정 보 보 호 학 회 지
제26권 제1호, 2016. 2

전자정부 소프트웨어의 보안성 강화를 위한 개발보안 제도 연구

박 양 환*, 김 민 경*

요 약

국내 개발보안 제도는 2012년 12월부터 '소프트웨어 개발보안 제도'를 의무화 하며 시작 되었다. 3년여가 지난 지금 시점에서 소프트웨어 개발보안 제도는 안정화가 되어가고 있지만 보완할 사항도 일부 있다. 이러한 상황에서 본 연구는 제도가 먼저 정착된 미국의 개발보안 제도의 동향과 국내에서 개발보안 제도가 정착되는 과정에서의 변화된 점에 대해서 분석한다. 또한 개발보안 제도를 더욱 활성화하기 위한 개선 방향에 대해서 제시한다.

Ⅰ. 서 론

대한민국 전자정부서비스는 UN(United Nations)이 실시한 전자정부 평가에서 3회 연속(2010년, 2012년, 2014년) 1위를 차지했고 공공데이터를 개방, 쌍방향 소통에 사용하는 등 적극적인 활용을 하고 있다. 또한 UN의 전자정부 지수는 '구축된 인프라 수준'뿐만 아니라 '국민들에게 서비스 제공'하는 형태의 항목을 포함한다는 점[1]에서 더욱 큰 의미가 있다. 하지만 국민들의 서비스 활용이 증가함에 따라서 정보를 보호해야 하는 중요성은 더욱 커지고 있다.

민간의 경우에도 KT 홈페이지 해킹사건(2014년 3월)[2]과 뿜뿌 해킹사건(2015년 9월)[3] 등을 통해서 보안에 대한 중요성은 입증되었다. 또한 KT 홈페이지 해킹사건의 경우는 '적절한 인증 없는 중요기능 허용'이라는 소프트웨어의 보안약점을 통해서 공격을 받았고 뿜뿌 해킹사건의 경우도 '취약한 암호화 알고리즘의 사용'이라는 소프트웨어 보안약점을 통해서 피해가 확대된 경우로 소프트웨어 보안을 체계적으로 설계 및 관리하는 방안이 요구 되고 있다.

미국은 2002년부터 연방정보보안관리법(Federal Information Security Management Act, FISMA)을 제정해 '시큐어 코딩'을 의무화했고[4] 대한민국의 경우에는 2012년 12월 "정보시스템 구축·운영지침"에 따라 '소프트웨어 개발보안 제도'를 의무적[5]으로 실행하도록 하고 있다. 하지만 아직 개발단계에 집중되어 있어 설계단계와 운영단계 등 SDLC(Software Development Life Cycle) 전반적으로 확대하여 안전한 소프트웨어를 개발하기 위한 노력이 필요하다.

따라서 본 논문에서는 안전한 소프트웨어 개발의 기반이 되는 제도에 대한 동향에 대해 분석한다. 먼저 국내보다 먼저 체계적으로 정착된 미국의 제도 동향에 대해서 설명하고 다음으로 국내의 제도 동향에 대해서 알아본다. 또한 국내 소프트웨어 개발보안 제도 의무화 이후 소프트웨어 개발보안 분야의 변화된 부분을 추진 현황 분석을 통해서 알아본다. 마지막으로 이러한 변화를 기반으로 소프트웨어 개발보안 제도를 더욱더 확대할 수 있는 방향성에 대해서 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 미국 소프트웨어 개발보안 제도에 대한 동향에 대해서 살펴보고 3장에서는 국내 소프트웨어 개발보안 제도의 동향에 대해서 분석한다. 이를 토대로 4장에서는 국내 소프트웨어 개발보안 제도의 추진 현황을 분석하고 개선 방향을 제시한다. 마지막으로 5장에서는 앞서 논의한 사항들에 대해 결론을 맺는다.

* 한국인터넷진흥원 사이버침해대응본부 인프라보호단 보안평가인증팀 (yhpark@kisa.or.kr, mkkim@kisa.or.kr)

Ⅱ. 미국 소프트웨어 개발보안 관련 제도 동향

2.1. 미국 정보보호 법률의 기본이 되는 FISMA

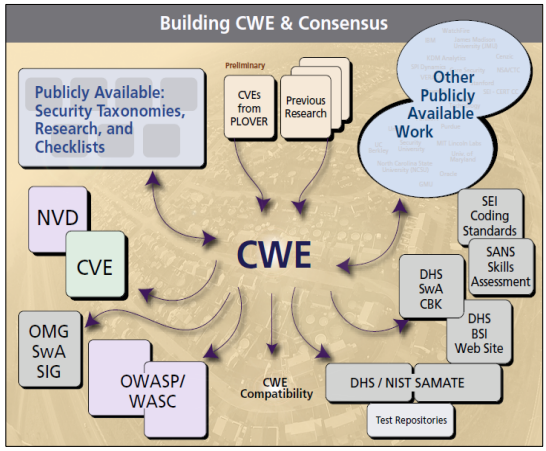
미국은 2002년도에 정부에서의 정보보안에 대한 기본법 역할을 하는 FISMA를 입법하였다. FISMA는 미국 공공 정보보호 분야의 대표적인 법률로서 모든 연방 정부기관은 FISMA를 준수하고 매년 1회 보안 프로그램의 유효성 및 개선계획에 관해서 관리예산처(Office of Management and Budget, OMB)와 미국 의회에 보고해야 한다. FISMA는 정부기관의 정보보안 노력을 감독할 권한과 책임을 부여하고 정부 정보자원의 안전을 위한 종합적 프레임워크 제공한다.

미국국립표준기술연구소(National Institute of Standards and Technology, NIST)는 FISMA에 의거 연방 정부의 정보보호와 관련 표준 및 지침서 작성의 책임이 있다.[6] 그 예로 SP(Speccial Publication) 시리즈를 발표하고 있으며 SP800-95에는 웹서비스에 대한 보안과 SP800-64에서는 정보보안에 대한 부분에서 소프트웨어 개발보안(Secure Coding)을 다루고 있다. 또한 SAMATE(Software Assurance Metrics & Tool Evaluation)프로젝트를 통해서 개발자 입장에서는 도구의 특성과 장단점을 파악할 수 있는 정보를 제공해 준다.

2.2. 미국 정부가 지원하는 소프트웨어 개발보안 관련 활동

미국의 국토안보부(Department of Homeland Security)는 다양한 방법으로 소프트웨어 보안을 지원하고 있다.

첫번째로 소프트웨어보증(Software Assurance, SwA) 프로그램[7]을 국토안보부의 국가사이버보안국(National Cyber Security Division, NCSD)을 통해서 후원하고 있다. SwA는 2003년 2월, 미국 부시 대통령이 Secure Cyberspace를 위한 국가전략으로 시작된 개념이다. 소프트웨어 보안 관리의 패러다임이 운영단계에서 취약점이 발견된 이후에 Patch를 적용하던 Patch Management에서 Software Assurance로 변화하였다. 소프트웨어 개발자들이 개발 초기단계부터 소프트웨어의 품질과 보안을 전체적으로 향상시킬 수 있도록 고안하도록 권고하고 있다.



〔그림 1〕 CWE의 배경과 영향도

두 번째로 CWE(Common Weakness Enumeration)[8]라는 보안약점 사전 정보를 관리하고 있다. CWE 역시 NCSD에서 지원하고 MITRE사에서 개발한다. CWE는 소프트웨어의 보안 약점을 다양한 관점에서 분류하여 모아 놓은 공식적인 사전으로 기존의 연구와 CVE(Common Vulnerabilities and Exposures) 등을 바탕으로 하고 있으며 NVD(National Vulnerability Database), SAMATE, OWASP 등에 영향을 미치고 있다.

미국의 국방부(Department of Defence)는 카네기멜론 대학(Carnegie Mellon University)의 SEI(Software Engineering Institute) 연구소를 지원을 하고 있다. SEI의 CERT[9]는 국토안보부(DHS)와 연계해 사이버보안와 관련한 다양한 연구하고 활용하고 있다. 소프트웨어 개발보안 분야로는 시큐어코딩 관련 표준을 제시하고 있다.

미국 상무부(Department of Commerce) 산하 연구기관인 NIST는 소프트웨어 보증 메트릭 및 도구평가 프

〔표 1〕 SEI CERT의 Secure Coding 표준

가이드 명	비고
SEI CERT C Coding Standard	C
CERT C++ Coding Standard	C++
SEI CERT Oracle Coding Standard for Java	Java
SEI CERT Perl Coding Standard	Perl
Android Secure Coding Standard	Android only, C,C++,Java

〔표 2〕 공개 정적분석 도구 (JAVA, C, C++, PHP)

도구 명	지원언어
Yasca	Java, C, C++, PHP
FindBugs, FindSecurityBugs, Jlint, LAPSE, PMD,	Java
BOON, Clang Static Analyzer, CQual, Csur, Smatch, Splint, UNO,	C
Cppcheck, Flawfinder, RATS	C, C++
PHP-Sat, Pixy, RATS	PHP

로젝트인 SAMATE[10]를 수행 하고 있다. SAMATE는 보안기능을 강조한 정적분석 도구를 평가할 수 있는 기준을 제시해 준다. 정적분석 도구를 평가할 수 있는 예제코드를 제공해 주고 상용 및 공개 도구에 대한 정보를 제공해 준다. SAMATE에서 발표한 C언어를 분석할 수 있는 공개 도구는 BOON, Clang Static Analyzer, Cppcheck, CQual, Flawfinder, Smatch, Yasca 등이 있고 JAVA언어를 분석할 수 있는 공개 도구는 FindBugs, FindSecurityBugs, Jlint, LAPSE, PMD 등이 있다.

Ⅲ. 국내 소프트웨어 개발보안 제도 동향

3.1. 국내 소프트웨어 개발보안 제도 대상

국내의 경우는 행정안전부를 중심으로 제도가 정착이 되고 있다. 행정안전부는 한국인터넷진흥원과 함께 2009년부터 소프트웨어 개발단계에서 소프트웨어 보안약점을 진단하여 제거하는 관련 연구를 진행하였다, 이 과정에서 2009년부터 2011년까지 전자정부 지원 사업을 대상으로 소프트웨어 보안약점 진단 시범사업을 수행하였고 연구결과와 진단 시범사업의 결과를 반영하여 2012년 6월 '행정기관 및 공공기관 정보시스템 구축·운영 지침'이 개정·고시되어 정보시스템 감리 대상 전자정부서비스는 개발보안을 적용하도록 의무화 하였다.

〔표 3〕 소프트웨어 개발보안 제도 대상

구분	내용	비고
대상	정보시스템 감리대상 정보화 사업 - 40억원 이상(2013.1)→20억원 이상(2014.1) →감리대상 전체(2015.1)	단계적 확대

제도의 안정적인 정착을 위해 정보화사업 규모에 따라 40억이상(2013년 1월), 20억 원 이상(2014년 1월), 감리대상 전체(2015년 1월)로 구분하여 소프트웨어 개발보안 의무화를 적용하였다.

3.2. 국내 소프트웨어 개발보안 제도 범위와 기준

행정안전부의 소프트웨어 개발보안의 범위는 신규개발의 경우는 소스코드 전체, 유지보수의 경우에는 유지보수로 인해 변경된 소스코드 전체를 대상으로 하고 있고 상용 소프트웨어로 정보시스템에 포함되는 경우는 제외를 하고 있다.

소프트웨어 보안약점 기준은 2012년 6월에는 43개 보안약점으로 최초 작성되고 2013년 8월에는 지침을 개정하여 47개로 조정하였다.

그리고 감리법인은 소프트웨어 보안약점을 진단 할 경우 '소프트웨어 보안약점 진단원'을 우선적으로 배치하도록 규정되어 있고 소프트웨어 보안약점 진단과정에서 진단 도구를 사용할 경우 '정보보호시스템 평가·인증 지침'에 따라 국가정보원장이 인증한 보안약점 진단 도구를 사용하여야 한다.

국내 소프트웨어 개발보안 제도는 대상이 감리대상으로 한정되어 있어 그에 따른 장점과 단점이 발생하고 있다.

장점으로는 감리대상 사업에 대해서는 개발보안을 강제화하여 초기 제도가 정착되는 시간을 단축 할 수 있었다.

반면에 단점으로는 감리 시점에 맞춰야하고 감리의

〔표 4〕 소프트웨어 개발보안 제도 범위와 기준

구분	내용	비고
범위	소스코드 (신규 개발 전체, 유지보수로 변경된 부분)	상용 SW 제외
기준	소프트웨어 보안약점 (지침 별표 3. SQL 삽입 등 47개 항목)	진단 기준
	소프트웨어 보안약점 진단원 (지침 별표 4. 자격 요건 및 관련 교육)	자격 기준
	소프트웨어 보안약점 진단도구 (지침 53조. 진단절차)	감리 도구 조건

짧은 수행기간 안에 개발보안을 같이 점검해야 해서 개발보안에 대한 성숙도가 높지 않은 상태에서 수행이 되었다. 그에 따른 부작용으로 프로젝트 종료 시점에 종료 감리에서 진단으로 인해 발주자·개발자·진단원 간의 어려움이 발생하기도 했다. 최근에는 프로젝트 개발 단계에서 수행사의 자가진단, 감리법인의 3자 진단 등을 통해 미리 수행하고는 있으나 전문인력, 예산, SDLC 전반에 대한 확대 어려움 등에 대한 문제점은 여전히 존재하고 있다.

Ⅳ. 국내 소프트웨어 개발보안 제도 추진 현황 및 개선 방향

4.1. 정보시스템 소프트웨어 보안약점 진단 현황

제도를 초기에 정착시키는 것은 어려운 일이다. 관련 예산, 기술연구가 선행 되어야 한다. 그리고 선도적으로 목표를 정하고 추진하는 것이 중요하다. 행정자치부는 행정기관 및 공공기관을 대상으로 신규로 개발되는 정보시스템에 대해서 시범적으로 보안약점을 진단하고 개선하였다.

2009년부터 2012년까지는 신규 구축 서비스에 대해서 진단을 실시하였다. 2009년에는 2개 서비스, 2010년에는 3개 서비스, 2011년에는 23개 서비스, 2012년에는 33개 서비스에 대해서 소스코드 보안약점을 진단하고 개선 조치하였다. 또한 2013년 부터는 운영 중인 정보시스템으로 진단 대상을 확대하였다. 2013년에는 161개 서비스, 2014년에는 31개 서비스 등에서 보안약점을 진단하고 개선 조치[11]를 하였다.

〔표 5〕 정보시스템 소프트웨어 보안약점 진단 추이 현황

구분	2009년	2010년	2011년	2012년	2013년	2014년
대상	2개	10개	23개	33개	161개	31개
	신규 개발 정보시스템				운영 정보시스템	

4.2. 모바일 전자정부 서비스 앱 보안성 검증 현황

정보시스템의 변화에 빠르게 대처하기 위해서 모바일 전자정부 서비스에 대해서도 소프트웨어 개발보안을 적용하였다. 2011년에 한국인터넷진흥원에 '모바일 전자정부 서비스 앱 보안성 검증센터'를 개소한 이후 2015년에는 '전자정부SW-IoT 보안센터'로 확장하였다. 그 결과 2011년에는 30건, 2012년에는 240건, 2013년에는 286건, 2014년에는 292건 등 총 848건에 대해서 소프트웨어 개발보안을 포함하여 전저정부 서비스 앱 검증을 하고 개선 조치[11]하였다.

〔표 6〕 모바일 전자정부 서비스 앱 보안성 검증 현황

구분	2011년	2012년	2013년	2014년
대상	30건	240건	286건	292건

4.3. 소프트웨어 개발보안 관련 가이드 개발 및 배포 현황

법에 따른 소프트웨어 개발보안 의무화는 상징적인 의미를 갖는다. 하지만 실무에서 직접 활용할 수 있는 구체적인 How to 방법을 제시하는 것 또한 더욱 중요하다.

행정자치부는 '행정기관 및 공공기관 정보시스템 구축·운영 지침'의 소프트웨어 개발보안 부분의 세부적인 부분을 가이드 하기 위해 안전한 코딩 기법을 이해관계자 역할별로 제시하였다. 발주자와 개발자를 위해서는 '소프트웨어 개발보안 가이드'와 '시큐어코딩 가이드'를 진단원을 위해서는 '소프트웨어 보안약점 진단 가이드'를 제시하였다.

2011년 6월에는 개발자 및 발주자들이 개발 과정에서 소프트웨어 보안약점을 파악하고 보안조치를 수행할 수 있도록 '소프트웨어 개발보안 가이드' 초판을 개발하였다. 또한 2012년 5월에는 소프트웨어 보안약점 진단원들이 보안약점 진단 수행 할 수 있도록 '소프트웨어 보안약점 진단 가이드' 초판을 개발 및 배포 하였다. 그리고 2013년 11월에는 개정된 '행정기관 및 공공기관 정보시스템 구축·운영 지침'에 따라 47개 소프트웨어 보안약점 기준 항목이 반영된 '소프트웨어 개발보안 가이드' 및 '소프트웨어 보안약점 진단 가이드' 개정판을 배포하였다.

2014년에는 모바일 분야에 대해서도 가이드를 확장하였다. 기관 자체적으로 안전한 모바일 서비스를 구축하고 취약점을 점검할 수 있도록 하기 위한 '모바일 대민서비스 구축 가이드'를 배포하고 모바일 대민서비스의 취약점 점검을 위해 '모바일 대민서비스 보안 취약점 점검가이드'를 개발 및 배포[11]하였다.

〔표 7〕 소프트웨어 개발보안 관련 가이드 개발 및 배포 현황

가이드(발행년도)	대상
소프트웨어 개발보안 가이드 (2011년 6월)	전자정부 개발자 및 발주자
시큐어코딩 가이드 (2011년 6월)	전자정부 개발자 및 발주자
소프트웨어 보안약점 진단 가이드 (2012년 5월)	전자정부 SW 보안약점 진단원
소프트웨어 개발보안 가이드(개정판) (2013년 11월)	전자정부 개발자 및 발주자
소프트웨어 보안약점 진단 가이드(개정판) (2013년 11월)	전자정부 SW 보안약점 진단원
모바일 대국민 전자정부 서비스 앱 소스코드 검증 안내서 (2014년 2월)	전자정부 개발자 및 발주자
모바일 대민서비스 구축 가이드 (2014년 10월)	전자정부 개발자 및 발주자
모바일 대민서비스 보안 취약점 점검 가이드 (2014년 10월)	전자정부 개발자 및 발주자

4.4. 소프트웨어 개발보안 관련 교육 및 전문가 양성 현황

제도를 정착시키기 위해서는 관련된 문화조성, 이해관계자 분석, 전문가 양성 또한 중요한 부분이다. 행정자치부는 소프트웨어 개발보안에 대한 인식제고 및 홍보를 위해서 2009년부터 공무원·개발자·감리원·민간기업 등을 대상으로 안전한 소프트웨어 개발방법론, 소스코드 수준에서의 보안약점 진단·제거 방법 등에 대한 개발보안 교육을 실시하고 있다.

그에 따라 2009년에는 109명, 2010년에는 266명, 2011년에는 1,019명, 2012년에는 2,262명, 2013년에는 2,544명, 2014년에는 2,001명에 대해서 교육을 실시하였다. 또한 소프트웨어 개발보안에 대한 전문가 양성을 위해서 전문교육을 실시하여 2012년 첫해 82명, 2013년 143명, 2014년 120명 등 총 345명의 소프트웨어 보안약점 진단원을 양성[11]하였다.

더불어 소프트웨어 개발보안 제도의 도입 성과를 공유하기 위해 2010년부터 소프트웨어 보안 컨퍼런스를 개최하고 있으며, 2014년 부터는 소프트웨어 개발보안 문화를 학생 때부터 조기인식 시키기 위해 대학생을 대상으로 '소프트웨어 개발보안 경진대회'를 매년 개최 하고 있다. 이것은 개발을 처음 접하는 대학생들에게 보안의 중요성을 인식시키는 의미 있는 일이라 할 수 있다.

〔표 8〕 소프트웨어 개발보안 교육 추진 현황

구분	2009년	2010년	2011년	2012년	2013년	2014년
교육생	109명	266명	1,019명	2,262명	2,544명	2,001명
진단원	-	-	-	82명	143명	120명

4.5. 국내 소프트웨어 개발보안 개선 방향

소프트웨어 개발보안은 광의적 의미로는 소프트웨어 개발 생명주기(SDLC)의 각 단계별로 요구되는 보안활동을 모두 포함하며, 협의적 의미로는 소프트웨어 개발 과정 중 소스코드 구현단계에서 보안약점을 배제하기 위한 '시큐어코딩(Secure Coding)'을 의미[12]한다. 초기에는 소스코드 구현단계에 집중을 했지만 MS-SDL(Security Development Lifecycle)[13], OWASP Secure SDLC[14] 등, 광의의 의미의 개발활동으로 확대되고 있다. 하지만 현재 국내는 제도의 초기 상태로 개발단계에 집중되어 있어 설계 및 운영 단계 등으로의 확대가 필요하다.

기획단계에서는 소프트웨어 개발보안에 대한 예산이 책정되어 있어야 하고 이것을 정량적으로 산정하기 위한 '소프트웨어 개발보안 대가산정 방식'이 필요하다. 현재 소프트웨어 개발보안 대가산정은 정보보안 컨설팅 비 안에 컨설턴트 등급별 산정방식으로 계산되고 있다.[15] 이는 일반적인 정보보호 컨설팅 대가 산정과 유사하여 실제 소프트웨어 개발보안에 필요한 대가와는 차이가 발생 하게 된다. 소프트웨어 개발보안은 소프트웨어 개발이라는 특성이 존재하는 분야인 만큼 기존의 FP(Function Point) 방식 등, 소프트웨어 특성을 고려한 현실적인 대가산정 방식이 필요하다.

요구분석 단계에서 보안요구사항을 구체화하여 '소프트웨어 개발보안 상세 요구사항 명세서'를 작성하여야 한다. 명세서에는 정보시스템이 만족해야하는 '보안약점 기준'을 명확히 명시하고 설계, 개발, 테스트 단계까지의 '소프트웨어 개발보안 요구사항 추적성'을 보장하여야 한다.

설계단계에서는 정보시스템의 구조를 이해하고 아키텍처 측면에서 예상되는 보안약점을 분석하고 대응을 '보안설계(Secure Design)'가 필요하다. Gary McGraw에 따르면, '보안문제의 50%를 설계결함[14]이 차지하

고 있다고 할 만큼 설계단계에 보안을 고려하는 것은 중요하다. 또한 설계결함 문제는 코드를 보는 것으로 찾을 수 없으며, 상위단계(Higher Level)의 이해가 필수적이다. 그리고 기존의 패턴중심의 룰(Rule)기반의 진단방법으로 발견할 수 없는 부분에 대해서 설계적인 고려를 해야한다. '적절한 인증 없는 중요기능 허용'이라는 보안약점은 패턴 중심의 정적분석 방법으로는 찾아내기 어려운 방법으로 설계단계에 고려를 하거나 동적분석 방법을 병행하여야 찾아 낼 수 있다.

운영단계에서는 변경되는 부분에 대한 관리가 중요하다. 구축 당시 개발보안이 적용되어 개발이 되어있어도 운영유지보수 하며 변경·추가 되는 부분에 대해서 적용하지 않는다면 한 번에 보안이 뚤리는 경우가 발생한다. 따라서 운영 프로세스 안에 '소프트웨어 개발보안 진단' 이라는 프로세스를 추가 하여 운영하여야 한다.

Ⅴ. 결 론

지금까지 미국과 국내의 소프트웨어 개발보안 제도의 동향, 국내 소프트웨어 개발보안 추진 현황을 분석한 후 개선방향을 제안하였다.

국내 소프트웨어 개발보안 제도는 개발단계 중심으로 초기에 빠른 시간안에 정착을 할 수가 있었지만 소프트웨어 개발보안의 효과성을 극대화 하기 위해서는 설계, 운영 단계 등 SDLC 전반으로 확대 하는 것이 최적의 제도의 방향성이다.

본 논문을 통해서 제시한 미국의 제도의 사례와 개선방향으로 제시한 '소프트웨어 개발보안 대가산정 방식', '소프트웨어 개발보안 상세 요구사항 명세서', '보안설계(Secure Design)', '운영프로세스에 소프트웨어 개발보안 진단 포함' 등이 제도의 방향에 포함될 때 개발보안의 확대에 큰 도움이 될 것이다.

국내 소프트웨어 개발보안은 2009년에 시작되고 2012년에 비로소 제도화 되었다. 2002년도에 제도화된 미국과 비교하면 아직은 걸음마 단계 상태이다. 제도 초기에 SDLC 전반에 존재하는 이해관계자(발주자, 개발자, 진단원 등)들이 긍정적으로 필요성을 인식하여 안전한 소프트웨어를 만드는 문화를 만들고 그 과정에서 소프트웨어 개발보안 제도가 적시에 지원 할 수 있도록 노력해야 할 것이다.

참 고 문 헌

[1] 국가지표체계, http://index.go.kr

[2] 연합뉴스, "KT 홈페이지 해킹…1천200만명 개인정보 털렸다", 2014.3.

[3] 전자신문, "뿜뿌 해킹, 개인정보 유출도 모자라 모바일 악성코드까지 유포", 2015.9.

[4] NIST, http://csrc.nist.gov

[5] 행정자치부, www.moi.go.kr

[6] 김성근,이재일, "secure coding 제도의 생태계 차원의 분석", *정보보호학회논문지* 22(5), pp.1206~1207, 2012.10

[7] Software Assurance, https://buildsecurityin.us-cert.gov/swa

[8] CWE, http://cwe.mitre.org/about/index.html

[9] SEI CERT, https://www.securecoding.cert.org

[10] SAMATE, https://samate.nist.gov

[11] 국가정보원, 미래창조과학부, 방송통신위원회, 행자부, 한국인터넷진흥원, 국가보안기술연구소, *2015 국가정보보호백서*, pp.90~95

[12] 행정자치부, *소프트웨어 개발보안 가이드*, pp.4, 2013.11

[13] MS-SDL, https://www.microsoft.com/en-us/sdl

[14] OWASP Secure SDLC, https://www.owasp.org/index.php/Secure_SDLC_Cheat_Sheet

[15] 한국소프트웨어산업협회, *SW사업 대가산정 가이드(2015년 개정판)*, pp.111

[16] Gary McGraw, *Software Security: Building Security In*, pp.139, 2006.2

〈저 자 소 개〉

박 양 환 (Park Yang Hwan)
2003년 2월 : 인천대학교 컴퓨터공학과 졸업
2016년 1월 : 고려대학교 정보보호학과 석사과정
2015년 5월~현재 : 한국인터넷진흥원 책임연구원
관심분야 : 소프트웨어 보안, 소프트웨어 개발보안, 웹 보안, 소프트웨어 아키텍처

김 민 경 (Kim Min Kyoung)
2000년 2월 : 울산대학교 전자공학과 석사
2000년 3월~현재 : 한국인터넷진흥원 보안평가인증팀장
관심분야 : 소프트웨어 보안, 소프트웨어 개발보안, 정보보호제품 성능평가