 모든 데이터베이스는 데이터베이스레벨의 인증 및 계정 관리 기능을 기본적으로 제공한다. 만일 해당 기업이나 조직이 보유한 데이터베이스가 단 하나라면 이것만으로도 충분할 것이지만 현실은 결코 그렇지 않으며, 다수의 실 사용자와 다수의 데이터베이스 계정이 혼재하는 형태가 가장 일반적이다. 이 경우 데이터베이스레벨의 사용자 관리만으로는 관리 및 보안상의 문제점이 발생한다. 이에 대한 해법은 데이터베이스레벨을 넘어서 기업/조직 레벨을 다룰 수 있도록 하는 인증 및 사용자 관리이다.

또한 아키텍처 비교 파트에서 기술한 접속 통제의 완전성 문제는 접근 제어에 그대로 직결된다. 즉, Gateway방식이나 Sniffing 방식의 경우 통제하지 못하는 접속이 있을 수 있다. 접속을 통제하지 못하는데 접근 제어가 가능할 리는 없다. 결론적으로 데이터베이스의 민감한 데이터의 보호를 위한 암호화 솔루션 아키텍처는 다음과 같은 기술들을 면밀히 검토 하고 도입을 고려하여야 한다. 완벽한 보안성, 데이터베이스의 부하, 가용성 이 세 가지의 부분들이 반드시 고려되어야 하고, 더불어 암호화, 접근제어, 감사의 세 가지 유형으로 구분한 데이터 보안 확보의 유형은 향후, 모바일 환경의 대두와 클라우드 컴퓨팅, 빅데이터의 확대로 인한 IT 환경의 변화에 따라 지금까지의 보안 솔루션 벤더와 기업 모두에게 새로운 도전이 될 것이며 이러한 기술적 변화를 적극적으로 수용하며 지속적인 투자 및 기능의 연구개발을 통해 발전시키는 동시에 보다 다양한 기술과의 융합을 통해 그 영역을 확대 시키는 것이야 말로 현 보안 솔루션의 과제라 할 수 있으며, 완벽한 보안 솔루션의 개발을 통해 점차적으로 증가하는 민감한 정보들의 데이터 보안 시장의 활용과 법제화된 개인정보의 보호법에 따라 데이터베이스 보안 분야의 비즈니스적인 관점에서 시장 확대 및 새로운 신규 시장이 형성 될 것으로 사료 된다. 향후 보다 진보된 데이터의 보안관리가 이루어지길 기대한다.

참 고 문 헌

[1] http://datalossdatabase.org/reportsPublicly Reported Data Breaches by DataLoss데이터베이스, 2005~2008.

[2] Report on IT Security priorities for 2009 by Forrester Research.

[3] http://download.oracle.com/docs/cd/E11882_01/network.112/e10746/asointro.htm#i1008719

Advanced Security Administrator's Guide.
[4] http://download.oracle.com/docs/cd/E11882_01/server.112/e10576/dvintro.htm#CEGBCJCB, Database Vault Administrator's Guide.
[5] http://download.oracle.com/docs/cd/E14472_01/doc.102/e14459/avadm_intro.htm#sthref30, Audit Vault Administrator's Guide.
[6] http://law.go.kr/LSW/lsSc.do?menuId=0&p1=&query=%EC%A0%95%EB%B3%B4%ED%86%B5%EC%8B%A0%EB%A7%9D+%EC%9D%B4%EC%9A%A9%EC%B4%89%EC%A7%84+%EB%B0%8F+%EC%A0%95%EB%B3%B4%EB%B3%B4%ED%98%B8+%EB%93%B1%EC%97%90+%EA%B4%80%ED%95%9C+%EB%B2%95%EB%A5%A0&x=3&y=9
[7] http://www.kisa.or.kr/jsp/public/laws/laws3.jsp
[8] http://olv.moazine.com/rviewer/index.asp
[9] http://blog.daum.net/kcc1335/1890
[10] http://dataloss데이터베이스.org
[11] 김정숙, "소프트웨어 보안을 위한 시큐어 코딩", 한국콘텐츠학회논문지, 제14권, 제1호, pp.386-399, 2014.
[12] 이병엽, 박준호, 유재수, "데이터베이스 규제 준수, 암호화, 접근제어 유형 분류에 따른 체크리스트 구현", 한국콘텐츠학회논문지, 제11권, 제2호, pp.61-68, 2011.

저 자 소 개

이 병 엽(Byoung-Yup Lee) 종신회원

- 1991년 2월 : 한국과학기술원 전산학과(공학사)
- 1993년 2월 : 한국과학기술원 전산학과(공학석사)
- 1997년 2월 : 한국과학기술원 경영정보공학(공학박사)
- 1993년 1월 ~ 2003년 2월 : 대우정보시스템 차장
- 2003년 3월 ~ 현재 : 배재대학교 전자상거래학과 교수

<관심분야> : XML, 지능정보시스템, 데이터베이스시스템, 전자상거래학

임 종 태(Jongtae Lim) 정회원

- 2009년 2월 : 충북대학교 정보통신공학과(공학사)
- 2011년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 박사과정

<관심분야> : 데이터베이스 시스템, 시공간 데이터베이스, 위치기반 서비스, 이동 P2P 네트워크, 소셜 네트워크 서비스, 빅 데이터 등

유 재 수(Jaesoo Yoo) 종신회원

- 1989년 2월 : 전북대학교 컴퓨터공학과(공학사)
- 1991년 2월 : 한국과학기술원 전산학과(공학석사)
- 1995년 2월 : 한국과학기술원 전산학과(공학박사)
- 1995년 3월 ~ 1996년 8월 : 목포대학교 전산통계학과 전임강사
- 1996년 8월 ~ 현재 : 충북대학교 전자정보대학 교수

<관심분야> : 데이터베이스 시스템, XML, 멀티미디어 데이터베이스, 분산 객체 컴퓨팅, 빅 데이터 등