# 『2023년 야누스(yANUs) 프로젝트』
# OWASP Top 10 기반 웹 취약점 보안 체크리스트 홈페이지
# 개발계획서

2023. 5. 12

| | | |
|---|---|---|
| 제출자 | 학교명 | 국립안동대학교 야누스 보안팀 |
| | 작성자 | 성명 : 김선혁, 안수윤, 김선민, 금정희 |

# 프로젝트 계획 요약

| 과 제 명 | OWASP Top10 기반 웹 취약점 보안 체크리스트 홈페이지 | | |
|---|---|---|---|
| 주 관 기 관 | 국립안동대학교 야누스 | 과제책임자 | |
| 참 여 기 업 | | 대 표 자 | |
| 개 발 분 야 | □ 앱 ■웹 | □ 기타 ( ) | |
| 개 발 기 간 | 2023년 5월 11일 ~ 2023년 12월 31일 (9개월) | | |

## 1. 기술 개발 목표

- 보안 전문가와 개발자들에게 웹 사이트의 보안 상태를 점검하고 개선할 수 있도록 OWASP Top 10 기반 취약점, 효과적 진단 방법, 대응 방안(보안 솔루션)에 대한 정보를 제공하는 것

## 2. 기술 개발 내용 및 방법

- Django 기반 웹 서비스 개발 (Python 기반 웹 프레임워크)
- OWASP Top 10 취약점에 대한 공격기법, 진단방법, 대응방안 연구
- OWASP Top 10 취약점을 기반으로 한 웹 사이트 보안 체크리스트를 개발
- 관련 정보 제공하는 홈페이지 구축


▲ 웹 서비스 구성도

## 3. 기술 개발시 예상효과 (기대효과)

- 보다 간편하고 체계적인 방법으로 웹 개발자들이 웹 사이트의 보안 상태를 점검하고 개선 가능

- 2023 최신 트렌드를 반영한 웹 보안 취약점을 사전에 예방하고 대응

- 보안 솔루션과 서비스 정보 제공으로 인해 개발자들이 보다 신속하고 효율적인 보안 조치를 취함

- 웹 보안 이슈를 보고하고 공유할 수 있는 커뮤니티 형성으로 보안 관련 지식과 정보를 널리 공유할 수 있음

# 프로젝트 수행자 정보

| | | | | |
|---|---|---|---|---|
| 팀 명 | | 야누스 보안팀 | 홈 페 이 지 | |
| 설립년월일 | | 2023년 3월 1일 | 참가자수(명) | 4 |
| 사업자등록번호 | | | 법인등록번호 | |
| 주 소 | | 경상북도 안동시 경동로 1375 | | |
| 대표자 | 성 명 | 김선혁 | 생년월일 | 2000.02.25 |
| | 전 화 | - | 핸 드 폰 | 010-8989-9460 |
| | 이메일 | hako5460@naver.com | 팩 스 | - |
| | 주 소 | - | | |
| 참가자 | 성 명 | 안수윤 | 생년월일 | 2004.10.11 |
| | 전 화 | - | 핸 드 폰 | 010-9386-6022 |
| | 이메일 | suyun6022@gmail.com | 팩 스 | - |
| | 주 소 | - | | |
| 참가자 | 성 명 | 김선민 | 생년월일 | 2004.01.09 |
| | 전 화 | - | 핸 드 폰 | 010-7250-3113 |
| | 이메일 | sunmini6077@gmail.com | 팩 스 | - |
| | 주 소 | - | | |
| 참가자 | 성 명 | 금정희 | 생년월일 | 1999.07.01 |
| | 전 화 | - | 핸 드 폰 | 010-9345-5349 |
| | 이메일 | dmsehd17@naver.com | 팩 스 | - |
| | 주 소 | - | | |
| 참가자 | 성 명 | - | 생년월일 | - |
| | 전 화 | - | 핸 드 폰 | - |
| | 이메일 | - | 팩 스 | - |
| | 주 소 | - | | |

# 1. 기술개발의 개요

## 1-1. 개발대상기술(또는 제품)의 개요 및 필요성

### ① 개요

OWASP Top 10에 명시된 취약점들을 기반으로 작성된 보안 체크리스트 홈페이지를 제작하고, 체크리스트 목록을 기반으로 웹 개발자들이 개발한 사이트의 취약점을 파악하고 진단 방법, 대응 방안을 최대한 자세하게 카테고리 별로 묶어 제시하여 웹 보안에 대한 이해와 인식을 높이는 데에 큰 도움을 기여하는 홈페이지 제작


### ② 목적

- 보안 체크리스트 홈페이지를 작성하여 웹 사이트의 취약점을 파악하고 대응 방안을 제시함

- 홈페이지의 체크리스트를 통해 웹 개발자는 자신이 개발한 웹 사이트의 보안 상태를 점검 하고, 대응 방안을 제공받는 데에 사용함

- 웹 개발자들은 보안에 대한 이해와 인식을 높이고, 보안 전문가들은 웹 사이트의 취약점을 검증하고 보완하는 데에 도움이 되는 정보들을 수집할 수 있음

- 웹 사이트의 신뢰성과 안정성을 높이고, 보안 위협에 대응할 수 있는 능력을 키우는 데에 기여할 수 있음

## 1-2. 기술 개발의 필요성

- 웹 사이트는 많은 정보와 데이터를 다루기 때문에 보안 위협에 노출될 가능성이 매우 높음또한 웹 애플리케이션 보안은 매우 중요한 이슈 중 하나로 인식되고 있음

- 이에 대한 이해와 적극적인 대응은 웹 사이트의 개발에 있어 신뢰성과 안정성을 보장하는 데에 필수적인 요소

- OWASP Top 10은 웹 어플리케이션에서 가장 많이 발생하는 취약점들을 정리한 목록으로, 3,4년에 한번씩 업데이트되어 최신 보안 이슈를 반영하고, 이 목록은 웹 개발자나 보안 전문가들이 보다 안전한 웹 사이트를 구축하고 운영할 수 있도록 도움이 되는 정보들을 제공함

## 1-3. 기술 개발시 예상효과 및 활용방안

- 최신 트렌드를 반영한 웹 취약점 탐지 및 대응 강화

- 보안 커뮤니티 형성을 통한 보안 지식, 최신 보안 이슈를 공유하고 협력

- 보안 업데이트 및 정책 시행을 강화하여 웹 사이트 보안을 지속 유지

- 개발자들의 자체적인 웹 사이트 보안 점검 수행

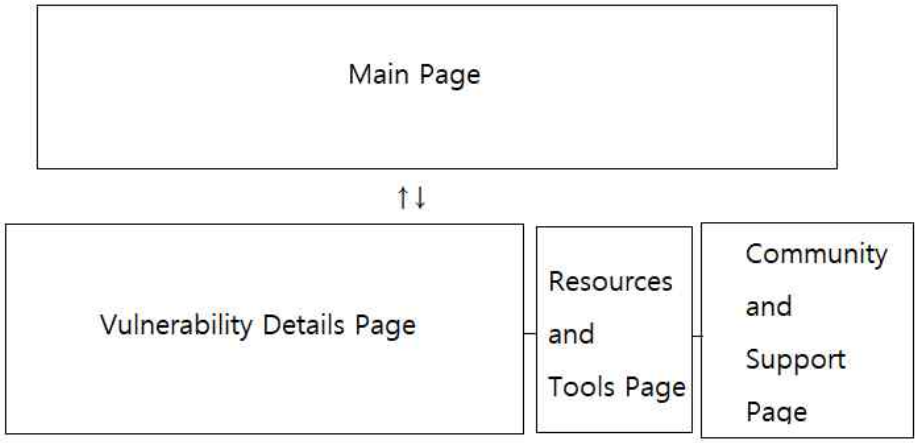
- 보안 체크리스트를 기반으로 한 보안 컨설팅 및 서비스 제공

# 2. 기술개발의 목표 및 개발내용

## 2-1. 개발목표 및 개발내용

○ **최종목표** : OWASP Top 10에 명시된 웹 취약점들을 기반으로 한 보안 체크리스트 홈페이지 제작

1) 개발 기술 시스템 구성




▲서비스 구성도

### (1) 메인 페이지

**1-1. 보안 체크리스트 홈페이지의 개발 목적과 개요를 간략히 설명하는 섹션**

- 사용자들에게 웹 개발 시 보안 체크리스트를 참고해야 하는 목적과 중요성을 전달하는 역할 수행

**1-2. OWASP Top 10 취약점 목록을 요약하여 표시하는 섹션**

- 각 취약점의 이름 + 아이콘을 나열하여 사용자들이 빠르게 최신 트렌드 웹 보안 취약점 파악 가능

**1-3. 보안 체크리스트 사용법 및 안내 섹션**

- 사용자들이 체크리스트를 어떻게 활용해야 하는지 & 각 취약점에 대한 진단 방법과 대응 방안을 어떻게 참고할 수 있는지를 상세히 안내

## (2) 보안 체크리스트 취약점 상세 페이지

### 2-1. 각 취약점에 대한 상세 설명과 원인을 제공하는 섹션

- 해당 취약점의 특징, 발생 원인, 공격자가 어떤 방법으로 취약점을 악용할 수 있는지 등을 자세히 설명

### 2-2. 공격 기법에 대한 설명과 예시를 보여주는 섹션

- 실제 공격 시나리오 또는 예시를 제공하여 취약점에 대한 이해 도움

### 2-3. 진단 방법과 진단 툴 안내 섹션

- 취약점을 진단하기 위한 방법과 사용 가능한 소프트웨어를 제시, 이를 참고하여 사용자들이 자체적으로 웹 사이트의 취약점 진단 가능

### 2-4. 대응 방안과 보완 조치를 제시하는 섹션

- 해당 취약점에 대한 대응 방안과 보완 조치 안내, 보안 솔루션 제시

## (3) 자료 및 도구 페이지

### 3-1. 보안 관련 자료, 리소스 제공 섹션

- 웹 보안에 관련된 가이드, 권장 사항, 보안 블로그 등의 자료를 제공

### 3-2. 웹 취약점 진단 및 대응을 위한 Tool을 안내하는 섹션

- 취약점 스캐너, 보안 테스트 도구 등을 소개하여 사용자들이 보다 쉽게 보안 검사를 수행

### 3-3. 추가 자료 섹션

- 관련 보안 자료나 레퍼런스, 연구 논문 등 제공

## (4) 커뮤니티 및 지원 페이지

### 4-1. 정보 및 의견을 공유할 수 있는 커뮤니티 섹션

- 사용자들이 정보를 공유하고 질문을 할 수 있는 공간 제공

### 4-2. 지원 및 문의를 위한 연락처 및 양식 제공

- 사용자들이 질문 및 문의를 할 수 있는 연락처 양식 제공

### 4-3. FAQ 섹션

- 보안 체크리스트 진단 중 자주 묻는 질문과 그에 대한 답변 제공

## ○ OWASP Top 10, Open Web Application Security Project

- OWASP Top 10은 웹 어플리케이션 보안에서 가장 많이 발생하는 취약점들을 정리한 목록으로, OWASP Top 10은 3,4년에 한번씩 정기적으로 업데이트되며, 2021년 기준으로 다음과 같은 취약점이 포함되어 있다.

| |
|---|
| 1. 인젝션 (Injection) |
| 2. 취약한 인증 및 세션 관리 (Broken Authentication and Session Management) |
| 3. 민감한 데이터 노출 (Sensitive Data Exposure) |
| 4. XML 외부 엔티티 주입 (XML External Entity Injection, XXE) |
| 5. 잘못된 접근 제어 (Broken Access Control) |
| 6. 보안 설정의 부재, 보안 구성의 오류 (Security Misconfiguration) |
| 7. 크로스 사이트 스크립팅(XSS, Cross-Site Scripting) |
| 8. 부적절한 역직렬화(Insecure Deserialization): |
| 9. 컴포넌트에 있는 알려진 취약점(Using Components with Known Vulnerabilities): |
| 10. 불충분한 로깅 및 모니터링(Insufficient Logging & Monitoring) |

### 1. 인젝션 (Injection):

원인: 사용자 입력이 필터링이나 검증 없이 직접 쿼리나 명령에 포함되어 실행될때 발생한다.
공격 기법: 공격자가 악성 코드를 입력하여 데이터베이스를 조작하거나 시스템에 접근할 수 있다.
진단 방법: 웹 애플리케이션에서 입력 필드를 찾아 입력 값에 SQL 구문을 삽입한 뒤, 쿼리 실행 결과에 대한 오류 메세지를 확인한다.
대응 방안: 사용자 입력값을 검증하고, 파라미터화된 쿼리를 사용하거나 안전한 API를 사용해야 한다.

### 2. 취약한 인증(Broken Authentication):

원인: 인증 및 세션 관리 기능이 제대로 구현되지 않아 공격자가 계정 정보를 탈취하거나 세션을 탈취할 수 있다.
공격 기법: 브루트 포스 공격, 세션 피싱, 쿠키 탈취 등을 통해 인증 정보를 획득한다.
진단 방법: 로그인 기능을 이용하여 로그인 성공 시 발급되는 쿠키나 세션 ID를 탈취하고, 다시 접속해 세션이 유효한지 확인한다.
대응 방안: 인증 및 세션 관리 기능을 강화하고, 2단계 인증, 비밀번호 정책, 세션 만료 등을 적용한다.

### 3. 민감한 데이터 노출(Sensitive Data Exposure):

원인: 웹 애플리케이션에서 민감한 데이터가 암호화되지 않거나 저장 및 전송 중 노출되어 있을 때 발생하는 취약점이다.
공격 기법: 다양한 방법으로 데이터를 탈취하거나 전송 중인 데이터를 가로채어 정보를 획득한다.
진단 방법: 웹 애플리케이션에서 HTTPS 프로토콜이 적용되지 않은 페이지를 찾아 개발자 도구로 데이터 전송을 캡처한다.
대응 방안: 민감한 데이터를 암호화하고, 저장 및 전송 중인 데이터를 안전하게 보호한다.

### 4. XML 외부 엔티티 주입 (XXE, XML External Entities):

원인: XML 처리기에서 외부 엔티티를 불러오는 기능이 활성화되어 있어 공격자가 외부 시스템에 접근할 수 있는 취약점이다.
공격 기법: XML 문서에 악성 외부 엔티티를 삽입하여 시스템 파일에 접근하거나 원격 코드 실행을 시도한다.
진단 방법: XML 파서에서 DTD(Entity) 태그가 사용되는지 확인하고, 악성 XML 데이터를 입력하여 서버 응답 결과를 확인한다.
대응 방안: XML 처리기에서 외부 엔티티를 불러오는 기능을 비활성화하고, 안전한 XML 파서를 사용한다.

### 5. 잘못된 접근 제어(Broken Access Control):

원인: 사용자의 권한을 제대로 확인하지 않아 공격자가 허가되지 않은 기능이나 데이터에 접근할 수 있다.
공격 기법: URL 조작, 권한 변경 요청, 다른 사용자의 권한을 이용해 정보를 탈취하거나 기능을 실행한다.
진단 방법: 로그인 후, URL 수정이나 요청 수정 등을 통해 권한 검증이 우회되는지 확인한다.
대응 방안: 사용자의 권한을 철저하게 검증하고, 접근 제어 목록(ACL) 및 역할 기반 접근 제어(RBAC)를 구현한다.

### 6. 보안 구성 오류(Security Misconfiguration):

원인: 웹 애플리케이션이나 서버, 데이터베이스 등의 구성이 잘못되어 취약점이 발생한다.
공격 기법: 기본 구성, 미사용 기능, 취약한 기능 등을 이용해 시스템에 침입한다.
진단 방법: 서버의 헤더, 디렉토리, 파일 권한 등을 확인하여 보안 설정이 잘못되었는지 확인한다.
대응 방안: 보안 설정 가이드를 따라 구성을 최적화하고, 정기적으로 구성을 검토하여 보안을 유지한다.

**7. 크로스 사이트 스크립팅(XSS, Cross-Site Scripting):**
원인: 사용자 입력값이 적절하게 필터링되지 않아 공격자가 악성 스크립트를 삽입할 수 있다.
공격 기법: 악성 스크립트를 삽입하여 다른 사용자의 브라우저에서 실행되게 하여 정보를 탈취하거나 조작한다.
진단 방법: 웹 애플리케이션에서 입력 필드를 찾아 악성 스크립트를 삽입하여 취약점이 발생하는지 확인한다.
대응 방안: 사용자 입력값을 적절하게 검증하고, 출력 시에 안전한 인코딩을 사용한다.

**8. 부적절한 역직렬화(Insecure Deserialization):**
원인: 역직렬화 과정에서 검증되지 않은 데이터가 시스템에 주입되어 악용된다.
공격 기법: 변조된 직렬화 데이터를 역직렬화하여 원격 코드 실행이나 권한 상승 등을 시도한다.
진단 방법: 역직렬화 가능한 객체를 찾아 변조된 데이터를 전송하여 시스템에 영향을 끼치는지 확인한다.
대응 방안: 직렬화 데이터의 무결성을 확인하고, 역직렬화 과정에서 위험한 객체 생성을 방지한다.

**9. 컴포넌트에 있는 알려진 취약점(Using Components with Known Vulnerabilities):**
원인: 웹 애플리케이션에 사용된 외부 라이브러리나 컴포넌트에 취약점이 존재한다.
공격 기법: 알려진 취약점을 이용해 시스템에 침입하거나 데이터를 탈취한다.
진단 방법: 사용 중인 라이브러리나 컴포넌트가 보안 취약점을 포함하고 있는지 확인하고, 패치나 업데이트 여부를 확인한다.
대응 방안: 사용 중인 외부 라이브러리나 컴포넌트의 보안 취약점을 주기적으로 확인하고, 필요한 경우 업데이트나 패치를 적용한다.

**10. 불충분한 로깅 및 모니터링(Insufficient Logging & Monitoring):**
원인: 시스템에서 발생하는 이벤트에 대한 로깅이 미흡하거나, 로그를 실시간으로 모니터링하지 않아 공격이나 침입을 감지하지 못한다.
공격 기법: 로그 조작이나 로그 분석의 지연을 이용해 공격을 숨기거나 지속시킨다.
진단 방법: 로깅이나 모니터링이 미흡한 시스템에서 로그를 확인하고, 실시간 모니터링 여부를 확인한다.
대응 방안: 로깅 정책을 강화하고, 로그를 실시간으로 모니터링하여 이상 징후를 감지하고 대응할 수 있도록 한다.

2) 개발 목표

| | |
|---|---|
| **개발자** | - OWASP Top 10 취약점에 대한 인식과 이해<br>- 웹 애플리케이션 개발 시 웹 보안 취약점을 방지하는 데 도움 제공<br>- 보안 체크리스트를 활용하여 보안 검토 및 테스트를 수행 |
| **웹사이트 운영자** | - 웹 사이트의 보안 상태를 객관적으로 확인할 수 있는 도구 제공<br>- OWASP Top 10 웹 취약점에 대한 각 대응 방안을 제시하여 보안 강화에 도움<br>- 취약점을 식별하고 보완하여 데이터 유출, 공격 및 손실 예방 |
| **일반 사용자** | - 정보 보안에 대한 인식 제고<br>- 안전한 웹 사이트를 이용하여 개인정보와 금융 정보를 보호할 수 있도록 안내<br>- 신뢰할 수 있는 웹 사이트를 식별하는 능력 향상 |

## 2-2. 개발기술 활용 계획

- 사이트 보안 체크리스트를 활용하여 자체적인 웹 보안 점검 및 대응 수행 가능
- 홈페이지 내의 보안 커뮤니티를 통한 최신 보안 이슈와 보안 관련 정보를 공유하고 협력
- 보안 체크리스트를 기반으로 한 보안 컨설팅 및 서비스 제공
- 공통적인 보안 요구사항과 체크리스트를 활용하여 웹 산업간의 협력과 표준화 제시 가능

## 2-3. 제품개발 계획

| | | |
|---|---|---|
| **1. 착수단계** | 기획, 설계 | OWASP Top 10 각 취약점 공격 기법, 진단 방안, 대응 방안 연구 및 분석 |
| | | 홈페이지 개발 일정 계획, 보안 체크리스트 구성 설계 |
| **2. 개발수행** | FrontEnd 개발 | UI 디자인 및 구현 |
| | | 클라이언트 측 기능 개발 |
| | BackEnd 개발 | DB 설계 및 구축 |
| | | 서버 측 로직 및 기능 구현 |
| | DB | DB 테이블 설계 및 관리 |
| | 시스템 관리자 개발 | 사용자 인증 및 권한 관리 |
| | | 로그인, 회원가입 기능 구현 |
| | 보안서버 개발 | 취약점 분석 및 보고서 생성 기능 구현 |
| **3. 테스트** | 단위 테스트 | 각 모듈 기능 독립적 테스트 |
| | 통합 테스트 | 모듈 간 상호작용 테스트 |
| | 시스템 테스트 | 전체 시스템 테스트 및 버그 수정 |
| **4. 안정화** | 버그 수정 및 최적화 | 테스트를 통한 버그 수정, 성능 향상 목적 |
| | 보완, 보안 강화 | OWASP Top 10을 고려하여 보안 취약점을 분석하고 대응 |
| | 사용성 검토 | UX 개선을 위한 인터페이스, 기능 검토및 조정 |
| **5. 종료단계** | 시스템 배포 | 안정화 시키고 실제 서버에 배포 |
| | 유지 및 보수 | 지속적인 유지보수와 업데이트 계획 수립 |
| | 프로젝트 평가 및 보고서 작성 | 최종 결과물 평가후 개선점 도출하여 보고서 작성 |

## 2-4. 연구개발 추진일정

| 일련번호 | 세부 개발내용 | 담당자 | 세부 추진 일정 (8개월) | | | | | | | | | | | | | | | | | | | | | | 비 고 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 5 | | | 6 | | | 7 | | | 8 | | | 9 | | | 10 | | | 11 | | | 12 | | | |
| 1 | 기술스택 역량 추가 강화, 세부 프로젝트 일정 계획 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | OWASP Top 10 각 취약점 공격기법, 진단방안, 대응방안 연구 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | UI 디자인 및 DB, 웹사이트 구조 설계 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | FrontEnd & BackEnd 개발 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | 보안 커뮤니티 및 협업 기능 구축 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | 최적화 및 테스트 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | 배포 및 유지보수 준비, 진행 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | 보고서 작성 | 공동 | | | | | | | | | | | | | | | | | | | | | | | | |

## 2-5. 역할분담

| 수행 인원 | 주요 담당 업무 | 기술개발 비중(%) |
|---|---|---|
| **김선혁** | - 전체 시스템 설계<br>- 계획 수립 및 진행방향 설정<br>- OWASP Top 10 각 취약점 분석<br>- DB 설계 및 구축<br>- 테스트 및 평가 | 25% |
| **안수윤** | - 전체 시스템 설계<br>- OWASP Top 10 각 취약점 분석<br>- FrontEnd 개발<br>- 테스트 및 평가 | 25% |
| **김선민** | - 전체 시스템 설계<br>- OWASP Top 10 각 취약점 분석<br>- FrontEnd 개발<br>- 테스트 및 평가 | 25% |
| **금정희** | - 전체 시스템 설계<br>- OWASP Top 10 각 취약점 분석<br>- BackEnd 개발<br>- 테스트 및 평가 | 25% |
| **총 계** | OWASP Top 10 취약점 보안 체크리스트 홈페이지 구축 | 100% |

## 3. 기대효과

| 구 분 | 내 용 |
|---|---|
| **기술적 측면** | - 웹 사이트 보안 체크리스트의 체계화, 자동화<br>- 취약점 탐지 및 대응의 신속성<br>- 개발자 및 관리자들의 웹 보안 인식과 교육의 증진<br>- 보안 대응 지침 제공 |
| **경제.산업적 측면** | - 취약점 사전 예방에 따른 보안 위험 감소로 인한 사고 비용 절감<br>- 웹 사이트의 안전성과 사용자들의 신뢰성 향상<br>- 고객들의 정보보호 요구 충족을 통한 기업의 시장 경쟁력 강화 |
| **활용방안** | - 웹 개발 및 유지보수<br>- 웹 보안 컨설팅 및 서비스 제공<br>- 기업 및 조직의 보안 강화<br>- 보안 업데이트 및 정책 시행 |