JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 25, No. 4, pp. 1079-1085, Apr. 2024

Check for updates

사용자 식별 기반의 데이터베이스 보안 시스템

이 석 우[1] · 강 민 욱[1] · 김 지 영[1] · 정 회 경[2*]
[1]배재대학교 컴퓨터공학과 박사과정
[2*]배재대학교 컴퓨터공학과 교수

Database Security System based on User Identification

Seok-Woo Lee[1] · Min-Wook Kang[1] · Ji-Young Kim[1] · Hoe-Kyung Jung[2*]

[1]Doctor's Course, Department of Computer Engineering, Paichai University, Daejeon 35345, Korea
[2*]Professor, Department of Computer Engineering, Paichai University, Daejeon 35345, Korea

[요 약]

데이터 유출은 심각한 손실을 초래하며, 최근 정보 유출 사고가 증가하여 보안의 중요성이 강조되고 있다. 이를 방지하기 위해 접근제어, 암호화, 데이터 마스킹, 데이터 손실 방지, DB 감사 등의 보안 기술이 적용되어 있다. 그러나 IBM과 포네몬 연구소의 2020년 보고서에 따르면, 조사된 기업 중 80%가 개인 정보 유출을 경험했으며, 보안 자동화 기술을 사용하지 않은 기업은 더 큰 손실을 보았다. 국내 기업에서 데이터 유출로 인한 피해액은 38억 원으로, 해킹 기술이 보안 기술을 앞서고 있다. 특히 웹애플리케이션 서버와 데이터베이스 서버 간의 보안 취약점이 문제이며, 웹방화벽의 한계도 지적된다. 본 논문에서는 이러한 문제를 해결하기 위해 웹애플리케이션 서버와 데이터베이스 서버 간의 통합 보안 관리 시스템을 제안한다. 이 시스템은 웹트랜잭션 사용자 식별과 데이터베이스 감사 추적을 통해 보안을 강화하며, 다양한 환경에 적용하여 데이터 유출에 의한 피해를 줄일 수 있음이 확인되었다.

[Abstract]

Data leaks cause severe losses, and recent increases in information-leak incidents emphasize the importance of data security. Various security technologies can be used to prevent data leaks, such as access control, encryption, data masking, data loss prevention, and database auditing. However, according to a 2020 report from IBM and the Ponemon Institute, 80% of the surveyed companies experienced personal information breaches, with greater losses suffered by those who did not use any security automation technologies. Domestic companies incur an average data loss of 3.8 billion won, with hacking technologies outpacing security measures. In particular, security vulnerabilities between web-application and database servers pose a critical issue, as well as the limitations of web firewalls. To solve this problem, this paper proposes an integrated security management system between web-application and database servers. This system strengthens security using web transaction user identification and database audit trails. The proposed system is expected to find significant applications in reducing data-leakage damages in various fields.

색인어 : SQL Injection, 데이터베이스 감사 정책, 데이터베이스 보안, 사용자 식별, 지능적 사이버 공격
Keyword : SQL Injection, Database Audit Policy, Database Security, User Identification, Intelligent Cyber Attacks

http://dx.doi.org/10.9728/dcs.2024.25.4.1079

Received 27 February 2024; **Revised** 22 March 2024
Accepted 15 April 2024

***Corresponding Author; Hoe-Kyung Jung**

Tel: +82-42-520-5640
E-mail: hkjung@pcu.ac.kr

Ⅰ. 서 론

DB에 저장된 제품 정보, 고객정보, 금융거래, 연구 결과, 비즈니스 전략 등은 기업의 중요 자산이다. 이러한 데이터의 훼손이나 유출은 재무적 손실, 기업 명성 훼손, 법적 책임, 운영 중단 등 문제를 일으킬 수 있다[1]. 이에 기업들은 DB 접근제어, 암호화, 마스킹, 손실 방지(Data Loss Prevention, DLP), 감사 등의 보안 기술을 적극적으로 도입하고 있다. 그럼에도 불구하고 데이터 유출 사고는 지속적으로 발생하고 있으며, 기업은 유출 사실을 늦게 알게 되거나 모르고 지나치는 경우가 많다[2]. DB 접근 제어는 성능 문제로 웹애플리케이션 서버와 DB와의 연동 시에는 제한적으로만 적용되며, DB암호화의 경우는 웹애플리케이션 레벨에서 이미 복호화된 데이터를 처리하므로 예외가 발생한다. 데이터베이스 감사 기능은 웹애플리케이션 사용자를 식별하지 못하며 감사 정책 위반 트랜잭션을 탐지하지만 차단하지는 못한다.

본 논문은 웹애플리케이션 사용자를 식별하고 이 정보를 DB와 연동하여, 감사 정책 위반 사용자를 차단하는 시스템을 설계 및 구현한다. 이 시스템은 웹트랜잭션 정보, 사용자 식별 정보, DB 감사 정보를 빅데이터 시스템에 저장하여, DB 보안 문제에 대한 종합적인 해결책을 제공한다. 이를 통해 보안 위협에 대응하고 DB 보안을 강화할 수 있는 의미 있는 개선 방안을 제시한다.

Ⅱ. 관련 연구

제2장에서는 기존의 DB 보안 문제점 및 개선 방안 연구에 관해 기술한다.

2-1 DB 보안 기술

DB 보안은 데이터의 가용성, 기밀성, 무결성을 확보하며, DB 및 애플리케이션에 대해 보안의 보장을 의미한다. DB에 보안을 위해 데이터 암호화, 데이터 접근제어, 데이터 손실 방지, 데이터 마스킹, DB 감사 등의 보안 기술이 적용되어 운영된다[3]-[5].

2-2 DB 접속 방식별 사용자 식별 방법

DBA(Database Administrator), 웹애플리케이션 서버, 일반 사용자 등 미들웨어에서 DB에 접속하여 데이터를 액세스하며, 접속 방식별로 DB에서 사용자를 식별하지 못할 수도 있고 식별할 수도 있다[6]. PC에서 직접 접속할 때는 사용자를 식별할 수 있고, 접근제어 시스템을 경유할 때는 사용자 식별이 안 되는 예도 있다. 특히 웹애플리케이션 서버에서 접속하는 경우 접속하는 사용자가 웹애플리케이션 서버로 인식되어 DB에서 사용자 식별이 안 된다[7].

2-3 웹애플리케이션에서 사용자 식별 방법

웹애플리케이션 서버로부터 DB에 접속하는 경우 DB에서는 사용자를 식별할 수 없고 웹애플리케이션 서버로 접속하는 사용자는 세션을 유지하기 위해 여러 방식으로 사용자 식별 방식을 제공한다[8],[9]. 웹애플리케이션에서 사용자를 식별하는 방법에서 쿠키, 토큰, 세션, 핑거 프린팅, IP주소 등을 이용하는 방식이 있으며, 다른 서비스의 계정 정보를 이용하여 식별하는 방식이 있다[10]-[13].

2-4 기존 DB 보안 기술의 문제점

DB에 저장된 데이터 보호를 위해 데이터 암호화, 데이터 접근제어, 데이터 손실 방지, 데이터 마스킹, DB 감사 등의 여러 가지 보안 기술이 적용되어 운영되지만, 웹애플리케이션 서버에서 접속하는 경우 표 1과 같이 기존 기술들이 소용없게 되는 문제가 있다.

표 1. 웹애플리케이션 접속 시 DB 보안의 문제점
Table 1. Problems with DB security when accessing web applications

Security technology	Problem
Access control	Since the user cannot be identified, it is difficult to set up access control and performance problems can occur
Data encryption	When accessed from a web application, it is ineffective because it is accessed in a decrypted state
Data masking	The technology used to build demo and test environments is not related to the operating environment
Data loss prevention	No effect as data encryption
DB audit	Since the user is not identified, it is impossible to know who caused the problem and cannot be blocked.

Ⅲ. DB 보안 시스템의 설계

제3장에서는 DB 보안 에이전트를 웹애플리케이션 서버에 설치하고 사용자의 세션 아이디를 식별하며, 식별된 사용자 세션 아이디를 DB에 연동하여 DB 감사 정책에 위배되는 사용자에 대해 웹애플리케이션 서버에서 차단하는 DB 보안 시스템에 대한 설계를 다룬다.

3-1 시스템 구성도

웹애플리케이션 서버에 DB 보안 에이전트를 추가하고 DB 보안 관리 시스템에서 DB에 연동하는 DB 보안 시스템을 구성하였다. 그림 1은 시스템의 전체 구성도로 본 연구에서 제

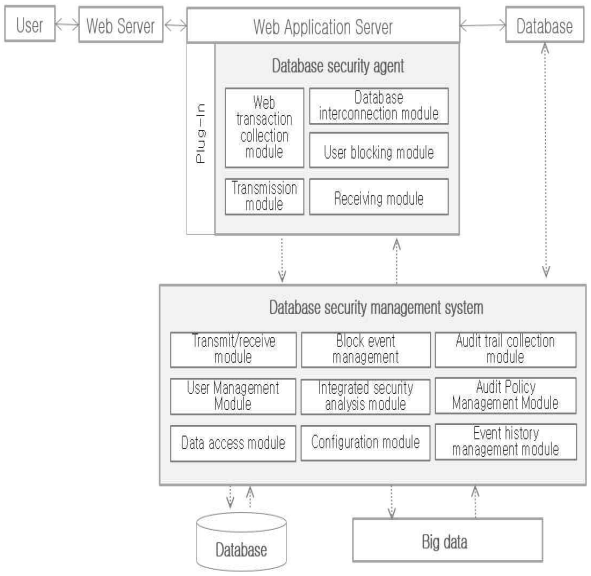


그림 1. 전체 시스템 구성도
Fig. 1. Entire system configuration diagram

안하는 DB 보안 시스템은 DB 보안 관리 시스템과 DB 보안 에이전트로 구분된다.

3-2 DB 보안 에이전트 설계

DB 보안 에이전트는 웹트랜잭션을 수집하는 수집모듈, 사용자 정보를 DB와 연동하는 DB연동 모듈, 수집한 데이터를 DB보안 관리 시스템으로 전송하는 전송모듈, 차단 대상 정보를 DB보안 관리 시스템으로 부터 수신하는 수신모듈, 차단 대상 웹트랜잭션을 차단하는 모듈 등으로 구성되며 이를 통해 웹애플리케이션 서버에서 사용자 식별 정보와 웹트랜잭션 정보를 수집하고 DB에 사용자 식별 정보에 대해 연동 작업을 수행하고, DB 보안 관리 시스템으로부터 전송되어 온 차단 명령을 수신하여 사용자를 차단하는 DB 보안 에이전트에 대

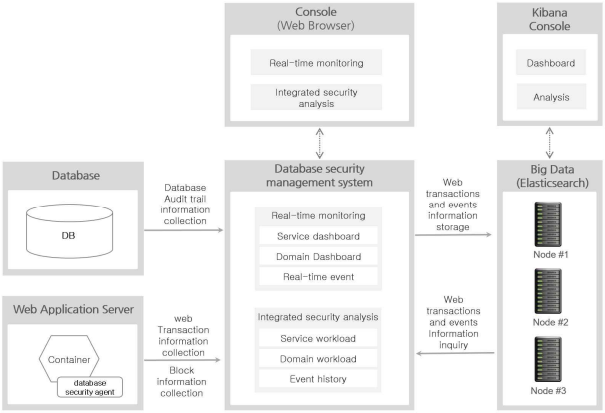


그림 2. DB 보안 관리 시스템 구성도
Fig. 2. DB security management system configuration diagram

해 설계하였다. DB 보안 에이전트는 웹애플리케이션 서버에 플러그인 방식으로 동작하고 비정상적인 사용자가 웹애플리케이션을 통해 DB에 접근하는 것을 차단한다.

3-3 DB 보안 관리 시스템 설계

그림 2는 DB 보안 관리 시스템의 구성도로 DB 보안 에이전트와 데이터 송수신, DB 및 빅데이터에 데이터 액세스, 차단 이벤트 관리, 사용자 관리, 환경 설정, 통합 보안 분석, 감사 정책관리, 이벤트 이력 관리, 감사 추적 정보 수집 등 9개의 모듈로 구성되고 사용자 관리, 환경 설정, DB 감사 정책관리, DB 감사 추적 정보 수집 및 차단 관리, 웹트랜잭션, 이벤트 관리, 차단 정보 수신 및 저장, DB 통합 보안 분석 등의 세부 기능이 포함한다.

Ⅳ. DB 보안 관리 시스템 구현

본 장에서는 제안하는 DB 보안 시스템을 구현 내용을 기술한다.

그림 3은 시스템의 구성도를 나타낸다. 웹애플리케이션 사용자는 DB와 웹애플리케이션을 통해 서비스 받도록 연동되며, DB와 웹애플리케이션 서버에 연동되는 DB 보안 관리시스템으로 구성된다. DB 보안 관리 시스템은 데이터 저장소로 Elasticsearch 빅데이터 저장소와 연동되도록 구현하였다.

4-1 구현 환경

표 2는 제안하는 시스템의 구현 환경이다. IntelliJ와 JDK를 통해 시스템의 개발 환경을 구축하여 DB 보안 웹애플리케이션과 DB 보안 관리 에이전트, 그리고 시험용 웹애플리케이션을 개발하였다. Jetty를 이용하여 개발한 웹애플리케이션을 구동한다. Derby DB에서는 환경 설정과 사용자 정보를 저장한다. Elasticsearch 클러스터 환경을 구축해 감사 추적 정보, 웹트랜잭션, 이벤트 정도를 저장하는 빅데이터 저장소

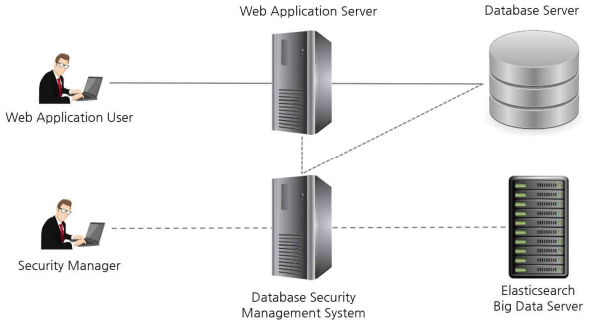


그림 3. 시스템 구성도
Fig. 3. System configuration diagram

표 2. 시스템 구현 환경
Table 2. System implementation environment

Type	Composition
Tools	IntelliJ, Derby DB, JDK, Kibana Tomcat, Jetty, Elasticsearch
OS	Linux Ubuntu 18.04.6 LTS, Windows 10 Pro

로 사용하였으며, Kibana를 이용해 빅데이터 분석 및 모니터링에 활용하였다. Tomcat을 이용하여 시험용 웹애플리케이션을 구동해 기능을 시험하였다.

4-2 DB 보안 에이전트 구현

DB 보안 에이전트 구현을 위해 자바 에이전트를 구현하였다. 그림 4는 JAVA 에이전트 요구사항에 따라 에이전트 메인 클래스인 JavaAgent.class 클래스와 MANIFEST.MF가 JVM에 적재될 때 재정의를 실행하는 AgentTransformer.class와 기타 에이전트 기능들이 정의된 클래스들에 대해 agent.jar 형식으로 압축되어 실행된다.

이렇게 구성된 에이전트는 사용자가 호출하여 웹애플리케이션 서버에서 처리가 시작되는 클래스에 대해 클래스 로더에 의해 메모리에 적재 될 때 클래스를 재정의하여 웹트랜잭션 정보가 수집될 수 있도록 구현되었고, 사용자의 식별 정보로는 HTTP 요청 헤더 클래스인 HttpServletRequest 객체에서의 getRequestedSessionId() 메소드를 호출하여 수집하였다. 수집된 정보는 웹애플리케이션에서 DB에 대해 SQL을 실행하기 바로 전에 DB 세션 정보 중에서 CLIENT_ID에 정보를 설정하는 프로시저를 호출함으로써 사용자 식별 정보를 연동하도록 구현하였다. 또한 DB 보안 관리 시스템에서 수신된 정보를 차단 목록에 넣고, 이후 호출에 대해 세션 아이디를 비교해 차단한다.

4-3 DB 보안 관리 시스템 구현

DB 보안 관리 시스템은 크게 감사 추적 정보 수집, 사용자 관리 및 환경 설정, 차단 이벤트 관리, 감사 정책관리, 이벤트

그림 4. JAVA agent
Fig. 4. JAVA agent

*Implementation screen
그림 5. 감사 정책관리 화면
Fig. 5. Audit policy management screen

이력 관리, DB와 빅데이터 데이터 액세스, 데이터 송수신 및 통합 보안 분석과 데이터 보안 에이전트 등의 9개 모듈로 구성된다. JAVA, CSS, JSP/Servlet 등으로 웹애플리케이션을 개발하여 jetty 웹애플리케이션 서버에서 구동될 수 있도록 구현하였다. 웹브라우저를 이용하여 접속해 웹 콘솔에 로그인하여 사용한다.

그림 5는 감사 정책관리 화면으로 DB에 설정된 감사 정책을 조회하고 정책을 삭제 및 추가한다. 그림 6은 수집된 웹트랜잭션 정보가 시각화 도구인 Kibana를 이용해 Elasticsearch에 저장된 내용을 조회한 화면이다.

그림 7은 통합 보안 분석을 위해 구현한 화면으로 Elaiarch의 Query DSL에 Aggregation을 적용해 웹트랜잭션 추이 분석 및 분포도가 가능하다.

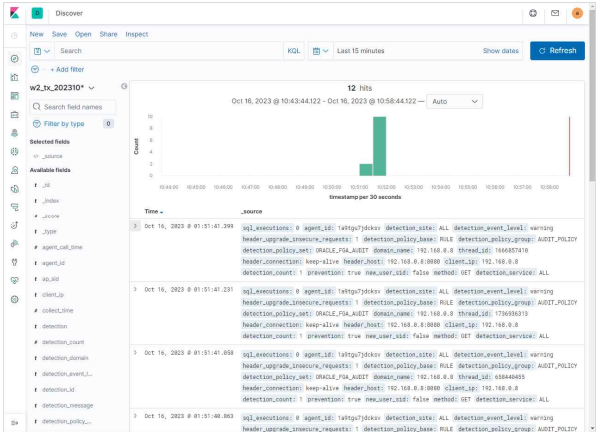
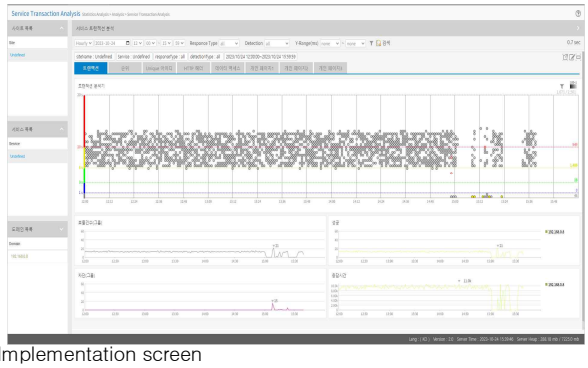


그림 6. 웹트랜잭션 정보 조회 화면
Fig. 6. Web transaction information inquiry screen



*Implementation screen
그림 7. DB 통합 보안 분석 화면
Fig. 7. DB integrated security analysis screen

V. 실험 및 고찰

5-1 시나리오

SQL Injection 공격 시 웹애플리케이션서버를 이용해 DB 내의 주요한 개인 정보를 탈취하는 비정상적인 사용자를 탐지하고 차단하는 기능을 시험하는 시나리오이다. 시험을 위해 사용한 테이블은 USER과 BOARD이다. BOARD 테이블에서 조회 조건에 해당하는 데이터를 조회하는 정상 SQL에 대해 SQL Injection을 통하여 BOARD 테이블과 개인 정보가 저장된 USERS 테이블에 대한 모든 데이터가 조회되도록 하는 시나리오를 구성하였다. 표 3과 4는 BOARD와 USER 테이블이다.

표 3. BOARD 테이블 데이터 목록
Table 3. BOARD table data list

ID	Title	Contents
user1	Israel and Hamas War	This war began on October 7, 2023.
user2	Russia and Ukraine War	This war began on February 24, 2022.
user3	American Independence Day	July 4, 1776 American Declaration of Independence

표 4. USERS 테이블 데이터 목록
Table 4. USERS table data list

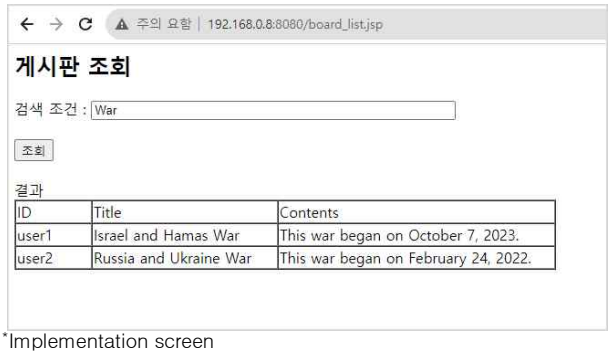
ID	Password	Mobile Number
user1	password1	010-1111-1111
user2	password2	010-1111-2222
user3	password3	010-1111-3333
user4	password4	010-1111-4444
user5	password5	010-1111-5555

5-2 실험

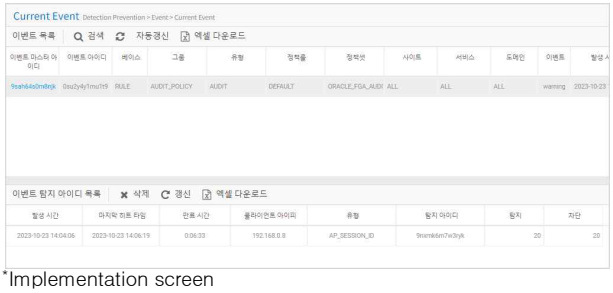
BOARD 테이블에서 검색하고자 하는 문자열을 입력하여 정상적인 조회가 되는지 확인하며, 검색하는 문자열에 SQL Injection 문자열을 추가하여 USERS 테이블에 있는 내용까지 조회되는지 확인하였다. 또한 SQL Injection으로 USERS 테이블이 조회 시 DB 감사 정책으로 설정한 USERS_TABLE_POLICY에 따라 감사 추적 정보가 발생하고, DB 보안 관리 시스템에서 추적 정보를 수집하여 차단 이벤트가 발생하며, DB 보안 에이전트로부터 차단 메시지가 전송되어 동일한 사용자의 요청에 대해 차단되는지 확인하였다.

그림 8은 사용자가 조회 조건을 입력하고 조회할 때 BOARD 테이블에서 전체 3건 데이터 중에서 'War' 문자열이 포함된 2건이 조회된다.

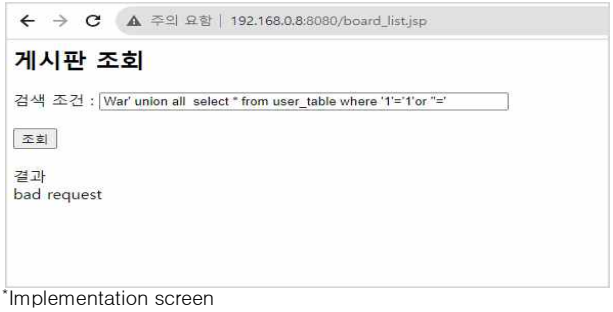
또한 감사 추적 정보 발생으로 인해 DB 보안 관리 시스템에서 차단 이벤트 발생이 확인되었다. 그림 9는 차단 이벤트 발생 상황으로 세션 아이디, 차단 시작 시각, 차단 건수 등의



*Implementation screen
그림 8. 감사 정책관리 화면
Fig. 8. Audit policy management screen



*Implementation screen
그림 9. 차단 이벤트 발생 화면
Fig. 9. Blocking event occurrence screen



*Implementation screen
그림 10. 사용자 차단 결과 화면
Fig. 10. User blocking result screen

정보가 관리된다.

그림 10은 DB 보안 에이전트에서 해당하는 사용자의 웹트랜잭션이 차단된 뒤에 조회 버튼을 클릭 시 DB 보안 에이전트에서의 차단 메시지인 'bad request'가 화면에 출력되어 정상 차단이 확인되었다.

5-3 고찰

웹애플리케이션에서 DB로 접속하는 환경에서 기존 DB 보안 기술이 적용되지 않는 문제를 해결하기 위해, 설계된 DB 보안 관리 시스템을 구현하고 시험하였다. 이 시스템은 웹애플리케이션 서버에 플러그인되어, 웹트랜잭션 정보 수집, 비정상 사용자 차단, 사용자 식별 정보 수집 및 DB 연동 등을 수행하는 보안 에이전트와 DB 감사 정책관리 및 통합 보안

분석 기능을 포함한다.

시험을 위해, DB에 사용자 정보와 게시판 데이터를 입력하고, 특정 DB 사용자가 사용자 정보 테이블에 대한 조회를 실행했을 때 감사 추적 정보가 발생하도록 설정하였다. 또한, 시험용 웹애플리케이션을 개발하여 SQL Injection 공격 시나리오를 포함한 조회 시험을 진행하였다. 이 과정에서 정상적으로 조회되는 데이터와 함께 개인 정보가 포함된 데이터가 조회됨을 확인하였고, 이에 따라 DB 감사 추적 정보가 생성되고, 해당 사용자에 대한 차단이 이루어짐을 확인했다.

이러한 시험 결과는 제안된 시스템이 DB 내에서 사용자 식별 및 감사 정책 위반 사용자를 효과적으로 차단하는 기능을 정상적으로 수행하며, 기대했던 목적이 달성됨을 보여주었다.

Ⅵ. 결론 및 향후 연구

데이터 보안의 필요성이 강조되고 있지만 기존 DB 보안 기술로는 해킹 기술의 발전에 대응하기 어렵다. 특히, 웹애플리케이션과 DB 서버 간의 연결에서 보안 취약점이 발생한다. 이러한 문제를 해결하기 위해, 본 논문에서는 웹애플리케이션을 통한 DB 접근 시 발생하는 보안 문제에 주목하고 새로운 보안 시스템을 제안했다.

본 논문에서 제안하는 DB 보안 관리 시스템은 웹애플리케이션 사용자를 DB에서 식별하고 보안 위반 사용자를 차단하는 기능을 포함한다. 또한, 보안에 관련되는 데이터를 빅데이터 시스템에 저장하고 분석하여 다양한 보안 위협에 대응한다. 시험 결과, 시스템은 SQL Injection 공격을 통해 민감 정보를 조회하는 시나리오에서도 사용자를 올바르게 식별하고 차단하는 데 효과적이었다.

실험을 통해, 본 연구에서 제안하는 시스템은 다양한 사용자 식별 정보를 활용하여 더욱 정확한 사용자 식별이 가능하며, DB 감사 정책을 다양화하여 보안 강화에 기여할 수 있음을 검증했다. 향후 연구로는 웹애플리케이션 모니터링과 본 연구 결과를 결합하여 통합 DB 보안 관리 시스템을 연구할 계획이다.

감사의 글

이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 지역지능화혁신인재양성사업임(IITP-2024-RS-2022-00156334). 본 과제(결과물)는 2024년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력기반 지역혁신 사업의 결과입니다(2021RIS-004).

참고문헌

[1] KOREA Data Agency. DB Security Purpose and Necessity [Internet]. Available: https://dataonair.or.kr/db-tech-reference/d-guide/db-security/ ?mod=document&uid=425.

[2] Tech World News. 2020 Global Corporate Data Leakage Status [Internet]. Available: https://www.epnc.co.kr/news/articleView.html?idxno=100975.

[3] P. Paul and P. Aithal, “Database Security: An Overview and Analysis of Current Trend,” *International Journal of Management, Technology, and Social Sciences,* Vol. 4, No. 2, pp. 53-58, October 2019. https://doi.org/10.2139/ssrn.3497728

[4] A. Mousa, M. Karabatak, and T. Mustafa, “Database Security Threats and Challenges,” in *Proceedings of 2020 8th International Symposium on Digital Forensics and Security,* Beirut, Lebanon, pp. 1-5, June 2020. https://doi.org/10.1109/ISDFS49300.2020.9116436

[5] M. ke, Computer Database Security and Oracle Security Implementation, Master’s Thesis, The University of Montana, Missoula: MT, June 2001. Available: https://scholarworks.umt.edu/cgi/viewcontent.cgi?article=6127&context=etd

[6] J. S. Han and D. C. Shin, “A Non-Agent Based Identification Scheme for Identifying Database Users in 3-tier System Environments,” *Journal of Information Technology Applications & Management,* Vol. 25, No. 2, pp. 147-159, June 2018. https://doi.org/10.21219/jitam.2018.25.2.147

[7] G.-M. Go, S.-J. Bu, and S.-B. Cho, “Learning Separate Expressions for User Queries to Detect Database Insider Attacks,” *KIISE Transactions on Computing Practices,* Vol. 27, No. 2, pp. 76-82, February 2021. https://.doi.org/10.5626/KTCP.2021.27.2.76

[8] S. Lee, S. Park, and H. Jung, “Web Monitoring Based Encryption Web Traffic Attack Detection System,” *Journal of the Korea Institute of Information and Communication Engineering,* Vol. 25, No. 3, pp. 449-455, March 2021. https://doi.org/10.6109/jkiice.2021.25.3.449

[9] S. Yoon, “Study on the Technical Security Factor for the Implementation of Secure DB System,” *Journal of the Korea Society of Computer and Information,* Vol. 19, No. 12, pp. 143-152, December 2014. https://doi.org/10.9708/jksci.2014.19.12.143

[10] H. Choi, H. Kim, H. Park, K. Chun, and H.-H. Lee, “Research on security requirements for personal information DB management technology,” *The Korea Institute of Information Security and Cryptology,* Vol. 18, No. 2, pp. April 76-86, 2008.

[11] W. Kim, S. Kang, K. Kim, and S. Kim, “Detecting Shell-

Code Using Entropy," *KIPS Transactions on Computer and Communication Systems*, Vol. 3, No. 3, pp. 87-96, March 2014. https://doi.org/10.3745/KTCCS.2014.3.3.87

[12] R. J. S. Raj, M. Viju Prakash, T. Prince, K. Shankar, V. Varadarajan, and F. Nonyelu, "Web Based Database Security in Internet of Things Using Fully Homomorphic Encryption and Discrete Bee Colony Optimization," *Malaysian Journal of Computer Science,* pp. 1-14, November, 2020. https://doi.org/10.22452/mjcs.sp2020no1.1

[13] S. Lee, A Study on the Database Security System Based on User Identification, Ph.D. Dissertation, Graduate School of Paichai University, Deajeon, December 2023.

이석우(Seok-Woo Lee)

2004년 : 한밭대학교(공학사-전자공학)
2020년 : 배재대학교 대학원
(공학석사-컴퓨터공학)
2024년 : 배재대학교 대학원
(공학박사-컴퓨터공학)
1996년～1997년: 육군중앙경리단
1997년～2000년: ㈜풍한산업
2000년～2005년: ㈜아이텍
2014년～현 재: ㈜엘리바이저 대표이사
※관심분야 : 보안(Security), 빅데이터(Big data), 인공지능(AI) 등

강민욱(Min-Wook Kang)

2009년 : 배재대학교
(공학사-정보통신공학)
2020년 : 배재대학교 대학원
(공학석사-컴퓨터공학)
2024년～현 재: 배재대학교 스마트ICT융합전공 박사과정
2019년～현 재: 성한주식회사 근무
※관심분야 : 계측(Instrumentation), 빌딩자동제어 (Building automatic control), 제어(Control) 등

김지영(Ji-Young Kim)

2000년 : 한양대학교 원자력공학과
(학사)
2009년 : 연세대학교 공학대학원
(석사-유비쿼터스컴퓨팅)
2000년～2014년: ㈜ 엘지씨엔에스
2014년～2017년: ㈜ 아이티센
2024년～현 재: 배재대학교 스마트ICT융합전공 박사과정
2018년～현 재: ㈜ 소프트아이텍
※관심분야 : 빅데이터(Big data), 인공지능(AI) 등

정회경(Hoe-Kyung Jung)

1985년 : 광운대학교
(공학사-컴퓨터공학)
1987년 : 광운대학교 대학원
(공학석사-컴퓨터공학)
1993년 : 광운대학교 대학원
(공학박사-컴퓨터공학)
1994년～현 재: 배재대학교 컴퓨터공학과 교수
※관심분야 : 머신러닝(Machine learning), 빅데이터(Big data), 임베디드 시스템(Embedded system), IoT 등