Product Introduction

U-SAFE Code v1.0

행정안전부 소프트웨어 개발보안 가이드 49개 항목 기반의
정밀 시큐어코딩 점검 솔루션

49개 보안약점 | 6개 언어 | Taint 흐름 추적 | 행안부 양식 보고서

www.unpl.co.kr | (주)유엔피플

TABLE OF CONTENTS

CONTENTS

소스코드 보안약점 점검의 새로운 기준,
U-SAFE Code v1.0의 주요 내용을
소개합니다.

01 **U-SAFE Code 소개**
도입 배경과 제품 개요

02 **핵심 분석 엔진**
49개 항목 · 6개 언어 · Taint 흐름 추적 · 정확도 검증

03 **주요 기능**
스캔 · 취약점 관리 · 조치 워크플로우 · 통계 · 보고서

04 **제품 특장점**
정밀성 · 준거성 · 협업성 · 편의성 · 보안성

05 **구성 및 기대효과**
배포 형태와 도입 기대효과

01 제품 소개

도입 배경 Why Secure Coding

공공 정보화사업은 **소프트웨어 개발보안 적용이 의무**이며, 구현 단계에서 **보안약점을 진단하고 제거**해야 합니다.

제도 요구사항

1. **「행정기관 및 공공기관 정보시스템 구축·운영 지침」**
   행정안전부 고시 — 정보화사업에 소프트웨어 개발보안을 적용하도록 규정
2. **「소프트웨어 개발보안 가이드 (2021)」**
   행정안전부·KISA — 7개 분야 49개 보안약점의 진단·제거 기준
3. **감리 · 검수 대응**
   구현 단계 보안약점 진단 결과와 조치 내역을 근거 자료로 제출

현장의 어려움

높은 도입·유지 비용

외산 상용 정적분석 도구는 라이선스와 유지보수 비용 부담이 큽니다.

과도한 오탐

단순 패턴 매칭은 이미 안전하게 처리된 코드까지 탐지해 검토 공수가 늘어납니다.

보고서 수작업

진단 결과를 행안부 양식에 맞춰 다시 정리하는 데 많은 시간이 듭니다.

조치 이력 관리 부재

담당자, 조치 상태, 재점검 결과를 별도 문서로 관리해야 합니다.

U-SAFE Code는 **탐지 → 검토 → 조치 → 검증 → 보고** 전 과정을 하나의 도구로 해결합니다.

01 제품 소개

U-SAFE Code v1.0 Product Overview

행정안전부「소프트웨어 개발보안 가이드」49개 항목을 기준으로 소스코드를 정적 분석하는 **설치형 시큐어코딩 점검 솔루션**입니다.

49 개
점검 항목 (7개 분야)

6 종
지원 언어

3 종
점검 기준 (행안부·OWASP·국정원)

10 단계
보안약점 조치 상태

행안부 49개 항목 완전 대응

SW 개발보안 가이드(2021) 7개 분야 49개 보안약점을 공식 번호 체계(1.1~7.2) 그대로 점검합니다.

Taint 기반 정밀 분석

외부 입력(Source)에서 위험 함수(Sink)까지 데이터 흐름을 파일 경계를 넘어 추적합니다.

조치까지 이어지는 협업

담당자 할당, 10단계 상태 관리, 정/오탐 분류로 조치 이력을 한곳에서 관리합니다.

행안부 양식 보고서 자동 생성

PDF 상세 보고서와 14열 로우데이터 Excel을 클릭 한 번으로 생성합니다.

02 핵심 분석 엔진

점검 범위 Coverage

7개 분야 49개 보안약점을 행안부 공식 번호 체계 그대로 점검하고, **6개 언어**의 소스코드를 구문 분석합니다.

분야	항목 수	대표 보안약점
1 입력데이터 검증 및 표현	17	SQL 삽입 · XSS · 경로 조작 · SSRF
2 보안기능	16	부적절한 인가 · 취약한 암호화 · 하드코드
3 시간 및 상태	2	경쟁조건(TOCTOU) · 무한 반복 · 재귀
4 에러처리	3	오류 메시지 정보노출 · 부적절한 예외 처리
5 코드오류	5	Null 역참조 · 자원 해제 · 역직렬화
6 캡슐화	4	세션 정보노출 · 남은 디버그 코드
7 API 오용	2	DNS lookup 보안결정 · 취약한 API

합계 **49** 개 항목

</> 지원 언어

Java
서버 웹 애플리케이션

JSP
화면 · 스크립틀릿

JavaScript
프런트엔드 스크립트

HTML
마크업 · 인라인 JS

XML
설정 · SQL Mapper

Python
스크립트 · 웹 앱

- tree-sitter 구문 트리 기반 분석
- OWASP Top 10 · 국정원 8종 자동 매핑

02 핵심 분석 엔진

Taint 흐름 추적 엔진 Data-flow Taint Analysis

외부 입력이 **어디에서 들어와 어디로 흘러가는지** 추적해, 실제로 위험한 흐름만 보안약점으로 판정합니다.

SOURCE
외부 입력
`request.getParameter()`

→

PROPAGATION
값 전파
대입 · 문자열 결합 · 메서드 인자

→

SANITIZER
정제 · 검증 확인
`PreparedStatement` · 인코딩

→

SINK
위험 함수
`executeQuery()` · `Runtime.exec()`

파일 간(cross-file) 흐름 예시

```
1  // UserController.java
2  String id = request.getParameter("id");          Source
3  userDao.find(id);                                파일 간 전달
4
5  // UserDao.java
6  String sql = "SELECT * FROM USERS"
7             + " WHERE ID='" + id + "'";           전파
8  stmt.executeQuery(sql);                          Sink
```

판정 **MOIS-01 SQL 삽입** · 심각도 **높음**

1 **PASS 1 프로젝트 요약**
전체 파일을 먼저 읽어 메서드별 입력·반환 흐름 요약(Taint Summary)을 만듭니다.

2 **PASS 1.5 파일 간 전파**
호출 인자를 따라 다른 파일·클래스로 오염 정보를 전달합니다.

3 **PASS 2 규칙 판정**
49개 규칙이 판정하고, 정제 여부로 심각도를 자동 분류합니다.

정제·검증이 확인된 흐름은 **보고하지 않아** 오탐을 줄입니다.

02 핵심 분석 엔진

탐지 정확도 검증 Accuracy Validation

분석 엔진은 **공인 벤치마크 · 항목별 테스트 케이스 · 실제 프로젝트** 3단계로 탐지 정확도를 검증합니다.

01

공인 벤치마크

2,740 케이스

OWASP Benchmark v1.2 (Java)

공개 정답지(expected results) 기준으로 탐지율·오탐률을 채점합니다.

02

항목별 테스트 케이스

453 케이스

49개 항목 · 취약 290 / 양호 163

항목마다 취약 코드와 안전한 코드를 함께 두어 탐지·오탐을 확인합니다.

03

실제 프로젝트 검증

Before/After

공공 정보화사업 실제 소스

엔진 개선 전후 결과를 비교해, 기존에 찾던 취약점을 놓치지 않는지 확인합니다.

엔진 개선 → 벤치마크 채점 → 항목별 케이스 → 실제 프로젝트 검증 → 배포

03 주요 기능

대시보드 Dashboard

전체 프로젝트의 점검 현황과 위험 수준을 **한 화면에서 파악**하고, 카드나 차트를 눌러 해당 목록으로 바로 이동합니다.

1 핵심 지표 카드

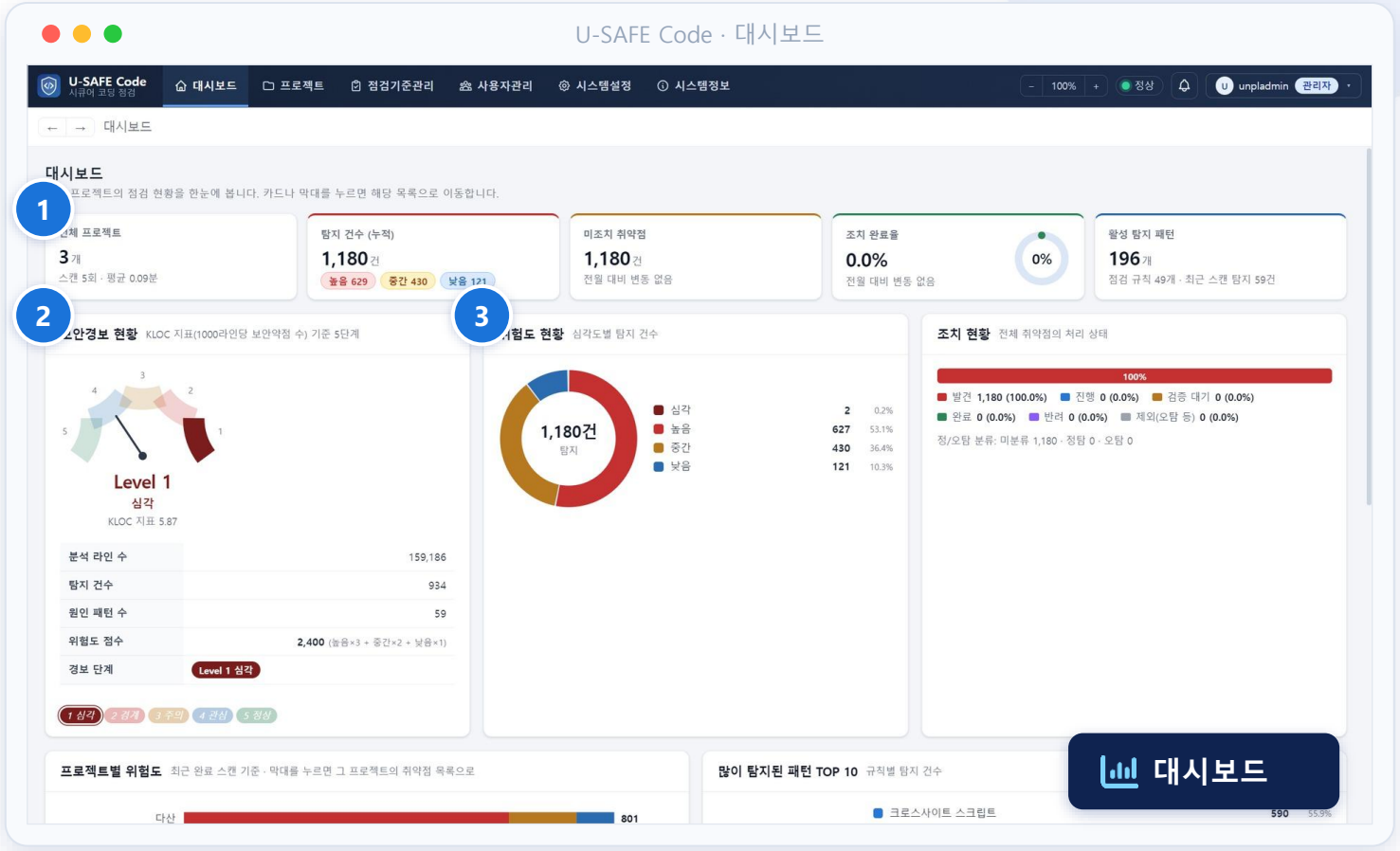
프로젝트 · 탐지 건수 · 미조치 취약점 · 조치 완료율 · 활성 탐지 패턴

2 보안경보 5단계

KLOC(1천 라인당 보안약점 수) 기준으로 심각 · 경계 · 주의 · 관심 · 정상을 표시

3 위험도 · 조치 현황

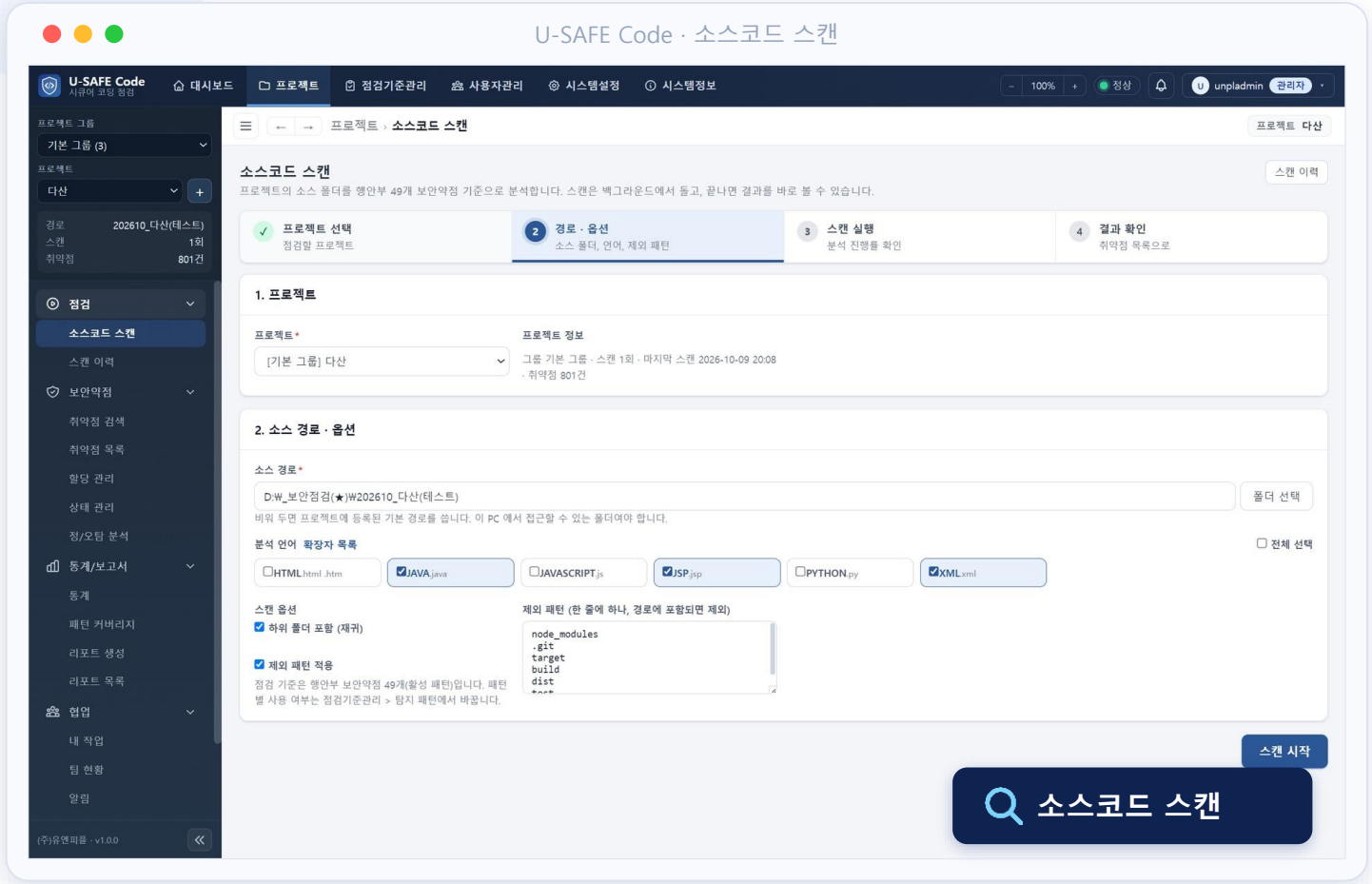
심각도 분포, 처리 상태, 프로젝트별 위험도와 많이 탐지된 패턴 TOP 10



03 주요 기능

소스코드 스캔 Source Code Scan

점검 대상 폴더를 선택하거나 **소스를 ZIP으로 업로드**하면, 4단계 마법사로 손쉽게 점검을 시작합니다.



STEP 1

프로젝트 선택

프로젝트·그룹 단위로 점검 대상을 관리합니다.

STEP 2

경로 · 옵션 설정

대상 언어를 고르고 node_modules 등 제외 경로를 지정합니다.

STEP 3

스캔 실행

49개 규칙과 Taint 엔진으로 전체 소스를 분석합니다.

STEP 4

결과 확인

회차별 이력을 저장하고 이전 회차 대비 신규/기존을 구분합니다.

서버형에서는 원격 PC가 **소스 ZIP 업로드**로 점검합니다.

03 주요 기능

취약점 검색 · 목록 Vulnerability Management

탐지된 보안약점을 **다양한 조건으로 검색**하고, 여러 건을 골라 **할당 · 상태 변경 · 정/오탐 분류를 한 번에** 처리합니다.

1 상세 조건 검색

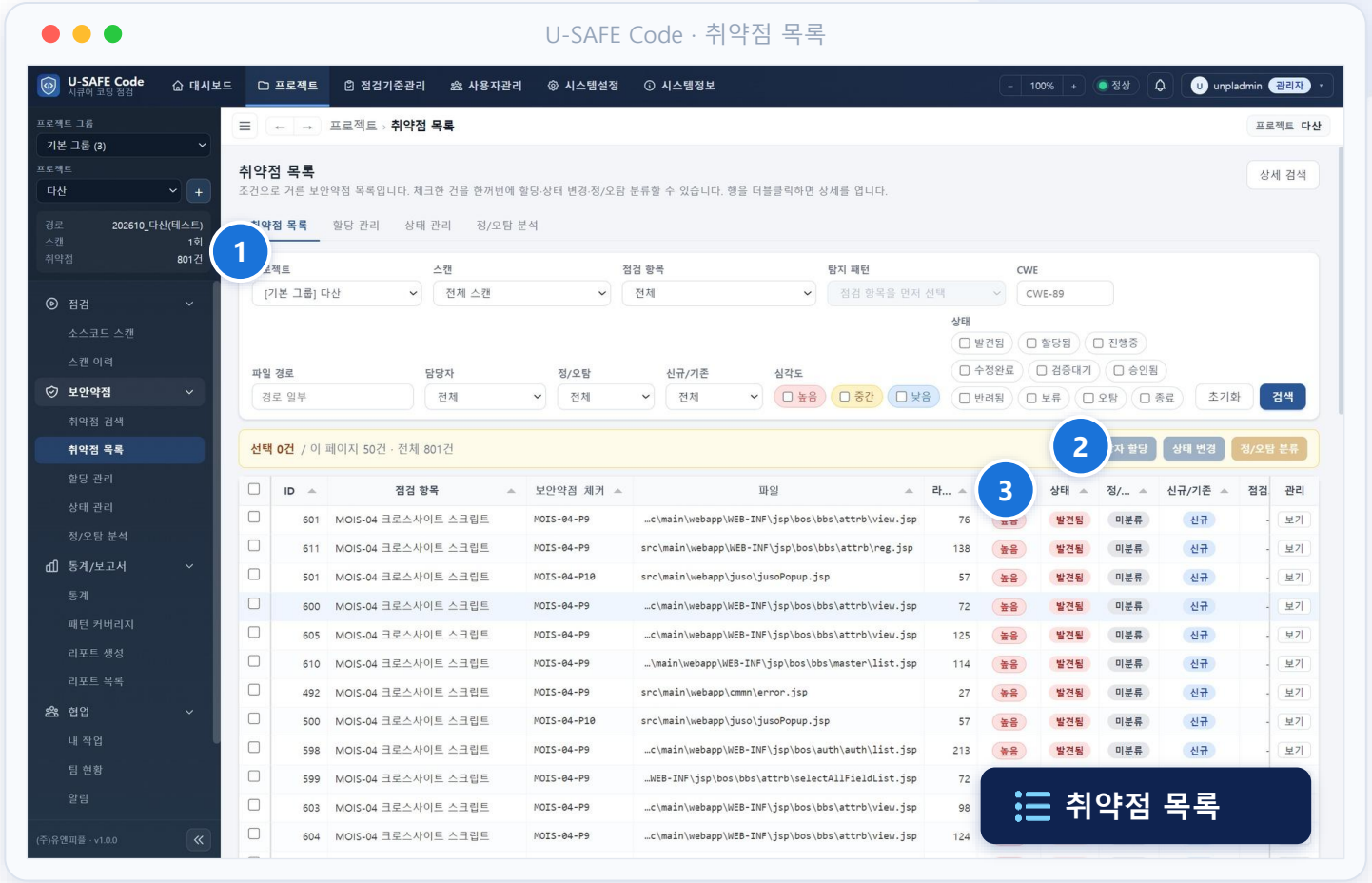
프로젝트 · 스캔 · 점검 항목 · 탐지 패턴 · CWE · 심각도 · 상태 · 담당자

2 일괄 처리

체크한 건을 한꺼번에 담당자 할당, 상태 변경, 정/오탐 분류

3 심각도 · 상태 배지

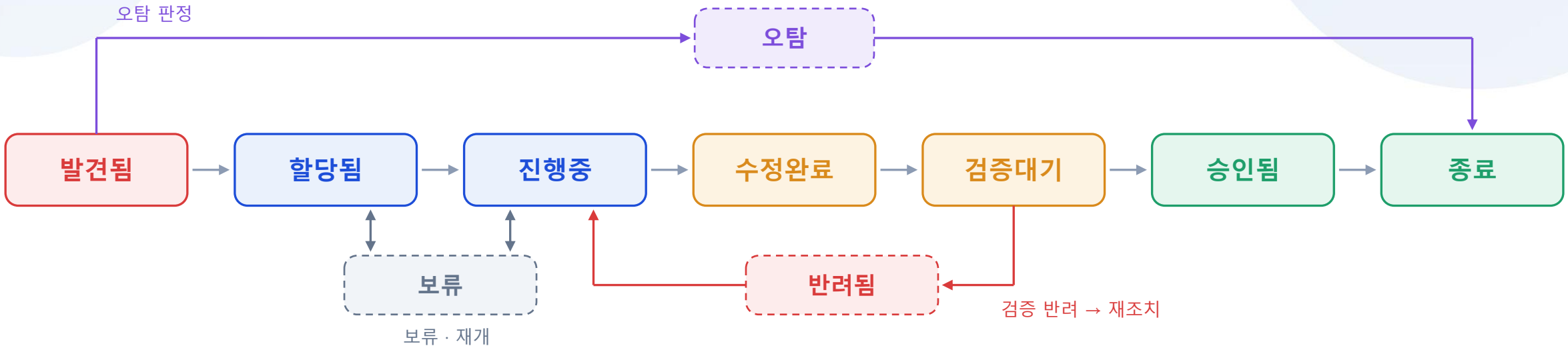
높음·중간·낮음, 처리 상태, 신규/기존 여부를 한눈에 구분하고 더블클릭으로 상세 확인



03 주요 기능

조치 워크플로우 Remediation Workflow

보안약점마다 **10단계 처리 상태**를 관리하고, 역할별로 허용된 상태 전환만 할 수 있도록 통제합니다.



역할별 상태 전환 권한

※ 뷰어는 조회만 가능

개발자

진행중 · 수정완료

선임 개발자

할당 · 승인 · 반려

감사자

할당 · 승인 · 반려 · 오탐 · 종료

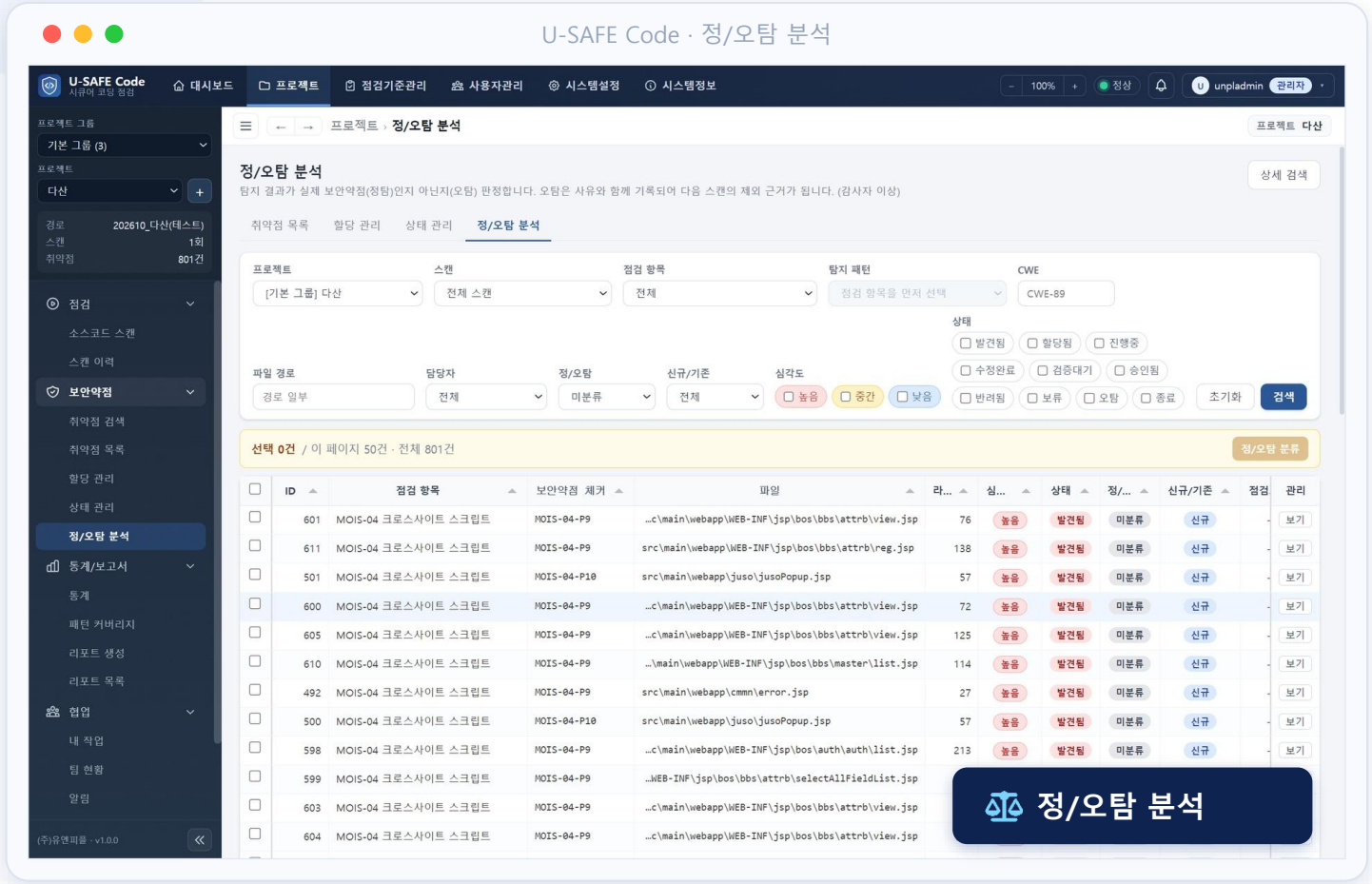
매니저 · 관리자

모든 상태 전환

03 주요 기능

정/오탐 분석 · 제외 관리 False-Positive Control

검토 결과를 **정탐 / 오탐으로 분류**하고, 제외 규칙을 등록하면 **다음 스캔부터 자동으로 반영**됩니다.



제외 패턴 관리

파일 / 디렉터리 제외

테스트 코드, 외부 라이브러리 등 점검이 필요 없는 경로

룰 제외

대상 시스템에 해당하지 않는 점검 규칙

화이트리스트

안전하다고 확인된 파일 · 코드 패턴

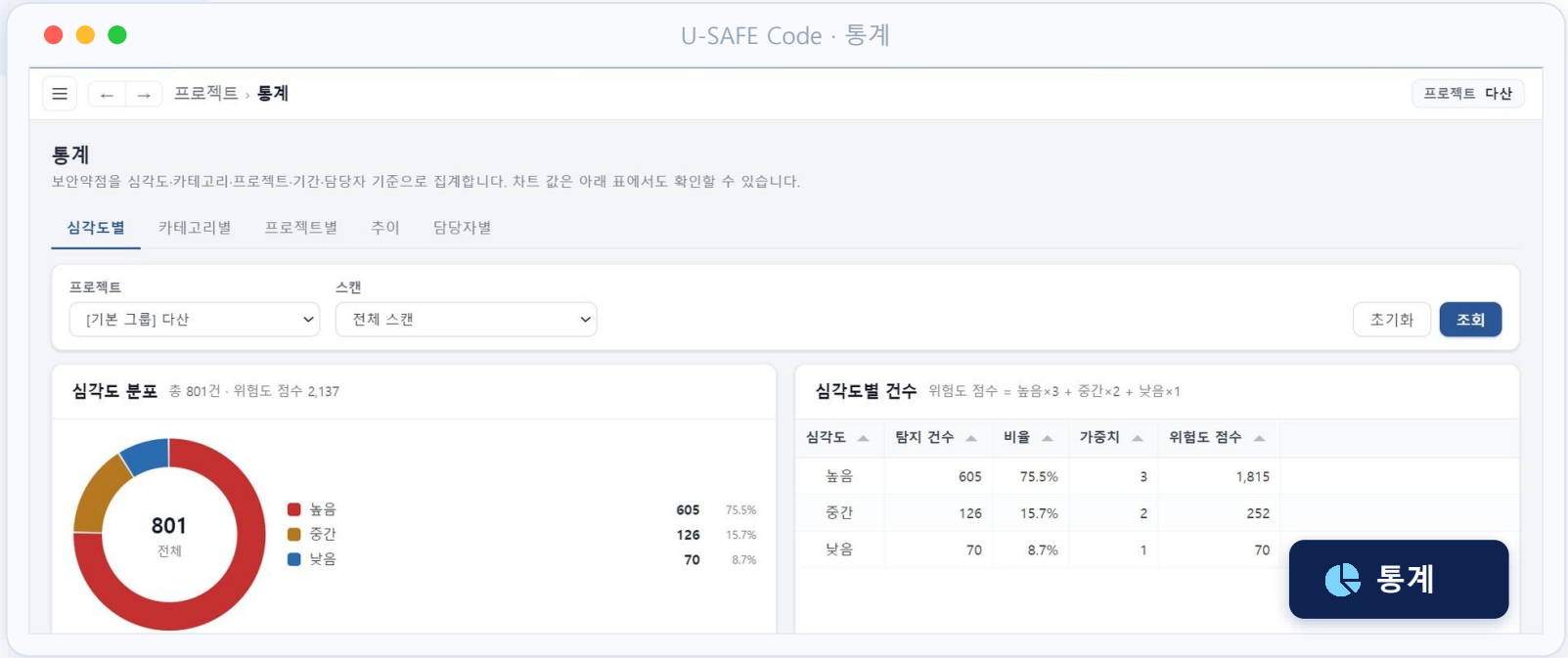
False Positive 등록

자동 제외를 켜면 같은 파일 · 라인 · 룰은 다음 스캔에서 자동 제외

03 주요 기능

통계 · 위험도 분석 Statistics

보안약점을 **심각도 · 카테고리 · 프로젝트 · 기간 · 담당자** 기준으로 집계하고, 위험도 점수로 조치 우선순위를 정합니다.



5가지 집계 관점

심각도별
높음 · 중간 · 낮음 분포와 위험도 점수

카테고리별
7개 분야 · 49개 항목별 탐지 현황

프로젝트별
프로젝트 간 위험 수준 비교

추이
회차 · 기간별 탐지와 조치 변화

담당자별
담당자별 할당 · 조치 진행 현황

위험도 점수 = 높음 × 3 + 중간 × 2 + 낮음 × 1

보안경보 5단계 (KLOC 기준)

1 심각 | 2 경계 | 3 주의 | 4 관심 | 5 정상

03 주요 기능

보고서 자동 생성 Report Automation

점검 결과를 **행정안전부 진단 양식의 PDF 상세 보고서와 Excel 로우데이터**로 클릭 한 번에 생성합니다.

실제 생성된 보고서



PDF 상세 보고서

표지 · 분석 요약 · 보안경보 · 항목별 결과, 탐지 위치 코드와 안전한 코드 비교까지 수록

Excel 로우데이터

행안부 14열 로우데이터 양식 그대로 제출하거나 추가 가공

점검 기준별 보고서

행안부 49개 · OWASP Top 10 · 국정원 8종 기준으로 다시 분류해 출력

상세본 · 요약본 선택

목적에 따라 전체 상세 보고서 또는 요약본을 생성

출력 형식 PDF Excel JSON

03 주요 기능

점검 기준 · 다중 기준 매핑 Multi-Standard

한 번의 스캔 결과를 **행안부 49개 항목 · OWASP Top 10(2021) · 국정원 웹 취약점 8종**으로 각각 분류해 보고합니다.

소스코드 1회 스캔

탐지 결과
보안약점 × CWE

재스캔 없이 **기준만 바꿔** 보고서 · 통계 생성

MOIS

행안부 49개 항목

1.1 SQL 삽입 ~ 7.2 취약한 API 사용

OWASP

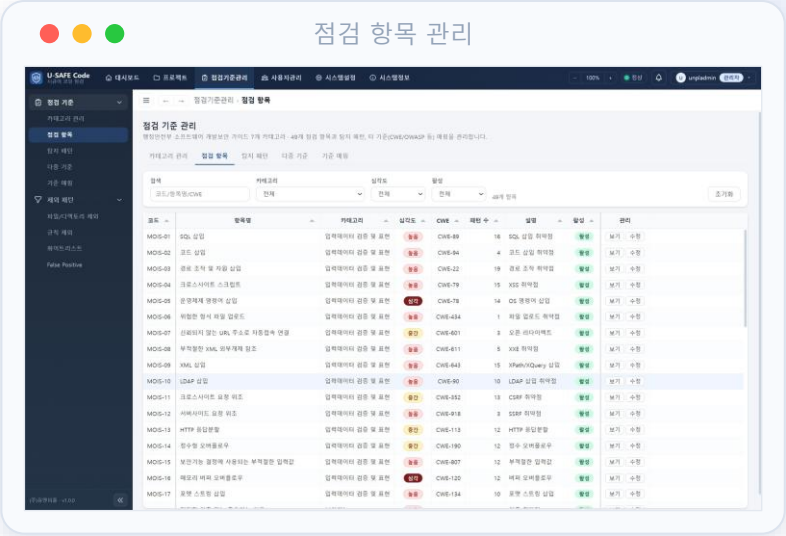
OWASP Top 10 (2021)

A01 접근 통제 ~ A10 SSRF

NIS

국정원 웹 취약점 8종

N1 SQL 인젝션 ~ N8 접근 제어



점검 기준 관리

- 카테고리 · 점검 항목 · 탐지 패턴 관리
- 항목별 활성화, 심각도, CWE 지정
- 기준 간 매핑 조회·편집

03 주요 기능

사용자 · 역할 권한 Role-based Access Control

6단계 역할 체계로 기능별 접근 권한을 통제하고, 보안 정책과 감사 로그로 사용 이력을 관리합니다.

Lv.5	**관리자**	모든 권한 · 시스템 설정 · 감사 로그 · 라이선스
Lv.4	**매니저**	프로젝트·사용자·팀 관리, 점검 기준 편집
Lv.3	**감사자**	할당·검증·반려·오탐 판정·종료
Lv.2	**선임 개발자**	보안약점 할당·검증·반려, 탐지 패턴 조회
Lv.1	**개발자**	할당된 보안약점 조치, 스캔 실행
Lv.0	**뷰어**	점검 결과·통계·보고서 열람

상위 역할은 **하위 역할의 권한을 모두 포함**합니다.



04 제품 특장점

제품 특장점 Key Strengths

정밀한 분석부터 조치 · 보고까지, **공공 시큐어코딩 점검에 필요한 기능**을 하나의 제품에 담았습니다.

정밀성 Precision

파일 간 Taint 추적과 정제 코드 인식으로 오탐 최소화

준거성 Compliance

행안부 49개 항목 번호 체계와 진단 보고서 양식 그대로

협업성 Collaboration

할당 · 10단계 상태 · 역할 권한으로 조치까지 관리

편의성 Usability

설치 파일 하나에 엔진 · DB 포함, 별도 구축 불필요

보안성 Security

소스를 외부로 보내지 않는 로컬 분석, 비밀 값 암호화 보관

분석 방식 비교

구분	U-SAFE Code	단순 패턴 매칭 방식
데이터 흐름(Taint) 추적	● 메서드 내부 + 파일 간 추적	△ 같은 줄·같은 파일 위주
정제 · 검증 코드 인식	● 안전하게 처리된 흐름은 미보고	✕ 패턴만 맞으면 탐지 → 오탐 증가
행안부 양식 보고서	● PDF 상세 · Excel 로우데이터 자동	✕ 결과를 양식에 맞춰 수작업 정리
조치 워크플로우	● 할당 · 10단계 상태 · 역할 권한	✕ 별도 문서·도구로 관리

05 구성 및 기대효과

제품 구성 · 배포 형태 Deployment

규모와 환경에 맞춰 **PC 설치형**과 **서버형(Windows · Linux)** 중 선택하며, 서버형은 웹 브라우저 접속도 지원합니다.

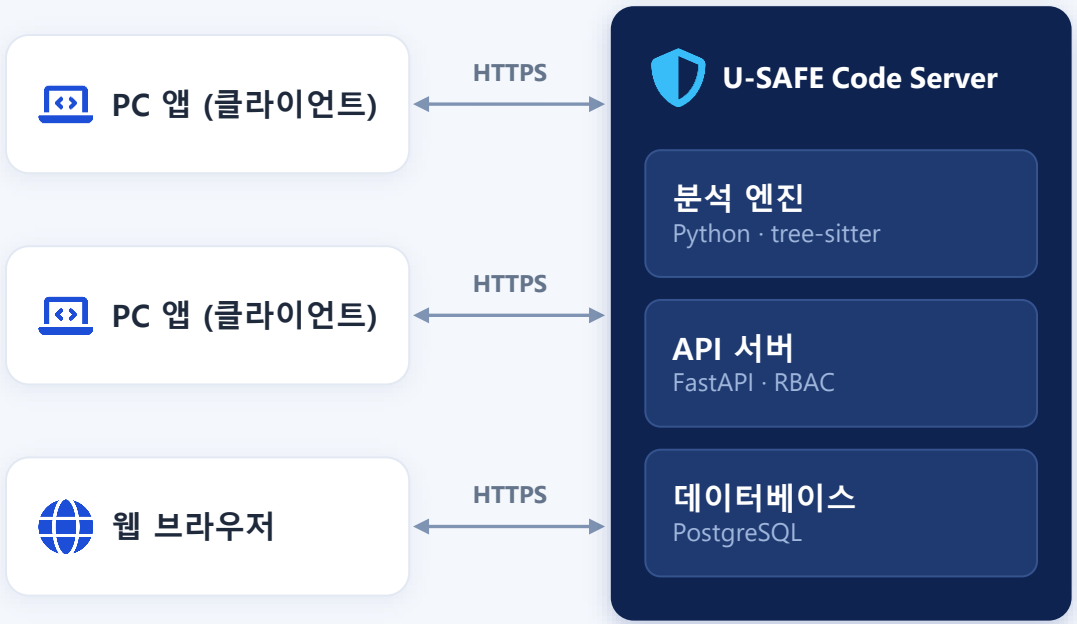
PC 설치형 Desktop

- 설치 파일 하나로 분석 엔진 · PostgreSQL 자동 구성
- 소스코드가 PC 밖으로 나가지 않는 로컬 분석
- 자동 업데이트 · 온라인/오프라인 라이선스 등록

서버형 Server

- 여러 PC · 사용자가 하나의 DB를 공유해 할당 · 협업
- Windows 서비스 / Linux systemd (Rocky · RHEL · Ubuntu)
- HTTPS 기본, PC 앱 · 웹 브라우저로 접속

서버형 구성도



운영 환경 Windows 10/11 (x64) · Windows Server · Rocky / RHEL · Ubuntu

도입 기대효과 Expected Benefits

U-SAFE Code 도입으로 **개발보안 의무 대응**과 **점검 · 조치 효율**을 함께 높일 수 있습니다.

01

개발보안 의무 대응

행안부 49개 항목 기준 점검과 공식 양식 보고서로 감리 · 검수에 바로 대응

02

진단 품질 향상

Taint 추적과 정제 코드 인식으로 오탐을 줄여 검토 공수 절감

03

조치 이력 가시화

담당자 · 상태 · 회차별 이력으로 조치 현황을 투명하게 추적

04

도입 · 운영 비용 절감

설치형 올인원 구성으로 별도 인프라 없이 합리적 비용으로 도입

05

시큐어코딩 내재화

개발 단계부터 상시 점검해 보안약점을 조기에 제거하는 문화 정착

Contact Us

감사합니다

안전한 소프트웨어의 시작, **U-SAFE Code v1.0**과 함께하세요.

Phone & Fax

02-855-7471

Fax 070-8650-0110

Online

idtong@unpl.co.kr

www.unpl.co.kr

Location

서울특별시 구로구 디지털로31길 38-9

803호 (구로동, 에이스테크노타워1차)