CODE-RAY XG

CodeRay 보안약점 분석 요약 보고서

OSS를 활용한 SASE 서비스 어플리케이션 개발 - 2 차

소프트웨어 개발보안 가이드 49개 항목

기관명	-
프로젝트 그룹 명	미니언즈
프로젝트 이름	OSS를 활용한 SASE 서비스 어플리케이션 개발
분석회차 (분석일)	2차 (2023-11-28)
보고서 출력일자	2023-11-28

담당부서 / 담당자	- / -
분석 엔진 버전	v6.0.2.16-r1

TRINITY SOFT

< 보고서 요약 >

보안약점 점검의 필요성

SW 개발 보안은 SW 개발 과정에서 개발자 실수, 논리적 오류 등으로 인해 SW에 내포될 수 있는 보안 약점 원인, 즉 보안 약점을 최소화 하는 한편, 사이버 보안 위협에 대응할 수 있는 안전한 SW를 개발하기 위한 일련의 보안 활동을 의미합니다. 광의적 의미로는 SW 개발 생명 주기(SDLC, Software Development Lifecycle)의 각 단계 별로 요구되는 보안 활동을 모두 포함하며, 협의적 의미로는 SW 개발 과정 중 소스코드 구현 단계에서 보안 약점을 배제하기 위한 '시큐어코딩(SecureCoding)'을 의미합니다. 최근의 사이버 공격은 침입차단 시스템 등 보안 장비를 우회하거나, 보안 패치가 발표되기 이전의 보안 약점을 악용하는 제로데이 공격,웹사이트 해킹 등이 많은 부분을 차지하고 있으며 사이버 공격의 약 75%가 SW자체의 보안 약점을 악용하는 것으로 웹사이트 공격이 대표적 예라고 할 수 있습니다.
사이버 공격을 선제적으로 예방 및 대응하기 위해서는 제품 출시 이전 단계인 SW 개발단계에서 보안 약점 을 제거하는 것이 가장 효과 적인데, 이는 미국 국립표준 기술연구 소(NIST)의 연 구 결과에 따르면 설계 과정에서 발생한 결함이 개발완료 후에 발견, 조치 되는 경우에는 설계 단계에서 결함을 수정 하는 비용대비 30배의 비용이 발생 한다고 합니다.또한 통합 과정에서 발생한 결함의경우, 20배의 수정비용이 발생하는등 개발 완료 이전에 보안 약점을 진단, 제거 하는 활동인 'SW개발 보안' 이 무엇보다 중요함을 나타내고 있습니다.

보안약점 점검 방식과 구성

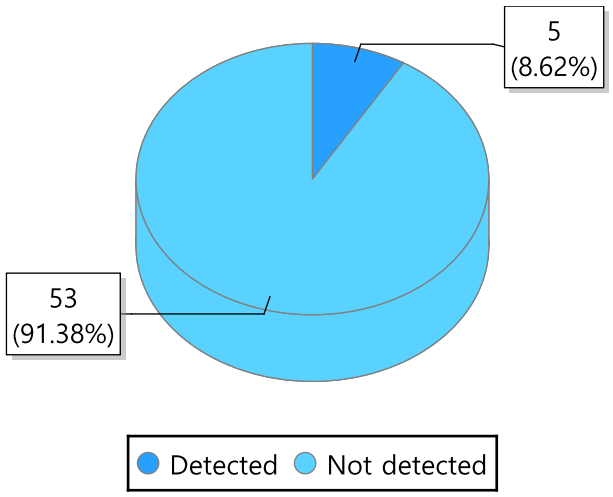
CODE-RAY XG는 정적분석 시큐어코딩 솔루션으로, SW를 실행하지 않고 소스 코드 수준으로 보안 약점을 분석합니다.정적분석 방법은 SW 개발 초기에 보안 약점 발견으로 빠르고 적은 비용으로 소스코드 수정이 가능한 장점이 있지만,컴포넌트간 발생할 수 있는 통합된 보안 약점 발견이 제한적일 수 있고 설계, 구조 관점의 보안 약점은 발견할 수 없습니다.
CODE-RAY XG 는 위와 같은 단점을 해결하기 위해 소스코드의 트리구조화 및 변수추적, 흐름 추적등의 기법으로 보안약점 발견의 확률이 높고, 서버환경 설정, 프레임워크 설정등의 보안약점 및 코드품질도 탐지해 SW 구조관점의 보안약점도 탐지 가능합니다.
분석유형 으로 '행정안전부 SW 보안약점 49개 기준','OWASP TOP10','CWE/SANSTOP 25','JAVA Code Convention','국정원 홈페이지 8대 보안약점', ' 전자금융감독규정 8대 보안약점'등의 기준에 의하여 점검 가능합니다.

위험도 분류 기준

심각	보안 설정 미흡으로인하여 조직의정보보호관리 및정보자산에 직접적인피해를 입을 수 있으며,중요한 자산 및 정보가노출될 수 있는 매우심각한 단계의 보안약점
높음	보안 설정 미흡으로 인하 여 조직의 정보보호관리 및 정보자산에 직접적인 피해를 입을 수 있으며, 중요한 자산 및 정보가 노출될 수 있는 단계의 보안 약점
중간	보안 설정 미흡으로인하여 조직의정보보호관리 및정보자산에 간접적,부분적인 피해를 입을수 있으며, 공격자에게정보가 제공될 수 있는단계의 보안 약점
낮음	조직의 정보보호관리 및 정보자산에 직접/간접적인 피해를 예상하기 어려우나, 소프트웨어 품질 면에서 취약할 수 있는 단계의 보안 약점

1. 프로젝트 분석결과 요약

보안약점검출 파일비율



분석 정보	
프로젝트 이름	OSS를 활용한 SASE 서비스 어플
분석회차 (분석일)	2차 (2023-11-28)
점검 기준	소프트웨어 개발보안 가이드 49개 항목
분석 언어	ANDROID,ASP,ASPX,C/C++,CS,DART,HTML,IOS,JAVA,JAVASCRIPT,JSP,KOTLIN,NEXACRO,PHP,PYTHON,SWIFT,VBSCRIPT,XML
탐지시간	00:00:08.475

분석 파일 정보	
분석 파일	58 건
분석 라인	3,805 라인
탐지파일	5 건
탐지원인	3 건
탐지결과	17 건

1.1 프로젝트 분석 보안경보 현황

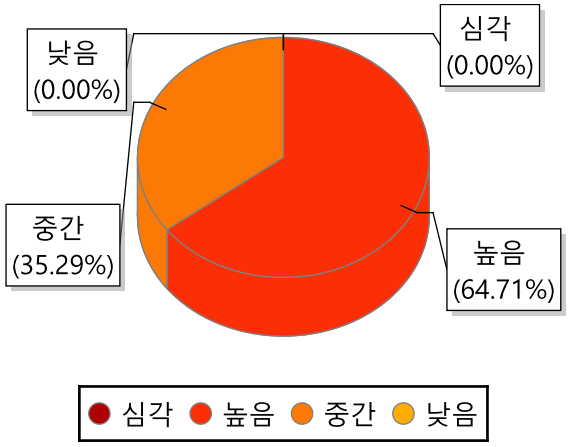
KLOC (Kilo Line Of Code) : 15 (1000 라인 당 15 건의 보안 약점 발생)

보안단계	5단계(정상)	4단계(관심)	3단계(주의)	2단계(경계)	1단계(심각)
보안경보 구간	0% ~	25% ~	50% ~	75% ~	100% ~
위험도 점수 구간	0 ~	128 ~	257 ~	385 ~	514 ~

* 보안경보 구간 : 소스코드의 총 라인수를 KLOC 15 기준으로 환산하여 5단계의 보안경보 구간 도출
* 보안경보 레벨 : 위험도 점수의 총 합이 속하는 보안경보 구간 및 위험도 점수의 구간을 확인하여 보안경보 레벨 도출

1.2 프로젝트 분석 위험도 현황

위험도별 탐지건수

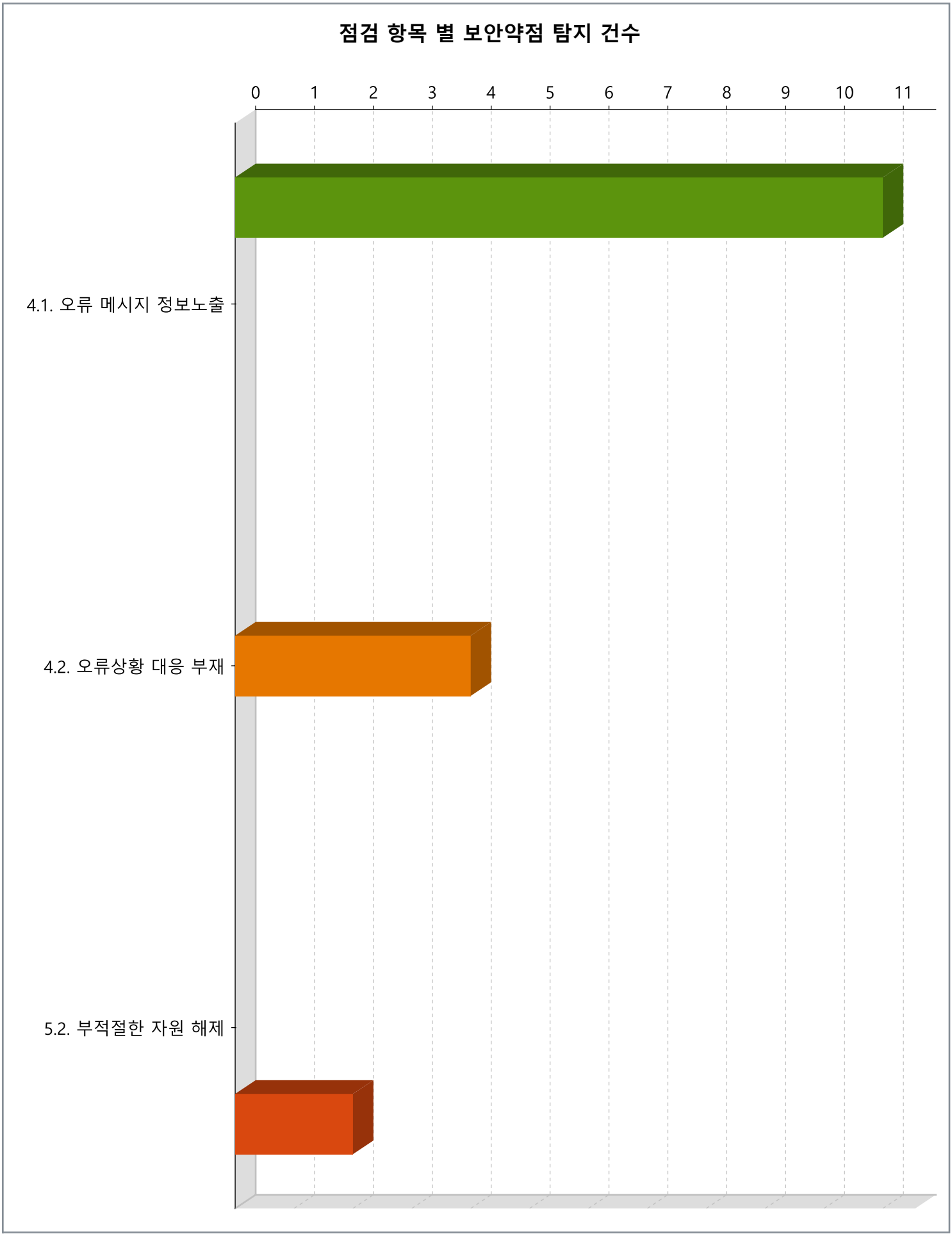


위험도 점수

위험도	탐지건수	위험도 가중치	위험도 점수
심각	0	9	0
높음	11	3	33
중간	6	2	12
낮음	0	1	0
탐지 건수	17		**45**

* 탐지건수 x 위험도 가중치 = 위험도 점수

1.3 프로젝트 점검 항목별 결과 요약



번호	점검 유형	원인패턴 건수	탐지건수
1	4.1. 오류 메시지 정보노출	1	11
2	4.2. 오류상황 대응 부재	1	4
3	5.2. 부적절한 자원 해제	1	2

2. 프로젝트 점검 항목별 결과

2.1 [4.1. 오류 메시지 정보노출]

보안약점ID	위험도	보안약점	원인패턴 건수	탐지건수
CWE-209	높음	민감한 정보가 포함 된 오류 메시지 생성	1	11

2.2 [4.2. 오류상황 대응 부재]

보안약점ID	위험도	보안약점	원인패턴 건수	탐지건수
CWE-390	중간	조치없이 오류 조건 감지	1	4

2.3 [5.2. 부적절한 자원 해제]

보안약점ID	위험도	보안약점	원인패턴 건수	탐지건수
CWE-404	중간	부적절한 리소스 종료 또는 해제	1	2