DNA Lab Security Scan Report

Date: 2026-01-05 20:23:59

Target: http://127.0.0.1:8080

16	15	1	74
Total Scans	Vulnerable	Safe	Total Findings

1. XSS (Cross-Site Scripting) — VULNERABLE

DETAILS

- [Dynamic] Reflected XSS 취약 (/board/list, param: keyword)
- [Header] CSP 헤더 미설정

RECOMMENDATION

출력값 검증 및 이스케이프 처리 | CSP (Content-Security-Policy) 헤더 설정

2. SQL Injection — VULNERABLE

DETAILS

- [Static] UserMapper.xml - select#login - SQL Injection 취약: ${username} (#{username} 사용 필요)
- [Static] UserMapper.xml - select#login - SQL Injection 취약: ${password} (#{password} 사용 필요)
- [Static] UserMapper.xml - select#findByUsername - SQL Injection 취약: ${username} (#{username} 사용 필요)
- [Static] UserMapper.xml - select#findByUsernameAndEmail - SQL Injection 취약: ${username} (#{username} 사용 필요)
- [Static] UserMapper.xml - select#findByUsernameAndEmail - SQL Injection 취약: ${email} (#{email} 사용 필요)
- [Static] UserMapper.xml - insert#save - SQL Injection 취약: ${username} (#{username} 사용 필요)
- [Static] UserMapper.xml - insert#save - SQL Injection 취약: ${password} (#{password} 사용 필요)
- [Static] UserMapper.xml - insert#save - SQL Injection 취약: ${nickname} (#{nickname} 사용 필요)
- [Static] UserMapper.xml - insert#save - SQL Injection 취약: ${email} (#{email} 사용 필요)
- [Static] UserMapper.xml - insert#save - SQL Injection 취약: ${role} (#{role} 사용 필요)
- [Static] UserMapper.xml - update#update - SQL Injection 취약: ${nickname} (#{nickname} 사용 필요)
- [Static] UserMapper.xml - update#update - SQL Injection 취약: ${email} (#{email} 사용 필요)
- [Static] UserMapper.xml - update#update - SQL Injection 취약: ${password} (#{password} 사용 필요)
- [Static] UserMapper.xml - update#update - SQL Injection 취약: ${id} (#{id} 사용 필요)
- [Static] UserMapper.xml - delete#delete - SQL Injection 취약: ${id} (#{id} 사용 필요)
- [Static] BoardMapper.xml - select#findAll - SQL Injection 취약: ${keyword} (#{keyword} 사용 필요)
- [Static] BoardMapper.xml - select#findByWriter - SQL Injection 취약: ${writer} (#{writer} 사용 필요)

- [Static] BoardMapper.xml - select#findByWriter - SQL Injection 취약: ${keyword} (#{keyword} 사용 필요)
- [Static] BoardMapper.xml - select#findByWriterOrAdmin - SQL Injection 취약: ${writer} (#{writer} 사용 필요)
- [Static] BoardMapper.xml - select#findByWriterOrAdmin - SQL Injection 취약: ${keyword} (#{keyword} 사용 필요)
- [Static] BoardMapper.xml - select#findById - SQL Injection 취약: ${id} (#{id} 사용 필요)
- [Static] BoardMapper.xml - insert#save - SQL Injection 취약: ${title} (#{title} 사용 필요)
- [Static] BoardMapper.xml - insert#save - SQL Injection 취약: ${content} (#{content} 사용 필요)
- [Static] BoardMapper.xml - insert#save - SQL Injection 취약: ${writer} (#{writer} 사용 필요)
- [Static] BoardMapper.xml - insert#save - SQL Injection 취약: ${filename} (#{filename} 사용 필요)
- [Static] BoardMapper.xml - insert#save - SQL Injection 취약: ${filepath} (#{filepath} 사용 필요)
- [Dynamic] 로그인 SQL Injection 취약 (payload: admin'#)
- [Dynamic] URL 파라미터 SQL Injection 취약 (/user/profile?id=)

RECOMMENDATION

PreparedStatement 사용 및 입력값 검증 | MyBatis에서 ${} 대신 #{} 파라미터 사용 | URL 파라미터 검증 강화

3. OS Command Injection VULNERABLE

DETAILS

- [Whitebox] AdminController.java:86 - OS Command Injection 위험 (사용자 입력이 명령어 실행에 사용됨)
- [Whitebox] AdminController.java:86 - 명령어 문자열 동적 조합 발견
- [Blackbox] OS Command Injection 취약 (/admin/system/ping): Windows 명령어 실행됨 (payload: 127.0.0.1 & ipconfig)

- [Blackbox] Blind Command Injection 취약 (/admin/system/ping): 시간 지연 확인 (7.1초)

RECOMMENDATION

사용자 입력을 명령어로 직접 실행 금지 | 명령어 하드코딩 및 파라미터 검증 강화 | Runtime.exec() 사용 금지 또는 화이트리스트 검증 | 명령어 실행 함수 사용 최소화 및 입력 검증 강화

4. CSRF (Cross-Site Request Forgery) VULNERABLE

DETAILS

- [Dynamic] CSRF 토큰 미사용: /board/write
- [Dynamic] CSRF 보호 없음: /user/profile (토큰 없이 작업 수행 가능)

RECOMMENDATION

Spring Security CSRF 보호 활성화 | POST 요청에 CSRF 토큰 추가

5. 약한 비밀번호 정책 VULNERABLE

DETAILS

- [Dynamic] 약한 비밀번호 허용됨 (예: 1234)

RECOMMENDATION

비밀번호 복잡도 검증 (최소 8자, 대소문자/숫자/특수문자 포함)

6. 불충분한 접근 제어 (인증/인가) VULNERABLE

DETAILS

- [Whitebox] UserMapper.xml - SQL Injection 취약 (로그인 쿼리)
- [Whitebox] BoardController.java - 게시글 조회 소유권 검증 부재 (IDOR 위험)
- [Blackbox] SQL Injection 인증 우회 성공 (payload: admin'#)

RECOMMENDATION

MyBatis에서 ${} 대신 #{} 사용 | PreparedStatement로 SQL Injection 방어

7. 취약한 비밀번호 복구 절차 VULNERABLE

DETAILS

- [Whitebox] DataInitializer.java - 안전한 토큰 생성 로직 부재
- [Whitebox] DataInitializer.java - 토큰 만료 시간 로직 부재
- [Whitebox] WebConfig.java - 안전한 토큰 생성 로직 부재
- [Whitebox] WebConfig.java - 토큰 만료 시간 로직 부재
- [Whitebox] AuthController.java - 토큰 만료 시간 로직 부재
- [Whitebox] UserMapper.java - 안전한 토큰 생성 로직 부재
- [Whitebox] UserMapper.java - 토큰 만료 시간 로직 부재
- [Blackbox] 비밀번호 재설정 토큰 검증 미흡: /password/reset
- [Blackbox] 예측 가능한 정보로 비밀번호 재설정 가능 (보안 질문 취약)

RECOMMENDATION

토큰 유효 시간 제한 (15분) | 보안 질문 제거 (토큰 기반 재설정만 허용) | 이메일 인증 토큰 사용 | UUID/SecureRandom으로 안전한 토큰 생성

8. 불충분한 세션 관리 VULNERABLE

DETAILS

- [Blackbox] 세션 쿠키에 HttpOnly 플래그 미설정

RECOMMENDATION

HttpOnly/Secure 쿠키 플래그 설정

9. 쿠키 변조 VULNERABLE

DETAILS

- [Blackbox] 쿠키 변조로 관리자 페이지 접근 성공: /admin/system (role=admin)
- [Whitebox] AdminController.java - 검증 없는 쿠키 기반 권한 제어 의심
- [Whitebox] AuthController.java - 검증 없는 쿠키 기반 권한 제어 의심

RECOMMENDATION

쿠키 서명(HMAC) 구현 또는 세션 기반 권한 관리 | 쿠키 값 변조 방지(서명) 적용

10. 파일 전송 취약점 VULNERABLE

DETAILS

- [Blackbox] 위험한 파일 업로드 가능: malware.exe (/libs/upload) - 200
- [Blackbox] 경로 조작으로 설정 파일 접근 성공: /download?path=../application.properties

RECOMMENDATION

파일명 난수화 및 확장자 화이트리스트 적용 | 다운로드 가능 경로를 화이트리스트로 제한 | 경로 정규화(normalize) 및 상위 디렉토리 이동(..) 차단 | 실행 가능한 확장자(.exe) 업로드 차단

11. 경로 조작 (Path Traversal) VULNERABLE

DETAILS

- [Blackbox] 시스템 파일 접근 가능 (/file/download)
- [Blackbox] 설정 파일 노출 확인 (/file/view)

RECOMMENDATION

화이트리스트 기반 경로 제한 | 파일 경로 정규화 및 검증

12. 에러 페이지 적용 미흡 VULNERABLE

DETAILS

- [Blackbox] 없는 경로가 메인/로그인 페이지로 리다이렉트됨 (Soft-404): http://127.0.0.1:8080/this_should_404_zzzz
- [Blackbox] 없는 경로가 메인/로그인 페이지로 리다이렉트됨 (Soft-404): http://127.0.0.1:8080/%ZZ
- [Blackbox] 없는 경로가 메인/로그인 페이지로 리다이렉트됨 (Soft-404): http://127.0.0.1:8080/error_test_1234

RECOMMENDATION

운영 환경에서 스택 트레이스 및 상세 에러 메시지 노출 차단 (Custom Error Page 적용) | 존재하지 않는 페이지 요청 시 404 상태 코드 반환

13. Format String 취약점 VULNERABLE

DETAILS

- [Whitebox] SupportController.java:37 - 포맷 문자열에 사용자 입력 연결
- [Blackbox] Format String 민감 정보 유출: SK-SHIELDUS-ADMIN-KEY-2025

RECOMMENDATION

String.format() 포맷 문자열 하드코딩 권장 | String.format()에서 민감 정보 제거

14. 불필요한 HTTP Method 악용 SAFE

RECOMMENDATION

안전 - 불필요한 HTTP Method가 차단되어 있음

15. SSRF (Server-Side Request Forgery) VULNERABLE

DETAILS

- [Whitebox] AdminController.java:86 - Command Injection 위험 (exec에 입력값 연결)
- [Blackbox] Command Injection/SSRF 취약: /admin/system/ping (payload: 127.0.0.1)
- [Blackbox] 내부망(Loopback) 접근 가능: /admin/system/ping

RECOMMENDATION

입력값 검증 및 명령어 인젝션 방어 | Runtime.exec() 사용 금지 또는 화이트리스트 검증 | 내부 IP 대역 접근 차단

16. Dynamic Scan — VULNERABLE

DETAILS

- [Dynamic] XSS found: http://127.0.0.1:8080/join
- [Dynamic] SQL Injection found: http://127.0.0.1:8080/join
- [Dynamic] CSRF token missing: http://127.0.0.1:8080/board/write
- [Dynamic] CSRF token missing: http://127.0.0.1:8080/update
- [Dynamic] CSRF token missing: http://127.0.0.1:8080/user/family/delete

RECOMMENDATION

Input HTML escaping (th:text) | Implement CSRF token | Use PreparedStatement