Journal of Knowledge Information Technology and Systems

ISSN 1975-7700

http://www.kkits.or.kr

A Study on the Activation of SW Security Test through Crowdsourcing

Ji-Eun Choi[1], Yu-Jin Jeon[2], Hwansoo Lee[*2]

[1]*Interdisciplinary Graduate Program in IT Law, Dankook University*
[2]*Department of Convergent Security, Dankook University*

ABSTRACT

Continuous cyber threats are on the rise for critical information from public institutions and private companies. However, the attacks by hackers are not predictable and continue to evolve, and there is a limitation in current security test methods to develop a reliable information system because of time and cost issues. As a new way to strengthen information system security, this study suggests a security test method with utilizing crowd SW testing technique, which is attracting attention as a new software testing methodology. Crowd SW security testing has the advantage of detecting the fault and strengthening security verification on various systems through a large number of people's participation. It can also be a solution to the time and cost problems posed by existing security testing. Utilizing the method in security test contribute to the improvement of information systems' reliability and quality through checking security vulnerabilities. In order to activate the cloud SW security test, it is necessary to apply the security test for each stage of development and to test public and leakage organizations. If crowd SW security testers are fostered at the national level, it will contribute to strengthening SW security and solving the manpower shortage problem of SW testing industry.

KEYWORDS: Crowdsourcing, SW security, Security testing, Web, Vulnerability

ARTICLE INFO: Received 28 September 2018, Revised 28 October 2018, Accepted 7 December 2018.

[*]Corresponding author is with the Department of Convergent Security, Dankook University, 152, Jukjeon-ro, Suji-gu, Yongin-si, Gyeonggi-do, KOREA.
E-mail address: hanslee992@gmail.com

1. 서 론

최근 공공기관 및 민간 기업 정보시스템의 보안 취약점을 이용한 사이버 공격이 지속적으로 발생하고 있다. 특히 사이버 공격의 약 75%가 SW 자체의 보안취약점을 악용하는 것으로 웹사이트 공격 비중이 높은 추세이다[1]. 웹사이트는 불특정 다수가 쉽게 접근할 수 있는 특성상 외부 공격에 쉽게 노출되고 웹서버, 웹 애플리케이션 서버 등 웹 관련 SW 자체의 보안취약점을 이용한 공격에 대한 대응이 어려운 실정이다. 또한, 사이버 공격은 공격유형이 정형화되어 있지 않기 때문에 이에 대응하기 위한 완전한 보안성 테스트는 존재할 수 없다. 따라서 SW 취약점 분석에서는 가급적 다양한 분야의 많은 항목을 점검과정에 포함시켜야 할 필요성이 있다.

현재의 SW 보안 테스트는 비용 및 시간의 제약으로 인하여 안정적인 시스템을 구축하는 데 한계가 있으며, 개발 이후 유지보수 또한 제대로 이루어지지 못하고 있는 실정이다. 최근 부각되고 있는 크라우드 테스팅은 이에 대한 대안을 제시한다. 크라우드 테스팅은 크라우드 소싱에 기반한 소프트웨어 테스팅 기법으로 제어된 환경에서 표준 프로세스에 따라 크라우드 테스터가 테스트를 수행한다. 기존 소프트웨어 테스팅에 비해 비용은 절감되고, 다수의 테스터 참여로 결과에 대한 신뢰성을 높여주기 때문에 최근 주목받고 있다. 더욱이 보안 분야에도 적용이 확대되어 가는 추세로 정보시스템의 보안성 강화에도 기여할 수 있다.

이에 본 연구에서는 크라우드 테스팅을 기반으로 한 보안성 테스팅 소개 및 확산 방안을 제시하여 향후 정보시스템의 보안성 강화에 기여하고자 한다. 이를 위해 2장에서는 국내외 SW 보안사고 대응 현황과 기술 동향을 살펴보고, 3장에서는 크라우드소싱 기반의 SW 보안테스트의 개념과 관련 사례를 통한 이의 적용 가능성을 살펴본다. 4장에서는 크라우드 SW 보안테스트 적용 방안 및 활성화를 위한 정책적 대안을 살펴보고, 5장에서 본 연구의 의의와 한계점을 제시한다.

2. SW 보안사고 대응 현황

2.1 국내 관련 법·정책

전 세계에서 매년 발생하는 해킹 등의 사이버 범죄로 인한 손실은 450조 원에 이르고 있으며. 민간기업의 경우 산업기술 등을 포함하는 기밀정보가 유출되어 유·무형적으로 상당한 피해를 보고 있을 뿐만 아니라 개인정보보호법에 따라 개인정보 유출에 대한 법적 책임까지 부담하고 있다[2]. 이는 기업 경쟁력 저하, 기업 이미지 하락, 수익감소 등 심각한 타격을 주어 결과적으로 산업 전반에 악영향을 미치게 된다[3].

정보시스템 보안사고의 약 75%가 SW의 취약점을 공격하여 발생함에 따라 최근 관련 법령에서는 SW 개발 보안성이 강화되는 추세이다. 공공부문 정보보안 관련 법률로는 「국가정보화 기본법」, 「전자정부법」, 「정보통신기반 보호법」, 「개인정보 보호법」과 대통령 훈령인 「국가사이버안전관리규정」 등이 있다. 또한, 민간부문 정보보안 관련 법률은 「국가정보화 기본법」, 「정보통신망 이용촉진 및 정보보호 등에 관한 법률」, 「정보통신기반 보호법」, 「전자서명법」, 「신용정보의 이용 및 보호에 관한 법률」 등이 있다.

국내의 경우에는 기관에 대한 보안인증을 통해 보안성 강화를 꾀하고 있는데 대표적으로 정보보호 관리체계(ISMS, Information Security Management System)와 공통평가 기준(CC, Common Criteria)이 있다. 정보보호 관리체계란 정보통신망의 안전성 및 신뢰성 확보를 위한 관리

적·기술적·물리적 보호조치를 포함하는 관리체계를 수립·운영하고 있는 자에 대한 인증제도로 종합적 관리체계가 인증심사기준에 적합한지 여부를 한국인터넷진흥원으로부터 인증받도록 하고 있다. 공통평가 기준은 IT 제품 및 사이트의 정보시스템에 대한 보안성을 평가하기 위한 평가 기준이다. 이는 국가 상호 간 평가결과 인정에 필요한 표준 평가 기준의 제정을 목적으로 미국, 유럽 등의 여러 평가 기준을 참조하여 개발하였다. CC는 1부 소개 및 일반 모델, 2부 보안 기능 요구사항(Security Function Requirement), 3부 보증사항(Security Assurance Requirement)으로 구성되어 있다.

2.2 국외 동향

국제표준화기구(ISO)는 취약점(Vulnerability)을 “하나 이상의 위협에 의해 잠재적으로 악용될 수 있는 자산 또는 통제의 약점”이라고 정의한다. 즉, 소프트웨어의 취약점은 소프트웨어가 가지고 있는 약점을 의미하고 실제로 공격자의 입장에서 가장 먼저 수행하는 것이 취약점 분석이다[4].

미국 국토안보부(DHS : Department of Homeland Security)가 관리하는 CWE(Common Weakness Enumeration)는 다양한 관점으로 수집된 보안취약점을 분석하여 사전식으로 분류한 체계이다. 해당 취약점에 대한 정의, 설명, 플랫폼, 예제코드 등의 정보를 제공하고 매년 25개의 CWE 항목을 발표하여 SW 개발에 참고할 수 있는 정보를 제공한다.

국제적인 오픈소스 웹 보안 비영리 기구(OWASP : The Open Web Application Security Project)는 웹 애플리케이션의 개발자, 설계자, 아키텍트, 운영자 혹은 기관들에게 보안 취약점으로 인한 영향을 알리고 개발시 취약점 점검 표준을 제공하는 것을 목적으로 <표1>과 같이 각 분야 전문가들의 의견을 수렴하여 4년에 한 번 가장 위험한 10가지의 보안취약점을 공개하고 있다[5].

표 1 OWASP Top 10 (2010 - 2017)[5]
Table 1. OWASP Top 10 (2010 - 2017)[5]

	2010년	2013년	2017년
A1	인젝션	인젝션	인젝션
A2	크로스 사이트 스크립팅(XSS)	인증 및 세션 관리 취약점	취약한 인증
A3	인증 및 세션 관리 취약점	크로스 사이트 스크립팅 (XSS)	민감정보 노출
A4	취약한 직접 객체 참조	취약한 직접 객체 참조	XML 외부 개체(XXE)
A5	크로스 사이트 요청 변조 (CSRF)	보안 설정 오류	취약한 접근 통제
A6	보안 설정 오류	민감 데이터 노출	보안 설정 오류
A7	불안정한 암호 저장	기능 수준의 접근 통제 누락	크로스 사이트 스크립팅(XSS)
A8	URL 접근 제한 실패	크로스 사이트 요청 변조 (CSRF)	불안정한 Deserialization
A9	미흡한 전송 계층 보호	알려진 취약점이 있는 컴포넌트 사용	알려진 취약점이 있는 컴포넌트 사용
A10	검증되지 않은 리다이렉트 및 포워드	검증되지 않은 리다이렉트 및 포워드	불충분한 로깅 및 모니터링

2.3 SW 취약점 진단 기술

보안성 테스트는 소프트웨어의 취약점을 분석하는 과정인데 이러한 취약점 분석은 내부에서 소프트웨어의 소스코드를 분석하는 방법인 정적분석(SAST : Static Application Security Testing)과 외부에서 공격자의 공격환경을 인위적으로 구축하여 취약점을 분석하는 동적 분석(DAST : Dynamic Application Security Testing)으로 나뉘어 진다[6]. 완성된 소프트웨어를 바탕으로 이루어지는 동적 분석과는 다르게 정적분석의 경우에는 개발자가 초기 개발단계에서부터 개발 중인 소프트웨어의 취약점을 실시간으로 분석하고 보완할 수 있다는 장점이 존재한다. 개발단계에서 보안취약점을 제거하는 것이 운영단계에서 제거하는 것보다 30배가량 비용이 절감된다는 미국 국립표준기술연구소(NIST)의 연구결과는 정적분석의 중요성을 단적으

로 보여주는 예이다. 정적분석을 수행함에 있어서 가장 확실하고 정확한 방법은 사람이 직접 소스코드를 점검하는 수동적인 방법이지만 이 방법은 시간 소모가 상당하므로 현실적으로 불가능하다[7]. 이 때문에 짧은 시간에 다양한 취약점을 분석해주는 자동화된 보안취약점 진단 도구를 사용하는 것이 일반적이다. 미국 NIST의 SAMATE 프로젝트는 보안취약점을 진단할 수 있는 공개용 진단 도구의 목록을 제공하고 있는데, 소스코드를 분석하는 보안 취약점 진단 도구의 유형은 <표 2>과 같다.

표 2 취약점 진단 도구
Table 2. Vulnerability Diagnosis Tool

구분		
Languages	c/c++	Cppcheck, Double Check, Flaw finder, Object Center 등
	c#	Attack Flow, dotTESTTM, Coverity, Roslyn Security 등
	JAVA	Attack Flow, bug Scout, Jtest®, Find Bugs 등
Network Scanner	ISS	통신 서비스, 운영체제, 라우터의 취약성 평가
	Nessus	Linux 및 기타 유닉스 계열 OS에 사용
	NT OBJECTives	웹 응용프로그램 취약점 스캐너인 NTOSpider를 제공
	Qualys	보안 취약성에 대한 네트워크 및 내부 시스템 검사
Web	Nikto, N-Stealth	웹 서버와 애플리케이션에 있는 취약점 데이터베이스를 이용하여 만든 취약점 스캐너
	Absinthe	SQL 인젝션 취약점 툴
	Sqlmap	오픈 소스 툴로 최근 SQL 인젝션 자동화 공격에 사용
	Acunetix	웹 애플리케이션의 취약점을 찾는 휴리스틱 웹 취약점 스캐너
	AppScan	위치파이어에서 만든 웹 애플리케이션 취약점 스캐너

2.4 기존 보안 취약점 테스트의 한계

기존 SW 보안테스트는 시간 및 인력의 한계, 예산 부족 등으로 인하여 많은 항목을 점검하는 것이 어렵고, 체계적인 테스트가 이루어지기에 한계가 있다 [8]. 테스트 인력의 부족과 보안성 자체점검, 관련 업체의 보안성 테스트 등은 평가의 신뢰성·객관성을 도출해내기 어렵다. 이는 아직까지 국내 기업들이 기업의 이익 창출과 관련된 활동에 집중하고 있으며, 보안은 비용으로 인식하고 있는 현재 상황에서 해결되기 어려운 문제이다. 특히 공공기관 및 대기업에 비해 보안에 투자할 여건이 되지 않는 중소기업의 경우에는 보안이 더욱 취약한데, 중소기업의 보안관리 비용은 평균 3,530만원으로 조사되며 이는 기업 전체 매출액의 0.24%에 불과한 수치이다[9]. 대기업 및 공공기업의 경우에도 인소싱의 테스팅을 진행할 때 개발 소프트웨어에 대한 이해도 및 전문성이 높은 장점이 있으나, 내부 테스터의 관리 비용이 높고, 소수 인력이 테스트를 진행하므로 시간이 많이 소요되며, 상대적으로 적은 보안성 검사가 진행되는 한계가 있다. 또한, 인소싱 테스팅은 개발자의 보안성 검토로 인하여 객관적인 결과를 도출해내기 어려운 점이 있다. 아웃소싱 테스팅의 경우 인소싱 테스팅에 비해 보안성과 관련한 객관적인 결과를 도출할 수 있지만, 비용 및 인력이 제한되어 있기 때문에 크라우드 테스팅에 비해 다양한 보안성 검토를 진행하지 못하게 된다. 이에 반해 크라우드 테스팅은 기존 인소싱 테스팅, 아웃소싱 테스팅에 비하여 비용을 줄이고, 시간을 단축하며 다양한 보안성 테스트의 결과를 가져온다.

3. 크라우드 소싱 기반 SW 보안테스트

3.1 크라우드 SW 테스트

크라우드 테스팅은 소수의 전문가보다 다수의 참여에 의한 판단이 객관적이고 나은 결과를 가져온다는 크라우드 소싱 기반의 테스팅 기법이다. 즉, 크라우드 테스팅이란 기업 및 조직이 프로토타입의 소프트웨어 제품에 대한 테스트를 기업 및 조직 내부의 인프라를 통해 자체적으로 진행하는 것이 아니라 다수의 전문가, 경력 단절자, 일반인 등 광범위하게 분포된 군중들에게 프로토타입의 소프트웨어를 사용하게 함으로써 피드백을 받는 신개념 아웃소싱 테스트 기법이다[10].

표 3 크라우드 테스팅 플랫폼[11]
Table 3. Crowd testing platform[11]

Task Domain	Platform	URL
Software Development	TopCoder	www.topcoder.com
	GetACoder	www.getacoder.com
Mobile App Development	AppStori	www.appstori.com
Small Coding Tasks	Bountify	www.bountify.co
Software Testing	uTest	www.utest.com
	99Tests	www.99tests.com
	TestBinds	www.testbinds.com
	Testbats	www.testbats.com
	Pay4bugs	www.pay4bugs.com
	Crowd Testers	www.crowdtesters.com.au
Mobile App Testing	TestFlight	www.testflightapp.com
	Mob4hire	www.mob4hire.com
	Testin	www.testin.com
Software Security Testing	Ce.Woo.Yun	ce.wooyun.org
	Bugcrowd	www.bugcrowd.com

Zogaj and Bretschneider (2013)는 독일의 크라우드 테스팅 업체인 'TestCloud'의 사례연구에 따라 크라우드 테스팅의 단계를 설명하였다[12]. 크라우드 테스팅은 첫 번째로 기능성, 성능, 안전, 유용성, 디자인 등 어떤 요소들을 테스팅 할 것인지 결정해야 한다. 두 번째로 기기, 운영체제, 브라우저 등 어떤 것이 테스팅의 대상이 될지 결정해야 한다. 세 번째 요소는 웹사이트 테스팅 시 테스터의 구성과 기간을 정해야 한다. 그 후 테스팅 가이드라인을 검토하여 테스팅 플랫폼을 실행한다. 여기서 테스트 될 대상 SW가 업로드되고 사람들은 이에 접근이 가능하다. 웹사이트 테스트가 실행되기 시작하면, 테스터들은 소프트웨어를 검토하고 버그가 있는지 확인하거나 디자인과 그 소프트웨어의 유용성을 평가할 수 있고, SW 개선을 위한 사항을 제시한다. 테스터가 버그를 찾아내면 전문 매니저들이 버그의 원인 등 취약점을 확인한다. 마지막으로 고객은 모든 버그가 등록되어있는 버그 리포트를 받고, 고객들은 모든 테스팅 과정을 추적하고 그들의 테스트 요구사항을 바꿈으로써 테스트 과정을 통제할 수 있다.

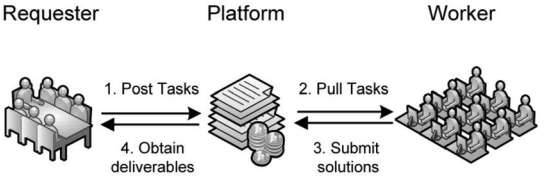


그림 1 크라우드 테스팅 단계[11]
Figure 1. Crowd testing process[11]

크라우드 테스팅은 비용적·시간적·결과적 측면에서 기존 소프트웨어 테스팅보다 효율적이다. 비용적 측면에서 크라우드 테스터에게 테스팅 성과비용을 지급해 고정된 고용비용을 절감하고, 실제 사용자의 다양한 테스트환경을 구축하므로 구축 비용 또한 감소한다. 시간적 측면에서 많은 테스터의 참여로 상대적으로 짧은 기간 동안 많은 요소를 검토할 수 있고, 기관에서 희망하는 테스팅 기간을 설정할 수 있다. 결과적 측면에서는 기업에서 원하는 다양한 조합의 프로젝트 환경 구축이 가능하기 때문에 소프트웨어 제품의 특성, 테스트 수행 기간, 테스트 인원, 경력 등을 고려하여 보안성 강화 테스트를 개별적인 특성에 맞게 진행할 수 있다[13]. 또한, 전문적인 테스팅 경력을 가진

크라우드 테스터 및 일반 테스터로부터 실제 환경에서 발생하는 다양한 결과가 도출될 수 있으며, 테스팅에 참여하는 테스터는 독립적으로 작업을 진행하기 때문에 결과에 대한 객관성을 확보할 수 있다[14].

표 4 크라우드 SW 테스팅 특징
Table 4. Crowd SW testing characteristics

구분		효과
강점	비용	· 고정된 고용비용 절감 · 테스트환경구축 비용 감소 · 관리 비용 절감
	시간	· 많은 테스터 참여로 시간 단축 · 테스팅 기간 설정 가능
	결과	· 예상하기 어려운 다양한 결과 도출 · 개별적 특성에 따른 다양한 조합의 환경 구축 및 다수의 결과 제공 · 테스팅 결과의 객관성 보장
약점	비용	· 테스터 확보 및 교육비용 필요 · 프로젝트에 배정된 테스터의 태도나 역량에 따라 추가 비용 발생 가능
	시간	· 테스터 변경 시 일정 연장 가능 · 테스팅 기간에 따른 결과 품질이 정비례하지는 않음
	결과	· 테스팅 결과 품질이 테스터 역량 및 구성에 의존적 · 결함의 원인에 대한 제시는 어려움

3.2 국내 사례

국내 크라우드 SW 테스트 대표 기업인 STA Consulting은 테스트 대상에 가장 적합한 방법으로 전문 인력 Pool을 활용하여 크라우드 테스팅을 제공하고 있는 국내 대표 기업이다[13]. 주로 SW의 사용성 및 품질 검증을 위한 테스트를 주로 수행하고 있어, 오류 최소화를 통한 간접적인 보안취약점 개선 효과는 있으나, 보안테스트 분야는 아직 활성화되어 있지 않다. 그러나 2016년에 수행한 한의학연구원의 모바일 SW(KOIN)에 대한 크라우드 테스팅과 일반 테스팅의 비교분석 결과를 살펴보면 그 실효성과 보안 분야 적용 가능성을 살펴볼 수 있다.

표 5 테스팅 결과 비교
Table 5. Comparison of testing results

구분		크라우드	일반
전체 결함 수		77	80
위험도	H	24	57
	M	43	19
	L	10	4
심각도	H	7	1
	M	32	58
	L	38	21

<표 5>와 같이 동일 SW에 대한 테스팅 결과 비교분석에 따르면 전체 발견 결함 건수에 있어서는 큰 차이가 없었다. 취약성이 높은 기능영역의 테스트에서는 소수의 전문 테스터들이 참여하는 일반 테스팅에서 더 많은 결함이 발견된 반면, 발견된 전체 결함에서 심각한 문제를 일으킬 수 있는 부분은 크라우드 SW 테스트에서 더 많이 발견되었다. 즉, 특정 영역에서의 테스트 품질은 일반 테스트가 우수할 수 있으나 SW 전반적인 품질 향상이나 안정성을 위해서는 크라우드 SW 테스트가 효과적일 수 있음을 보여준다.

3.3 해외 사례

크라우드 보안 테스팅은 크라우드 소싱에 기반을 둔 소프트웨어 테스팅 기법으로 제어된 환경에서 표준 프로세스에 따라 크라우드 테스터가 SW 보안 테스트를 수행하는 것이다. 기존 소프트웨어 테스팅에 비해 비용, 시간, 결과 품질 등에서 여러 장점이 있으나, 반대로 완성되지 않은 SW가 외부 테스터에 의해 평가됨에 따라 치명적 결함이 공개될 수가 있다는 점에서는 한계가 있을 수 있다. 따라서 이러한 관리를 위해서 테스터를 전문적으로 관리 공급하는 플랫폼의 역할 중요하며 해외의 경우 이러한 플랫폼이 점차 증가하고 있다.

3.3.1 Bugcrowd

Bugcrowd는 모바일, 데스크톱 및 웹사이트 보안성 테스트를 위한 다양한 그룹의 전문가를 테스터로 구성하고 있다. 현재 테스터의 수는 32,000명에 달한다. Bugcrowd는 취약점 분석의 기준을 위험정도에 따라 Critical, High, Medium, Low 4단계로 구분하고 있다. ‘Bugcrowd’의 2014년 10월 8일~22일 진행된 학습관리시스템을 운영하는 C사의 크라우드 테스팅 결과에 따르면 63명의 테스터는 322개의 테스팅 결과물을 제출하였으며 그중 59개의 보안과 관련된 문제점이 발견되었다. P1(Critical)에 해당하는 위험성이 높은 취약점은 발견되지 않았으나, P2(High)에 해당하는 취약점 27가지, P3(Medium)에 해당하는 취약점 8가지 등이 발견되었다. 발견된 취약점 중 위험성이 높은 P2(High)에 해당하는 것으로 캘린더, 파일 업로드 등에서 발견된 다수의 “저장 크로스사이트 스크립팅(stored XSS)”과 “반사 크로스사이트 스크립팅(reflected XSS)”, “이메일과 관련한 CSRF(Cross Site Request Forgery) 및 bypass 취약점” 등이 발견되었다. 발견된 보안취약점들은 점검을 통해 수정되거나, 보안에 영향을 주지 않는 버그로 판단하여 시스템 보안의 안전성을 개선하였다[15].

3.3.2 Openbugbounty

Openbugbounty(www.openbugbounty.org)는 2014년에 사이트 보안성 테스트를 위해 설립된 비영리 커뮤니티 재단이다. 총 3,964명의 테스터를 보유하고 있으며 테스터는 자신의 경력 및 프로필과 연락처를 커뮤니티에 자율적으로 등록한다. 웹사이트 운영자는 각 테스터의 프로필을 확인하고 이메일 또는 트위터를 통해 테스터에게 취약점 분석을 의뢰한다. 테스터는 의뢰받은 웹사이트에 대하여 자율적으로 보안성 테스트를 진행하고 취약점 분석 결과를 커뮤니티에 보고한다. 커뮤니티는 취약점이 보고되는 즉시 웹사이트 소유자와 구독자들에게 이를 통지하게 된다. 테스터에 의해 보고된 취약점에 대한 기술적인 세부사항은 바로 공개되지 아니하고 웹사이트 소유자가 보안성 패치를 진행할 시간을 제공한다. 웹사이트 소유자가 취약점 패치를 적용하였다면 보고서가 제출된 지 30일 이후에 테스터에 의하여 공개할 수 있고, 패치를 적용하지 않았다면 90일 이후에 공개할 수 있다. 취약점에 대한 세부사항의 공개 여부는 테스터의 재량이지만 일단 공개된 정보는 테스터가 삭제할 수 없다. 현재까지 공개된 테스터의 Report 수는 총 168,480건이며, 총 77,547건의 웹사이트 취약점을 파악하여 웹사이트 소유자들이 취약점을 패치하는 데 도움을 주었다. 파악한 취약점 중 약 66.24%의 취약점이 크로스 사이드 스크립팅(XSS) 이며, 이 중에는 WordPress 및 Amazon과 같은 대기업 또한 포함되어 있다.

4. 크라우드 SW 보안테스트 활성화 방안

4.1 보안테스트 적용 방안

마이크로소프트사는 7단계에 거쳐 보안성을 강화하는 방안을 제시하고 있다. 1) 교육(Traning) 단계는 소프트웨어 개발 보안 교육을 진행하고, 2) 분석(Requirement) 단계는 보안 및 정보보호에 대한 요구 수립, 소프트웨어의 질 향상 및 버그 정의, 보안 및 프라이버시 위험을 평가하도록 한다. 또한, 3) 설계(Design) 단계에서 설계 요구 분석, 공격의 영역 분석 및 위협모델링을 구축하고, 4) 구현(Implementation) 단계에서는 도구명세, 금지된 함수사용의 제한, 정적분석을 진행한다. 5) 시험(Verification) 단계에서는 동적 테스팅 및 퍼즈 테

스팅과 공격영역을 검토하여야 하며, 6) 운영(Release) 단계에서는 사고 대응계획 및 최종 보안성을 검토하고 기록을 보관하게 된다. 7) 개발 이후에 사고가 발생할 경우 사고 대응(Response)을 수행한다. 이러한 SW 개발단계에서 크라우드 SW 보안 테스팅이 적용될 경우 보안성 향상에 기여할 수 있다. 특히 시험(Verification) 단계 및 운영(Release) 단계에서 보안취약점과 관련한 다양하고 유의미한 결과를 나타낼 수 있으며, 사고 발생 후 대응(Response) 단계에서는 취약점을 검토하여 향후 보안사고의 재발을 방지하는 역할을 할 수 있다[16].

표 6 SW 개발단계별 적용 방안
Table 6. Application in SW development process

단계	실행
1. Training	· Core security training
2. Requirement	· Establish security requirements · Create quality gates/bug bars · Perform security and privacy risk assessments
3. Design	· Establish design requirements · Perform attack surface analysis/reduction · Use threat modeling
4. Implementation	· Use approved tools · Deprecate unsafe functions · Perform static analysis
5. Verification	· Perform dynamic analysis · Perform Fuzz testing · Conduct attack surface review
6. Release	· Create ac incident response plan · Conduct final security review · Certify release and archive
7. Response	· Execute incident response plan

4.2 정책적 지원 방안

먼저 공공기관 및 유출기관에 대한 SW 보안테스트 지원을 통해 활성화 가능할 것이다. 현행 법제에서는 사전점검에 관한 규정은 비교적 체계적으로 구성되어 있지만, 사후적 조치에 관한 규정은 부족한 현실이다. 「정보통신망 이용촉진 및 정보보호 등에 관한 법률」에서 이용자의 정보보호를 위해 침해사고의 예방 및 확산 방지를 위하여 취약점 점검, 기술 지원 등의 필요한 조치를 하도록 규정하고 있다. 그러나 사후 보안과 관련된 테스트 등을 구체적으로 명시하고 있지 않으므로 공공기관 및 유출기관에 보안성 점검을 의무화하도록 하여야 한다. 또한, 소프트웨어의 성격에 따라 크라우드 테스팅을 적용하는 방안을 고려할 수 있다. 다만 기업의 규모 및 유출 정도에 따라 유형을 분류하여 의무조치의 차별성을 두어야 할 것이다.

둘째로, 중소기업의 경우 보안테스트에 비용 투자가 어려우므로 이에 대한 지원을 통해서도 활성화 가능할 것이다. 중소기업의 보안성 강화를 위한 다양한 지원 사업이 진행되고 있다. 산업기술보호협회가 진행하는 “산업기술보호 보안닥터 사업”은 보안전문가가 중소기업의 보안관리 체계 및 분야별 보안활동에 존재하는 취약점을 점검하고, 보안성을 강화할 수 있는 방안을 제시한다. 또한, 보안사고가 발생한 기업에는 보안 현황을 점검하고, 보안대응 방안 및 교육을 지원하고 있다. 또한, 과학기술정보통신부가 지원하는 ‘소프트웨어 프로슈머 평가사업’은 경력 3년 이상의 소프트웨어 개발 및 테스트 경력자와 일반 사용자로 구성된 테스터가 중소기업 및 스타트업의 소프트웨어를 미리 사용하고 소프트웨어 전반에 걸친 이용만족도, 편의성, 기능 완성도, 디자인 등 품질과 시장성을 평가하고, 결함을 개선하도록 한다. 사업을 통해 소프트웨어의 결함이 43% 개선된 것으로 조사된다[17]. 이와 유사하게 중소기업의 보안점검을 위해 크라우드 기법이 활용된다면 보안 사고를 예방하는데 도움이 될 수 있다. 특히 크라우드 테스팅 사업은 많은 전문가 및 일반인 테스터들의 참여를 통해 중소기업·스타트업의 정보시스템에 대한 품

질향상과 더불어 관심을 제고하기 때문에 기업 운영에도 도움이 될 것으로 보인다.

셋째로, 최근의 일자리 확대와 관련하여 인력양성 사업 추진을 통해서도 크라우드 SW 보안 테스팅을 활성화할 수 있을 그것으로 예상한다. 크라우드 테스터는 교육을 통해 전문적인 테스터로 성장할 수 있으므로 일반인들도 참여할 수 있는 장점이 있다. 기존에는 일부 기업들이 이와 유사한 형태로 크라우드 테스터를 교육하여 필요한 기관에 매칭시켜주는 유료사업을 일부 운영하고 있으나, 크라우드 테스팅의 활성화를 위해서는 정부 차원의 확대된 인력양성 지원이 필요하다. 크라우드 테스터 인력양성의 지원은 테스터의 증가와 크라우드 테스팅을 활용하는 기관의 증가를 가져오며, 이는 소프트웨어 보안과 관련한 새로운 시장을 구축하고, 소프트웨어 산업 발전이라는 종합적인 효과가 있게 된다. 특히 크라우드 테스팅은 보안전문가뿐만 아니라 일반 테스터도 참여할 수 있기 때문에 경력단절 여성이나, 소프트웨어에 관심이 있는 자 또한 관련된 교육을 받고, 참여하여 새로운 일자리 창출에도 기여할 수 있다[18]. 크라우드 테스팅 산업의 발전은 많은 테스터의 참여와 크라우드 테스팅 활용 기업의 증가를 통해 보안성 검증 비용이 낮아지는 반면 소프트웨어의 보안성은 강화되는 효과를 가져온다.

5. 결 론

본 논문에서는 기존 보안성 테스트의 시간적·비용적 한계를 개선하려는 방안으로 크라우드 테스팅을 제시하였다. 크라우드 테스팅은 크라우드소싱에 기반을 두어 다수의 전문가 및 일반인이 테스트에 참여함으로써 기존 인소싱 테스팅과 아웃소싱 테스팅에 비하여 비용을 줄이고, 시간을 단축하며 다양한 보안성 테스트의 결과를 가져오기 때문에 소프트웨어 보안성과 함께 품질 향상에까지 기여한다. 크라우드 테스팅의 적용을 위해서는 우선 보안성 테스트 전반에 걸친 보안요건이 강화될 필요성이 있으며, 유출기관의 사고 재발을 방지하기 위한 테스팅의 의무화를 구체적으로 규정하는 것이 필요하다. 또한, 정부 차원의 크라우드 테스팅 플랫폼을 구축하여 크라우드 테스터 및 크라우드 테스팅 활용기관의 증가를 통하여 소프트웨어 보안과 관련한 새로운 시장을 구축하고, 결과적으로 보안성 테스트와 관련한 비용부담을 낮출 수 있다. 또한, 중소기업의 보안사고 예방을 위해 크라우드 테스팅을 지원하는 사업을 운영하여 보안 위험을 줄이고 결과적으로 국내 산업을 보호하는 효과를 가져올 수도 있다.

하지만 본 연구는 크라우드 테스팅을 적용하기 위한 구체적 방법론 및 시스템 유형에 대한 깊은 논의가 이루어지지 못한 한계가 있다. 본 연구에서는 사례 분석으로만 논의와 결론을 이끌고 있어 정량적 분석을 통한 연구결과의 객관성을 확보하지 못하였다는 점 또한 개선되어야 할 부분이다. 마지막으로 크라우드 테스팅 자체가 가지는 개방성으로 인해 보안성 검증에 다소 약점이 있을 수 있는데 이러한 부분에 대한 충분한 논의가 이루어지지 못한 것 또한 본 연구의 한계점이다. 따라서 향후 본 연구가 가지는 한계점을 보완할 수 있는 다양한 정책적 연구의 진행과 함께 정보시스템 품질 향상 및 보안성 강화에 기여할 수 있는 구체적인 논의가 이루어질 필요가 있다.

References

[1] T. Lanowitz, *Now is the time for security at application level,* Gartner, pp. 1-8, 2005.

[2] N. Losses, *Estimating the global cost of*

cybercrime, Centre for Strategic & International Studies, pp. 1-24, 2014.

[3] J-S. Jo, *Cyber-security enhancement strategy for the protection of corporate information assets*, Journal of global business research, Vol. 23, No. 3, pp. 1-22, 2011.

[4] S-J. Jang, and E-S. Choi, *A study on implementation of vulnerability assessment tool on the web*, Korean Institute of Information Scientists and Engineers Conference Proceeding. Vol. 34, No. 1D, pp. 82-85, 2007.

[5] OWASP, OWASP Top 10, 2017.

[6] J-H. Bang, and R. Ha, *Evaluation methodology of diagnostic tool for security weakness of e-GOV software*, The Journal of The Korean Institute of Communication Sciences. Vol. 38, No. 10, pp. 335-343, 2013.

[7] J. Yoon, and W-T. Sim, *An automatic network vulnerability analysis system using multiple vulnerability scanners*, Journal of KIISE : Computing Practices and Letters. Vol. 14, No. 2, pp. 246-250, 2008.

[8] Audit researcher, *Research on vulnerability check of public sector information system security control*, pp. 1-157, 2015.

[9] Small and Medium Business Institute, *KOSBI SME FOCUS*, pp. 14-17, pp. 1-24, 2014.

[10] B-S. Chai, S-G. Park, H-Y. Kwon, and C-K. Jeong, *Crowd sourcing as a control mechanism of software test outsourcing*, Korea Society of IT Services 2015 Spring Conference, pp. 83-86, 2015.

[11] K. Mao, K. Capra, M. Harman, and Y. Jia, *A survey of the use of crowdsourcing in software engineering*, UCL Department of Computer science, pp. 1-36, 2015.

[12] S. Zogaj, and U. Bretschneider, *Crowdtesting with test cloud-managing the challenges of an intermediary in a crowdsourcing business model,* Proceedings of the 21st European Conference on Information Systems, ECIS2013, pp. 1-15, 2013.

[13] STA Consulting, STA crowd testing service, pp. 1-27, 2014.

[14] X. Peng, *Collaborative software development platforms for crowdsourcing,* IEEE Software Technology, pp. 1-7, 2014.

[15] Bugcrowd, *CANVAS by instructure bugcrowd flex program results,* pp. 1-7, 2014.

[16] https://www.microsoft.com/en-us/sdl/process/requirements.aspx, Dec. 12, 2018.

[17] http://www.asiae.co.kr/news/view.htm?idxno=2016082309332023839, Aug. 2016.

[18] http://www.dt.co.kr/contents.html?article_no=2016060302109960718004, Jun. 2016.

크라우드소싱을 통한 SW 보안 테스트 활성화 방안 연구

최지은[1], 전유진[2], 이환수[3]

[1] *단국대학교 IT법학협동과정 석사*

[2] *단국대학교 융합보안학과 석사과정*

[3] *단국대학교 융합보안학과 조교수*

요 약

공공기관 및 민간 기업의 중요정보를 대상으로 하는 지속적인 사이버 위협이 증가하고 있다. 그러나 해커들로부터의 공격은 예측하기 어렵고 지속적으로 진화하고 있는 상황이며 현재의 보안성 테스트 방법은 비용 및 시간의 제약으로 시스템을 안정적으로 보호하는 데 한계가 있다. 이에 본 연구에서는 최근 새로운 소프트웨어 테스팅 기법으로 주목받고 있는 크라

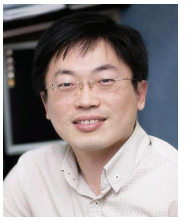
우드 테스팅 기법을 활용하여 정보시스템 보안성을 강화할 수 있음을 제안한다. 크라우드 테스팅은 다수의 사람이 참여하여 다양한 시스템상에서의 탐색을 통해 결함을 발견하고 보안성을 강화하여 검증할 수 있는 장점이 있다. 또한, 기존 보안성 테스트의 문제점으로 제기되는 시간 및 비용문제에 대한 해결책이 될 수 있다. 보안테스트에 크라우드 기법을 적용하는 것은 취약성 확인을 통한 소프트웨어의 안정성과 품질 향상에 기여할 수 있다. 크라우드 테스팅이 활성화되기 위해서 SW 단계별 보안성 테스트의 적용과 공공기관 및 유출기관에 대한 테스팅 의무화가 필요하다. 국가적 차원에서 크라우드 SW 보안 테스터 육성이 이루어진다면 SW 보안성 강화와 함께 인력 부족 문제 해결에도 기여할 수 있을 것이다.

감사의 글

이 논문은 2018년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2018S1A5A8027174)

Ji-Eun Choi received the M.S. degree in the Interdisciplinary Graduate Program in IT Law from Dankook University in 2017. She is currently working at Lotte Insurance. Her reserch interests include Crowd-sourcing, Intellectual Property, IP insurance, etc.
E-mail address: genie9281@naver.com

Yu-Jin Jeon is currently a M.S. student in the Department of Convergence Security in Dankook University. She graduated from the Department of Chemical Engineering. Her reserch interests include Industrial Security, Intellectual Property, etc.
E-mail address: yujini2473@naver.com



Hwansoo Lee received the Ph.D. degree in the Department of Business and Technology Management from KAIST. He currently works as an assistant professor in the department of convergent security at Dankook University. His research focuses on information security & privacy, electronic commerce, and enterprise information systems.
E-mail address: hanslee992@gmail.com