Software Requirements Specification

1. Introduction (소개)

1.1. Purpose (목적)

본 문서의 목적은 DB 침입 탐지 시스템(Database Intrusion Detection System, DB-IDS)에 대한 요구사항을 명확히 정의하여, 설계(SDS) 및 구현 단계에서 참조할 수 있는 기준을 제시하는 것이다.

DB-IDS는 데이터베이스에 대한 비정상적인 접근 시도를 탐지하고 관리자에게 알림을 제공하여, 데이터베이스의 보안을 강화하는 것을 목표로 한다.

1.2. Scope (범위)

- 시스템 이름: DB 침입 탐지 시스템 (DB-IDS)
- 시스템 기능
  - 모든 SQL 쿼리를 로깅한다.
  - SQL Injection, DROP TABLE, UNION SELECT 등 패턴 기반 공격을 탐지한다.
  - 평소와 다른 쿼리 빈도, 행 반환 수, 쿼리 실행 시간 등을 이용한 행동 기반 이상 탐지를 수행한다.
  - 비인가 계정의 시스템 테이블 접근, DELETE/INSERT 시도를 권한 기반 탐지한다.
  - 탐지 이벤트 발생 시 관리자에게 실시간 알림을 전송한다.
  - 관리자는 웹 기반 대시보드를 통해 로그 및 탐지 이벤트를 조회할 수 있다.
- 시스템 범위
  - DB 서버 앞 단의 프록시 형태로 동작한다.
  - 관리자는 웹 UI를 통해 시스템을 제어하고 결과를 확인한다.
- 시스템 장점

    - 데이터베이스 보안 사고를 조기 탐지하여 데이터 손실 및 피해를 최소화할 수 있다.
    - 기존 DB 로그만으로는 파악하기 어려운 이상 행위 패턴을 시각적으로 제공한다.
    - 관리자가 실시간으로 보안 위협을 인지할 수 있어 운영 효율성을 높인다.
- 시스템 목적
    - 데이터베이스에 대한 침입 시도를 신속하게 탐지하고, 관리자에게 알림을 제공함으로써 정보 자산을 보호한다.
    - 개발 및 운영 환경에서 보안성을 강화하고, 내·외부 보안 감사에 활용할 수 있는 근거 데이터를 제공한다.

1.3. Definitions, acronyms and abbreviations (용어 및 약어 정의)

- IDS (Intrusion Detection System): 네트워크나 시스템에서 발생하는 이벤트를 모니터링하여 비인가된 접근이나 이상 행위를 탐지하는 시스템.
- DB-IDS (Database Intrusion Detection System): 데이터베이스 쿼리와 접근 패턴을 감시하여 침입 시도를 탐지하는 특화된 IDS.
- SQL (Structured Query Language): 관계형 데이터베이스 관리 시스템(RDBMS)에서 데이터 질의와 조작에 사용되는 표준 언어.
- SQL Injection: 사용자가 입력한 데이터를 검증하지 않고 SQL 문에 직접 포함시켜 실행하는 보안 취약점을 악용한 공격 기법.
- Proxy (프록시): 클라이언트와 서버 사이에서 통신을 중계하는 중간 계층. DB-IDS는 주로 프록시 기반으로 동작하여 쿼리를 가로채고 분석함.
- Pattern-based Detection (패턴 기반 탐지): 미리 정의된 공격 시그니처나 금지된 SQL 구문을 기반으로 탐지하는 기법.
- Behavior-based Detection (행동 기반 탐지): 정상 동작 프로파일과 비교하여 비정상적 행동(쿼리 빈도 급증, 데이터 반환량 폭증 등)을 탐지하는 기법.
- Authorization-based Detection (권한 기반 탐지): 특정 사용자나 계정이 데이터베이스 자원에 접근할 수 있도록 허용하는 과정.
- Alert (알림): 보안 이벤트가 발생했음을 관리자에게 통지하는 메시지. 이메일, Slack으로 전달 가능.

- Dashboard (대시보드): 관리자에게 탐지 이벤트, 로그, 통계 정보를 시각화하여 제공하는 UI.
- RDBMS (Relational Database Management System): 관계형 데이터 모델을 기반으로 데이터를 저장·관리하는 시스템. MySQL, PostgreSQL, Oracle 등이 포함됨.
- Admin (관리자): DB 침입을 탐지하기 위해 시스템을 사용하는 실 사용자.
- False Positive (오탐): 정상적인 쿼리를 침입으로 잘못 탐지하는 경우.
- False Negative (미탐): 실제 침입 시도를 탐지하지 못하는 경우.

1.4. References (참고자료)

- IEEE Std 830-1998, *IEEE Recommended Practice for Software Requirements Specifications*
- Halmstad University (2022), *Anomaly Detection in SQL Databases*
- GreenSQL (오픈소스 DB 보안 프록시)
- OWASP SQL Injection Cheat Sheet, OWASP Foundation
- ISO/IEC 27001:2013, *Information Security Management*

1.5. Overview (개요)

이 문서의 나머지 부분에서는 DB-IDS의 기능 요구사항, 비기능 요구사항, 시스템 인터페이스, 사용자 특성, 제약사항 등을 상세히 설명한다.

이를 통해 개발자는 설계 단계에서 필요한 기준을 확보할 수 있고, 테스트 단계에서는 요구사항 충족 여부를 검증할 수 있다.

2. Overall Description (전반적인 기술사항)

본 장에서는 데이터베이스 침입 탐지 시스템(DB-IDS)의 일반적인 특성과 외부 환경, 그리고 제품 요구사항에 영향을 미치는 요인들을 설명한다. 이 장은 상세 요구사항을 직접 기술하지 않고, 3장에서 다룰 구체적 요구사항에 대한 배경을 제공한다.

2.1. Product Perspective (제품 관점)

DB-IDS는 독립적으로 동작할 수 있으나, 실제로는 데이터베이스 관리 시스템(RDBMS) 앞단에 위치한 프록시 또는 미들웨어로서 동작한다.

2.1.1. 시스템 인터페이스 (System Interfaces)

- 데이터베이스 서버: DB-IDS는 DB 서버와 직접 연결되어 모든 SQL 요청을 감시한다.
- 관리자 대시보드: 관리자는 웹 UI를 통해 로그, 탐지 이벤트를 확인한다.
- 알림 채널: 이메일, Slack을 통해 탐지 이벤트를 전달한다.

2.1.2. 사용자 인터페이스 (User Interfaces)

- 웹 기반 대시보드(UI)
  - 로그인 화면 (관리자 인증)
  - 실시간 쿼리 로그 모니터링 화면
  - 탐지 이벤트 목록/상세 화면
  - 사용자별/시간대별 통계 그래프
  - 알림 설정 메뉴 (이메일, Slack 등)

2.1.3. 하드웨어 인터페이스 (Hardware Interfaces)

- 지원 하드웨어:
  - x86 기반 서버 및 클라우드 VM (AWS EC2 t3.medium)
  - 최소 2 Core CPU, 4GB RAM
- 지원 장비:
  - 네트워크 인터페이스 카드(NIC) 1Gbps 이상
  - 스토리지 SSD 20GB 이상 (로그 저장용)

2.1.4. 소프트웨어 인터페이스 (Software Interfaces)

- 운영체제: Linux 기반(Ubuntu 20.04+)
- 시각화 및 대시보드: React 19+
- 백엔드(IDS): SpringBoot 3.5+
- 내부 데이터베이스: SQLite 3.35+

- 대상 데이터베이스: MySQL 8.0+
- 오브젝트 스토리지: Amazon S3
- API 연동: Slack Webhook API, SMTP 서버
- 사용 목적: 데이터 수집, 분석, 알림 전송, 관리자 UI 제공

2.1.5. 통신 인터페이스 (Communications Interfaces)

- DB 통신: TCP/IP, 포트 3306(MySQL)
- 웹 대시보드: HTTPS (TLS 1.2 이상)
- 알림 연동: HTTPS REST API (Slack), SMTP (메일 서버)
- 내부 통신: 로그 수집기 → 탐지 엔진 → 저장소 → 대시보드 간 gRPC/REST 통신
- 로그 백업/아키이빙 통신: HTTPS

2.1.6. 메모리 제약사항 (Memory Constraints)

- 최소 4GB RAM 필요
- DB 로그 저장소는 일일 평균 1GB 이상 필요
- 탐지 엔진은 쿼리당 50MB 이하 메모리를 사용해야 함 (성능 보장 목적)

2.1.7. 운영 (Operations)

- 운영 모드: 쿼리를 실시간 분석 및 조회 가능하고, 관리자에게 즉시 알림
- 백업 방안: 로그 및 탐지 이벤트 데이터는 매일 S3로 증분 백업하며, 버전닝을 활성화한다. 수명주기 정책으로 30일 이후 비용 효율적인 스토리지 클래스로 자동 이전한다.

2.1.8. 사이트 적용 요건 (Site Adaptation Requirements)

- 설치 환경:
  - 클라우드 환경: VM/Container로 배포
- 초기화 요구사항:
  - 모니터링 대상 DB 연결 정보 입력
  - 관리자 계정 등록
  - 알림 채널 설정

    - S3 버킷명/리전, 라이프사이클 정책, 버전닝 옵션 설정
    - IAM 역할/정책 연결

2.2. Product functions (제품 기능)

DB-IDS가 제공해야 할 주요 기능은 다음과 같다. 이는 설계를 설명하기 위함이 아니라, 제품이 수행할 기능의 논리적 관계를 상위 수준에서 요약한 것이다.

- 쿼리 로깅 기능
    - 모든 SQL 쿼리(SELECT, INSERT, UPDATE, DELETE 등)를 기록한다.
    - 사용자 ID, 실행 시간, 반환 행 수 등의 메타데이터를 포함한다.
- 패턴 기반 탐지 기능
    - SQL Injection, DROP TABLE, UNION SELECT 등 사전에 정의된 금지 패턴을 탐지한다.
- 행동 기반 탐지 기능
    - 정상 패턴과 비교하여 쿼리 빈도 급증, 반환 행 수 폭증 등 이상 행위를 탐지한다.
- 권한 기반 탐지 기능
    - 권한이 없는 계정이 민감한 테이블에 접근하거나 삭제/삽입 명령을 수행하려 할 경우 탐지한다.
- 알림 및 차단 기능
    - 비정상 쿼리가 탐지되면 관리자에게 실시간으로 알림(이메일, Slack 등)을 전송한다.
    - 필요 시 해당 세션을 강제 종료하거나 차단할 수 있다.
- 관리자 대시보드 기능
    - 탐지 이벤트 및 로그를 시각화하여 제공한다.
    - 사용자별/시간대별 통계 차트를 제공한다.

2.3. User Characteristics (사용자 특성)

DB-IDS의 주요 사용자는 다음과 같은 특성을 가진다.

- DB 관리자 (Primary User)

    - 교육 수준: 대학 학사 이상의 IT 관련 전공자 또는 이에 준하는 지식 보유자
    - 경험: DBMS 운영 경험 1년 이상
    - 기술적 전문 지식: SQL 기본 지식, 서버 운영 경험, 보안 정책 이해 능력
- 보안 담당자 (Secondary User)
    - 교육 수준: 보안 관련 자격 또는 교육 이수자
    - 경험: 네트워크 및 시스템 보안 관리 경험 1년 이상
    - 기술적 전문 지식: IDS/IPS 기본 지식, 보안 이벤트 분석 능력
- 일반 개발자 (Occasional User)
    - 교육 수준: 소프트웨어 개발자
    - 경험: DB 연동 애플리케이션 개발 경험
    - 기술적 전문 지식: SQL 문법 이해, 간단한 로그 분석 능력
    - 사용 목적: 테스트 환경에서 시스템 탐지 결과 검증

2.4. Constraints (제약 사항)

- 규제 정책 (Regulatory policies)
    - 개인정보 보호법 및 ISMS 보안 규정 준수한다.
    - 로그에는 주민번호, 계좌번호 등 민감정보를 직접 저장하지 않는다.
- 하드웨어 제한 (Hardware limitations)
    - 최소 2 Core CPU, 4GB RAM 환경에서 정상 동작해야 한다.
    - DB 서버에 과도한 성능 저하를 유발하지 않아야 한다.
- 소프트웨어 인터페이스 (Software interfaces)
    - Spring Boot 3.5+, React 19+, MySQL 8.0+, Ubuntu 20.04+ 환경에서 호환되어야 한다.
    - Slack(HTTPS REST API), SMTP(메일) 연동이 가능해야 한다.
    - S3 서비스 가용성에 따라 백업 작업이 영향 받을 수 있다.
- 운영 제약 (Operational constraints)

    - 탐지 지연은 1초 이내여야 하며, 장애 발생 시 1시간 이내 복구 가능해야 한다.
    - 시스템은 동시에 다수의 요청을 병렬로 처리할 수 있어야 한다.
- 신뢰성 요구 (Reliability requirements)
    - 오탐(False Positive) ≤ 5%, 미탐(False Negative) ≤ 2%를 목표로 한다.
- 보안 고려 (Security considerations)
    - 탐지 로그는 관리자 외 접근 불가해야 하며, 무결성을 보장해야 한다.

2.5. Assumptions and dependencies (가정 및 의존성)

- 가정
    - 데이터베이스 서버는 정상적으로 동작하며, DB-IDS는 해당 서버와의 네트워크 연결이 가능하다.
    - 관리자는 기본적인 SQL 지식을 가지고 있으며, 대시보드 사용에 어려움이 없다.
    - 탐지 규칙은 사전에 정의된 패턴 및 정책을 기준으로 한다.
- 의존성
    - DB-IDS는 특정 DBMS(MySQL 8.0)에 의존한다.
    - 외부 알림 채널(Slack, SMTP 등)의 가용성에 따라 알림 기능이 영향을 받을 수 있다.
    - 운영 환경(OS, 하드웨어)에 따라 성능이 달라질 수 있다.
    - AWS S3 및 관련 IAM 서비스의 가용성
    - VPC → S3 접근 경로 구성

2.6. Apportioning of Requirements (단계별 요구사항)

아래 요구사항은 향후 버전에서 확장될 수 있으며, 현재 MVP 범위에는 포함되지 않는다.

- 멀티 DBMS 지원: PostgreSQL, Oracle 등 추가 지원 (현재는 MySQL만 지원).
- 자동 차단 기능: 탐지된 공격 세션을 자동으로 종료하거나 방화벽 연동.
- 머신러닝 기반 탐지: 이상 탐지 알고리즘을 AI 모델 기반으로 고도화.
- 고급 리포팅 기능: 시각화 대시보드 외에도 PDF 보고서 자동 생성.
- 다국어 지원: 영어, 일본어 등 다국어 UI 확장.

3. Specific Requirements (구체적 요구사항)

3.1. External Interface Requirements (외부 인터페이스 요구사항)

DB-IDS는 데이터베이스 서버, 관리자, 외부 알림 채널과 상호작용하며, 각 인터페이스에 대한 입력 및 출력, 데이터 형식, 타이밍 요구사항은 다음과 같다.

3.1.1. 사용자 인터페이스 (User Interfaces)

- 로그인 화면
  - 목적: 관리자 인증 수행
  - 입력: 사용자 ID, 비밀번호
  - 출력: 인증 성공 시 대시보드 진입, 실패 시 오류 메시지
  - 형식: 웹 브라우저 HTML5, HTTPS POST 요청
  - 제약: 비밀번호 최소 8자, 대소문자+숫자 포함
  - 타이밍: 응답 2초 이내
- 실시간 모니터링 화면
  - 목적: 현재 실행되는 SQL 쿼리 및 탐지 이벤트 시각화
  - 출력 항목: 사용자 ID, 실행 시간, SQL 요약, 탐지 여부(정상/이상)
  - 형식: 표(Table) + 그래프 시각화
  - 타이밍: 5초 주기 갱신
- 탐지 이벤트 상세 화면
  - 목적: 특정 이벤트 로그 상세 확인
  - 출력 항목: 사용자 ID, 이벤트 ID, 원본 쿼리, 탐지 유형(패턴/행동/권한), 심각도(Level), 발생 시각
  - 형식: JSON 기반 API 호출 결과를 UI에 표시
  - 타이밍: 클릭 시 2초 이내 응답
- 알림 설정 화면

    - 목적: 관리자별 알림 채널 설정
    - 입력: 이메일 주소, Slack Webhook URL
    - 출력: 저장 성공/실패 메시지
    - 형식: 웹 폼(Form)
    - 타이밍: 저장 요청 후 3초 이내 완료

3.1.2. 하드웨어 인터페이스 (Hardware Interfaces)

- 클라우드 컴퓨팅 리소스
    - 목적: DB-IDS가 실행될 가상화된 인스턴스 환경 제공
    - 형식: AWS EC2, GCP Compute Engine, 또는 동등한 VM 서비스
    - 최소 스펙: vCPU 2개, 메모리 4GB, 디스크 20GB SSD
    - 제약: 인스턴스는 모니터링 대상 DB와 동일 VPC/네트워크 내에 위치해야 함
- 네트워크 인터페이스 (가상 NIC)
    - 목적: DB 서버와 IDS 간 SQL 트래픽 송수신
    - 형식: 클라우드 제공 가상 NIC (1Gbps 이상)
    - 제약: 지연시간 < 10ms, VPC 내 보안 그룹/방화벽 규칙에 따라 포트(3306 등)가 열려 있어야 함

3.1.3. 소프트웨어 인터페이스 (Software Interfaces)

- 운영체제 (OS)
    - 이름: Ubuntu
    - 버전: 20.04 LTS 이상
    - 소스: Canonical 배포판
    - 목적: IDS 실행 환경 제공
    - 제약: Docker 환경 호환성 유지
- 내부 데이터베이스
    - 이름: SQLite (경량 RDB)

    - 버전: SQLite 3.35+
    - 소스: 오픈소스 배포판 (SQLite)
    - 목적: IDS 내부 상태 관리
    - 인터페이스 데이터 형식: SQL 텍스트 스트림(UTF-8 인코딩)
    - 프로토콜/포트:
        - SQLite: 파일 기반 접근
    - 제약:
        - 대상 DB(모니터링되는 MySQL)와는 별도로 운영되어야 하며, IDS 내부 관리용으로만 사용된다.
        - 저장 용량은 수 MB~수십 MB 수준을 목표로 하며, 주기적으로 백업/아카이브 가능해야 한다.
- 대상 데이터베이스
    - 이름: MySQL
    - 버전: 8.0 이상
    - 소스: Oracle MySQL Community Edition
    - 목적: 모니터링 대상 DB
    - 인터페이스 데이터 형식: SQL 텍스트 스트림 (UTF-8 인코딩)
    - 프로토콜/포트: TCP/IP 3306
    - 제약: 대상 DB 성능 저하 최소화
- 백엔드(IDS 엔진)
    - 이름: Spring Boot
    - 버전: 3.5+ (Java 21 LTS)
    - 소스: Spring 공식 배포
    - 목적: 탐지 룰 적용(패턴/행동/권한), 이벤트 생성 및 저장, 알림 트리거
    - 데이터 연동:
        - REST API (JSON, UTF-8)

        - SSE(Web API EventSource)
    - 보안: HTTPS(TLS 1.2+) 권장, 관리자 RBAC 및 감사 로그 기록
- 시각화 및 대시보드
    - 이름: React
    - 버전: 19+
    - 소스: Meta/오픈소스
    - 목적: 실시간 모니터링 UI, 탐지 이벤트/통계 시각화
    - 데이터 연동:
        - REST API(JSON, UTF-8)
        - SSE — 실시간 이벤트 테이블 갱신
    - 호스팅 형식: 정적 파일(HTML/CSS/JS)
- 외부 API 연동
    - Slack Webhook
        - 목적: 보안 이벤트 알림 전송
        - 메시지 형식: JSON
        - 프로토콜: HTTPS POST
        - 타이밍: 탐지 후 1초 이내 전송
    - SMTP 서버
        - 목적: 이메일 알림 발송
        - 메시지 형식: MIME (Plain Text/HTML)
        - 프로토콜/포트: TCP 587(TLS)
        - 타이밍: 탐지 후 10초 이내 발송
- 오브젝트 스토리지(Amazon S3)
    - 메시지/오브젝트 형식: GZIP 압축된 JSON/CSV(UTF-8), 메타데이터 태그 포함
    - 타이밍: 일일 배치

- 보안: SSE-S3/SSE-KMS, 버킷 정책 최소 권한, 접근 로그(서버 액세스 로그/CloudTrail)

3.1.4. 통신 인터페이스 (Communications Interfaces)

- DB 서버와의 통신
  - 프로토콜: TCP/IP
  - 포트: 3306 (MySQL)
  - 데이터 형식: SQL 쿼리 문자열 (UTF-8)
  - 요구사항: 탐지 지연 ≤ 1초
- 대시보드와 브라우저 간 통신
  - 프로토콜: HTTPS (TLS 1.2 이상)
  - 포트: 443
  - 데이터 형식: JSON API 응답, HTML5 화면
  - 요구사항: 응답 시간 ≤ 2초
- 내부 모듈 간 통신
  - Collector → Detection Engine → Storage → Dashboard
  - 프로토콜: gRPC 또는 REST
  - 데이터 형식: JSON 이벤트
  - 요구사항: 모듈 간 지연 ≤ 500ms

3.2. Functional Requirements (기능 요구사항)

FR-1. 쿼리 로깅 기능

- 시스템은 모든 SQL 쿼리를 기록해야 한다. (SELECT, INSERT, UPDATE, DELETE 포함)
- 로그에는 사용자 ID, 실행 시간(ISO 8601 형식), SQL 원문 및 요약, 반환 행 수, 실행 결과 상태가 포함되어야 한다.
- 입력 검증: 모든 SQL 요청은 UTF-8 인코딩을 따라야 하며, 비정상 인코딩은 거부된다.

- 동작 순서: SQL 요청 수집 → 로그 기록 → DB 서버 전달 → 응답 반환 → 대시보드 반영.
- 비정상 대응: 로그 저장소가 가득 차면 오래된 로그를 자동 압축/아카이브한다.
- 출력: 로그는 대시보드에서 조회 가능하며, 필터링/검색/다운로드 기능을 제공한다.

FR-2. 패턴 기반 탐지 기능

- 시스템은 사전에 정의된 금지 패턴(SQL Injection, DROP TABLE, UNION SELECT 등)을 탐지해야 한다.
- 입력 검증: 사전에 등록된 화이트리스트/블랙리스트 규칙은 반드시 형식 검증 후 반영해야 한다.
- 동작 순서: SQL 요청 → 패턴 검사 → 정상 시 DB 전달 / 이상 시 이벤트 기록 + 알림 전송.
- 비정상 대응: 패턴 검사 모듈 오류 발생 시 경고 이벤트를 발생시킨다.
- 출력: 탐지 이벤트는 대시보드에 표시되며, 알림 채널로 즉시 전송된다.

FR-3. 행동 기반 탐지 기능

- 시스템은 정상 동작 프로파일과 비교하여 비정상 행위를 탐지해야 한다.
- 탐지 조건:
  - 쿼리 실행 시간 ≥ 5초 → 느린 쿼리 경고
  - 반환 행 수 ≥ 10,000건 → 대량 조회 경고
  - 사용자별 초당 쿼리 횟수 ≥ 100건 → 폭주 탐지
- 동작 순서: SQL 요청 → 실행 메타데이터 수집 → 정상 범위와 비교 → Score 계산 → 이벤트 판정
- 변환 규칙:

  ```
  Score = α*(쿼리 빈도 편차) + β*(반환 행 수 편차) + γ*(실행 시간 편차)
  ```

  (α=0.4, β=0.3, γ=0.3)

  Score ≥ 0.8 → 이상 탐지 이벤트 발생
- 비정상 대응:
  - 탐지 엔진 메모리 초과 시 최근 탐지 규칙만 유지하고 경고 이벤트를 발생시킨다.

    - 행동 기반 탐지 모듈 오류 발생 시 경고 이벤트를 발생시킨다.
- 출력: 탐지 이벤트는 대시보드에 표시되며, 알림 채널로 즉시 전송된다.

FR-4. 권한 기반 탐지 기능

- 시스템은 권한이 없는 계정의 비인가 접근을 탐지해야 한다.
- 입력 검증: 관리자 권한 정보는 RBAC(Role-Based Access Control)에 따라 사전 등록되어야 한다.
- 동작 순서: SQL 요청 → 사용자 권한 확인 → 정상 권한일 경우 통과 / 위반 시 이벤트 기록 + 알림 전송
- 비정상 대응: DB 권한 체크 실패 시 경고 이벤트를 발생시킨다.
- 출력: 탐지 이벤트는 대시보드에 표시되며, 알림 채널로 즉시 전송된다.

FR-5. 알림 기능

- 탐지 이벤트 발생 시 사전에 등록된 채널(Slack, Email)로 알림을 전송해야 한다.
- 입력 검증: 이메일 주소는 RFC 5321 형식을 따라야 하며, Slack URL은 HTTPS 형식이어야 한다.
- 동작 순서: 이벤트 생성 → 알림 메시지 구성 → 채널 전송
- 비정상 대응: 알림 전송 실패 시 3회 재시도한다.
- 파라미터 영향: 알림 전송 시간 SLA는 1~10초 내에 완료되어야 한다.
- 출력: 알림 메시지에는 탐지 유형, 심각도, 사용자 ID, 발생 시각, 원본 쿼리가 포함된다.

FR-6. 관리자 대시보드 기능

- 관리자는 웹 기반 UI를 통해 시스템 상태를 조회 및 제어할 수 있다.
- 기능:
    - 실시간 쿼리 로그 확인
    - 탐지 이벤트 목록/상세 조회
    - 사용자별·시간대별 통계 그래프 제공
    - 알림 채널 설정 관리

- 입력 검증: 관리자 로그인 시 ID/비밀번호 규칙을 검증한다.
- 동작 순서: 로그인 인증 성공 시 대시보드 진입 → 기능 메뉴 접근 → 요청 결과 표시
- 비정상 대응: 인증 실패 시 오류 메시지를 반환한다.
- 출력: 대시보드는 HTTPS를 통해 제공되며, 모든 데이터는 JSON 기반 API 응답으로 갱신된다.

FR-7. 로그 관리 및 백업 기능

- 시스템은 로그 및 탐지 이벤트 데이터를 주기적으로 백업해야 한다.
- 입력 검증: 백업 대상 경로와 주기 설정 값은 관리자가 지정해야 하며, 형식 검증이 필요하다.
- 동작 순서: 로그 저장 → 주기적 백업 작업 실행 → 성공/실패 여부 기록
- 비정상 대응: 로그 백업 실패 시 관리자 알림 전송 및 재시도를 수행한다.
- 출력: 오래된 로그는 아카이브 파일로 압축된다.

3.3. Performance Requirements (성능 요구사항)

PR-1. 응답 시간 (Response Time)

- SQL 쿼리 탐지 지연은 1초 이내여야 한다.
- 관리자 대시보드에서의 조회 응답은 2초 이내에 반환되어야 한다.
- 탐지 이벤트 발생 시 알림은 10초 이내에 전달되어야 한다.

PR-2. 처리량 (Throughput)

- 시스템은 초당 1,000건 이상의 SQL 요청을 처리할 수 있어야 한다.
- 로그 기록과 탐지는 이 처리량에서도 정상적으로 동작해야 한다.

PR-3. 동시성 (Concurrency)

- 시스템은 100명 이상의 동시 사용자 요청을 처리할 수 있어야 한다.
- 관리자 대시보드는 동시에 10명 이상의 접근을 지원해야 한다.

PR-4. 자원 사용 (Resource Utilization)

- 최소 사양(2 vCPU, 4GB RAM)에서도 안정적으로 동작해야 한다.
- 탐지 엔진은 쿼리당 50MB 이하의 메모리만 사용해야 한다.
- 로그 저장소는 일일 평균 1GB 이상의 데이터를 저장할 수 있어야 한다.

PR-5. 가용성 및 복구 (Availability & Recovery)

- 시스템은 월 기준 99% 이상의 가용성을 유지해야 한다.
- 장애 발생 시 1시간 이내 복구 가능해야 한다.
- 로그 데이터는 장애 상황에서도 손실되지 않아야 한다.

PR-6. 신뢰성 (Reliability)

- 오탐(False Positive)은 5% 이하, 미탐(False Negative)은 2% 이하여야 한다.
- 알림 전송 성공률은 99% 이상이어야 한다.

3.4. Logical Database Requirements (논리적 데이터베이스 요구사항)

3.4.1. 데이터 저장 요구사항

- 쿼리 로그(QueryLog)
  - 저장 항목: LogID, UserID, 실행시간, SQL 원문/요약, 반환 행 수, 실행 상태(성공/실패)
  - 목적: 모든 SQL 요청을 기록하여 분석 및 감사에 활용
- 탐지 이벤트(DetectionEvent)
  - 저장 항목: EventID, 관련 LogID, 탐지유형(패턴/행동/권한), 심각도(Level), 발생 시간, 원본 쿼리
  - 목적: 비정상 행위 탐지 기록 및 알림 트리거
- 관리자 계정(User)
  - 저장 항목: AdminID, PW 해시, 이메일, 최근 로그인 시간
  - 목적: 대시보드 접근 인증 및 권한 관리
- 알림 기록(NotificationLog)

- 저장 항목: NotifyID, EventID, 채널(Slack/Email), 전송 결과(성공/실패), 전송 시간
- 목적: 알림 성공률 추적 및 장애 원인 분석

3.4.2. 논리적 데이터 관계

- AdminUser ↔ QueryLog (1:N)

  하나의 관리자(AdminUser)는 여러 개의 쿼리 로그(QueryLog)를 생성할 수 있다.

- QueryLog ↔ DetectionEvent (1:0..1)

  하나의 쿼리 로그는 0개 또는 1개의 탐지 이벤트(DetectionEvent)를 가진다.

  모든 탐지 이벤트는 반드시 하나의 쿼리 로그에 연결된다.

- DetectionEvent ↔ NotificationLog (1:N)

  하나의 탐지 이벤트는 여러 알림(NotificationLog)과 연결될 수 있다.

- ArchiveLog ↔ QueryLog (1:N)

  하나의 아카이브 파일(ArchiveLog)에는 여러 개의 쿼리 로그(QueryLog)가 포함될 수 있으며,

  각 쿼리 로그는 0개 또는 1개의 아카이브 파일에 연결될 수 있다.

3.4.3. 무결성 및 일관성 요구사항

- 고유성 제약 (Uniqueness)
  - QueryLog.LogID, DetectionEvent.EventID, AdminUser.AdminID, NotificationLog.NotifyID는 모두 고유해야 한다.
- 참조 무결성 (Referential Integrity)
  - DetectionEvent.LogID는 반드시 QueryLog.LogID를 참조해야 한다.
  - NotificationLog.EventID는 반드시 DetectionEvent.EventID를 참조해야 한다.
- 데이터 일관성 (Consistency)
  - 비밀번호는 평문으로 저장하지 않고 반드시 해시(SHA-256 이상) 방식으로 저장해야 한다.
- 보안 제약 (Security Constraints)
  - 탐지 이벤트 및 로그 데이터는 관리자 외 접근 불가해야 한다.

- 민감정보(사용자 비밀번호)는 암호화 저장해야 한다.

3.5. Design Constraints (설계 제약사항)

3.5.1. 운영체제 및 플랫폼 제약

- 운영체제는 Linux 기반 (Ubuntu 20.04 LTS 이상)을 사용해야 한다.
- 서버는 x86-64 아키텍처에서 동작해야 하며, 클라우드 VM을 지원한다.
- 최소 하드웨어 사양은 2 vCPU, 4GB RAM, 20GB SSD를 충족해야 한다.

3.5.2. 개발 언어 및 프레임워크 제약

- 백엔드 탐지 엔진은 Java 21 (LTS) + Spring Boot 3.5+를 사용해야 한다.
- 프론트엔드 대시보드는 React 19+ 기반으로 작성되어야 한다.
- 내부 데이터베이스는 SQLite 3.35+를 사용하여 관리 데이터를 저장해야 한다.
- 대상 DBMS는 MySQL 8.0 이상을 지원해야 한다.

3.5.3. 표준 및 규격 준수

- 보안 통신은 반드시 HTTPS (TLS 1.2 이상)를 사용해야 한다.
- 비밀번호는 평문 저장이 금지되며, SHA-256 이상의 해시 알고리즘으로 저장해야 한다.
- 로그 및 이벤트 데이터는 UTF-8 인코딩을 준수해야 한다.
- 이메일 발송은 SMTP (RFC 5321)를 따라야 한다.

3.5.4. 보안 및 규제 준수

- 개인정보 보호법 및 ISMS 보안 규정을 준수해야 한다.
- 로그에는 주민등록번호, 계좌번호 등 민감정보를 직접 저장해서는 안 된다.
- 탐지 이벤트 및 로그는 관리자 외에는 접근할 수 없어야 하며, 무결성이 보장되어야 한다.

3.5.5. 운영 환경 제약

- DB-IDS는 모니터링 대상 DB와 동일 네트워크(VPC) 내에서 배치되어야 한다.
- 시스템 장애 발생 시 1시간 이내 복구(Mean Time To Recovery ≤ 1h)가 가능해야 한다.

- 로그 백업은 하루 1회 이상 수행되어야 하며, 보관 정책은 최소 30일을 보장해야 한다.

3.5.6.외부 연동 제약

- 알림 전송은 Slack Webhook API (HTTPS POST) 및 SMTP 메일 서버(TCP 587, TLS)를 통해 이루어져야 한다.
- 외부 API 장애 발생 시 최대 3회까지 재시도 후 장애 로그를 기록해야 한다.
- 외부 서비스 의존성(Slack, SMTP 등)으로 인한 가용성 저하 가능성을 고려해야 한다.

3.6. Software System Attributes (소프트웨어 시스템 속성)

3.6.1. 신뢰성 (Reliability)

- 시스템은 월 기준 99% 이상의 가용성을 유지해야 한다.
- 탐지 오탐(False Positive)은 5% 이하, 미탐(False Negative)은 2% 이하를 목표로 한다.
- 로그 및 탐지 이벤트 데이터는 장애 상황에서도 손실되지 않아야 한다.
- 장애 발생 시 1시간 이내 복구(Mean Time To Recovery ≤ 1h)가 가능해야 한다.

3.6.2. 가용성 (Availability)

- 시스템은 24시간 365일 상시 운영 가능해야 한다.

3.6.3. 보안성 (Security)

- 모든 통신은 HTTPS (TLS 1.2 이상)를 사용해야 한다.
- 관리자 인증은 ID/비밀번호 조합을 사용하며, 비밀번호는 SHA-256 이상의 해시 알고리즘으로 저장해야 한다.
- 관리자 계정은 5회 이상 로그인 실패 시 잠금 상태로 전환되어야 한다.
- 탐지 로그와 이벤트 데이터는 관리자 외에는 접근할 수 없어야 한다.

3.6.4. 유지보수성 (Maintainability)

- 시스템은 모듈화된 구조로 구성되어야 한다.
- 주요 기능은 독립적으로 수정·배포 가능해야 한다.

- 로그 및 설정 파일은 사람이 읽을 수 있는 형식으로 관리되어야 한다.
- 에러 메시지와 예외 로그는 표준화된 형식으로 기록되어야 한다.

3.6.5. 확장성 (Scalability)

- 단일 인스턴스로 초당 1,000건의 SQL 요청을 처리할 수 있어야 하며, 필요 시 수평 확장을 통해 처리량을 늘릴 수 있어야 한다.
- 탐지 규칙은 버전 관리가 가능해야 한다.
- 대시보드의 로그 조회 기능은 데이터 증가에 대비해 페이지네이션 및 인덱싱을 지원해야 한다.

3.6.6. 이식성 (Portability)

- DB-IDS는 Docker 기반 컨테이너 환경에서도 실행 가능해야 한다.
- 클라우드 환경(AWS EC2, GCP Compute Engine 등)에서 지원 가능해야 한다.
- 기본 지원 DBMS는 MySQL 8.0 이상이며, 향후 PostgreSQL, Oracle 등 다른 RDBMS로 확장 가능해야 한다.

4. Supporting Information (추가 정보)

4.1. Table of Contents and Index (목차와 인덱스)