|  |
| --- |
| **스터디 2조 OyesMall** |
| **시나리오 기반 모의침투 결과보고서** |

|  |  |  |
| --- | --- | --- |
|  | 팀명: |  |
|  | **팀장:** |  |
|  | **팀원:** |  |
|  |  |  |
|  |  |  |
|  |  |  |

**목차**

[1. 개요 9](#_Toc191031107)

[1.1. 모의해킹 정의 9](#_Toc191031108)

[1.2. 수행일정/수행내역 9](#_Toc191031109)

[1.3. 수행 대상 (수정필요) 9](#_Toc191031110)

[1.4. 수행 단계별 방법 9](#_Toc191031111)

[1.5. 점검 항목 10](#_Toc191031112)

[1.6. 점검 도구 (수정필요) 11](#_Toc191031113)

[2. 결과 요약 13](#_Toc191031114)

[2.1. 총평 13](#_Toc191031115)

[3. 상세 수행 내역 14](#_Toc191031116)

[3.1. 환경 구성 14](#_Toc191031117)

[3.2. 정보 수집 15](#_Toc191031118)

[3.3. 계정정보 추측 및 대입 16](#_Toc191031119)

[3.3.1. 취약한 패스워드 설정 여부 16](#_Toc191031120)

[3.3.2. 어플리케이션/장비 기본 패스워드 설정 여부 18](#_Toc191031121)

[3.4. 인증 우회 20](#_Toc191031122)

[3.4.1. 쿠키 재사용 (Replay Attack) 여부 20](#_Toc191031123)

[3.4.2. 클라이언트 인증 우회 여부 (Javascript 우회) 21](#_Toc191031124)

[3.5. 파라미터 조작 23](#_Toc191031125)

[3.5.1. 필드 값 조작에 따른 검증 여부 23](#_Toc191031126)

[3.6. XSS (CSRF) 취약점 24](#_Toc191031127)

[3.6.1. 악의적인 스크립트 필터링 여부 (POST 메소드) 25](#_Toc191031128)

[3.6.2. XST, TRACE 옵션 허용 여부 26](#_Toc191031129)

[3.7. 관리자 페이지 추측 27](#_Toc191031130)

[3.7.1. 페이지 내 관리자 페이지 링크 여부 27](#_Toc191031131)

[3.7.2. 관리자 페이지 접근 여부 28](#_Toc191031132)

[3.8. 페이지 내 중요 정보 노출 29](#_Toc191031133)

[3.8.1. 중요 개인정보 노출 여부 29](#_Toc191031134)

[3.8.2. 쿠키 값 내 중요정보 유출 여부 30](#_Toc191031135)

[3.8.3. 중요정보 평문 전송 31](#_Toc191031136)

[3.9. 파일 다운로드 취약점 32](#_Toc191031137)

[3.9.1. 입력 값 검증 미흡으로 파일 다운로드 공격 여부 32](#_Toc191031138)

[3.10. 파일 업로드 취약점 33](#_Toc191031139)

[3.10.1. 입력 값 검증 미흡으로 파일 업로드 공격 여부 33](#_Toc191031140)

[3.11. SQL Injection 35](#_Toc191031141)

[3.11.1. SQL Injection 허용 여부 35](#_Toc191031142)

[3.12. 최신 취약점 미패치 38](#_Toc191031143)

[3.12.1. 보안에 취약한 오래된 어플리케이션 사용 여부 38](#_Toc191031144)

[3.13. 부적절한 서버 설정 39](#_Toc191031145)

[3.13.1. 서버 보안 설정 여부 39](#_Toc191031146)

[4. 취약점 대응방안 41](#_Toc191031147)

[4.1. 계정정보 추측 및 대입 41](#_Toc191031148)

[4.1.1. 취약한 패스워드 설정 여부 41](#_Toc191031149)

[4.1.2. 어플리케이션/장비 기본 패스워드 설정 여부 41](#_Toc191031150)

[4.2. 인증 우회 41](#_Toc191031151)

[4.2.1. 쿠키 재사용 (Replay Attack) 여부 41](#_Toc191031152)

[4.2.2. 클라이언트 인증 우회 여부 (Javascript 우회) 42](#_Toc191031153)

[4.3. 파라미터 조작 42](#_Toc191031154)

[4.3.1. 필드 값 조작에 따른 검증 여부 42](#_Toc191031155)

[4.4. XSS (CSRF) 취약점 42](#_Toc191031156)

[4.4.1. 악의적인 스크립트 필터링 여부 (POST 메소드) 42](#_Toc191031157)

[4.4.2. XST, TRACE 옵션 허용 여부 43](#_Toc191031158)

[4.5. 관리자 페이지 추측 43](#_Toc191031159)

[4.5.1. 페이지 내 관리자 페이지 링크 여부 43](#_Toc191031160)

[4.5.2. 관리자 페이지 접근 여부 43](#_Toc191031161)

[4.6. 페이지 내 중요 정보 노출 43](#_Toc191031162)

[4.6.1. 중요 개인정보 노출 여부 43](#_Toc191031163)

[4.6.2. 쿠키 값 내 중요정보 유출 여부 44](#_Toc191031164)

[4.6.3. 중요정보 평문 전송 44](#_Toc191031165)

[4.7. 파일 다운로드 취약점 44](#_Toc191031166)

[4.7.1. 입력 값 검증 미흡으로 파일 다운로드 공격 여부 44](#_Toc191031167)

[4.8. 파일 업로드 취약점 44](#_Toc191031168)

[4.8.1. 입력 값 검증 미흡으로 파일 업로드 공격 여부 44](#_Toc191031169)

[4.9. SQL Injection 44](#_Toc191031170)

[4.9.1. SQL Injection 허용 여부 44](#_Toc191031171)

[4.10. 최신 취약점 미패치 45](#_Toc191031172)

[4.10.1. 보안에 취약한 오래된 어플리케이션 사용 여부 45](#_Toc191031173)

[4.11. 부적절한 서버 설정 45](#_Toc191031174)

[4.11.1. 서버 보안 설정 여부 45](#_Toc191031175)

**그림 목차**

[그림 1‑1 모의해킹 수행 단계 10](#_Toc191031176)

[그림 3‑1 모의해킹 환경 14](#_Toc191031177)

[그림 3‑2 모의해킹 홈페이지 (OyesMall) 15](#_Toc191031178)

[그림 3‑3 Nikto를 통한 기본적인 스캔 16](#_Toc191031179)

[그림 3‑4 취약한 패스워드 회원가입 시도 17](#_Toc191031180)

[그림 3‑5 취약한 패스워드 회원가입 성공 17](#_Toc191031181)

[그림 3‑6 SQL Injection 취약점을 이용한 데이터베이스 조사 18](#_Toc191031182)

[그림 3‑7 서버에서 사용 중인 데이터베이스 목록 18](#_Toc191031183)

[그림 3‑8 SQL Injection 취약점을 이용한 oyesmall 테이블 조사 18](#_Toc191031184)

[그림 3‑9 데이터베이스 oyesmall의 테이블 목록 19](#_Toc191031185)

[그림 3‑10 SQL Injection 취약점을 이용한 admin\_tb의 데이터 조사 19](#_Toc191031186)

[그림 3‑11 데이터베이스 oyesmall의 테이블 admin\_tb의 데이터 19](#_Toc191031187)

[그림 3‑12 로그인 전 개발자 도구 쿠키 20](#_Toc191031188)

[그림 3‑12 로그인 전 Burp Suite 쿠키 20](#_Toc191031189)

[그림 3‑13 로그인 후 개발자 도구 쿠키 21](#_Toc191031190)

[그림 3‑13 로그인 후 Burp Suite 쿠키 21](#_Toc191031191)

[그림 3‑14 회원가입 페이지 응답 소스코드 165~187 21](#_Toc191031192)

[그림 3‑15 회원가입 페이지 응답 소스코드 189~214 22](#_Toc191031193)

[그림 3‑16 회원가입 페이지 응답 소스코드 216~240 22](#_Toc191031194)

[그림 3‑17 회원가입 페이지 응답 소스코드 248~261 23](#_Toc191031195)

[그림 3‑18 회원가입 요청 파라미터 24](#_Toc191031196)

[그림 3‑19 변조된 값으로 회원가입 성공 24](#_Toc191031197)

[그림 3‑20 악성 스크립트 업로드 시도 25](#_Toc191031198)

[그림 3‑21 악성 스크립트 업로드 성공 25](#_Toc191031199)

[그림 3‑22 탈취한 사용자 정보 리스트 26](#_Toc191031200)

[그림 3‑23 탈취한 사용자 정보 26](#_Toc191031201)

[그림 3‑24 HTTP 메서드로 TRACE 사용 27](#_Toc191031202)

[그림 3‑25 관리자 페이지 링크 위치 28](#_Toc191031203)

[그림 3‑26 OyesMall 관리자 페이지 28](#_Toc191031204)

[그림 3‑27 sqlmap을 이용하여 테이블 정보 조회 28](#_Toc191031205)

[그림 3‑28 sqlmap 결과 29](#_Toc191031206)

[그림 3‑29 회원정보 관리 페이지 개발자 모드 30](#_Toc191031207)

[그림 3‑30 쿠키 값 내 중요 정보 노출 31](#_Toc191031208)

[그림 3‑31 중요정보 평문 전송 31](#_Toc191031209)

[그림 3‑32 사용자가 업로드한 파일 32](#_Toc191031210)

[그림 3‑33 다운로드 시 입력 값 32](#_Toc191031211)

[그림 3‑34 시스템 파일 다운로드 시도 33](#_Toc191031212)

[그림 3‑35 시스템 파일 다운로드 성공 33](#_Toc191031213)

[그림 3‑36 악성 파일 업로드 시도 34](#_Toc191031214)

[그림 3‑37 업로드 시 입력 값 34](#_Toc191031215)

[그림 3‑38 파일 확장자 변경 34](#_Toc191031216)

[그림 3‑39 악성 파일 업로드 성공 35](#_Toc191031217)

[그림 3‑40 오류 발생을 위한 특수문자 전송 35](#_Toc191031218)

[그림 3‑41 특수문자로 인한 오류 발생 36](#_Toc191031219)

[그림 3‑42 SQL Injection 취약점을 이용한 데이터베이스 조사 36](#_Toc191031220)

[그림 3‑43 서버에서 사용 중인 데이터베이스 목록 37](#_Toc191031221)

[그림 3‑44 SQL Injection 취약점을 이용한 oyesmall 테이블 조사 37](#_Toc191031222)

[그림 3‑45 데이터베이스 oyesmall의 테이블 목록 37](#_Toc191031223)

[그림 3‑46 SQL Injection 취약점을 이용한 admin\_tb의 데이터 조사 38](#_Toc191031224)

[그림 3‑47 데이터베이스 oyesmall의 테이블 admin\_tb의 데이터 38](#_Toc191031225)

[그림 3‑48 취약한 웹 서버 소프트웨어 사용 39](#_Toc191031226)

[그림 3‑49 서버 보안 설정 미흡 40](#_Toc191031227)

**표 목차**

[표 1‑1 모의해킹 진단 일정 9](#_Toc191031228)

[표 1‑2 모의해킹 수행 범위 9](#_Toc191031229)

[표 1‑3 수행 단계 설명 10](#_Toc191031230)

[표 1‑4 점검 항목 11](#_Toc191031231)

[표 1‑5 점검 도구 12](#_Toc191031232)

[표 2‑1 모의해킹 진단 내역 13](#_Toc191031233)

[표 3‑1 계정정보 추측 및 대입 점검 항목 16](#_Toc191031234)

[표 3‑2 인증 우회 점검 항목 20](#_Toc191031235)

[표 3‑3 파라미터 조작 점검 항목 23](#_Toc191031236)

[표 3‑4 XSS (CSRF) 취약점 점검 항목 24](#_Toc191031237)

[표 3‑5 관리자 페이지 추측 점검 항목 27](#_Toc191031238)

[표 3‑6 페이지내 중요 정보 노출 점검 항목 29](#_Toc191031239)

[표 3‑7 파일 다운로드 취약점 점검 항목 32](#_Toc191031240)

[표 3‑8 파일 업로드 취약점 점검 항목 33](#_Toc191031241)

[표 3‑9 SQL Injection 점검 항목 35](#_Toc191031242)

[표 3‑10 최신 취약점 미패치 점검 항목 38](#_Toc191031243)

[표 3‑11 부적절한 서버 설정 점검 항목 39](#_Toc191031244)

# 개요

## 모의해킹 정의

본 모의해킹 진단은 웹 서비스의 관련된 모든 정보 자산에 대해 취약점을 도출/분석 하여 대책을 수립하기 위한 것이다. 시나리오 기반으로 해커와 동일한 환경과 조건, 기술을 가지고 모의 침투를 실행하여 취약점을 발견하고, 발견된 취약점을 점검하여 사전 예방을 통한 보안 현황 확인과 대응 방안 확립을 목적으로 한다.

## 수행일정/수행내역

본 모의해킹은 2025년 02월 20일부터 ~ 2025년 02월 21일까지 2일간 진행이 되었다. Task 별 자세한 일정은 표 1‑1과 같다.

|  |  |
| --- | --- |
| 02월 20일(목) | 02월 21일(금) |
| 환경분석  모의해킹 수행 | 모의해킹 수행  보고서 작성 |

표 ‑ 모의해킹 진단 일정

## 수행 대상

본 모의해킹은 아래 표 1‑2의 OyesMall 웹 서비스를 대상으로 점검을 진행했다.

|  |  |  |
| --- | --- | --- |
| 대상 도메인 | 대상 IP 정보 | 서비스 |
| http://210.95.67.235:8183/ | 210.95.67.235 | ASP 웹 서비스 |

표 ‑ 모의해킹 수행 범위

## 수행 단계별 방법

본 모의해킹은 단계별로 정보 수집부터 보고서 작성까지 아래 그림 1‑1의 과정을 통해 진행된다.



그림 ‑ 모의해킹 수행 단계

각 수행 단계별 요약 설명은 아래 표 1‑3과 같다.

|  |  |
| --- | --- |
| 수행 단계 | 설명 |
| 정보 수집 | 대상에 대한 서버/네트워크/서비스에 대한 불필요한 서비스 접근 가능성, 외부에서 파악할 수 있는 정보들을 수집하는 단계 |
| 취약점 분석 | 각 네트워크 구간별로 적합한 취약점 스캔 도구를 이용하여 발생할 수 있는 취약점에 대한 정보를 수집하는 단계 (단, 네트워크 장비/서비스에 장애를 유발할 수 있는 경우에는 제외) |
| 침투 테스트 | 취약점 정보 수집 및 분석 단계를 통해 획득한 정보를 기반으로 수동 점검하여 시스템 내부까지 침투할 수 있는지 시나리오 기반으로 접근하는 단계 |
| 실제 공격 | 취약점이 확인되었을 때 공격에 의한 시스템 보안 위협이 시스템 및 비지니스 측면에서 어느 정도의 영향을 미칠 수 있는지 분석하는 단계 |
| 보고서 작성 | 도출된 취약점에 대한 총평/영향도/상세분석/보안가이드가 포함된 보고서를 작성하는 단계 |

표 ‑ 수행 단계 설명

## 점검 항목

점검 항목은 OWASP TOP 10, SANS TOP 25, KISA 48대 취약점 항목 등을 기반으로 제작된 자사의 취약점 점검 방법론을 이용하여 진행된다.

|  |  |  |  |
| --- | --- | --- | --- |
| **순번** | **분류** | **코드** | **점검 항목** |
| 1 | 계정정보 추측 및 대입 | BP-001 | 취약한 패스워드 설정 여부 |
| BP-002 | 어플리케이션/장비 기본 패스워드 설정 여부 |
| 2 | 인증 우회 | BP-003 | 쿠키 재사용 (Replay Attack) 여부 |
| BP-005 | 클라이언트 인증 우회 여부 (Javascript 우회) |
| 3 | 파라미터 조작 | BP-007 | 필드 값 조작에 따른 검증 여부 |
| 4 | XSS (CSRF) 취약점 | BP-008 | 악의적인 스크립트 필터링 여부 (POST 메소드) |
| BP-010 | XST, TRACE 옵션 허용 여부 |
| 7 | 관리자 페이지 추측 | BP-014 | 페이지 내 관리자 페이지 링크 여부 |
| BP-015 | 관리자 페이지 접근 여부 |
| 8 | 페이지내 중요 정보 노출 | BP-016 | 중요 개인정보 노출 여부 |
| BP-017 | 쿠키 값 내 중요정보 유출 여부 |
| BP-019 | 중요정보 평문전송 |
| 10 | 파일 다운로드 취약점 | BP-023 | 입력 값 검증 미흡으로 파일 다운로드 공격 여부 |
| 11 | 파일 업로드 취약점 | BP-024 | 입력 값 검증 미흡으로 파일 업로드 공격 여부 |
| 14 | SQL Injection | BP-027 | SQL Injection 허용 여부 |
| 15 | 최신 취약점 미패치 | BP-028 | 보안에 취약한 오래된 어플리케이션 사용 여부 |
| 16 | 부적절한 서버 설정 | BP-029 | 서버 보안 설정 여부 |

표 ‑ 점검 항목

## 점검 도구

본 모의해킹을 수행하면서 사용된 도구는 아래 표 1‑5와 같다.

|  |  |  |
| --- | --- | --- |
| 도구 이름 | 용도 | 사이트 |
| Nmap | 취약점 분석 | http://nmap.org/ |
| DirBuster | 디렉터리 스캔 | http://sourceforge.net/projects/dirbuster/ |
| Nikto | 취약점 분석 | https://cirt.net/Nikto2 |
| BXSS | 침투 테스트 | https://bxsshunter.com/ |
| SQLMap | SQL Injection 테스트 | www.sqlmap.org/ |
| BurpSuite | 프록시 | http://portswigger.net/Burp/ |
| Exploit DB | 취약점 분석 | https://www.exploit-db.com/ |
| Foxy | 프록시 | https://getfoxyproxy.org/ |

표 ‑ 점검 도구

# 결과 요약

## 총평

본 모의해킹 진단 결과 총 17개의 취약점이 발견되었다. 주요 취약점은 아래 표 2‑1과 같다.

|  |  |
| --- | --- |
| 취약점 | 요약 |
| 계정정보 추측 및 대입 | 단순한 패스워드 설정으로 계정정보 추측 가능 |
| 인증 우회 | OyesMall 웹 서비스는 세션 토큰을 영구적으로 사용하고 있으며, 쿠키 재사용 공격에 취약한 것으로 확인 |
| 파라미터 조작 | 회원가입 페이지에서 파라미터 변조로 비밀번호 변조 여부 확인 |
| XSS 취약점 | 게시판 서비스에 악성 스크립트 삽입 및 다른 사용자 권한 획득이 가능하며, 바이러스 배포가 가능 |
| 관리자 페이지 추측 | 메인 페이지내 관리자 로그인 페이지 링크 여부 확인 |
| 페이지내 중요 정보 노출 | 개인정보수정 페이지에 불필요한 중요 개인정보가 노출 |
| 파일 다운로드 | 입력 경로를 체크하지 않는 파일 다운로드 스크립트를 이용하여 웹 사이트 주요 파일 다운로드 |
| 파일 업로드 | Q&A 게시판에 별도의 검증 절차가 없이 웹셸 업로드 확인 |
| SQL Injection | 관리자 페이지, 상품관리, 게시판내 검색 페이지에서 SQL Injection 취약점 발견 |
| 최신 취약점 미패치 | 취약점 점검 도구인 네서스(Nessus)를 이용하여 점검한 결과 총 19개의 웹 취약점이 발견 |
| 부적절한 서버 설정 | 잘못된 서버 설정으로 인한 클릭재킹, XSS 등이 가능 |

표 ‑ 모의해킹 진단 내역

# 상세 수행 내역

## 환경 구성

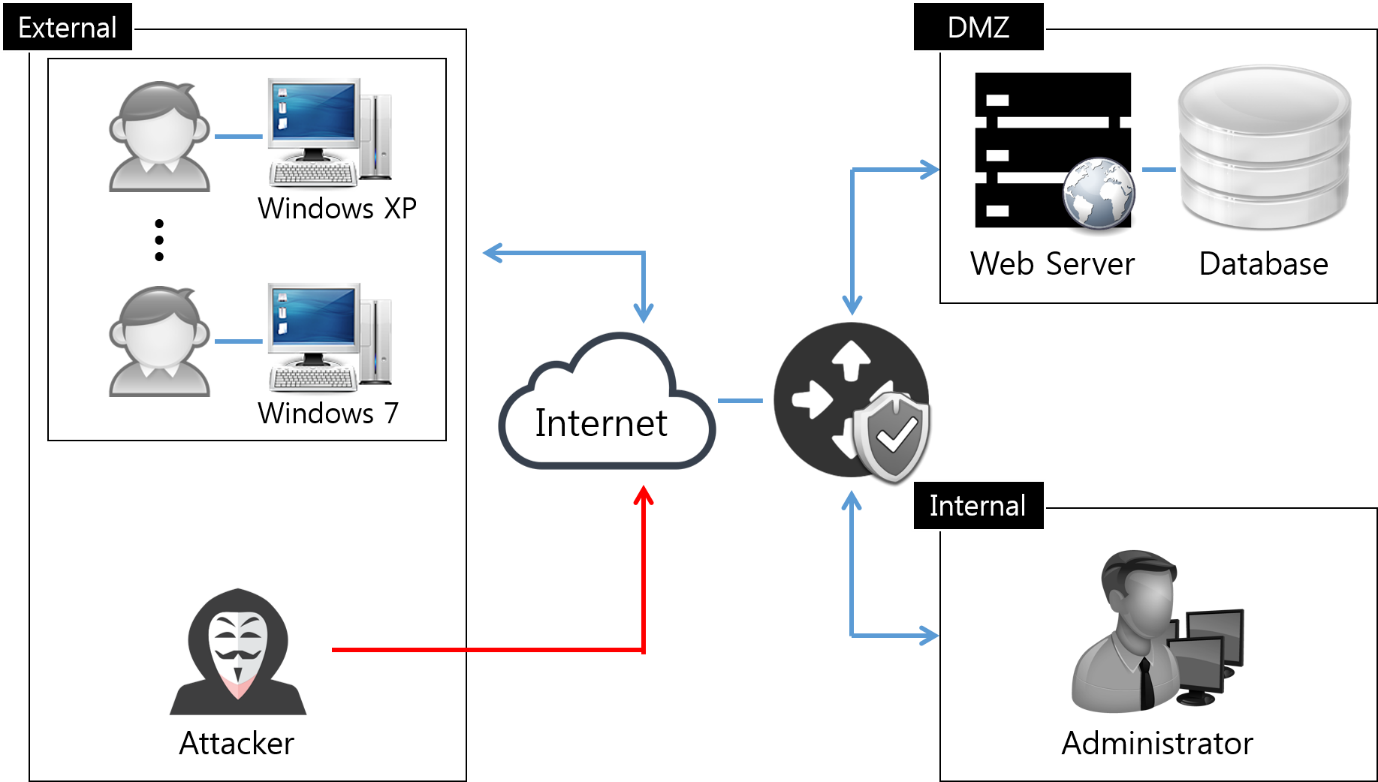


그림 ‑ 모의해킹 환경

위 그림 3‑1은 본 모의해킹 보고서의 환경 구성이다. 공격 대상서버는 ASP로 구성된 쇼핑몰이며, MSSQL을 사용한다. 공격자는 OyesMall ASP 쇼핑몰을 대상으로 다양한 시나리오 공격을 통해 모의해킹을 수행하며, 도출된 취약점을 바탕으로 대응 방안을 적용해본다.

* 환경은 시나리오 따라 달라질 수 있음

## 정보 수집

텍스트, 스크린샷, 소프트웨어, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 모의해킹 홈페이지 (OyesMall)

위 그림 3‑2는 시나리오 기반 모의해킹을 수행할 ASP로 구성된 OyesMall 쇼핑몰이다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ Nikto를 통한 기본적인 스캔

위 그림 3‑3은 nikto는 웹 서비스 취약점을 자동 점검해주는 Perl로 작성된 오픈 소스 도구이다. nikto의 스캔 결과로 보안에 취약한 오래된 어플리케이션을 사용하고 있는 점, 보안 헤더가 설정되지 않은 점, HTTP에서 TRACE메서드가 사용 가능한 점을 발견했다.

## 계정정보 추측 및 대입

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 계정정보  추측 및 대입 | BP-001 | 취약한 패스워드 설정 여부 |
| BP-002 | 어플리케이션/장비 기본 패스워드 설정 여부 |

표 ‑ 계정정보 추측 및 대입 점검 항목

### 취약한 패스워드 설정 여부

계정과 동일한 패스워드 생성 또는 단순 패스워드가 생성 가능하도록 웹 로직이 구현되어 있을 경우에 발생할 수 있는 취약점이다. 공격자가 무차별 대입 공격을 실행하면 사용자의 계정 및 패스워드가 노출될 수 있으며, 테스트 용도로 사용된 계정 또한 삭제되지 않을 경우 침해사고가 발생할 수 있다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 취약한 패스워드 회원가입 시도

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 취약한 패스워드 회원가입 성공

위 그림 3‑4, 그림 3‑5를 보면 test 라는 취약한 패스워드를 넣어도 아무런 검증 없이 그냥 넘어간다. 사전에 있는 단어나 아이디에 포함된 단어 등 유추하기 쉬운 패스워드는 자동화된 툴을 사용하면 간단하게 찾을 수 있다.

### 어플리케이션/장비 기본 패스워드 설정 여부

초기 설치 시 제공하는 기본 계정과 패스워드를 그대로 사용할 경우 발생할 수 있는 취약점이다. 기본 패스워드는 자동 스캔 도구를 통해 쉽게 노출될 위험이 있으며, 공격자가 이를 악용하면 중요 정보가 탈취될 가능성이 높다.

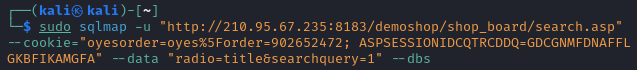


그림 ‑ SQL Injection 취약점을 이용한 데이터베이스 조사

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 서버에서 사용 중인 데이터베이스 목록

위 그림 3‑6, 그림 3‑7은 sqlmap을 활용하여 서버에서 사용 중인 데이터베이스 목록을 조회한 모습이다.

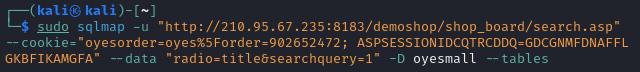


그림 ‑ SQL Injection 취약점을 이용한 oyesmall 테이블 조사

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 데이터베이스 oyesmall의 테이블 목록

위 그림 3‑8, 그림 3‑9은 sqlmap을 이용해 데이터베이스 oyesmall 내 테이블 목록을 확인한 모습이다.

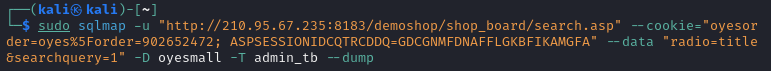


그림 ‑ SQL Injection 취약점을 이용한 admin\_tb의 데이터 조사

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 데이터베이스 oyesmall의 테이블 admin\_tb의 데이터

위 그림 3‑10, 그림 3‑11은 sqlmap을 사용하여 데이터베이스 oyesmall 내 테이블 admin\_tb의 데이터를 확인한 모습이다. 이를 통해 ID가 admin, PW가 admin인 관리자 계정 정보를 확인했다. SQL이 취약할 경우, 공격자가 SQL Injection 탐지 도구를 활용하면 정보가 쉽게 유출될 수 있다.

## 인증 우회

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 인증 우회 | BP-003 | 쿠키 재사용 (Replay Attack) 여부 |
| BP-005 | 클라이언트 인증 우회 여부 (Javascript 우회) |

표 ‑ 인증 우회 점검 항목

### 쿠키 재사용 (Replay Attack) 여부

여러 세션에서 동일한 쿠키를 재사용할 수 있도록 웹 로직이 구현되어 있을 경우 발생할 수 있는 취약점이다. 공격자는 탈취한 쿠키를 활용해 사용자의 인증 세션을 이용할 수 있으며, 추가적인 인증 절차 없이 정상 사용자로 가장할 수 있다. 이로 인해 계정 탈취, 권한 상승, 민감한 정보 접근 등의 보안 사고가 발생할 위험이 크다.

텍스트, 소프트웨어, 웹 페이지, 웹사이트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 로그인 전 개발자 도구 쿠키

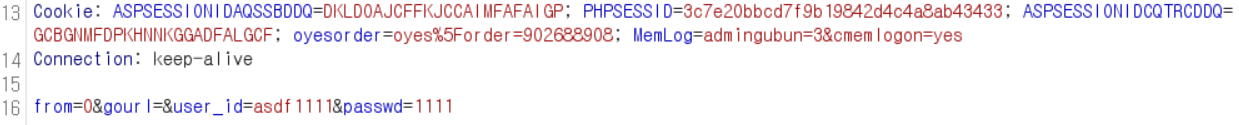


그림 ‑ 로그인 전 Burp Suite 쿠키

텍스트, 스크린샷, 소프트웨어, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 로그인 후 개발자 도구 쿠키

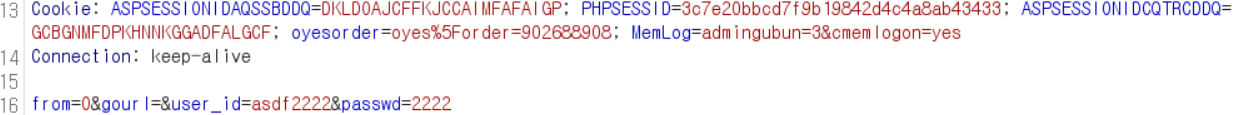


그림 ‑ 로그인 후 Burp Suite 쿠키

위 그림 3‑12, 그림 3‑13, 그림 3‑14, 그림 3‑15를 보면 F12 개발자 도구로 확인한 로그인 전, 후 쿠키가 동일하다. 이는 세션 관리가 적절하게 이루어지지 않았다는 의미이며, 보안 취약점을 초래할 수 있다.

### 클라이언트 인증 우회 여부 (Javascript 우회)

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원가입 페이지 응답 소스코드 165~187

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원가입 페이지 응답 소스코드 189~214

텍스트, 소프트웨어, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원가입 페이지 응답 소스코드 216~240

텍스트, 폰트, 번호, 스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원가입 페이지 응답 소스코드 248~261

위 그림 3‑16, 그림 3‑17, 그림 3‑18, 그림 3‑19과 같이 OyesMall의 회원가입 페이지는 클라이언트 인증만 존재한다. Burp Suite를 이용해 회원가입 응답 페이지를 수정할 경우 클라이언트 인증을 우회하여 주소, 전화번호, 핸드폰번호, 직업, 이메일 필드를 인증 없이 회원가입이 가능하다.

## 파라미터 조작

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 파라미터 조작 | BP-007 | 필드 값 조작에 따른 검증 여부 |

표 ‑ 파라미터 조작 점검 항목

### 필드 값 조작에 따른 검증 여부

사용자가 입력하는 파라미터에 대한 검증이 제대로 이루어지지 않을 경우 발생할 수 있는 취약점이다. 공격자는 요청 값을 변조하여 시스템을 악용하거나 비정상적인 요청을 지속적으로 발생시켜 서비스 장애를 유발할 수 있다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원가입 요청 파라미터

웹 사이트에서 회원가입 시 올바른 정보로 기입한 뒤 Burp Suite를 이용하여 위 그림 3‑20처럼 비밀번호 값을 1로 변경한다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 변조된 값으로 회원가입 성공

위 그림 3‑21를 보면 별다른 검증 없이 패스워드가 1로 가입된다.

## XSS (CSRF) 취약점

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 인증 우회 | BP-008 | 악의적인 스크립트 필터링 여부 (POST 메소드) |
| BP-010 | XST, TRACE 옵션 허용 여부 |

표 ‑ XSS (CSRF) 취약점 점검 항목

### 악의적인 스크립트 필터링 여부 (POST 메소드)

사용자가 입력하는 데이터를 검증하지 않아 악의적인 스크립트가 포함된 입력값이 그대로 처리될 경우 발생할 수 있는 취약점이다. 공격자는 악성 스크립트를 삽입하여 사용자의 세션 탈취, 피싱, 권한 없는 요청 수행 등을 유도할 수 있다. 이를 악용하여 사용자의 개인 정보가 유출되거나, 관리자가 의도하지 않은 명령을 실행하게 되어 시스템에 장애를 일으킬 위험이 있다.

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 악성 스크립트 업로드 시도

위 그림 3‑22을 보면 Q&A 게시판에 bxsshunter에서 제공하는 스크립트를 삽입하여 게시글 업로드를 시도하고 있다.

텍스트, 스크린샷, 소프트웨어, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 악성 스크립트 업로드 성공

위 그림 3‑23를 보면 악성 스크립트가 필터링 되지 않고 업로드에 성공했다.

텍스트, 소프트웨어, 컴퓨터 아이콘, 웹 페이지이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 탈취한 사용자 정보 리스트

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 탈취한 사용자 정보

위 그림 3‑24, 그림 3‑25을 보면 bxsshunter 사이트에서 게시글에 접근한 사용자의 정보를 확인할 수 있다. 이렇게 획득한 화면 스크린샷, 쿠키, 세션 ID, 접속 IP 등을 이용하여 개인정보 유출 등의 피해가 발생할 가능성이 높다.

### XST, TRACE 옵션 허용 여부

웹 서버에서 TRACE 메소드가 활성화되어 있어 XST(Cross-Site Tracing) 공격이 가능할 경우 발생할 수 있는 취약점이다. 공격자는 XSS과 결합하여 피해자의 브라우저에서 실행되는 악성 스크립트를 이용해 쿠키 값을 추출한다. 이를 통해 세션 하이재킹이나 CSRF 공격을 수행하여 계정 탈취, 정보 유출 등 보안 사고가 발생할 위험이 높아진다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ HTTP 메서드로 TRACE 사용

위 그림 3‑26를 보면 HTTP 메서드로 TRACE가 사용되고 있다.

## 관리자 페이지 추측

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 관리자 페이지  추측 | BP-014 | 페이지 내 관리자 페이지 링크 여부 |
| BP-015 | 관리자 페이지 접근 여부 |

표 ‑ 관리자 페이지 추측 점검 항목

### 페이지 내 관리자 페이지 링크 여부

관리자 페이지로 이동할 수 있는 링크가 일반 사용자 페이지에 노출되어 있을 경우 발생할 수 있는 취약점이다. 공격자가 별도의 인증 절차 없이 관리자 페이지에 접근이 가능하거나, 약한 인증 방식이 적용되면 쉽게 접속하여 중요 설정 변경 등의 보안 사고로 이어질 수 있다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 관리자 페이지 링크 위치

위 그림 3‑27를 보면 좌측 상단에 위치한 Red를 통해 관리자 페이지로 접근할 수 있다. 관리자 계정 정보는 3.3.2에서 획득했기 때문에 인증에 성공하여 관리자 페이지 내부로 접속할 수 있다.

스크린샷, 텍스트, 디자인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ OyesMall 관리자 페이지

위 그림 3‑28은 Red를 통해 접근한 관리자 페이지 로그인 화면이다.

### 관리자 페이지 접근 여부

웹 어플리케이션에서 별도의 보안 조치 없이 관리자 페이지에 직접 접근할 수 있을 경우 발생할 수 있는 취약점이다. 공격자가 자동화된 스캔 도구를 활용하여 관리자 페이지를 통한 주요 정보 접근이 가능해진다.

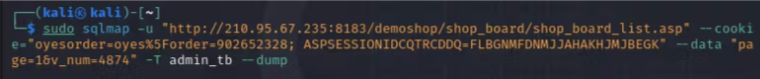


그림 ‑ sqlmap을 이용하여 테이블 정보 조회

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ sqlmap 결과

위 그림 3‑29, 그림 3‑30을 보면 관리자 ID, PW가 admin, admin인 것이 확인된다. 이렇게 취득한 계정 정보로 쉽게 관리자 페이지에 접속하여 관리자 권한을 획득할 수 있다.

## 페이지 내 중요 정보 노출

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 페이지 내 중요  정보 노출 | BP-016 | 중요 개인정보 노출 여부 |
| BP-017 | 쿠키 값 내 중요정보 유출 여부 |
| BP-019 | 중요정보 평문 전송 |

표 ‑ 페이지내 중요 정보 노출 점검 항목

### 중요 개인정보 노출 여부

사용자의 개인정보가 암호화되지 않은 상태로 노출될 경우 발생할 수 있는 취약점이다. 공격자는 단순한 웹 크롤링이나 브라우저 개발자 도구를 이용해 노출된 개인정보를 쉽게 획득할 수 있으며, 이를 통해 피싱 공격, 금융 사기 등의 보안 사고로 이어질 가능성이 높다. 특히, 로그 파일이나 HTML 소스코드 내에 개인정보가 포함된 경우, 검색 엔진을 통해 무단으로 수집될 위험도 존재한다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 회원정보 관리 페이지 개발자 모드

위 그림 3‑31를 보면 회원정보 관리 페이지에서 F12 개발자 모드로 확인하면 중요 개인 정보가 평문으로 노출된다.

### 쿠키 값 내 중요정보 유출 여부

쿠키 값에 민감한 정보(사용자 인증 토큰, 세션 ID 등)가 암호화되지 않은 채 저장되거나 전송될 경우 발생할 수 있는 취약점이다. 공격자가 네트워크를 가로채거나 XSS 공격을 통해 탈취한 쿠키 값으로 정보 유출, 계정 탈취 등이 발생할 수 있다.

**텍스트, 스크린샷, 소프트웨어, 웹 페이지이(가) 표시된 사진

자동 생성된 설명**

그림 ‑ 쿠키 값 내 중요 정보 노출

위 그림 3‑32을 보면 세션 ID, 로그인 정보가 평문으로 노출되어 전송된다. 중요 데이터를 전송할 때는 반드시 암호화를 하여 하이재킹 위험을 줄일 필요가 있다.

### 중요정보 평문 전송

중요한 정보가 암호화 없이 평문으로 전송될 경우 발생할 수 있는 취약점이다. 공격자는 중간자 공격을 통해 사용자의 로그인 정보, 결제 정보 등을 탈취할 수 있다. HTTPS 대신 HTTP를 사용하는 경우, 데이터가 암호화되지 않아서 네트워크를 통해 전송되는 동안 노출될 위험이 크다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 중요정보 평문 전송

위 그림 3‑33은 로그인 요청 시 비밀번호 및 개인정보를 평문으로 전송하는 모습이다. 암호화 없이 그대로 전송되므로 중간자 공격에 취약하다. 민감한 정보 및 개인 정보는 암호화하여 전송하고, 데이터베이스 저장 시에도 해싱 또는 암호화 기법을 적용하는 것이 필요하다.

## 파일 다운로드 취약점

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 파일 다운로드  취약점 | BP-023 | 입력 값 검증 미흡으로 파일 다운로드 공격 여부 |

표 ‑ 파일 다운로드 취약점 점검 항목

### 입력 값 검증 미흡으로 파일 다운로드 공격 여부

파일 다운로드 기능에서 입력 값에 대한 검증이 제대로 이루어지지 않을 경우 발생할 수 있는 취약점이다. 공격자가 서버에 저장된 시스템 관련 파일을 다운로드 받을 수 있다. 이로 인해 원격 코드 실행, 시스템 침해, 서버 손상 등의 피해가 발생할 수 있다.

텍스트, 폰트, 소프트웨어, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 사용자가 업로드한 파일

Burp Suite에서 Intercept on 상태로 위 그림 3‑34에 보이는 shell(1).cer 파일을 다운로드 시도한다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 다운로드 시 입력 값

Burp Suite에서 그림 3‑35과 같이 다운로드 시 입력값을 확인할 수 있다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 시스템 파일 다운로드 시도

시스템 파일 shop\_download.asp을 다운로드 받기 위하여 Burp Suite에서 입력값을 그림 3‑36처럼 변경한다.

텍스트, 라인, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 시스템 파일 다운로드 성공

Intercept off하면 그림 3‑37와 같이 시스템 파일이 다운로드된다.

## 파일 업로드 취약점

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 파일 업로드  취약점 | BP-024 | 입력 값 검증 미흡으로 파일 업로드 공격 여부 |

표 ‑ 파일 업로드 취약점 점검 항목

### 입력 값 검증 미흡으로 파일 업로드 공격 여부

파일 업로드 기능에서 입력 값에 대한 검증이 제대로 이루어지지 않을 경우 발생할 수 있는 취약점이다. 공격자가 업로드할 파일의 형식이나 확장자를 조작하여 악성 파일을 업로드할 수 있다. 이후 이를 다운로드하고 실행시키면 원격 코드 실행, 시스템 침해, 서버 손상 등으로 이어질 수 있다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 악성 파일 업로드 시도

위 그림 3‑38을 보면 악성 파일의 확장자를 업로드가 가능한 jpg로 변경하여 정상 파일로 위장했다. Burp Suite에서 Intercept on 상태로 Submit을 시도한다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 업로드 시 입력 값

Burp Suite에서 그림 3‑39과 같이 업로드 시 입력값을 확인할 수 있다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 파일 확장자 변경

Burp Suite에서 입력값을 그림 3‑40처럼 변경하여 정상파일로 위장한 악성 파일을 원래 확장자로 복구한다.

텍스트, 소프트웨어, 폰트, 아쿠아이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 악성 파일 업로드 성공

Intercept off하면 그림 3‑41와 같이 시스템 파일이 업로드된다

## SQL Injection

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| SQL Injection | BP-027 | SQL Injection 허용 여부 |

표 ‑ SQL Injection 점검 항목

### SQL Injection 허용 여부

사용자가 입력한 값이 직접 SQL 쿼리문에 포함될 경우 발생할 수 있는 취약점이다. 공격자가 이를 조작하여 원래 의도된 커리 외의 다른 명령을 실행할 수 있다. 데이터베이스에서 민감한 정보를 탈취하거나, 데이터베이스의 내용을 변경하는 등의 피해가 발생할 가능성이 높다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 오류 발생을 위한 특수문자 전송

위 그림 3‑42을 보면 SQL 문법 오류를 발생시키기 위해 의도적으로 특수문자 전송을 시도하고 있다.

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 특수문자로 인한 오류 발생

위 그림 3‑43처럼 데이터베이스가 반환하는 오류 메시지를 분석함으로써 데이터베이스 구조, 테이블명 등을 파악할 수 있다. 좀 더 쉽게 파악하기 위해서 sqlmap을 사용했다.

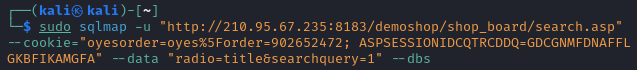


그림 ‑ SQL Injection 취약점을 이용한 데이터베이스 조사

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 서버에서 사용 중인 데이터베이스 목록

위 그림 3‑44, 그림 3‑45은 sqlmap을 활용하여 서버에서 사용 중인 데이터베이스 목록을 조회한 모습이다.

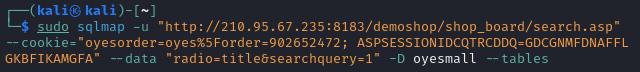


그림 ‑ SQL Injection 취약점을 이용한 oyesmall 테이블 조사

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 데이터베이스 oyesmall의 테이블 목록

위 그림 3‑46, 그림 3‑47은 sqlmap을 이용해 데이터베이스 oyesmall 내 테이블 목록을 확인한 모습이다.

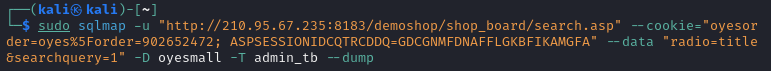


그림 ‑ SQL Injection 취약점을 이용한 admin\_tb의 데이터 조사

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 데이터베이스 oyesmall의 테이블 admin\_tb의 데이터

위 그림 3‑48, 그림 3‑49은 sqlmap을 사용하여 데이터베이스 oyesmall 내 테이블 admin\_tb의 데이터를 확인한 모습이다. 이를 통해 ID가 admin, PW가 admin인 관리자 계정 정보를 확인했다. SQL이 취약할 경우, 공격자가 SQL Injection 탐지 도구를 활용하면 정보가 쉽게 유출될 수 있다.

## 최신 취약점 미패치

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 최신 취약점  미패치 | BP-028 | 보안에 취약한 오래된 어플리케이션 사용 여부 |

표 ‑ 최신 취약점 미패치 점검 항목

### 보안에 취약한 오래된 어플리케이션 사용 여부

오래된 어플리케이션이나 시스템에서 최신 보안 패치가 적용되지 않을 경우 발생할 수 있는 취약점이다. 보안 패치가 미적용된 시스템은 기존에 발견된 취약점들이 여전히 존재하기 때문에 공격자가 이를 활용할 수 있다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 취약한 웹 서버 소프트웨어 사용

위 그림 3‑50을 보면 보안에 취약한 Microsoft-IIS/6.0을 사용하고 있다. 지원이 종료되어 보안 업데이트가 제공되지 않기 때문에, 새로운 취약점이 발견되어도 패치가 불가능하다.

## 부적절한 서버 설정

|  |  |  |
| --- | --- | --- |
| 분류 | 코드 | 점검 항목 |
| 부적절한  서버 설정 | BP-029 | 서버 보안 설정 여부 |

표 ‑ 부적절한 서버 설정 점검 항목

### 서버 보안 설정 여부

서버 보안 설정이 약하거나 불필요한 서비스가 활성화되어 있는 경우 발생할 수 있는 취약점이다. 공격자가 서버에 대한 불법적인 접근을 시도하거나 시스템을 악용할 수 있다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

그림 ‑ 서버 보안 설정 미흡

위 그림 3‑51를 보면 X-Frame-Option 헤더와 X-Content-Type-Options 헤더가 설정되지 않았다. X-Frame-Option 헤더가 설정되지 않으면 iframe을 삽입하여 클릭재킹 공격이 가능해진다. X-Content-Type-Options 헤더가 설정되지 않으면 MIME 타입을 자동으로 결정하여 실행할 수 없어 XSS 공격이 가능해진다.

# 취약점 대응방안

취약점 대응방안은 안전행정부와 한국인터넷진흥원에서 발행한 주요정보통신기반시설 기술적 취약점 분석 평가 방법 상세가이드를 기반으로 작성했다. 대응방안은 상세 수행 내역에서 도출된 취약점과 순서는 동일하며, 관리적, 기술적, 물리적 그리고 소스코드 관점의 보안을 설명한다.

## 계정정보 추측 및 대입

### 취약한 패스워드 설정 여부

취약한 패스워드를 사용할 경우 Brute Force (무차별 대입 공격), Dictionary Attack (사전 공격), Rainbow Table Attack (레인보우 테이블 공격) 등에 쉽게 노출될 위험이 있다. 이는 대/소문자, 특수문자를 포함한 8자 이상의 비밀번호를 사용하여 방지할 수 있다. 비밀번호 검증은 클라이언트와 서버 양쪽에서 적용하여 우회가 불가능하도록 해야 한다. 또한, 데이터베이스에 저장할 때 해시 함수를 적용하여 평문 저장을 방지하며 데이터베이스 정보가 유출되더라도 사용자의 개인정보가 보호될 수 있도록 해야 한다.

### 어플리케이션/장비 기본 패스워드 설정 여부

기본 패스워드는 어플리케이션, 네트워크 장비, 서버, 데이터베이스 등의 초기 설정 계정 정보를 의미한다. 공격자는 기본 패스워드를 이용해 시스템에 무단 접근하여 데이터 유출, 시스템 제어, 악성코드 설치 등의 공격을 수행할 수 있다. 따라서 기본 패스워드는 보안 정책에 따라 강력한 패스워드로 변경하고, 사용하지 않는 기본 계정은 비활성화 또는 삭제가 필요하다.

## 인증 우회

### 쿠키 재사용 (Replay Attack) 여부

쿠키에 저장된 세션값이 재사용될 경우, 공격자가 탈취한 쿠키를 이용해 사용자의 인증을 우회할 수 있다. 이를 방지하기 위해 서버에서는 HttpOnly 속성을 적용해 자바스크립트에서 쿠키 접근을 차단하고, Secure 속성을 설정해 HTTPS 환경에서만 쿠키가 전송되도록 해야 한다. 또한 쿠키의 유효기간을 설정해 일정 시간이 지나면 세션을 자동 만료시키고, 사용자가 재로그인할 때마다 새로운 쿠키값을 발급해야 한다. 보안성을 높이기 위해 쿠키 대신 Server Side Session방식을 사용하는 것도 좋은 방법이며, 중요 정보는 AES, SEED와 같은 안전한 암호화 알고리즘을 적용해 보호해야 한다.

### 클라이언트 인증 우회 여부 (Javascript 우회)

클라이언트 사이드에서만 인증을 하는 경우 우회가 가능하여 보안 취약점이 발생한다. OyesMall 사이트의 경우 회원가입시 인증 우회가 가능한 취약점이 발견되었으며, 이를 해결하기 위해 서버에서도 전달된 값을 강제적으로 검증하는 코드가 필요하다. 또한, 브라우저에서 페이지 소스를 확인하여 인증 로직을 분석할 수 없도록 클라이언트 코드를 난독화하는 등의 추가적인 보안 조치가 필요하다.

## 파라미터 조작

### 필드 값 조작에 따른 검증 여부

회원가입 시 클라이언트에서 필수 입력 필드를 Burp Suite를 활용하여 제거할 수 있는 취약점이 발견되었다. 이를 방지하기 위해 서버는 클라이언트에서 필수 필드가 누락되었더라도 서버 측에서 다시 한 번 검증하고, 검증되지 않은 요청은 거부할 수 있도록 해야 한다.

## XSS (CSRF) 취약점

### 악의적인 스크립트 필터링 여부 (POST 메소드)

XSS 취약점이 허용될 경우, 공격자는 다른 사용자의 쿠키탈취, 피싱 사이트 유도 등 다양한 공격을 수행하게 할 수 있다. 이를 방지하기 위해 <script></script> 태그, 이벤트 핸들러 등을 차단하고, 특수문자를 이스케이프 처리해야 한다. 또한, 화이트리스트 기반의 입력값 검증을 통해 악성 코드 삽입을 예방할 수 있다.

### XST, TRACE 옵션 허용 여부

TRACE 메서드를 허용할 경우, XST 공격에 취약할 수 있다. 공격자는 XST를 이용하여 사용자의 쿠키, 세션 정보를 탈취할 수 있기 때문에, IIS 웹 서버 관리 콘솔에서 TRACE 메서드를 비활성화하고 WAF에서 TRACE메서드를 필터링 하도록 설정해야 한다.

## 관리자 페이지 추측

### 페이지 내 관리자 페이지 링크 여부

관리자 페이지 URL을 동적으로 생성하여 고유하고 예측 불가능하게 만들어 공격자가 관리자페이지의 위치를 추측하거나 접근하는 것을 어렵게 만든다. 또한, 관리자 페이지에 대한 접근을 특정 IP 주소 범위(회사 내부 네트워크)로 제한하는 것도 효과적이다.

### 관리자 페이지 접근 여부

관리자 계정이 탈취당할 경우, 공격자는 관리자 페이지에 접근하여 권한 상승을 통해 악의적인 행동을 할 수 있다. 이를 방지하기 위해 임의의 URL 경로를 사용하여 외부에 쉽게 노출되지 않도록 해야 한다. 또한, IP를 화이트 리스트 방식으로 설정해야 한다. 고정 IP에서만 접근을 허용하여 외부 접근을 차단한다. 보안을 강화하기 위해 추가적인 2단계 인증 절차를 요구하는 것도 필요하다.

## 페이지 내 중요 정보 노출

### 중요 개인정보 노출 여부

주민등록번호와 같은 민감한 정보를 처리할 때는 반드시 암호화하여 저장해야 한다. 이를 통해 데이터베이스가 유출되더라도 개인정보가 보호될 수 있다. 또한, HTML 소스에서 개인정보가 직접적으로 노출되지 않도록 해야하며, 개인정보는 백엔드에서만 처리하고, 프론트엔드에서는 사용하지 않도록 해야 한다. 개인정보를 마스킹 처리하는 것도 중요한 보안 조치이다. 이러한 방식은 불필요하게 노출되는 것을 방지하고, 보안을 강화하는 데 도움을 준다.

### 쿠키 값 내 중요정보 유출 여부

쿠키를 통한 정보 전송 시 HTTPS를 사용하여 암호화된 연결을 통해 데이터를 전송해야 한다. 또한, 쿠키에 httpOnly 플래그를 설정하면 자바스크립트에서 해당 쿠키에 접근할 수 없게 되어 XSS 공격을 차단할 수 있다. 세션 타임아웃을 설정하는 것도 도움이 된다. 세션이 일정 시간 동안 활동이 없으면 자동으로 로그아웃 되도록 설정하여 세션 하이재킹 위험을 줄일 수 있다.

### 중요정보 평문 전송

중요 정보는 암호화된 채널을 통해 전송되어야 한다. SSL(TLS)과 같은 암호화된 통신 프로토콜을 사용하여 데이터를 보호해야 한다. 또한, 전송되는 정보의 불필요한 노출을 막기 위해 암호화된 형태로 처리해야 한다.

## 파일 다운로드 취약점

### 입력 값 검증 미흡으로 파일 다운로드 공격 여부

입력 값 검증이 미흡하면 공격자가 파일 다운로드 경로를 조작하여 중요 파일을 다운로드 할 수 있는 취약점이 발생한다. 다운로드 경로에서 상위 디렉터리 참조(../)를 사용하여 상위 디렉터리로 접근할 수 있다. 이를 방지하기 위해 입력 값에서 상위 디렉터리 참조(../)를 필터링하여 경로 탐색 공격을 차단해야 한다. 또한, 사용자가 다운로드할 수 있는 파일 형식을 제한하여 안전한 파일만 다운로드하도록 한다.

## 파일 업로드 취약점

### 입력 값 검증 미흡으로 파일 업로드 공격 여부

입력 값 검증이 미흡하면 공격자가 악성 파일을 서버에 업로드하고 이를 실행시킬 수 있는 취약점이 발생한다. 파일이 서버에 업로드될 디렉터리를 제한하여 파일의 권한을 설정한다. 또한, 업로드된 파일이 악성 코드를 포함하고 있는지 검사한다.

## SQL Injection

### SQL Injection 허용 여부

SQL Injection을 허용할 경우 공격자는 악의적인 SQL 쿼리를 삽입하여 데이터베이스에 접근할 수 있다. 이를 방지하기 위해 사용자 입력을 철저하게 검증하고, 허용된 입력만 처리해야 한다. 오류 메시지는 최소화하여 시스템의 구조나 데이터베이스 정보를 파악하지 못하도록 해야 한다. SQLMAP을 방지하기 위해서는 WAF를 사용하여 헤더를 차단하고, 화이트 리스트 기반으로 특정 키워드 패턴을 차단해야 한다. 서버에서 prepared statements를 사용하여 SQL 쿼리와 데이터를 분리하여 방지할 수 있다.

## 최신 취약점 미패치

### 보안에 취약한 오래된 어플리케이션 사용 여부

보안에 취약한 오래된 어플리케이션을 사용할 경우, 공격자는 이미 알려진 취약점을 악용해 시스템에 침투할 수 있다. 이를 방지하기 위해 항상 최신 버전의 소프트웨어를 사용하고, 정기적으로 업데이트 및 패치를 적용해야 한다.

## 부적절한 서버 설정

### 서버 보안 설정 여부

서버 보안을 강화하려면, 클릭재킹 공격을 방지하기 위해 X-Frame-Options 헤더를 설정하고, 스크립트 삽입 공격(XSS)을 예방하기 위해 X-Content-Type-Options 헤더를 설정해야 한다. 또한, 쿠키 탈취를 방지하기 위해 쿠키의 httpOnly 플래그를 활성화해야 한다.