SBOM 도구 실증 결과보고서

2023. 12.

kisia 한국정보보호산업협회

SBOM 도구 실증 결과보고서

※ 이 보고서는 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 보고서임(No. 2022-0-00277, SW공급망 보안을 위한 SBOM 자동생성 및 무결성 검증기술 개발 과제)

SBOM 도구 실증 결과

1. 실증 개요

국내 유/무료 SBOM 실증 도구를 활용하여 현황 파악 및 본 과제의 SBOM 자동생성 기술 개발에 활용할 수 있는 분석 자료 수집을 목적으로 국내 정보보호기업 3개 社를 대상 SBOM 툴을 활용한 실증 조사를 실시하였다.

2. 기업별 실증 세부 결과

실증은 기업 별로 현재 시중에 유통되고 있는 실증도구 현황 및 장단점을 비교하여 소프트웨어 개발 시장에서 필요로 하는 SBOM 자동 생성 기술 개발을 가능하게 하는 데에 의의를 두었으며, 기업별 실증 세부 결과는 다음과 같다.

2-1. A사 실증 세부 결과

실증 관련 주요 내용 및 결과

- 실증절차

  (1) 소스 리포지토리에서 최신 소스 다운로드 받은 후, 개발 코드 정리

  (2) SBOM 스캐너 툴을 이용하여 프로젝트별로 스캐닝 파일을 생성

  (3) 프로젝트별로 생성된 스캐닝 파일을 분석 서버로 업로드하여 분석

- 실증 관련 주요 내용 및 결과

  (1) 10여 개 프로젝트 생성 및 12회 누적 분석 실행

  (2) 취약점 리스트 공유 및 검증 플랫폼 필요

실증 시 애로사항 및 개선사항

- 취약점 탐지 오류 및 라이선스 조치

  분석 결과에서 일부 매칭률이 낮아 라이브러리 오탐이 발생하거나, 라이선스 표기에 오탐이 존재하는 것으로 확인되기도 하였다.

- 분석 시 용량 제한

  분석에 사용된 시스템의 라이선스 정책으로 인해 특정 용량 이하로 소스 코드만 분리해내는 작업이 번거로웠다. 빌드 패키징 상의 이슈로 소스와 바이너리가 혼합된 구성에서 불편한 선별작업이 요구되었으며, 스캐너가 구동될 때 소스 코드 파일이 아닌 경우 용량 계산에서 제외되도록 수정이 될 수 있다면 분석 스캔 데이터를 만들 때 조금 더 용이할 것으로 보인다.

- Debug/Release 분기 처리

  개발시점에서만 사용되고 최종 빌드에는 포함하지 않는 경우도 있을 수 있으므로 빌드 후 포함되지 않는 코드에 대한 취약점의 제외 기능이 있으면 좋을 것 같다.

- 상용라이브러리에 대한 분석추가

  오픈소스 라이브러리 뿐만 아니라 상용 라이브러리에 대한 분석 추가 기능도 보완되었으면 좋겠다.

실증결과를 통한 취약점 관리 방향

- 취약점 리스트 공유 및 검증

  사용된 모듈 및 라이브러리는 버전이 업데이트 될 경우 제공된 기능 및 사용 방법이 변경될 수도 있다. 이에 따라 취약점으로 보고된 경우에도 바로 적용하지 않고, 취약점 리스트를 개발자와 공유하여 실제 사용 여부 및 버전 업데이트 가능 여부를 한 번 더 확인하여 검증하도록 하였다.

- 모듈 및 라이브러리 버전 업데이트

  버전 업데이트가 가능한 경우 업데이트를 진행하여 취약점 이슈를 해결한다. 더 이상 버전 업데이트를 지원하지 않는 모듈이거나 라이브러리인 경우 사용 가능한 다른 버전을 검토하여 교체 작업을 진행하도록 하였다.

- 자동화된 빌드 툴과의 통합

  최근의 소프트웨어는 점점 더 복잡해지고 있으며 개발 이후의 업무 프로세스 즉, 빌드 및 자동화 테스트 등이 Jenkins 같은 CI/CD 툴에 의해 지원되는 경우가 많았다. 실증에 사용된 툴이 그러한 기능을 지원할 수 있으면 CI/CD 상의 파이프라인에 SBOM 실증 단계를 포함하여 패키지 빌드 시 자동으로 확인할 수 있도록 구성하면 좋을 것 같다.

SBOM 관련 정책/기술적 개선사항

- 정책적 개선사항

법률적으로 제도화하는 것에 앞서, 공급망보안이라는 새로운 사이버보안 영역에 대한 중요성 인식과 SBOM활성화를 위한 노력들을 제도화하고 이를 통해서 발생한 공급망 보안 취약점에 대해, 정부, 기관 그리고 기업이 함께 대응할 수 있는 제도가 우선되어야 한다. 그리고 나서, 이런 정책적 기반아래, 법률적으로 제도화하는 것이 바람직하다.

- 기술적 개선사항

실증에 사용되었던 SBOM 도구에서 적지만 일부의 오탐 결과가 포함되어 분석에 어려움을 겪은 바가 있어 기본적인 SBOM 도구에 대한 정탐률을 높이기 위한 기술적 개선이 필요해 보인다.

일반적인 정적 및 동적 분석 도구들이 소스 코드 스캔을 기본적으로 수행하는 것으로 볼 때 그러한 도구들에 SBOM 기능이 추가되는 것이 SBOM만 있는 솔루션보다는 보다 시장에서 경쟁력이 있을 것으로 판단된다.

2-2. B사 실증 세부 결과

실증 관련 주요 내용 및 결과

- 실증절차

먼저 실증 체크 리스트를 작성해 도구 평가를 위한 항목으로 정확성, 범용성, 운영환경, 사용성, DB 업데이트, 기술지원을 선정하였다. 다음으로 SBOM 도구로 오픈소스 취약점을 분석하기 위해 제품의 소스코드 중 CC인증에서 지정하고 있는 인증효력유지 허용 대상의 공개용 소프트웨어 목록에서 3rd party 소프트웨어 55종을 골라 대상으로 선정하였다. 마지막으로 해당 제품의 검색 엔진에 해시파일을 업로드하고 분석을 실행하였다.

실증시 애로사항 및 개선사항

- 운영환경

도구는 기본적으로 효율적인 점검을 위해 개발 플로우에 긴밀하게 적용 가능해야 하는데 그러기 위해서는 개발 프로세스 상에 SBOM 점검을 포함하고, 개발 과정에서 자동으로 SBOM 점검이 가능하도록 구축해야 한다.

하지만 퍼블릭 클라우드 환경의 개발 환경을 사용하는 경우를 제외하고는 개발 환경은 일반적으로 폐쇄망으로 구성되어 있으므로, SBOM 도구는 On-premise 운영환경으로 구축해야 한다. 하지만 실증 시에는 On-premise 운영환경이 지원되지 않아서, 실제 제품 개발에 적용했을 때의 효과를 점검하지는 못하였다.

폐쇄망이라 수동으로 DB를 업데이트해야 하나, 자동 업데이트만큼 자주 수행하기엔 한계가 있으므로, 주요 DB 업데이트 발생 시 긴급 업데이트하도록 알람을 띄워주는 등의 체계도 필요해 보인다.

- 기술 지원

해당 제품의 경우, 설치 및 점검하는 과정에서 기술 지원을 받을 수 있었으나, 추가 검토도구에 대한 기술지원도 필요하다.

- SBOM 도구의 사용성

SBOM 도구는 기존의 오픈소스 점검 도구와 취약점 점검 도구의 합이므로 점검된 오픈소스와 그의 취약점 관리에 대한 기능이 분리되어 관리되면 좋겠다. 그리고 결과를 표 형식으로 나열해 두어, 각 항목별로 분석을 해야 결과를 알 수 있었기에 DB에서 점검된 각 도구들의 하위 컴포넌트 간 의존성을 보여주는 기능이 강화되었으면 한다.

실증결과를 통한 취약점 관리방향

현재 개발 단계에서는 오픈소스 점검, 소스코드의 취약점 점검, 해킹 팀을 통한 바이너리 취약점 점검을 수행하고 있다. 이 과정은 제품의 새 버전이 발생할 때마다 개발 프로세스 상에서 각 단계를 수행하는 기준점으로 적용하고 있다.

SBOM 도구를 통해서 발견되는 취약점 역시, 기존 개발 프로세스 상에서 관리하는 취약점에 추가하여 관리할 예정이다.

추가적으로 제품에 사용되는 오픈소스 소프트웨어의 취약점을 빠르게 대응하기 위해, SBOM을 주기적으로 자동수행하고 취약점이 발견된 경우 알람을 띄울 수 있도록 SBOM 도구와 개발 도구 간의 연계가 필요하다. 현재는 이러한 연동 기능을 미제공하는 것으로 보이는데, 이후에 실사용 시에는 개발 도구와의 연동 협업이 필요하다.

SBOM 관련 정책/기술적 개선사항

SBOM 관리 기관 지정

SBOM 활성화를 위해 필요한 기관으로는, SBOM 체계 및 표준을 관리하는 기관과 SBOM 활성화를 담당하는 기관이다. 그리고, SBOM 보고서에는 사용중인 오픈소스 목록과 취약점 정보가 포함되어 있기 때문에 관리하지 않으면 제로데이 공격에 악용될 수 있으므로 각 개발사가 소프트웨어의 SBOM 점검을 하고, 생성된 보고서를 제출한 후 제출 증명서를 발급받을 수 있는 기관이 필요하다.

일관성있는 정책 운영

금융보안원 보안과 관련된 공공·관계기관에서 SBOM 및 공급망 보안 체계 도입을 추진하고 있다. 공급망 보안 강화와 SBOM의 도입의 활성화는 긍정적이지만, 각 기관별 정책이 일관성 없다면 소프트웨어 개발사에게 부담이 가중된다. 그러므로, 여러 기관에서 이루어지고 있는 SBOM 관련 정책에 대해서도 앞서 언급한 SBOM 관리 기관에서 일관성을 조율할 필요가 있다.

취약점 대응 방안

SBOM이 효과를 보이려면 발견된 취약점을 빠르게 대응할 수 있도록 협력사와 파트너 공급망 보안 관리 체계를 확보하는 것이 중요하다. 앞서 언급한 제 3의 기관에서 각 소프트웨어의 SBOM 보고서를 취합하고, 그 보고서를 기반으로 오픈소스를 사용하는 소프트웨어의 통합 데이터베이스를 구축한다면, 제로데이 취약점이 발견될 때 영향 받는 제품을 즉시 조회하고, 대응 지침을 전달하여 대응하도록 할 수 있다.

오픈소스의 신뢰성 확보

오픈소스에서 취약점이 발견되는 경우, 대응해야 하는 주체가 불명확한 경우가 있다. 각 오픈소스를 제공하는 커뮤니티 등에서 취약점을 제거한 신규 버전을 제공하는 경우에는 그 패치를 적용하면 되지만, 그렇지 않은 경우에는 사용자(소프트웨어 개발업체)가 직접 취약점을 제거할 수 있는 방안을 고려해야 한다. 그러나, 이 경우에는 소스코드에 직접 변경을 발생시키므로, 오픈소스의 라이선스 정책에서 허용하는 범위내의 오픈소스만 수정을 할 수 있다.

또한, 발견된 취약점을 제거한 버전을 제공한다고 하더라도, 오픈소스를 제공하는 제공처가 신뢰할 수 있는 곳이 아니라면 해당 오픈소스를 사용하는데 리스크가 생긴다. 그러므로, 협력사와 파트너 공급망 보안 관리 체계를 확보하는 것이 필요하다.

SBOM 도구 실증 결과

- SBOM 관리를 위한 교육 및 가이드라인 제공

  SBOM의 개념과 SBOM을 관리하는 방법에 대한 교육이 필요하다. 특히, SBOM 관리를 위해 업체에서 갖추어야 할 사항을 Best practice 등의 구체적인 사례로 제시해주어야 한다.

- SBOM 도구의 지원

  개발 환경에 따라 SaaS 서비스 혹은 폐쇄망에서 사용할 수 있도록 SBOM 도구 설치 장비의 대여 또는 단기 라이선스 부여 등의 지원 방안이 필요하다.

- 다양한 점검 방식 제공

  실제 오픈소스를 소스코드 외의 라이브러리로 사용하는 경우도 있으므로, 바이너리 점검이 가능한 도구의 지원도 필요하다. 특히 소프트웨어 제품을 개발하는 과정에서 타 업체와의 기술 협업 등으로 3rd party일부 모듈을 사용하는 경우엔 소스코드 없이 라이브러리 형태로 사용하기 때문에 이런 경우를 고려하여 바이너리 점검이 필요하다.

- 오픈소스의 의존성 관리

  오픈소스 소프트웨어 간 의존성이 관리되지 않으면, 취약점이 발견되는 경우 간접적으로 끼치는 영향을 파악하기 어렵다. 그러므로, 오픈소스 소프트웨어를 분석할 때에는 SBOM에서 컴포넌트 간의 의존성도 파악할 수 있어야 한다. 또한 오픈소스 소프트웨어의 의존성을 관리하는 데이터베이스를 구축하여, 소프트웨어 개발사와 사용자들이 함께 참고할 수 있도록 제공되어야 한다.

2-3. C사 실증 세부 결과

실증 절차

[Build Server 내의 소스코드를 최신버전으로 Update → Build Server의 제품 Build 및 제품 운영에 필요한 3rd S/W 및 Library 확인 → 스캐너로 Build Server 스캔 → 결과 값을 서버로 업로드 및 결과 확인 → 결과 검토 및 적용 → 사용자 정의 룰 추가] 순으로 진행하였다.

실증 시 애로사항 및 개선사항

- 점검도구의 사용 환경 제한

  내부 보안정책으로 개발 소스코드의 외부 반출이 불가능하지만 분석서버가 클라우드 환경으로 제공되어, 스캐너를 통한 결과 점검 시 사용자가 작업을 위해 추가 작업을 진행하였다.

- 스캐너로 Build Server 스캔 시 개발팀의 지원 필요

  스캐너를 통해 스캔을 할 경우 Build 환경이 지원된 서버에서만 스캔이 가능하여, Build Server에 접근이 가능한 개발팀의 지원이 필요하였다.

- 개발팀의 취약점 관련 인식 변화

  기존에는 3rd SW의 취약점을 고객사의 요청에 있는 경우에만 대응을 하였으나, 이번 검증을 통해 개발 단계에서 제품의 취약점을 파악하고자 하는 인식의 변화가 생겼다.

- 업무 프로세스 개선

  기존에는 [제품 개발 → 제품 패키징 → 제품 테스트 및 버그수정 → 제품 릴리즈]까지의 단계를 거쳤으나, 실증을 통해 [제품 개발 → 제품 패키징 → 제품 테스트 및 취약점 점검 → 제품 릴리즈]의 단계로 취약점 프로세스를 실무에 적용하고 변경하였다.

실증결과를 통한 취약점 관리방향

- 주기적인 취약점 점검 및 대응

  주기적인 취약점 점검을 수행하여 소스코드 및 3rd SW의 취약점 확인 해당 취약점에 대응할 수 있는 프로세스를 정립한다.

- 취약점 데이터베이스 관리

  발견된 취약점에 대하여 데이터베이스를 생성하고 중요도에 따라 취약점에 대한 우선순위를 설정한다.

- 이슈관리 시스템을 통한 취약점 대응

  중요도에 따라 정리된 취약점을 이슈관리 시스템에 등록하여 취약점 점검 대응을 관리한다.

- 정기적인 운영체제 및 미들웨어의 보안 패치 관리

  운영체제 및 미들웨어의 보안 패치 적용을 통해 고객사 대상으로 운영체제 및 미들웨어 보안 패치를 권고한다.

- 보안 관점의 개발자 교육 시행

  개발자들의 제품 개발 시 취약한 소스코드를 사용하지 않도록 시큐어 코딩 관련 교육을 주기적 시행하고, 기존에 수행하지 않았던 소스코드 리뷰 과정을 통해 소스코드 취약점 및 버그를 점검할 수 있도록 개발 프로세스 재정립한다.

SBOM 관련 정책/기술적 개선사항

- 가독성 좋은 형태로의 문서화 기능 지원

  표준문서인 CycloneDX와 SPDX는 다양한 파일 확장자를 제공하는데 해당 파일을 확인하기 위해서는 그 확장자에 맞는 reader tool을 이용하여 시각화 된 SBOM 목록을 확인하기 위해 1단계가 더 진행되었다.

- 취약점 상세 내역 확인 기능 지원

  SBOM의 도구에서 최종 배포된 Software에 대한 검출을 하며, 배포되어 사용 중인 Software의 버전 체크 및 취약점 개수를 알려주고, 신규 버전으로 업데이트를 권고 하지만 검출된 버전의 취약점에 대한 내용을 추가로 확인 할 수 있는 기능이 제공되지 않아 어떤 취약점이 있는지 확인이 안 되며, 경우에 따라서는 제품에서 사용 안하는 기능과 상관없는 취약점의 불필요한 업데이트가 진행 될 수 있다.

○ 정책적 개선사항

현재 SBOM은 ①오픈소스 관리 및 취약점관리, ②공급망 보완 대응, ③SW 공급망 관리의 3개 영역을 모두 다루고 관리를 해야하는 상황이다. 그러므로 소프트웨어 산업 및 소프트웨어를 사용하는 모든 정부, 기관 그리고 기업에서 SBOM를 기반으로 하는 소프트웨어 개발 및 관리 체계를 가질 수 있는 범 국가적인 정책으로 확대해야 할 것이다.

기타 사항

SBOM에는 CYCLONEDX와 SPDX 2종류의 표준 형식이 있는데 두 형식이 각각 다른 정보를 취급하므로 비교를 위해 두 가지 모두 점검이 필요하다. 기대했던 내용은 22년도의 log4j 사건처럼 3rd SW에 log4j가 있어서 취약점이 발견되는 것인데, 이번 SBOM 실증 사업에서는 3rd SW가 다른 3rd SW를 가져와 쓰는 내용을 확인하기 어려웠다. 추후에 개발될 SBOM 도구에는 해당 기능이 더 자세하게 지원되었으면 한다.

3. 요약 및 시사점

국내 주요 정보보호 기업 3개社가 참여하여 SBOM 툴을 활용한 실증 조사 실시 결과 요약 및 시사점은 다음과 같다.

구 분	A사	B사	C사
SBOM 실증 주요 내용	▸ 10여개 프로젝트 생성 및 12회 누적 분석 실행 ① 소스 리포지토리에서 최신 소스 다운 ② SBOM 스캐너 툴로 스캐닝 파일 생성 ③ 프로젝트별로 분석 서버에 업로드, 분석	▸ CC인증 효력이 유지되는 오픈소스 목록 55개 중 11개 OSS 선정 ▸ 해당 제품의 검색 엔진에 해시 파일을 업로드하고 분석 실행	▸ 자사제품 엔진부(15,602개 파일), 콘솔부(1,642개 파일) 총 17,244개 파일 분석
정책적 개선사항	▸ 법률적으로 제도화하는 등 의무화를 빠른 시일 내에 수행하여 SW 공급망 보안에 대한 정책적 개선 필요	▸ 정부의 SBOM 일관된 정책 운영을 위한 관리 기관 일원화 (취약점 악용 우려하여 생성된 SBOM을 제3기관 제출 증명서로 발급받는 체계 등) ▸ 유료도구 지원, SBOM 관리를 위한 교육 및 Best Practice 가이드 제공	▸ 기업들이 SBOM을 왜 해야 하는지 의문을 가지게 되는 상황 이므로 이에 대한 정책적인 답변이 필요하고 SBOM에 쉽게 접근할 수 있도록 정책적 지원 역시 필요
기술적 개선사항	▸ SBOM 도구에 대한 정탐율을 높이기 위한 기술적 개선 필요 ▸ 일반적인 정적 및 동적 분석 도구들에 SBOM 기능 추가 필요	▸ 취약점 추가 시 알림 팝업 기능 ▸ 소스코드, 바이너리 등 다양한 점검방식 지원 ▸ 오픈소스 의존성 도식화 강화 필요 ▸ 수정된 오픈소스 탐지 기술 강화 필요	▸ 가독성 좋은 형태의 문서화 기능 지원 ▸ 취약점 상세 내역 확인 기능 지원 ▸ 오탐율 개선 필요
총평 및 시사점	▸ **국내 주요 정보보호기업 3개 社를 대상으로 SBOM 툴을 활용한 실증 조사 실시** - SBOM 도구 활용, 보안제품 및 응용 SW 대상 실증 진행 (SBOM 분석 및 실증 시 애로사항, 개선사항을 통해 향후 SBOM 도입에 필요한 방향성 제시) ▸ 정부의 SBOM 일관된 정책 운영을 위한 관리 기관 일원화 또는 다부처 협력체계 강화, 취약점 우선순위를 확인할 수 있는 통합 DB, SBOM 플랫폼 필요, 유료도구 지원, Best Practice 가이드, SBOM 관리를 위한 교육 등 **SBOM에 대한 거부감을 줄이는 정책 지원 필요** ▸ SBOM 도구의 정탐율 향상, 취약점 상세 내역 확인, 다양한 점검 방식 지원 등 **SBOM툴의 기능 향상을 위한 기술적 지원 필요** ▸ **이번 실증을 통해 개발팀 내 취약점 인식 변화 및 자사 개발 프로세스에 변화를 겪은 기업이 존재하여, SBOM 정책이 단순 관리 불편 요소가 아닌, 보안 인식 제고에 도움되는 사례임을 확인할 수 있었음**		

[표 10] SBOM 실증 조사 결과 요약 및 시사점

이 보고서는 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 보고서임
(No.2022-0-00277, SW 공급망 보안을 위한 SBOM 자동생성 및 무결성 검증기술 개발 과제)

SBOM 도구 실증
결과보고서

kisia 한국정보보호산업협회