한국컴퓨터정보학회 하계학술대회 논문집 제23권 제2호 (2015. 7)

설계 단계의 보안 코딩 지침

신성윤[O], 이상원[*], 이현창[*]
[O]군산대학교 컴퓨터정보통신공학부
[*]원광대학교 정보전자상거래학부
e-mail:s3397220@kunsan.ac.kr[O], {sangwonlee, hclglory}@wku.ac.kr[*]

Security Coding Guide of Design Phase

Seong-Yoon Shin[*O], Sang-Won Lee[**], Hyun-Chang Lee[**]
[*]OSchool of Computer & Information Communication Engineering, Kunsan National University
[**]Division Computer and Electronic Commerce(Institure of Convergence and Creativity),
Wonkwang University

● Abstract ●

본 논문에서는 S/W 개발 보안 지침을 알려준다. S/W 개발 보안에서 S/W의 보안 취약점 유형에 대하여 설명한다. S/W 보안 취약점 유형인 입력 데이터 검증 및 표현, API 악용, 보안 특성, 시간 및 상태, 에러처리 코드품질, 그리고 캡슐화에 대하여 설명하도록 한다. 즉, 본 논문에서는 보안 취약점에 대한 소스코드 레벨에서의 대응조치에 대한 가이드를 제시하고자 한다.

키워드: 보안 지침(security guide), 취약점(vulnerability), 소스 코드 레벨(source code level)

I. Introduction

S/W의 설계란 사용자의 요구사항을 새로 개발하는 시스템에서 어떻게 충족시킬 것인가의 해결 방안을 찾고 이를 구체화하는 단계로 사용자 요구사항에 대한 분석 결과와 실제 프로그래밍 언어로 구현하는 단계를 연결해주는 역할을 담당한다[1].

행정안전부의 전자정부 소프트웨어 개발 운영자를 위하여 2012년에 발표한 소프트웨어 개발보안 가이드를 개선하기 위한 방안을 제안한 논문[2]도 있었다.

본 논문에서는 이 같은 S/W 설계 단계에 포함 되어야할 보안 코딩 지침을 제시한다.

II. S/W Development Security Guide

행정기관 등이 안전한 소프트웨어를 개발하여 각종 사이버위협으로부터 예방 · 대응코자 하여 만든 것이다. SW 개발단계부터 보안약점을 제거하는 'SW 개발보안' 의무제가 시행되며 이에 따른 관련 가이드를 보급한 것이다.

가이드 주된 내역서
- 개발할 때 참고 : 소프트웨어 개발보안 가이드
. 언어별 시큐어코딩 가이드 : JAVA, C, Android-JAVA
- 점검할 때 참고 : 소프트웨어 보안약점 진단가이드

※ SW 개발보안을 반영한 "정보시스템 구축 · 운영 지침(행안부 고시)" 개정

ㅇ 활용 : S/W 개발보안을 수행하는 행정 · 공공기관 정보시스템 개발자 및 유지보수자, 담당공무원

- 진단 : 진단원 및 감리원, 사업자 자체 SW보안약점 잔존여부 진단 등

III. S/W Security Vulnerability Type

유형	내용
입력데이터 검증 및 표현	• 프로그램 입력값에 대한 검증 누락 또는 부적절한 검증, 데이터의 잘못된 형식지정으로 인해 발생할 수 있는 보안약점 (예) SQL 삽입, 경로 조작 및 자원 삽입, 크로스사이트 스크립트 등
보안기능	• 보안기능(인증, 접근제어, 기밀성, 암호화, 권한 관리 등)을 적절하지 않게 구현시 발생할 수 있는 보안약점 (예) 부적절한 인가, 중요정보 평문저장, 하드코드된 비밀번호 등
시간 및 상태	• 동시 또는 거의 동시 수행을 지원하는 병렬 시스템, 하나 이상의 프로세스가 동작되는 환경에서 시간 및 상태를 부적절하게 관리하여 발생할 수 있는 보안약점 (예) 경쟁조건: 검사시점과 사용시점(TOCTOU) 등
에러처리	• 에러를 처리하지 않거나, 불충분하게 처리하여 에러정보에 중요정보(시스템 등)가 포함될 때 발생할 수 있는 보안약점 (예) 오류상황 대응 부재, 오류 메시지를 통한 정보노출 등
코드오류	• 타입 변환 오류, 자원(메모리 등)의 부적절한 반환 등과 같이 개발자가 범할 수 있는 코딩오류로 인해 유발되는 보안약점 (예) Null Pointer 역참조, 부적절한 자원 해제 등
캡슐화	• 중요한 데이터 또는 기능성을 불충분하게 캡슐화 하였을 때 인가되지 않은 사용자에게 데이터 누출이 가능해지는 보안약점 (예) 제거되지 않고 남은 디버거 코드, 시스템 데이터 정보노출 등
API 오용	• 의도된 사용에 반하는 방법으로 API를 사용하거나, 보안에 취약한 API를 사용하여 발생할 수 있는 보안약점 (예) DNS lookup에 의존한 보안결정 등

IV. Conclusions

본 논문에서는 S/W 개발 보안 지침을 전달해 주었다. 특히, S/W 개발 보안 단계에서 S/W의 보안 취약점 유형에 대하여 개략적으로 설명하였다. S/W 보안 취약점 유형인 입력 데이터 검증 및 표현, API 악용, 보안 특성, 시간 및 상태, 에러처리 코드품질, 그리고 캡슐화에 대하여 설명하도록 한다. 즉, 본 논문에서는 보안 취약점에 대하야 개괄적으로 설명 하였으며, 우리가 S/W를 개발할 때 소스코드 레벨에서의 대응조치에 대한 가이드를 제시하였다.

References

[1] http://www.happycampus.co.kr/doc/3614795/

[2] Kyung Sook Han,Taehwan Kim, Ki Young Han, Jae Myung Lim,Changwoo Pyo, "An Improvement of the Guideline of Secure Software Development for Korea E-Government," Journal of the Korea Institute of Information Security and Cryptology, Vol. 22, No. 5, pp.1179-1189, 2012