디지털산업정보학회 논문지 제17권 제4호-2021년 12월
http://dx.doi.org/10.17662/ksdim.2021.17.4.077

Spring Boot 기반의 오픈소스 소프트웨어 보안 취약점 및 패치 정보 제공 웹 어플리케이션 개발*

심 완** · 최 웅 철***

Spring Boot-based Web Application Development for providing information on Security Vulnerabilities and Patches for Open Source Software

Sim, Wan · Choi, WoongChul

〈Abstract〉

As Open Source Software(OSS) recently invigorates, many companies actively use the OSSes in their business software. With such OSS invigoration, our web application is developed in order to provide the safety in using the OSSes, and update the information on the new vulnerabilities and the patches at all times by crawling the web pages of the relevant OSS home pages and the managing organizations of the vulnerabilities.

By providing the updated information, our application helps the OSS users and developers to be aware of such security issues, and gives them to work in the safer environment from security risks. In addition, our application can be used as a security platform to greatly contribute to preventing potential security incidents not only for companies but also for individual developers.

Key Words : Open Source Software(OSS), Common Vulnerabilities and Exposures(CVE), Common Vulnerability Scoring System(CVSS), Common Weakness Enumeration(CWE)

Ⅰ. 서론

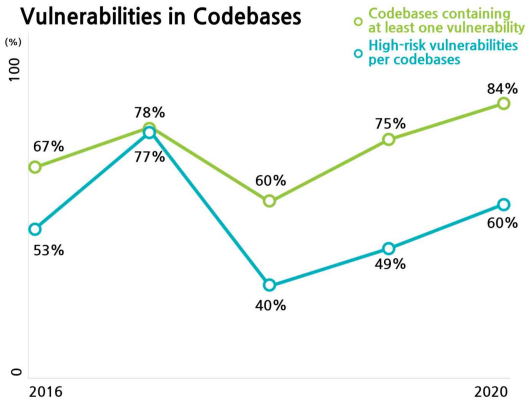
오픈소스 소프트웨어(Open Source Software, OSS)는 커뮤니티를 통한 빠르고 유연한 개발, 높은 호환성과 유연성, 비용 절감을 장점으로 전 세계 개발자와 더불어 기업에서까지 사용되고 있다. 많은 개발자와 전문가들이 OSS 개발에 참여하기 때문에 신뢰성과 안정성이 높이 평가되지만, 보안성은 별개의 문제다. 1,200개 이상의 상용 코드베이스를 대상으로 조사한 시놉시스의 2019 오픈소스 보안과 위험 분석 보고서에 따르면, 스캔 된 코드베이스 중 96%는 OSS를 포함했으며, 평균적으로 298개의 OSS 컴포넌트를 사용한 것으로 나타났다. 또한, 60%의 코드베이스에서 적어도 하나 이상의 OSS 취약점을 가지고 있다고 설명했다[1]. 추가로 2021년 발표된 자료에서는 현재 그

* 이 논문은 2019년도 광운대학교 교내학술연구비 지원에 의해 연구되었음
** 광운대학교 컴퓨터소프트웨어학과(제1저자)
*** 광운대학교 컴퓨터소프트웨어학과 교수(교신저자)

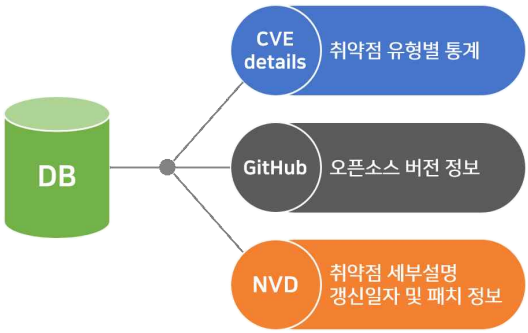
비율이 84%로 나타나, 매년 증가하는 추세다[2]. Michael Meike는 오픈소스 WMCSs(Web Content Management Systems)를 예로 들며, 만약 공격자가 어떤 OSS의 보안 취약점을 발견한다면, 해당 OSS가 사용되고 있는 모든 어플리케이션이 공격당할 위험이 있다고 경고했다[3]. 다시 말해 기업의 관리체계가 없는 무분별한 OSS 사용은 오히려 해킹 위협에 따른 보안 사고를 초래할 수 있다. 실제로 원자력 발전소에서는 디지털 기술이 발전하고 아날로그 장비의 노후화, 부품 단종에 따른 유지보수의 어려움 등의 문제에 따라 디지털 기반의 설비와 통신망의 사용이 원자력 발전소의 계측제어시스템에 점차 확대되었다[4]. 이처럼 보안 사고가 해당 국가 전체에 막대한 피해를 가져다 줄 수 있는 원전과 같은 곳은 더욱 각별한 주의가 필요하다.

미국에서는 국토안보부의 책임 하에, MITRE社, NIST를 중심으로 국가 취약점 DB인 NVD(National Vulnerability Database)를 운영하여 효율적으로 통합 관리하고 있다[5]. 이로써 SCAP(Security Content Automation Protocol) 표준 기반 데이터를 통해 취약점 관리, 보안 측정 및 규정 준수의 자동화를 가능하게 한다. 취약점 식별자 체계인 CVE(Common Vulnerabilities and Exposures)를 통해 소프트웨어와 펌웨어의 취약점들을 파악하고 분류할 수 있다. 취약점 분석 사이트인 'CVE Details'에서는 벤더, 제품, 버전별 검색 기능을 통해 원하는 CVE 정보를 쉽고 빠르게 찾을 수 있다. 그러나 NVD와 CVE-details 모두 취약점 관리를 위한 실질적인 기능을 제공하진 않는다. 조회만 가능하기 때문에 새로운 취약점 정보를 빠르게 파악할 수 없어, 보안 사고에 즉각적인 조치가 어렵다.

본 논문에서는 웹 크롤링을 활용해 실시간으로 OSS에 대한 CVE 정보를 수집하여 DB를 구성하고, 새로운 보안 취약점이 보고되는 즉시 사용자에게 알리는 OSS 보안 취약점 및 패치 정보 제공 웹 어플리케이션을 소개하고자 한다. 본 어플리케이션은 최근 OSS 활성화에 따라 OSS 기반 환경에서 안전성을 확보하고 보안 취약점 정보를 상시 모니터링이 가능하게 하도록 고안되었다. 최신 정보를 이메일을 통해 알려줌으로써 서비스 개발자가 사용 중인 OSS를 항상 최신으로 유지하고 보안 위협으로부터 안전할 수 있도록 돕는다.



<그림 1> 시놉시스에서 조사한 오픈소스 취약점 비율[2]



<그림 2> 크롤링을 통한 취약점 및 패치 정보 DB 구현

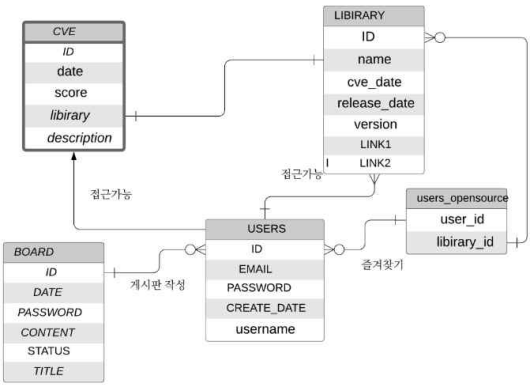
II. 구현 내용과 기능

본 어플리케이션은 국내 기업에서 많이 활용되는 OSS 80여 개를 선정하여, 크롤링을 통해 깃허브

(GitHub)에서는 해당 OSS의 버전 정보, NVD와 CVE-details에서는 취약점에 관한 전반적인 정보를 수집해 DB에 저장한다.

Spring Boot를 통해 서버와 DB를 관리하고, ReactJS를 통해 UI/UX를 구성하여 웹 어플리케이션을 구현하였다. 검색을 통해 원하는 OSS를 선택하여, 보고된 CVE 리스트와 세부 정보를 확인할 수 있고, 날짜 또는 점수 순으로 정렬하여 조회할 수도 있다. 크롤링을 통해 얻은 OSS 취약점 데이터와 회원, 게시글 정보는 MySQL을 이용하여, 관계형 데이터베이스를 구축해 분산하여 저장하였다. 기능에 맞게 테이블들을 연관 지어 다음과 같이 설계하였다.

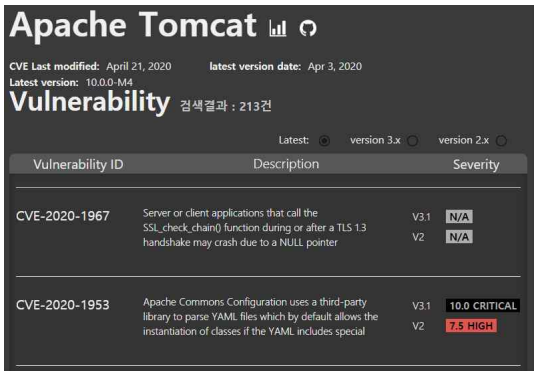
또한, 가입된 회원이 지정한 OSS에 대하여 크롤링을 통해 신규 취약점 보고가 확인될 시, 이메일 알림을 발송하도록 구현하였다.



<그림 3> 어플리케이션 전체의 ERD 다이어그램

2.1 취약점 정보 페이지

이 페이지를 통해 사용자는 원하는 OSS를 검색하여 모든 CVE 취약점 정보를 확인할 수 있다. GitHub을 참고하여, 현재 해당 OSS의 가장 최신 버전을 먼저 확인하여, 사용자가 현재 사용 중인 OSS를 최신화하는 것에도 큰 도움을 준다. 이와 더불어 현재까지 공개된 모든 CVE 취약점을 보고일자 순, 위험도(severity) 순으로 정렬하여 사용자에게 제공한다. NVD와 CVE-details에서 제공한 데이터를 본 어플리케이션 내에서 조회 및 검색이 모두 가능하기 때문에, 여러 OSS의 취약점 정보를 쉽고 빠르게 확인할 수 있다.



<그림 4> 취약점 정보 페이지 캡처 화면

2.2 취약점 세부 정보 페이지

데이터는 NVD에서 제공하는 취약점 정보를 기반으로 구성되었으며, 사용자는 취약점에 대한 설명과 패치 정보를 확인할 수 있다. 이를 통해 해당 취약점에 대한 자세한 정보와 더불어 해당 취약점을 실제로 시연해 볼 수 있는 코드나 해결하기 위한 구체적인 방안을 확인할 수 있다. 또한, 취약점 위험도 평가 척도인 CVSS(Common Vulnerability Scoring System)를 제공한다. CVSS는 미국의 비영리조직인 FIRST(Forum of Incident Response and Security Teams)에 의해 관리되고 있으며 전반적인 소프트웨어 및 하드웨어 취약점에 대한 주요 특징을 파악하고 심각성을 반영하는 수치 점수 산출 방법이다[6]. 사용자는 CVSS 점수를 참고하여, 취약점 관리 프로세스를 적절하게 평가하고, 우선순위를 지정할 수 있을 것이다.

<그림 5> CVE 세부내용 조회 캡처 화면

추가적인 리포트를 원하는 사용자들을 위해 해당 CVE에 관련된 더 자세한 정보, 보안 약점 정보인 CWE(Common Weakness Enumeration) 관련 사이트의 링크를 제공한다.

2.3 즐겨 찾기

앞서 언급한 3개의 사이트에서 취합한 정보를 하나의 페이지에서 출력하고, 정보의 출처가 되는 사이트의 링크를 두었다. 사용자는 이메일을 통해 회원가입할 수 있다. 가입한 회원은 즐겨찾기 기능을 통해 원하는 OSS에 대한 간략한 정보를 메인 페이지에서 확인할 수 있다. 또한, 최근 7일 이내 업데이트가 있는 OSS를 강조 표시하여 더 쉽게 최신 정보를 받을 수 있도록 하였다.

또한 즐겨찾기로 등록한 OSS에 대해, 지속적인 크롤링을 통해 신규 취약점 보고가 업데이트되면 등록된 이메일에 알림 메시지를 발송한다. 이를 통해, 보다 쉽게 OSS를 관리하고 최신화할 수 있다.

2.4 게시판

현재 DB에 추가된 80여 개의 OSS 이외에 원하는 OSS에 대해서 게시판 기능을 통해 추가 요청할 수 있다.

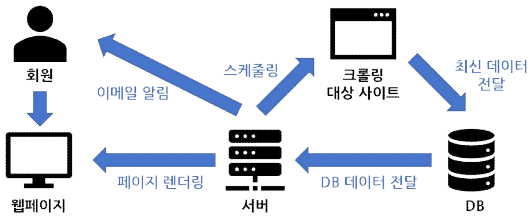
<그림 6> 즐겨찾기 기능 캡처 화면

<그림 7> 게시판 기능 캡처 화면

게시글은 기본적인 제목과 내용 이외에도 상태를 확인할 수 있도록 하였다. 상태는 관리자가 진행 상황에 따라 '접수', '처리 중', '완료' 3가지 상태 중 하나로 표시한다. 이를 통해 자신이 요청한 OSS 추가 작업의 진행 상황을 실시간으로 확인할 수 있다. 이외에도 기타 문의 사항을 게시판을 통해 작성한다.

2.5 이메일 알림기능

Spring Boot의 스케줄링 기능을 통해 주기적으로 새로운 정보를 크롤링한다. 이를 통해 DB 내의 취약점 정보를 최신 데이터로 유지할 수 있도록 하였다. NVD에서 새로운 보안 취약점이 추가되거나, 깃허브(Github)에서 최신 업데이트 정보가 추가되면, 업데이트되었다는 정보를 담아 해당 OSS를 즐겨찾기 등록한 회원에게 이메일로 알려준다. 이를 위해 Java에서 제공하는 API를 활용하였다.



<그림 8> 어플리케이션 시스템 도식도

<표 1> 상위 3개의 보안취약점

순위	OSS	CVE/CVSS/CWE
1	cron-utils	CVE-2021-41269 CVSS: 10.0 CRITICAL CWE: CWE-94
2	vm2	CVE-2021-23449 CVSS: 10.0 CRITICAL CWE: CWE-915
3	ssh2	CVE-2020-26301 CVSS: 10.0 CRITICAL CWE: CWE-78

Ⅲ. 향후 연구 방향

OSS 보안 취약점과 패치 정보에 대한 데이터는 현재 CVE-details와 NVD만을 크롤링하고 있다. 그러나 Guido Schryen은 NVD가 CVE 취약점에 대한 포괄적인 데이터베이스를 제공하는 반면에, NVD 분석가가 추가한 취약점 특성에 대해선 좀 더 주의를 기울여야 한다고 설명했다[7]. 따라서, 더 다양한 OSS 보안 취약점 리포트 제공 사이트를 크롤링하여 사용자에게 전달함으로써, 더 상세하고 정확한 분석 자료를 통해 어플리케이션의 가치를 높일 수 있을 것이다. 김규형에 따르면, 과거 화상회의 서비스 업체 줌(Zoom)은, 과거 보안 업데이트 당시 보안 문제점들을 꾸준한 업데이트를 통해 해결하고 있음을 블로그를 통해 전달하였다[8]. 이와 같은 사례를 통해 알 수 있듯이, 더 많은 자료 공개를 통해 사용자가 자신의 소프트웨어의 신뢰도를 높이는데 기여할 수 있을 것이다.

대표적인 OSS 취약점 관리 업체인 WhiteSource가 2021년 11월까지 공개한 최근 위험도 상위 3개의 보안 취약점 데이터베이스 자료에 따른 관련 CVE/CVSS/CWE는 다음과 같다[9].

OSS에 있어 위협이 되는 것은 취약점 문제만이 아니다. 오픈소스 라이센스 충돌에 대한 위협도 존재한다. 시놉시스의 2021 오픈소스 보안과 위험 분석 보고서에 따르면, 2020년에 검사된 코드베이스의 65%에 라이센스가 충돌하는 요소가 포함됐다. 특히 그중, 약 3/4 정도의 코드 베이스가 GNU General Public License(GPL) 및 그 변형에 대해 라이센스 충돌이 있었다. 예시로, 다수의 GNU 프로그램에 대한 저작권을 가지고 있는 FSF는, 위반자에 대해 직접 시정을 요구할 수 있는 지위에 있다. 그리하여 이들은 위반사항을 수집하고, 증거를 확보한 뒤 위반 기업들로 하여금 관련 소프트웨어, 제품, 웹사이트 등을 통해 관련 라이센스와 소스코드를 제공하도록 조치를 취하기도 한다[10]. 따라서, 직접 코드 베이스 전체의 라이센스와 충돌하는 OSS가 포함된 경우와 같이 언제나 잠재적인 라이센스 충돌의 위험성에 대해 경계해야 할 필요가 있다. 실제로, 국내 기업을 포함한 14개 기업에서 BusyBox의 오픈소스 라이센스인 GPL 조건을 위반하여, 막대한 금전적 배상금을 지급하는 조건으로 합의한 사례가 존재한다.

따라서 오픈소스 라이센스 및 특허에 대한 충돌을 검사하는 기능을 추가함으로써, 기업의 잠재적인 법적 및 금전적 피해를 예방할 수 있을 것이다. 다양한 산업 군에서 OSS가 널리 사용되고 있는 만큼, 이에 대한 중요성은 날로 커질 것이다.

Ⅳ. 결론 및 기대효과

OSS 관리자를 대상으로 한 스니크(Snyk)의 설문조

사에 따르면, OSS 관리자의 44%가 OSS에 대한 취약점 분석을 수행하지 않는다고 한다[11]. OSS의 편리함에 가려 보안에 안전할 것이라는 안일한 인식, 취약점 분석 역량 부족이 주요 원인일 것이다.

현재 선행 관리체계로써 AhnLab, Black Duck 등이 있지만, 보안 취약점에 대한 분석력이 상대적으로 떨어지고, 실질적인 패치 정보 제시에 부족함이 있다. 또한, CVE의 업데이트 사항을 즉각적으로 알 수 없다는 단점이 있다. 이에 따라 여러 기업들은 이미 자체적인 취약점 분석 부서를 신설하거나, 분석 시스템을 개발하여 운용 중에 있다[11].

- 마이크로소프트는 Semmle의 취약점 분석 기술을 Github에 통합시켜 자사 소스코드의 취약점을 검사한다.
- 구글은 세계 최고의 취약점 분석 팀(Project Zero)을 운영하여 OSS 프로젝트를 포함한 유명 소프트웨어의 취약점을 분석하여 공개한다.
- 삼성은 AVAS(Automated Vulnerability Analysis System)을 개발하여 OSS의 취약점 분석을 자동화하고 있다.

본 어플리케이션은 주기적인 스케줄링을 통해 OSS의 보안 취약점 현황을 주시하고, 지속적인 관리가 가능하게 하여, 취약점에 대한 빠른 탐지 및 대응을 할 수 있도록 한다. 또한, 사용자가 즐겨찾기 기능을 통해 등록한 OSS에 대한 최신 업데이트 정보를 실시간으로 확인할 수 있으므로, 사용하는 OSS를 항상 최신 버전으로 유지할 수 있도록 돕는다. 이를 통해 기업 입장에서는 이를 보안 플랫폼으로써 활용하여, 잠재적인 보안 사고를 예방하는 데 큰 도움이 될 것으로 기대한다.

참고문헌

[1] Synopsys Cybersecurity Research Center, “2019 Open Source Security and Risk Analysis,” Synopsys, 2019, p.4.

[2] Synopsys Cybersecurity Research Center, “2021 Open Source Security and Risk Analysis Report,” Synopsys, 2021, p.10.

[3] M. Meike · J. Sametinger · A. Wiesauer, “Security in Open Source Web Content Management Systems,” IEEE Security & Privacy, Vol. 7, No. 4, 2009, pp.44-51.

[4] 정성민, “원전 안전계통의 사이버보안 위협 및 대응,” 디지털산업정보학회 논문지, 제16권, 제1호, 2020, pp.99-109.

[5] 김동진 · 조성제, “국가 DB 기반의 국내외 보안취약점 관리체계 분석,” Internet and Information Security, 제1권, 제2호, 2010, pp.130-147.

[6] 김민철 · 오세준 · 강현재 · 김진수 · 김휘강, “공개 취약점 정보를 활용한 소프트웨어 취약점 위험도 스코어링 시스템,” 정보보호학회 논문지, 제28권, 제6호, 2018, pp.1449-1461.

[7] Guido Schryen, “Is Open Source Security a Myth?,” Communications of the ACM, Vol. 54 No. 5, 2011, pp.130-140.

[8] 김규형 · 최윤성, “줌의 보안 취약점 분석과 보안 업데이트 결과 비교,” 디지털산업정보학회 논문지, 제16권, 제4호, 2020, pp.55-65.

[9] “Open Source Vulnerability Database,” WhiteSource, 2021년 11월 22일, https://www.whitesourcesoftware.com/vulnerability-database, 2021년 11월 22일 접속.

[10] 이철남, “오픈소스 라이센스 분쟁사례의 분석과 기업의 대응방안에 관한 고찰,” 정보과학회지, 제26권, 제7호, 2008, pp.22-36.

[11] 류원옥 · 조수형 · 이승윤, “오픈소스 보안취약점 관리를 통한 안전한 오픈소스 사용 방안,” 2021년도 한국통신학회 하계종합학술발표회 논문집, 2021, pp.854-855.

■ 저자소개 ■

심 완
Sim, Wan

2015년 3월~현재
광운대학교 컴퓨터소프트웨어학과

관심분야 : 웹, 데이터네트워크, 보안
E-mail : 0420syj@naver.com

최 웅 철
Choi, WoongChul

2002년 9월~현재
광운대학교 컴퓨터소프트웨어학과
교수
2001년 5월 Univ. of Illinois, Ph.D.,
Computer Science
1991년 2월 서울대학교 컴퓨터공학과(공학석사)
1989년 2월 서울대학교 컴퓨터공학과(공학사)

관심분야 : 데이터네트워크, 클라우드컴퓨팅,
네트워크 보안
E-mail : wchoi@kw.ac.kr

논문접수일 : 2021년 11월 24일
수 정 일 : 2021년 12월 10일
게재확정일 : 2021년 12월 20일