쿼리 은폐 기법을 활용한 위링크의 데이터베이스 보안 구현

임복출*, 김인구**, 김순곤***
*지니네트웍스(주), **(주)위컴즈, ***중부대학교 컴퓨터·게임학과
e-mail:Wiseman.Lim@gmail.com, king@wecoms.com, sgkim@jbm.ac.kr

Implementation of Database Security using the Query Hiding Technique on the WeLink

Bock-Chool Lim*, In-Koo Kim**, Soon-Gohn Kim***
*Geninetworks Corp.
**Wecoms Corp.
***Dept. of Computer and Game Science, Joongbu University

요 약

기존에 PC나 서버에 설치하여 제공되던 방식에서 클라우드 컴퓨팅 환경의 서비스 방식이 일반화되어 가고 있다. 클라우드 환경에서 발생하는 데이터는 형태와 양이 기존 방식처럼 관리하는 것은 불가능에 가깝다. 이러한 시대적 흐름에 발맞춰 데이터의 처리 및 저장과 관련된 기술의 중요성도 더욱 커져가고 있다. 본 논문에서는 중소기업을 위한 프레임워크인 위링크(WeLink)내에서 데이터 저장, 조회, 수정, 삭제 등을 위한 보안방안을 모색하였다. 또한 프레임워크 기반으로 데이터 보안을 위한 쿼리나 데이터베이스에 상관없이 서비스를 제공할 수 방법을 제안하였다.

1. 서론

오늘날 기업들은 다양한 서비스 제공을 위해 고객의 많은 정보를 보유하고 있는 가운데 정보 유출로 인한 피해가 발생하고 있어, 보안에 대한 요구가 커지고 '데이터베이스 보안'의 필요성 또한 날로 높아지고 있다[1]. 데이터베이스 보안은 데이터베이스와 해당 오브젝트에 대하여 사용자의 행위의 허용유무가 필요하다[2].

본 논문에서는 소형 웹서비스가 필요한 중소기업형 프레임워크인 WeLink에 적용하기 위한 데이터베이스 보안 기법과 실제 구현한 사례를 살펴보겠다.

2. 관련 연구

본 장에서는 기업에서의 데이터베이스 보안 위협과 데이터베이스 보안 관련 기술에 대하여 기술한다.

2.1. 기업에서의 데이터베이스 보안 위협

기업의 많은 정보를 보유하고 있는 데이터베이스는 천재지변과 같은 물리적 손상 외에 악의적인 의도로 취약점을 주로 공격하는 외부 위협과 내부의 인가자와 비인가자에 의한 내부 위협으로 나눌 수 있다[1]. 공격 유형도 데이터베이스 자체의 공격부터 SQL을 이용한 공격과 Script의 약점을 이용한 Cross site scripting도 있다[3].

2.2. 데이터베이스 보안 관련 기술

데이터베이스 보안 위협으로부터 기업의 소중한 데이터를 지키기 위하여 데이터베이스 보안 솔루션을 제공하거나 정보통신망법 상 개인정보보호 강화조치를 시행한다. Gateway 방식, Sniffing 방식, Hybrid 방식등의 보안 아키텍처를 적용한 솔루션을 적용하기도 하고 데이터 암호화를 통한 데이터베이스 구축을 시행하기도 한다[4,5]. 보안의 기본은 비밀유지, 데이터의 완전성, 유효성을 보장하는 것이다[6]. 또한 물리적, 논리적 데이터베이스의 완전성뿐만 아니라 접근 제어, 사용자 인증 등을 요구하며, 사람을 비롯한 어플리케이션, 네트워크, 운영체제 등의 레벨별로 보장되어야 한다[7].

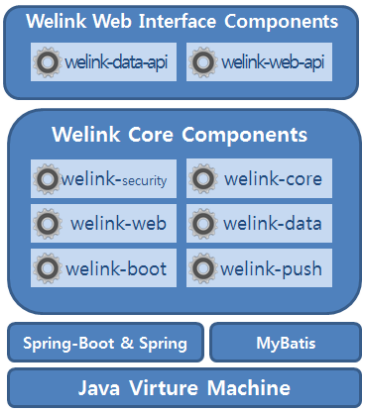
3. WeLink의 데이터베이스 보안 설계 및 구현

3.1. 프레임워크와 쿼리 은폐 기법 적용

WeLink는 Spring-Boot 기반, Embedded WAS형태의 경량 Web Application Framework이며, Database의 보안을 보장하고자 Spring - Security를 적용하였다. 간단한 설정만으로 데이터베이스 보안을 위한 흐름제어, 접근제어, 완전성보장, 암호화, 인증/권한회득 등[8]의 다양한 기법을 제공한다. 웹 어플리케이션을 위한 데이터베이스는 로그인, 인증, 프로그램 제어 등의 모듈로 구성할 수 있다[9].

다음 [그림 1]은 WiLink Stack 구성도이다. Layer구조를 가지며, Based Components와 Core와 Interface로 구성되어 있다. 개발자는 End-point에 해당되는 Interface Components를 이용하여 쿼리의 새로운 기술없이 데이터

베이스에 접근 및 조회, 수정, 삭제 등을 수행할 수 있다.



[그림 1] WeLink Stack

3.2. WeLink의 쿼리 은폐 기법 구현

Java기반의 일반적 프로그래밍에서는 다음 <표 1>과 같이 데이터베이스 접속 관리와 데이터 조작어 모음을 별도 작성 및 관리를 한다.

<표 1> XML기반의 설정

```
<configuration>
  <environments default="development">
    <environment id="development">
      <transactionManager type="JDBC"/>
        <dataSource type="POOLED">
          <property name="driver" value="com.mysql.jdbc.Driver"/>
          <property                                          name="url"
value="jdbc:mysql://test.service.com:3306/database?characterEncoding=UTF-8"/>
          <property name="username" value=""/>
          <property name="password" value=""/>
        </dataSource>
    </environment>
  </environments>
  <mappers>
    <mapper resource="service-crud.xml"></mapper>
  </mappers>
</configuration>

<mapper namespace="service-crud">
    <select id="read" parameterType="long" resultType="readEntity">
        select * from service where pk = #{p_pk}
    </select>
</mapper>
```

하지만 WeLink 프레임워크를 이용하면 서비스 레벨에서는 end-point에 요청할 path, http 1.1 method, request param, action, action param만 정의를 하면 된다. 정의된 규격에 의해 WeLink Web을 통하여 자동 RESTFul API로 변환되어 해당 기능을 구현할 수 있다.

<표 2> WeLink의 쿼리 은폐 기법

```
<endpoints path="/service">
  <endpoint path="/selectServiceAll" method="get">
    <actions>
      <action    name="list"    type="query"    query-id="service.selectServiceAll"
query-operation="select" datasource-id="mysql">
      </action>
    </actions>
  </endpoint>
</endpoints>
```

4. 결론

본 논문에서는 중소기업형 웹 어플리케이션 프레임워크인 WeLink에서 데이터베이스 보안을 위하여 구현한 WeLink-Secutiry에 대하여 살펴보았다. 데이터베이스 보안은 개인정보 유출 등의 사고로 인해 더더욱 필요한 사항이며 기술적 성숙도도 점점 높아질 것으로 기대된다.

향후 연구과제는 클라우드 컴퓨팅 환경에서 WeLink 프레임워크를 적용하기 위한 방안을 모색하며, SaaS 기반의 서비스 제공을 위한 아키텍처 구조를 변경 및 제공할 것이다.

감사의 글

이 연구는 미래창조과학부(공고 제 2016-0218호) '2016년 글로벌 SaaS 육성 프로젝트(GSIP) - 사업 기업형 기반 SaaS' 과제(과제번호 : S0180-16-1038)의 결과임.

참고문헌

[1] FSA 금융보안연구원, "DB 암호화 기술 가이드", 2014.12

[2] Deepika, Nitasha Soni. "Database Security: Threats and Security Techniques." International Journal of Advanced Research in Computer Science and Software Engineering. Volume 5, Issue 5(May 2015): 621-624

[3] Varunkumar, K. A., et al. "Various Database Attacks and its Prevention Techniques." International Journal of Engineering Trends and Technology (IJETT) 9

[4] 이병엽, 박준호, 김미경, 유재수. "데이터베이스 규제 준수, 암호화, 접근제어 유형 분류에 따른 체크리스트 구현." 한국콘텐츠학회논문지, v11, no.2(2011.2): 61-68

[5] 이병엽, 임종태, 유재수. "데이터베이스 암호화 솔루션 구현 및 도입을 위한 기술적 아키텍처." 한국콘텐츠학회논문지, v14, no.6(2014.6): 1-10

[6] Sandhu, Ravi S., and Sushil Jajodia. "Data and database security and controls." Handbook of information security management, Auerbach Publishers (1993): 1-37

[7] Gaikwad, Tejashri R., and A. B. Raut. "A Review on Database Security." International Journal of Science and Research (IJSR). Volume 3, Issue 4(April 2014): 372-374

[8] Almutairi, Abdulrahman Hamed, and Abdulrahman Helal Alruwaili. "Security in database systems." Global Journal of Computer Science and Technology Network, Web & Security 12 (2012): 9-14

[9] Zhu Yangqing, Yu Hui, Li Hua, and Zeng Lianming. "Design of A New Web Database Security Model1." Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium(2009): 292 - 295