韓國IT서비스學會誌
第21卷 第4號
2022年 8月, pp.63-74

Journal of Information Technology Services
https://doi.org/10.9716/KITS.2022.21.4.063

# OHDSI OMOP-CDM 데이터베이스 보안 취약점 및 대응방안

이경환* · 장성용**

# OHDSI OMOP-CDM Database Security Weakness and Countermeasures

Kyung-Hwan Lee* · Seong-Yong Jang**

■ Abstract ■

Globally researchers at medical institutions are actively sharing COHORT data of patients to develop vaccines and treatments to overcome the COVID-19 crisis. OMOP-CDM, a common data model that efficiently shares medical data research independently operated by individual medical institutions has patient personal information (e.g. PII, PHI). Although PII and PHI are managed and shared indistinguishably through de-identification or anonymization in medical institutions they could not be guaranteed at 100% by complete de-identification and anonymization. For this reason the security of the OMOP-CDM database is important but there is no detailed and specific OMOP-CDM security inspection tool so risk mitigation measures are being taken with a general security inspection tool. This study intends to study and present a model for implementing a tool to check the security vulnerability of OMOP-CDM by analyzing the security guidelines for the US database and security controls of the personal information protection of the NIST. Additionally it intends to verify the implementation feasibility by real field demonstration in an actual 3 hospitals environment.

As a result of checking the security status of the test server and the CDM database of the three hospitals in operation, most of the database audit and encryption functions were found to be insufficient.

Based on these inspection results it was applied to the optimization study of the complex and time-consuming CDM CSF developed in the "Development of Security Framework Required for CDM-based Distributed Research" task of the Korea Health Industry Promotion Agency.

According to several recent newspaper articles, Ramsomware attacks on financially large hospitals are intensifying. Organizations that are currently operating or will operate CDM databases need to install database audits(proofing) and encryption (data protection) that are not provided by the OMOP-CDM database template to prevent attackers from compromising.

Keyword : Security, Requirements, OMOP-CDM, STIG, PostgreSQL, Weakness, Tool, Check

Submitted : June 25, 2022 1st Revision : August 12, 2022 Accepted : August 22, 2022

* 서울과학기술대 IT정책전문대학원 박사과정
** 서울과학기술대 IT정책전문대학원 교수, 교신저자

# 1. 서 론

관찰 의료 데이터의 오픈소스 커뮤니티 표준을 제시하는 OMOP-CDM(Observational Medical Outcomes Partnership Common Data Model)은 의료 관련 조치 방안(절차, 의료 정책 변경, 약물 노출 등)과 이런 조치 방안이 유발한 이익과 악영향 결과(상태 발생, 절차, 약물 노출 등) 간의 연관성을 식별하고 평가하는 연구 수행을 지원하기 위해 설계되었다(OHDSI, 2021). 관측 의료 데이터 과학 및 정보학 기관인 OHDSI(Observational Health Data Sciences and Informatics)는 분산 연구망(DRN Distributed Research Network)에서 임상학적 특성, 인구학적 평가 및 예측을 지원하기 위해 오픈소스 도구로 구성된 정보기술 생태계를 구축하여 제시하였다.

OHDSI는 생태계에서 데이터 분석 기능을 제공하는 ATLAS/WebAPI를 통하여 인증과 인가의 기본적인 보안 체계를 제시하였지만, 저장된 자체 OMOP-CDM 데이터베이스에 관한 보안 체계는 제시하지 않고 있다.

OMOP-CDM의 목적상 데이터베이스, 테이블 및 컬럼의 속성들이 일반에게 공개되어 있어 개인정보, PII(Personally Identifiable Information) 및 PHI(Protected Health information)를 공격자가 쉽게 침해할 수 있는 표적이 되기 때문에 데이터베이스 보안 권고사항이 필요하다.

COVID-19로 인하여 백신이나 치료제 연구를 위하며 많은 병원과 제약회사가 자체 병원의 코호트 자료를 더욱 많이 교류 공유하는 상황에서, 만일 하나의 병원이라도 기밀성, 무결성 및 가용성이 문제가 있는 연구 자료를 교류 공유한다면 그 자료를 바탕으로 개발한 백신이나 치료제의 신뢰성에 나쁜 영향을 줄 수도 있다.

본 연구에서는 최고의 보안 체계를 구현하고 있는 미국군병원을 포함한 미국국방산하기관 공통데이터모델의 데이터베이스 보안 지침 STIG(Security Technical Implementation Guides)(DISA, 2022)를 분석하여 OMOP-CDM 환경에서 강력하고 효율적인 데이터베이스 보안 체계를 적용하고 구현하기 위한 보안 점검 툴을 제시하고자 한다. 또한 실제 병원 환경에서 보안 점검 툴을 실증하여 구현 타당성 및 현장 CDM 데이터베이스의 보안 실태를 검증하고자 한다.

구현된 보안 점검 툴을 구현하고 활용하여 실제 운영 중인 3개 병원의 CDM 데이터베이스에서 실증하여 구현 타당성을 검증하고, 보안 상태를 점검한 결과, 모든 병원이 데이터베이스 감사 및 암호화 기능이 대부분 미비한 것으로 드러났다. 감사 기능인 pgaudit 만이라도 적절하게 구성하면 보안 태세가 40%정도 향상되리라 예상된다. 또한 본 연구에서 분석 제시한 여러 가지 보안 권고사항을 구현하면 CDM 데이터베이스의 보안 태세가 강화되리라 예상된다.

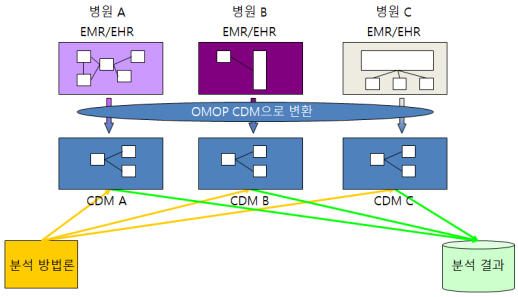
# 2. 연구의 배경

여러 산업분야에서 빅데이터의 이점을 추구하기 위해 공통데이터모델(윤현아, 2021)을 제안하여 데이터를 공유하고 있지만, 구체적인 보안 지침을 제공하지는 않아 랜섬웨어 같은 보안 위험에 처해 있다(Kan, 2017; Wilson, 2021).

오딧세이 컨소시엄은 OMOP-CDM 기반의 분산 연구망(DRN)을 이끌고 나가는 국제적인 비영리 연합체로, 전 세계 산학연 기관들이 자유롭게 참여하여 활동하고 있으며, 비교효과분석, 의료품질검증, 약물 안전성 평가, 경제성 평가 등 보건의료 빅데이터를 활용한 새로운 의학 근거들을 창출하는 공동연구를 수행하고 있다. 또한, 컨소시엄에 참여하고 있는 많은 연구자와 개발자들이 공통된 데이터 구조와 의미를 기반으로 실행되는 다수의 오픈소스 분석 툴들을 개발하고 배포하고 있다. OHDSI는 [그림 1]과 같이 분산 네트워크 기반의 접근법을 추구하고 있다(출처: https://www.ohdsi.org/data-standardization/the-common-data-model/).

OMOP-CDM은 공통데이터모델이라 개인정보를 비식별화 및 익명화를 해서 보안 조치를 하더라

도 테이블 구조가 공개되어 있어 나쁜 의도를 가진 공격자가 쉽게 데이터베이스 구조를 파악하고 공격할 수 있어 특별한 보안 권고사항이 필요하다.




[그림 1] OMOP-CDM 개념 모델

실제 세계의 공격을 관측하여 적대적 전술과 기술에 관하여 일반에 공개한 지식-기반 시스템인MITRE ATT&CK의 분석 기능인 Navigator를 활용하여 의료 기관 공격자를 분석하면 Orangeworm, APT41, Tropic Trooper, Fin4, menuPass 및 Deep Panda 같은 공격자가 지금까지 주로 공격하는 해커이고, 가장 많이 사용하는 공격 기술은 데이터 탈취가 목적인 “스피어피싱 공격”으로 판명되었다.

[그림 2]는 MITRE ATT&CK Navigator를 활용하여 의료 기관을 타겟으로 하는 공격자의 공격 기술의 빈도를 분석한 결과로, 공격 회수 별로 표시된 공격 기술이다. 회수가 많을수록 색깔이 짙게 나타나도록 표시하였다.

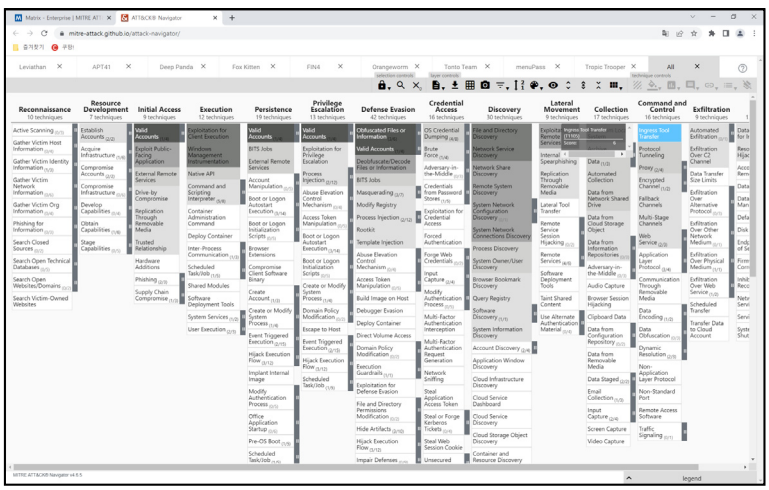
의료기관인 경우에 공격자는 데이터 탈취가 목적인 “Ingres Tool Transfer” 기술을 가장 많이 사용한다고 분석되었다.

그러므로 의료기관은 데이터 보안에 더욱더 관심을 집중해야 된다.

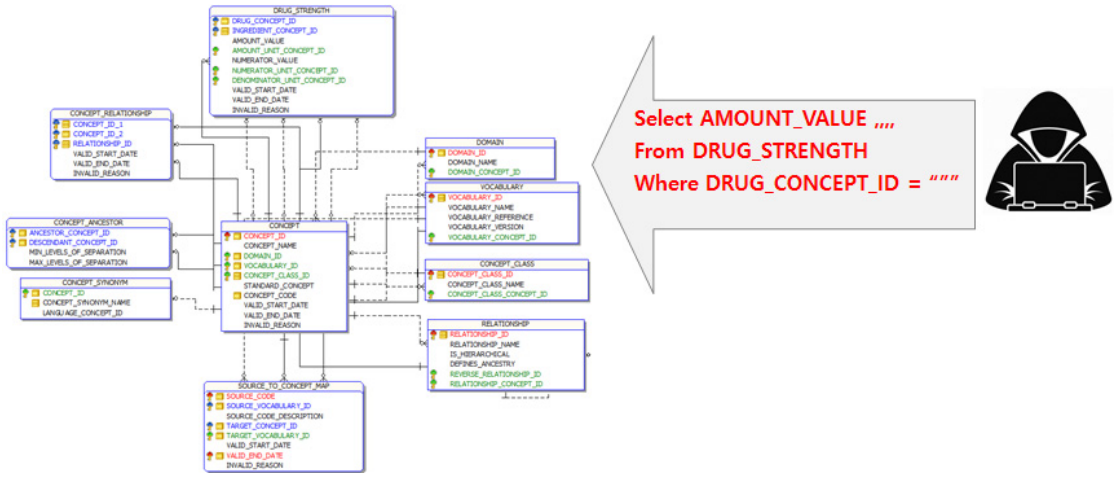
일반적으로 데이터베이스는 접근통제와 암호화를 통하여 보안 권고사항을 강구하고 있지만, 접근통제는 완벽하게 구현하지 않는 한 허점이 있을 수 있고, 암호화는 트랜잭션 속도 때문에 아주 중요한 데이터가 아니 경우에는 전체 데이터에 구현하기는 불가능하다. 그러므로 데이터를 담은 데이터베이스 구조라도 안전하도록 반복적으로 위협을 점검해야 한다.

예를 들어 OMOP-CDM의 경우, 그림 3과 같은 각 테이블의 관계도까지도 공개되어 있기 때문에 공격자에게 공격 표면이 많이 노출되어 있는 상태이다. 공격자가 공개된 자료로 쉽게 공격할 스크립트를 만들 수도 있다.

이런 이유로, OMOP-CDM 데이터베이스의 현재 보안 상태를 점검하고 취약점을 조치할 수 있는 권고사항 및 보안 권고사항을 제시하는 툴이 필요하다.




[그림 2] 의료기관 공격 기술

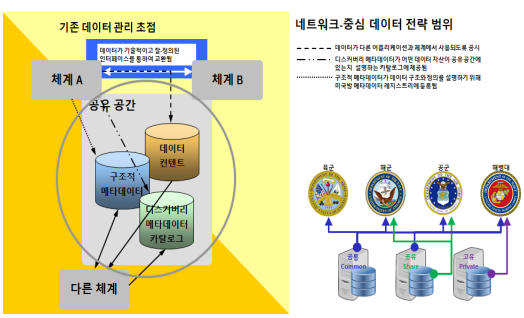



[그림 3] OMOP-CDM 테이블 공격 형태

## 3. 관련 연구

공통데이터모델은 여러 다른 기관에서 공통적인 테이블 구조에 각자의 고유 데이터를 적재하여 분석하는 모델이다. 전세계적으로 가장 보안이 필요하고 강력한 미국국방산하기관을 지원하는 네트워크-중심 데이터 전략(NCDS, Net-Centric Data Strategy)(DoD CIO, 2007)에서 보안기술 구현지침 STIG를 수립하여 보안 지침을 제공하고 있어, 본 연구에서 벤치마킹하여 CDM 환경에 맞는 보안 기술을 분석한다.

아래 그림처럼 NCDS는 공유 공간에 모든 공통, 공유 및 고유 데이터의 구조적 메타데이터, 카탈로그 및 원본 데이터를 등록하고 관리하여 각 군기관 간에 데이터 소통을 원활하게 만든다(DoD CIO, 2007).




[그림 4] NCDS 개념 모델

STIG는 미국방 정보 보증(IA, Information Assurance) 및 IA-적용 장치/시스템을 위한 구성 표준으로, 1988년부터 DISA(Defense Information Systems Agency)가 STIG를 제공하여 미국방 보안 시스템의 보안 태세를 개선하는 필수 역할을 담당하고 있다. STIG는 악성 컴퓨터 공격에 취약할 수 있는 정보시스템/소프트웨어를 엄격 통제하는 기술 지침을 가진다. STIG는 전체 보안을 개선하기 위해 네트워크, 서버, 컴퓨터 및 논리적 설계 내에 보안 프로토콜을 표준화하는 사이버보안 요구사항 지침이다. 이런 지침이 구현되면, 취약성을 줄이기 위해 소프트웨어, 하드웨어, 물리적/논리적 아키텍처의 보안을 개선할 수 있다.

일반적으로 기관에서 보안 권고사항을 수립할 때 ISO 27001(ISO, 2013)과 NIST SP 800-53(NIST, 2017)을 참조하여 보안 통제를 구현한다. 그러나 유럽을 중심으로 국제적으로 인정받는 ISO 27001은 범용적으로 사용하기 위한 목적을 가지기 때문에 너무 포괄적이고 개념적인 내용만을 제시하기 때문에 현업 실무자가 구체적인 보안 권고사항을 구현하기 힘들다. 그나마 NIST 800-53은 미국의 실용적 사상으로 보안 담당자에게 실제적인 보안 통제를 제시하고는 있지만, 보안 카테고리(즉, 접근 통제, 인지 및 훈련 및 물리적 보안 등) 별로 보안 통제 대책을

제시하기 때문에 보안 권고사항이 필요하다고 식별된 보안 대상(예, 애플리케이션, 데이터베이스 및 네트워크 등)에 관한 사항으로 분류되지 않아 실무자가 어려움을 느낀다.

이런 ISO 27001과 NIST SP 800-53의 단점을 극복하기 위해, 미국방 STIG는 구체적으로 네트워크-기반 공격을 최소화 하고 공격자가 네트워크나 기계 자체에서 시스템 접근을 방지하는 방법을 제공한다. 또한 STIG는 소프트웨어 업데이트나 취약성 패치 같은 유지보수 프로세스도 제시하며 애플리케이션, 네트워크, 라우터, 방화벽, DNS 및 스위치 등의 보안 요구사항과 대책을 제시한다. STIG는 NIST SP 800-53(NIST, 2014)의 보안 권고사항에 따라 작성되었다.

예를 들어, STIG는 다음과 같은 보안 요구사항으로 점검 대책을 제시한다:

- 요구사항 ID: V-214048
- 제목: 취약성 평가대로 조직-정의 함수, 포트, 프로토콜 및 서비스를 금지 및 제한해야 한다.
- 점검 사항: DBA 권한으로 다음과 같은 SQL을 실행한다:
- $ sudo su - postgres
- $ psql -c “SHOW port”
- 만일 현재 정의된 포트 구성이 금지된 것이라면, 지적 사항이다.
- 보안 권고사항: 다음 명령은 PGDATA/PGVER 환경 변수를 사용한다. 데이터베이스의 리스닝 포트를 변경하려면, DBA 권한으로 postgresql.conf 파일의 설정을 다음과 같이 변경한다:

  $ sudo su - postgres
  $ vi $PGDATA/postgresql.conf
  원하는 포트로 port 파라미터를 변경하고, 데이터베이스를 재시작한다:
  # SYSTEMD 서버
  $ sudo systemctl restart postgresql-${PGVER?}
  # INITD 서버
  $ sudo service postgresql-${PGVER?} restart
  참고) psql은 디폴트로 port 5432를 사용한다. psql로 port를 명시하거나 PGPORT 환경 변수를 설정하여 변경될 수 있다:
  $ psql -p 5432 -c “"SHOW port”
  $ export PGPORT=5432

위와 같은 사항으로 STIG는 <표 1>과 같이 여러 가지 데이터베이스(Oracle, SQLServer, PostgreSQL 등)에 관한 보안 요구사항을 제시하여 보안을 점검하고 조치할 것을 지시한다.

현재 국내 병원의 대부분 CDM DB는 2017년 산업통상자원부의 “선행 공통데이터모델 분산형 바이오헬스 통합 데이터망 구축 기술개발” 사업으로 선정된 기업에서 3년간 39개 병원(상급종합병원 및 종합병원)에 CentOS 운영체제를 기반한 PostgreSQL로 CDM DB를 구현한 것으로 조사되어 본 연구에서는 구현 환경으로 CentOS 운영체제 기반 PostgreSQL CDM DB를 선정하였다.

〈표 1〉 STIG PostgreSQL 보안 요구사항 예제

| ID | 요구사항 |
|---|---|
| V-214048 | 취약성 평가대로 조직-정의 함수/포트/프로토콜/서비스를 금지/제한해야 한다. |
| V-214049 | 이벤트 결과(성공/실패)를 판단하기 충분한 정보를 가진 감사 레코드를 생성해야 한다. |
| V-214050 | DB의 보안-관련 소프트웨어 업데이트는 인가자/공급자가 명시한 기간 내에 설치되어야 한다. |
| V-214051 | DB가 생성한 감사 정보는 인가되지 않은 변경으로부터 보호되어야 한다. |
| V-214052 | 모든 사용자/그룹/역할/원칙에 관하여 계정 관리와 자동화를 제공하는 조직-수준 인증/접근 메커니즘에 통합되어야 한다. |
| V-214053 | 비-특권 사용자에게 공격자가 악용할 수 있는 정보를 노출하지 않고 교정 활동에 필요한 정보를 제공하는 에러 메시지를 제공해야 한다. |
| V-214054 | DB 소프트웨어 모듈을 변경하는 권한은 제한되어야 한다. |
| … | |

# 4. OMOP-CDM 보안 점검 툴

일반적으로 데이터베이스 보안 점검은 요구사항을 수동 명령 방식으로 SQL을 실행하여 점검하거나 기본적으로 제공하는 데이터베이스 보고서를 생성하거나 여러 가지 평가 방법(Kaddoura and Haraty)으로 수동/자동으로 확인하여 수행된다.

<표 2>는 STIG PostgrSQL 데이터베이스 보안 요구사항의 점검 방법을 요약한 목록이다.

〈표 2〉 PostgreSQL 요구사항 점검 예제

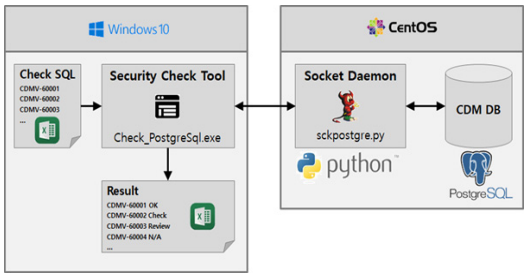
| ID | 점검 사항 |
|---|---|
| V-214048 | As the database administrator, run the following SQL:<br>$ psql -c "SHOW port"<br>If the currently defined port configuration is deemed prohibited, this is a finding |
| V-214049 | If audit records exist without the outcome of the event that occurred, this is a finding. |
| V-214050 | If PostgreSQL is not at the latest version and the evaluated version has CVEs (IAVAs), then this is a finding. |
| V-214051 | If the permissions are not 0600, this is a finding. |
| V-214052 | If there are any records with a different auth-method than gss, sspi, or ldap, that are not documented and approved, this is a finding. |
| V-214053 | If client_min_messages is not set to error, this is a finding. |
| V-214054 | If any files are not owned by root or have permissions allowing others to modify (write) configuration files, this is a finding. |
| … | |

보안 점검 사항 개수가 소량일 경우에는 문제가 없을 수 있으나, 109개인 PostgreSQL OMOP-CDM 보안 점검 요구사항을 수동으로 하나씩 점검하기에는 시간과 노력이 많이 소모되기 때문에 가능하다면 자동으로 점검을 수행하는 것이 점검 담당자에게 편리성을 제공할 수 있다. 그러나 어떤 경우(자동화 불가능)에는 수동으로만 가능한 사례도 있을 수 있다.

본 연구에서는 간편 일괄 점검 방식을 제시하여 데이터베이스나 사용자 환경에 따라 손쉽게 보안점검을 수행할 수 있는 모델을 제시한다.

## 4.1 보안 점검 툴 개념도

본 연구에서는 서버에 직접적으로 보안 점검 툴을 설치하기 보다는 최소의 통신 프로그램인 소켓 데몬을 설치하고 Windows 10 환경에서 점검 툴을 실행하는 것이 안전하다고 판단되어(안전 진단되어야 하는 서버에 설치되는 프로그램의 최소화) 그림 5와 같은 개념 모델을 구성하였다.



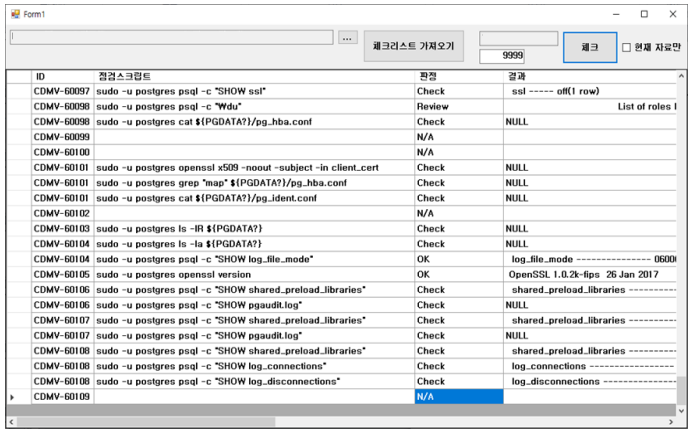

[그림 5] 평가 툴 개념도

보안 점검 툴(Check_PostgreSql.exe)은 점검사항(엑셀 형식)을 입력받아 각 스크립트를 CentOS에 실행되는 소켓 데몬(sckpostgre.py)으로 전송하고 소켓 데몬은 스크립트를 CDM DB에 실행하여 결과(엑셀 형식)를 출력하고 분석하는 방식을 구현하였다.

## 4.2 점검 애플리케이션 구현

점검 애플리케이션은 CDM 데이터베이스에서 리턴된 점검 결과에 따라 다음과 같이 판정한다:

- OK: 명확하게 정상적인 결과
- Check: 명확하게 비정상적인 경우
- Review: 비정상적일 수 있거나 여러 스크립트 실행이나 스크립트 결과가 많아 검토가 필요한 경우
- N/A: 스크립트로 자동 점검이 불가한 항목




[그림 6] 평가 툴 실행 화면

〈표 3〉 점검사항 파일 예제

| ID | 점검 스크립트 | 기준 |
|---|---|---|
| CDMV-60001 (V-214048) | sudo -u postgres psql -c "SHOW port" | If rtn="5432" then "OK" else "Check" |
| CDMV-60002 (V-214049) | sudo -u postgres psql -d cdm -c "CREATE Table stig_test(id INTEGER NOT NULL)" | If rtn="NULL" then "OK" else "Check" |
| | sudo -u postgres psql -d cdm -c "INSERT INTO stig_test(id) VALUES (0)" | If rtn="INSERT 0 1" then "OK" else "Check" |
| | sudo -u postgres psql -c "ALTER TABLE stig_test ADD COLUMN name text" | If rtn="ALTER TABLE" then "OK" else "Check" |
| | sudo -u postgres psql -c "UPDATE stig_test SET id = 1 WHERE id = 0" | If rtn="UPDATE 0" then "OK" else "Check" |
| | sudo -u nopostgres psql -c "INSERT INTO stig_test(id) VALUES (1)" | If rtn="NULL" then "OK" else "Check" |
| | sudo -u nopostgres psql -c "ALTER TABLE stig_test DROP COLUMN name" | If rtn="NULL" then "OK" else "Check" |
| | sudo -u nopostgres psql -c "UPDATE stig_test SET id = 0 WHERE id = 1" | If rtn="NULL" then "OK" else "Check" |
| | sudo -u postgres psql -c "DROP TABLE stig_test" | If rtn="DROP TABLE" then "OK" else "Check" |
| | sudo-upostgrescat${PGDATA?}/pg_log/postgresql-Sun.log | If rtn="NULL" then "Check" else "Review" |
| CDMV-60003 (V-214050) | sudo -u postgres psql -c "SELECT VERSION()" | If rtn="NULL" then "Check" else "Review"" |
| | sudo -u postgres rpm -qa \| grep postgres | If rtn="NULL" then "Check" else "Review" |
| | sudo -u postgres apt-cache policy postgres | If rtn="NULL" then "Check" else "Review" |
| CDMV-60004 (V-214051) | sudo-upostgresgrep"log_file_mode"${PGDATA?}/postgresql.conf | If rtn ="0600" then "OK" else "Check" |
| | sudo-upostgresgrep"log_directory"${PGDATA?}/postgresql.conf | If rtn = "NULL" then "Check" else "Review" |
| | sudo-upostgresls-la${PGDATA?}/pg_log | If rtn = "NULL" then "Check" else "Review" |
| CDMV-60005 (V-214052) | sudo -u postgres cat ${PGDATA?}/pg_hba.conf | If rtn = "NULL" then "Check" else "Review" |
| CDMV-60006 (V-214053) | sudo -u postgres psql -c "SELECT current_setting('client_min_messages')" | If rtn = "error" then "OK" else "Check" |
| ... | | |

[그림 6]은 평가 툴을 실행한 화면으로 점검 스크립트와 그 결과에 따른 판정 상황을 보여준다.

<표 3>은 STIG가 제시한 점검 사항을, SQL (Standard Query Language)로 점검하는 스크립트와 기준을 분석하고 설정하여 평가 툴에 적용한 예제이다.

대부분 SQL 리턴은 정확할 경우 명확한 응답을 주지만 오류가 있을 경우에는 무응답인 경우가 많아 각 요구사항의 SQL을 분석하여 기준을 설정하였다.

# 5. 보안 검증 및 보안 권고사항

## 5.1 테스트 환경

테스트는 상급종합병원 기준인 500 병상 이상을 충족하는 3개 병원의 CDM 데이터베이스 서버를 대상으로 동일한 클라이언트에서 구현하였다.

<표 4>는 테스트 된 클라이언트와 3개 병원의 데이터베이스 서버의 구성 환경을 요약하였다.

〈표 4〉 테스트 환경

| | 클라이언트 | 병원 A 서버 | 병원 B 서버 | 병원 C 서버 |
|---|---|---|---|---|
| 병상 | | 884 병상 | 905 변상 | 608 병상 |
| 서버 모델 | LG 17ZB995-GP75ML | HP DL360G10 | HP DL360G10 | HP DL360G10 |
| 운영 체제 | Windows 10 Pro | CentOS 7.6 | CentOS 7.6 | CentOS 7.6 |
| CPU | Intel Core i7-10510U 1.80GHz | Xeon 2.3GHz 12 core | Xeon 2.3GHz 12 core | Xeon 2.3GHz 12 core |
| 메모리 | 40GB | 128 GB | 128 GB | 128 GB |
| 하드 디스크 | SSD 510GB | SSD 240GB | SSD 240GB | SSD 240GB |
| 데이터베이스 | | PostgreSQL v13.0 | PostgreSQL v13.0 | PostgreSQL v13.0 |

## 5.2 테스트 결과

A 병원, B 병원 및 C 병원의 CDM 서버는 국가공공사업(39개 병원 대상)으로 지정된 동일한 업체에서 설치 및 유지관리하고 기능 추가가 없기 때문에, 점검한 결과 모두 동일한 결과가 나왔고 나머지 36개 병원도 동일한 결과가 예상된다.

점검 결과 항목을 살펴보면 비정상적인 항목이 76%로 다수를 차지하고 항목별 결과를 유형 분류하면 다음과 같다:

- 정상적인 결과를 가진 항목(OK)은 총 109개 항목 중 8개(7%)
- 비정상적인 결과를 가진 항목(Review: 21, Check: 62)은 총 109개 항목 중 83개(76%)
- 자동 스크립트로 점검 불가한 항목(N/A)은 총 109개 항목 중 18개(17%)

<표 5>는 테스트 결과를 병원과 판정 결과별로 요약한 표이다.

〈표 5〉 테스트 결과

| 기준 | 병원 A CDM 서버 | | 병원 B CDM 서버 | | 병원 C CDM 서버 | |
|---|---|---|---|---|---|---|
| | 개수 | % | 개수 | % | 개수 | % |
| OK | 8/109 | 7% | 8/109 | 7% | 8/109 | 7% |
| Review | 21/109 | 19% | 21/109 | 19% | 21/109 | 19% |
| Check | 62/109 | 57% | 62/109 | 57% | 62/109 | 57% |
| N/A | 18/109 | 17% | 18/109 | 17% | 18/109 | 17% |

## 5.3 문제점 분석 및 보안 권고사항

3개 병원의 테스트 결과로 문제점을 분석하여, 취약점을 가진 CDM 데이터베이스를 안전하게 운영하기 위해 다음과 같은 보안 권고사항을 제시한다.

STIG는 제시한 요구사항의 해결 방법을 기본적으로 제시하지만, 별도 문서로 권고사항도 제시한다.

예를 들어, STIG PostgreSQL 요구사항 V-214049인 경우,

- 요구사항: 이벤트 결과(성공/실패)를 판단하기 충분한 정보를 가진 감사 레코드를 생성해야 한다.
- 권고사항: pgaudit PostgreSQL을 사용하면, DB의 다양한 양상을 감사하도록 구성될 수 있다. 로깅이 활성화되면, 모든 에러, 거부 및 요청 실패가 로깅된다.

다음 명령은 PGDATA/PGVER 환경 변수를 사용한다.

pgaudit과 로깅이 활성화되면, postgresql.conf 파일에 다음과 같은 구성을 설정한다. DBA 권한으로 다음과 같이 실행한다:

```
$ sudo su - postgres
$ vi ${PGDATA?}/postgresql.conf
pgaudit.log_catalog='on'
pgaudit.log_level='log'
pgaudit.log_parameter='on'
pgaudit.log_statement_once='off'
pgaudit.log='all, -misc'
```

다음으로 postgresql.conf의 로깅 구성을 조정한다.

```
$ sudo su - postgres
$ vi ${PGDATA?}/postgresql.conf
log_line_prefix = '< %m %u %d %e: >'
log_error_verbosity = default
```

마지막으로 시스템 관리자 권한으로 PostgreSQL을 재시작한다:

```
# SYSTEMD SERVER ONLY
$ sudo systemctl reload postgresql-${PGVER?}
# INITD 서버
$ sudo service postgresql-${PGVER?} reload
```

각 점검 규칙에 위반된 항목에 권고사항을 적용하면 PostgreSQL CDM 데이터베이스를 안전하게 운영할 수 있다.

<표 6>은 각 요구사항 별 권고사항을 정리한 요약 표이다.

<표 6>에서 권고하는 조치 사항을 보안 권고사항 별(pgaudit, openSSL, pgcrypto, RLS 및 기타)로 유형 분류하고, NIST SP 800-60(NIST, 2004)이 제시하는 각 기밀성/무결성/가용성 분류를 근거로, 보안 권고사항으로 강화될 수 있는 CIA(Confidentiality, Integrity and Availability)를 분석하면 <표 7>과 같다.

각 병원이 자체에서 연구 개발하여 CDM을 운영하는 것이 아니라 국가 공공사업으로 지정된 한 업체에서 유지관리하고 있기 때문에, 추가적인 보안 활동이 미비하여 많은 취약점을 가지고 있다고 분석된다.

〈표 6〉 보안 권고사항 예제

| ID | 점검 규칙 | 권고 사항 |
|---|---|---|
| CDMV-60001 (V-214048) | 취약성 평가대로 조직-정의 함수/포트/프로토콜/서비스를 금지/제한해야 한다. | “Port 5432”로 설정. |
| CDMV-60002 (V-214049) | 이벤트 결과(성공/실패)를 판단하기 충분한 정보를 가진 감사 레코드를 생성해야 한다. | 감사 로깅 활성화를 위해 추가 툴 “pgaudit” 설치. |
| CDMV-60003 (V-214050) | DB의 보안-관련 소프트웨어 업데이트는 인가자/공급자가 명시한 기간 내에 설치되어야 한다. | 패치 설치. |
| CDMV-60004 (V-214051) | DB가 생성한 감사 정보는 인가되지 않은 변경으로부터 보호되어야 한다. | 로그 파일 보호, “log_file_mode= 0600” 설정. |
| CDMV-60005 (V-214052) | 모든 사용자/그룹/역할/원칙에 관하여 계정 관리와 자동화를 제공하는 조직-수준 인증/접근 메커니즘에 통합되어야 한다. | “${PGDATA?}/pg_hba.conf” 파일 수정. |
| CDMV-60006 (V-214053) | 비-특권 사용자에게 공격자가 악용할 수 있는 정보를 노출하지 않고 교정 활동에 필요한 정보를 제공하는 에러 메시지를 제공해야 한다. | “client_min_messages= error” 설정. |
| CDMV-60007 (V-214054) | DB 소프트웨어 모듈을 변경하는 권한은 제한되어야 한다. | DBA 권한으로, PGDATA 구성 파일의 허가와 소유권을 변경, “$ chmod0600 ${PGDATA?}/postgresql.conf”. |
| ... | | |

〈표 7〉 보안 권고사항별 CIA 유형 분류

| | pgaudit | openSSL | pgcrypto | RLS | 기타 |
|---|---|---|---|---|---|
| 요구사항 | 44/109 | 7/109 | 4/109 | 3/109 | 51/109 |
| CIA 영향 | 기밀성<br>무결성<br>가용성 | 기밀성 | 기밀성 | 기밀성 | 기밀성<br>무결성<br>가용성 |
| 개선 | 40% | 6% | 4% | 3% | 47% |

보안의 첫 단계는 무엇을 보호할 것인지 결정하는 자산 식별부터 시작된다. 현재 미국에서 새로 발간되거나 개정되는 모든 보안 지침은 식별, 보호, 탐지, 대응 및 복구의 5 단계를 가진 NIST CSF(Cyber Security Framework)(NIST, 2018)를 준수하도록 법으로 규정하고 있다.

현재 각 병원에서 실제 환자를 다루지 않는 연구 데이터인 CDM을 중요한 자산으로 식별하지 않고 보호 대상으로 고려하지 않아 이런 문제가 발생한다고 분석된다.

데이터베이스 감사 로그나 데이터 암호화에 관한 취약점을 가지게 되면 악성 공격자의 최종 목표인 데이터 유출이나 데이터베이스 손상이 쉬워지기 때문에 이런 취약점을 제거하는 보안 권고사항을 강구해야 한다.

## 6. 결 론

본 연구에서 CDM 데이터베이스를 점검한 결과 OMOP 사이트에서 제공해주는 데이터베이스 템플릿 스크립트로 데이터베이스를 구성한 이후에, 각 병원에서 CDM 데이터베이스를 보안 자산으로 식별하지 않아 추가 보안 권고사항을 마련하지 않았기 때문에 대부분 CDM 데이터베이스에서 많은 취약점을 가지고 있다.

COVID-19 질병이 정점을 달하고 있고 어느 정도 안정되리라 예상되지만, 향후에 치료제를 연구하거나 다른 유사한 질병을 빠르게 대처하기 위해 지금보다 더 많은 코호트 연구 자료가 공유 교류되기 때문에 CDM 데이터베이스는 중요 자산으로 식별되어야 한다.

CDM 데이터베이스의 보안 권고사항을 강구하기 위해 pgaudit 같은 추가 유틸리티를 설치하여 구성하고, 강력한 보안 체계를 요구하는 미국방 STIG 데이터베이스 요구사항을 충족하면 기밀성, 무결성 및 가용성 측면에서 CDM 데이터베이스 보안 태세를 최소 40%(109개 항목 중 44개 해당) 이상 향상시킬 수 있는 것으로 분석되었다(<표 7> 참조).

- 감사 로깅을 활성화하기 위해 pgaudit을 설치하고 환경에 맞게 관련 파라미터를 설정(기밀성, 무결성 및 가용성 40% 향상).
- 안전한 통신을 위한 openSSL을 설치하고 구성(기밀성 6% 향상).
- 데이터 암호화를 위한 pgcrypto를 설치하고 환경에 맞게 관련 파라미터 설정(기밀성 4% 향상).
- 로우-수준 보안을 강화하기 위해 RLS(Row-Level Security) 적용(기밀성 3% 향상).
- 구성 파일 같은 데이터베이스 환경 설정 변경(기밀성, 무결성 및 가용성 47% 향상)

CDM 의료 연구 데이터는 가명화나 익명화로 개인정보를 보호하지만(김강한, 2021; Hammond et al., 2012), CDM 데이터베이스의 연구 데이터가 조작되어 교류된다면, 특히 데이터 무결성이 훼손된다면 의학 연구 발전에 심각한 문제가 발생하리라 본다.

본 연구에서 개발된 애플리케이션 및 분석 결과는 한국보건산업진흥원의 “CDM 기반 분산 연구에 필요한 보안 프레임워크 개발“과제에서 응용되었고 실증 평가하기 위해 한국기계전기전자시험연구소의 규격에 따른 시험을 받아 통과(성적서 번호 SW2021-00101)하였으며, 확산 보급을 위해 GitHub 웹사이트(https://github.com/khlee830/khlee830/Check_PostgreSql.zip)에 업로드하였다.

개인정보보호에 대한 보안 점검은 시간과 비용이 많이 소요되고 개별 병원의 환경이 다르고, 개별적 보안 사항이라 자동으로 수행하기는 힘든 점이 아쉽지만 본 연구의 결과를 바탕으로 CDM 데이터베이스 구성의 보안 태세만이라도 현재 보다 더 강화되

도록 각 병원에서 DoD STIG을 근거한 본 논문의 보안 권고사항을 강구해야 한다.

## 참고문헌


김강한, “가명화 개인건강정보 보호 관련 기본권 보장에 관한 연구”, *세계헌법연구*, 제27권, 제2호, 2021, 27-73.

윤현아, “공통 데이터 모델을 이용한 성인 조현병 환자의 항정신병 약물 처방 패턴 분석”, *한국보건사회약료경영학회지*, 제9권, 제2호, 2021, 111-120.

DISA, “PostgreSQL 9.X Security Technical Implementation Guide (STIG) Overview Version 2, Release 2”, 2022, Available at: https://www.stigviewer.com/stig/postgresql_9.x/.

DoD CIO, “Department of Defense Net-Centric Data Strategy”, 2007, Available at: https://dodcio.defense.gov/Portals/0/documents/DoD_NetCentricServicesStrategy.pdf.

Hammond, K. W., Efthimiadis, E. N., and Laundry, R. J., “Efficient De-identification of Electronic Patient Records for User Cognitive Testing”, IEEE, 2012, Available at: https://ieeexplore.ieee.org/document/6149163.

ISO, “ISO/IEC 27001:2013 Information technology - Security techniques - Information security management systems - Requirements”, 2013, Available at: https://www.iso.org/standard/54534.html.

Kaddoura, S. and Haraty, R. A., “A Parallelized Database Damage Assessment Approach after Cyberattack for Healthcare Systems”, 2021, Available at: https://www.researchgate.net/publication/350540292_A_Parallelized_Database_Damage_Assessment_Approach_after_Cyberattack_for_Healthcare_Systems.

Kan, M., “A ransomware attack is spreading worldwide, using alleged NSA exploit: UK's National Health Service was among the organizations hit by the Wanna Decryptor ransomware on Friday”, 2017, Available at: https://www.computerworld.com/article/3196378/a-ransomware-attack-is-spreading-worldwide-using-alleged-nsa-exploit.html.

NIST, “Framework for Improving Critical Infrastructure Cybersecurity Version 1.1”, 2018, Available at: https://nvlpubs.nist.gov/nistpubs/cswp/nist.cswp.04162018.pdf.

NIST, “SP 800-53r4 Security and Privacy Controls for Federal Information Systems and Organizations”, 2014, Available at: https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/archive/2015-01-22.

NIST, “SP 800-60r1 Guide for Mapping Types of Information and Information Systems to Security Categories”, 2004, Available at: https://csrc.nist.gov/publications/detail/sp/800-60/vol-1-rev-1/final.

OHDSI, “The Book of OHDSI”, 2021, Available at: https://ohdsi.github.io/TheBookOfOhdsi/.

Wilson, R. “Emerging ransomeware threats: An anticipatory ethical anaylsis”, 2021, Available at: https://ieeexplore.ieee.org/document/9629211.

## About the Authors

**이 경 환 (khlee@onthelive.kr)**

현재 ㈜온더라이브에서 CTO로 근무 중이고, KISIA에서 융합보안(의료)를 강의하고 있습니다. 연세대학교 공대에서 건축공학을 전공하였고 대학원에서는 컴퓨터공학을 전공했습니다. 최근에 서울과학기술대학교 산업정보시스템 전공으로 공학박사 학위를 취득했습니다. 주요 연구분야는 사이버보안통제와 융합보안(의료/제조/에너지)입니다.

**장 성 용 (syjang@seoultech.ac.kr)**

현재 서울과학기술대 산업정보시스템공학과 교수로 재직 중이고, 서울대학교 산업공학전공으로 학사, 석사 및 박사를 취득했습니다. 주요 연구분야는 전자상거래, 시뮬레이션, TOC 제약이론입니다.