https://doi.org/10.14400/JDC.2017.15.3.175

하이브리드 분석 기반의 스마트 퍼징 시스템 설계

김만식, 강정호, 전문석
숭실대학교 컴퓨터학과

A Design of Smart Fuzzing System Based on Hybrid Analysis

Mansik Kim, Jungho Kang, Moon-seog Jun
Dept. of Computer Science & Engineering, Soongsil University

요 약 전 세계적으로 IT 산업이 발전함에 따라 소프트웨어 산업 또한 크게 성장하였으며, 사회전반에 걸쳐 일상생활에서부터 금융과 공공 기관까지 영향력을 미치고 있다. 특히 ICT 기술의 활성화로 인해 소프트웨어 산업은 더욱 고도화 되고, 다양한 기능과 기술을 공유하게 되었다. 그러나 이렇게 성장하는 소프트웨어 산업과 비례하여 제공되는 서비스에 치명적인 위협을 가할 수 있는 다양한 보안 위협 또한 크게 증가 하였다. 이미 OpenSSL 하트블리딩 취약점으로 전 세계적으로 큰 이슈를 일으켰으며, 그밖에도 이란의 원자력 발전시설, 미국의 에너지 기업들이 소프트웨어 취약점으로 인해 많은 피해를 입었다. 본 논문에서는 응용프로그램 보안 사고의 큰 비중을 차지하고 있는 소프트웨어 취약점을 효과적으로 탐지·식별 할 수 있는 블랙박스, 화이트박스 테스트를 연계한 하이브리드 퍼징 시스템을 제안한다.

주제어 : 스마트 퍼징, 블랙박스 테스트, 화이트박스 테스트, 하이브리드 분석, 소프트웨어 취약점

Abstract In accordance with the development of IT industry worldwide, software industry has also grown tremendously, and it is exerting influence on the general society starting from daily life to financial organizations and public institutions. However, various security threats that can inflict serious threat to provided services in proportion to the growing software industry, have also greatly increased. In this thesis, we suggest a smart fuzzing system combined with black box and white box testing that can effectively detect·distinguish software vulnerability which take up a large portion of the security incidents in application programs.

Key Words : Smart Fuzzing, Black box test, White box test, Hybrid analysis, Software Vulnerability

1. 서론

IT기술이 발달함에 따라 일상생활에서부터 금융이나 공공 기관 까지 사회전반에 걸쳐 많은 IT 서비스들이 출시되고 있다 [1, 2, 3]. 그러나 이렇게 제공되는 서비스의 질과 양이 다양해짐에 따라 이를 위협하는 공격 또한 비례로 증가하고 있으며, 실제로 2014년 4월에 보안 패치 이전의 보안취약점을 활용한 OpenSSL 하트블리드 제로데이 공격과, 2010년 7월의 이란 원자력 발전시설을 공격한 스턱스텟, 2011년 미국의 에너지 기업을 공격한 나이

*본 논문은 중소기업청에서 지원하는 2016 년도 산학연협력 기술개발사업(No. C0364522)의 연구수행으로 인한 결과물임을 밝힙니다.
Received 29 December 2016, Revised 8 February 2017
Accepted 20 March 2017, Published 28 March 2017
Corresponding Author: Moonseog Jun
(Dept. of Computer Science, Soongsil University)
Email: mjun@ssu.ac.kr

ISSN: 1738-1916

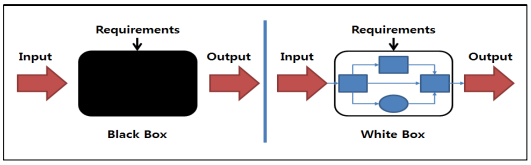
트 드래곤 등과 같이 기업이나 공공 기관의 데이터를 탈취하기 위하여 지속으로 취약점을 공략하는 APT공격, 2010년 Paypal 사이버 공격과 같이 항상 외부에 노출되어 있어 공격의 대상이 되는 웹 사이트 공격 등 많은 서비스들이 SW 자체 보안 취약점으로 공격을 당하였다[4]. 또한 Gartner의 보고서에 따르면 SW 보안사고의 75%는 취약점을 가지고 있는 응용프로그램에서 발생한다고 하였으며, Symantec의 보고서에 따르면 2012년 한해 동안 발견된 취약점이 5,291개에 이른다고 한다[5, 6].

이러한 이유로 SW 보안 취약점을 예방하기 위해 다양한 노력이 진행되고 있으며, 한 예로 국외의 MITRE는 미국 국토보안부내 국가 사이버 보안국의 지원으로 CWE(Common Weakness Enumeration)를 설립하여 SW 취약점을 다양한 관점에서 분류하고 있으며, CERT(Computer Emergency Response Team)에서는 C, C++, JAVA, Perl 언어로 개발되는 어플리케이션에 대하여 보안 취약점을 제품 출시 전에 제거할 수 있도록 시큐어 코딩 가이드를 제공하고 있다[7, 8, 9, 10]. 또한 국내에서는 2012년 5월에 발표한 행정안전부의 시큐어 코딩 의무화 법안에 따라 40억원 이상의 정보화 사업에서는 행정안전부가 제공하는 시큐어 코딩 가이드라인에 따라 43개의 보안 취약점을 반드시 제거해야한다. 그러나 이러한 노력에도 불구하고 아직까지 많은 SW들이 알려지거나, 알려지지 않은 보안 취약점으로 인해 위협을 받고 있으며, 이를 방지하기 위하여 다양한 기법들이 제안되고 있다.

퍼징은 이러한 노력 중 취약점을 발굴하기 위한 방법으로 1989년에 Wisconsin-Madison 대학의 Barton Miller 교수 연구실에서 개발된 일종의 SW 보안 테스트 기법이다[11]. 무작위로 SW에 데이터를 입력하여 에러가 발생하는 경우를 탐지함으로써 취약점을 검출한다. 그러나 무작위로 도출된 입력 값 기반의 퍼징은 특정 취약점에 대한 도출이 어렵고, 많은 시간을 소모하는 문제점이 있다. 본 논문에서는 이러한 문제점을 해결하고 효율성을 높이기 위하여, 블랙박스 테스트와 화이트박스 테스트를 연계한 하이브리드 분석 기반의 스마트 퍼징 시스템을 설계 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 보안 취약점을 검사하기 위한 화이트박스 테스트, 블랙박스 테스트, 퍼징을 간단히 소개하고, 3장에서는 스마트 퍼징 시스템을 구성하기 위한 구성요소를 정의한다. 4장에서는 제안하는 스마트 퍼징 시스템의 구조와 동작 과정을 설명하고 5장에서 결론을 설명한다.

2. 관련 연구



[Fig. 1] Black box testing and White box testing

2.1 블랙박스 테스트

블랙박스 테스트는 테스터가 SW가 무엇을 하는지만 알고서 수행하는 테스트로, 테스터는 SW 내부적으로 어떻게 동작하는지 모른다. 오직 [Fig. 1]의 왼쪽 그림과 같이 SW 기능과 요구사항을 기반으로 입력 값에 대응하는 결과 값만 도출하여 테스트를 한다. 블랙박스 테스트는 소스코드의 정보와 기술적 스킬을 요구하지 않고 큰 시스템에 유용하다는 장점이 있지만, 짧은 기간 동안 모든 가능한 입력 값을 테스트하기엔 무리가 있고, 논리적 오류를 검출하지 못하며, 명확한 기능적 사양에 대한 지식 없이 테스트 케이스를 만들기 어렵다는 단점이 있다.

2.2 화이트박스 테스트

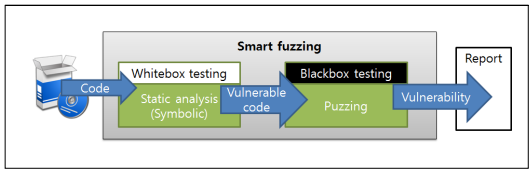
화이트박스 테스트는 테스터가 SW의 코드에 접근하여 검사하는 테스트로, 테스터는 SW의 내부적 동작 과정을 알 수 있다. 그러나 [Fig. 1]의 오른쪽 그림과 같이 SW 입력 값이 어떻게 결과 값을 도출하는지 확인 할 수 있지만, 코드의 동작에 맞추어 테스트를 수행하다가 객관적으로 테스트를 하지 못할 위험이 있다[12]. 화이트박스 테스트는 코드에 접근 할 수 있어 모든 코드를 명확히 확인할 수 있고 숨어 있는 코드에 대한 에러를 식별할 수 있으며, 입력 값을 추측하기 쉽고, 테스트 시나리오를 짜기 쉽다는 장점이 있으나, 화이트박스 테스트를 수행하기 위해 숙련된 스킬이 필요하고, 비용이 비싸며, 테스트 사양을 충족시켰다고 보장할 수 없다는 단점이 있다.

2.3 퍼징

퍼징은 SW에 유효하지 않은 값이나 무작위로 선출한

값을 입력하여 기대하지 않은 결과를 도출하여 에러와 잠재되어 있는 취약점을 식별하는 보안 테스트이다. 퍼징은 또한 테스트하기 위해 입력 값을 선출하는 방법에 따라 종류가 나뉘는데, SW에 대한 지식 없이 무작위로 입력 값을 선출하는 블랙박스 퍼징과, SW의 코드와 수행 내용의 지식을 기반으로 입력 값을 선출하는 화이트박스 퍼징, 블랙박스 퍼징과 화이트박스 퍼징의 장점을 결합하여 하이브리드 분석을 수행하는 그레이박스 퍼징이 있다[13]. 본 논문은 이 그레이박스 퍼징의 확장된 개념으로 스마트 퍼징을 설계하였다.

3. 스마트 퍼징 구성요소



[Fig. 2] Proposed Smart Fuzzing Structure

스마트 퍼징은 기존 퍼징과 달리 소프트웨어 취약점을 발견하기 위한 자동화된 방법으로 우선 퍼징을 수행하고자 하는 대상 SW에 대한 데이터 모델을 생성하여야 하며, 데이터 파일 및 SW 자체에 대한 분석을 자동으로 수행한다[14]. 제안하는 스마트 퍼징은 기존의 퍼징과 달리 [Fig. 2]와 같이 화이트박스 테스트의 Static analysis을 통해 취약점 정보와 입력 파일 구조를 모델링화 시키고, 모델링된 데이터 구조와 취약코드를 블랙박스 테스트의 Dynamic analysis를 통해 연계 Fuzzing하여 입력 데이터를 추출하여 취약점을 탐지하고 리포트 한다. 제안하는 스마트 퍼징 시스템의 모든 과정을 자동화 연계 처리하여 정확한 취약점을 식별하기 위해서는 다음과 같은 구성 요소가 필요하다.

3.1 Static Analysis Engine

Static Analysis Engine은 소스코드의 보안약점을 분석하기 위해 분석 대상 소스 파일이 업로드가 되면, 업로드 된 소스 파일 점검 후 분석 결과를 생성한다[15]. SW 취약점 분석 대상으로 SANS 연구소, MITRE, 그리고 미국 및 유럽의 여러 유수 소프트웨어 보안 전문가들이 협력해서 만든 25가지 보안약점 목록인 CWE SANS TOP 25와, 주요 웹 애플리케이션 보안 취약점으로 인한 영향을 알리기 위해 OWASP에서 발표하고 있는 보안약점 리스트 OWASP Top10, 안전한 S/W 개발을 위해 필수 진단하여 제거해야 하는 행정안전부에서 제시한 S/W 보안약점 등을 기준 삼아 Static Analysis을 수행한다[16, 17].

3.2 Dynamic Analysis Engine

Dynamic Analysis Engine은 Dynamic Analysis을 수행하는 Fuzzer로서 스마트퍼징을 수행할 대상 SW에 Static Analysis로 도출된 취약점을 기반으로 입력 값을 선출하여 Dynamic Analysis을 수행한다[18]. 무작위로 선출된 입력 값이 아니라 취약점 의심이 높은 입력 값을 대상으로 수행함으로써 짧은 기간 동안 높은 정확도를 가지고 작업을 수행하여 취약점을 도출 할 수 있으며, Static Analysis가 가지고 있는 논리적 오류 검출만의 한계를 극복한다.

3.3 Vulnerability Analysis Platform

Vulnerability Analysis Platform은 스마트 퍼징을 수행하기 위하여 정보를 종합하고 관리자가 다양한 소스코드를 관리하고 모니터링 할 수 있는 플랫폼이다. 기본적으로 스마트 퍼징을 수행하는 각 구성 요소의 입력 및 결과 값을 산출하고 저장하고, 다양한 소스코드 및 SW에 대하여, 스마트 퍼징을 하나의 시스템에서 종합적으로 처리하기 위해 데이터를 모델링화 하여 각 구성요소 간에 유기적으로 업무를 수행할 수 있도록 하고, 최종적으로 스마트 퍼징 수행 결과를 report한다. 뿐만 아니라 <Table 1>와 같이 사용자 편의를 위한 종합 관리 서비스를 제공함으로써 사용자가 스마트 퍼징 과정을 모니터링 하고 관리할 수 있도록 한다.

〈Table 1〉 Vulnerability Analysis Platform Functions

Function	Content
Integrated platform services	- Vulnerability analysis Search service to retrieve information from the entire field of the data
Dashboard	- Service represented by Figure to be able to analyze the data in a visual - Service that represents the total number of vulnerabilities, the number of registered projects, and the total number of revised vulnerabilities

Monitoring	- Checking the detailed analysis - Confirming results through intuitive charts with detailed numerical analysis results

3.4 Data set DB

Data set DB는 Vulnerability Analysis Platform에서 수행한 스마트 퍼징 대상 SW의 모델링화된 데이터 셋을 저장하고 Static Analysis로부터 Dynamic Analysis을 통해 도출된 내역 및 결과를 저장하며, Dynamic Analysis Engine이 스마트 퍼징을 수행할 때 데이터를 제공한다. 또한 각 수행 내역 및 결과 값을 저장하여 최종적으로 Vulnerability Analysis Platform이 사용자에게 스마트 퍼징 수행 결과를 report 할 수 있게 해준다.

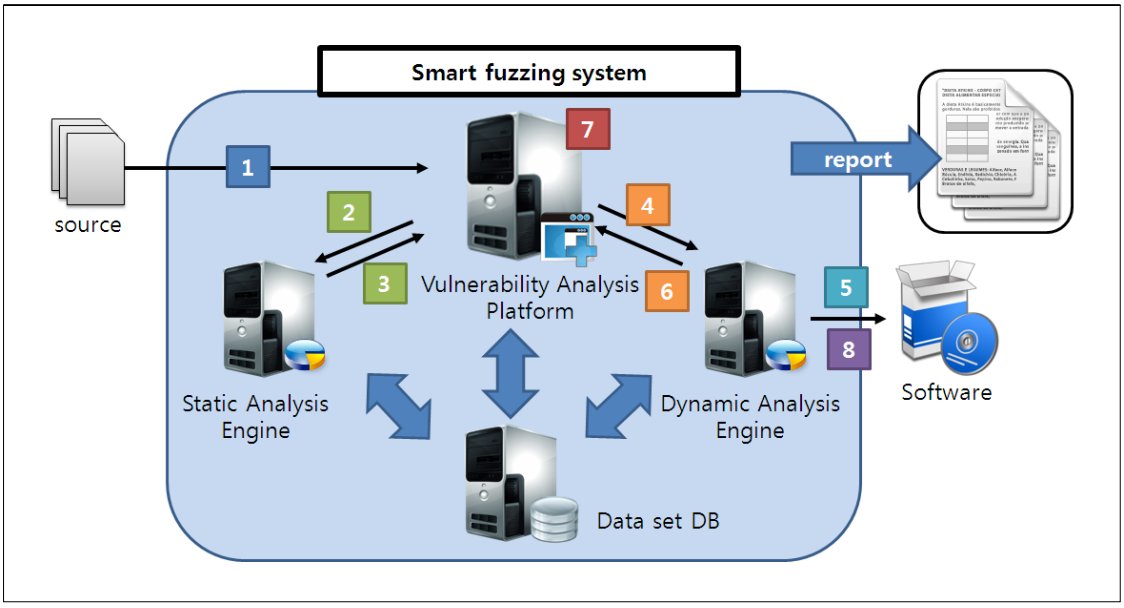
4. 제안하는 시스템 구조

제안하는 스마트 퍼징 시스템은 [Fig. 3]와 같이 Static Analysis Engine, Vulnerability Analysis Platform, Data set DB, Dynamic Analysis Engine으로 이루어져 있으며 소스코드와 퍼징 대상 정보, 퍼징 대상의 SW를 기반으로 스마트 퍼징을 수행하여 결과를 report 한다. 스마트 퍼징 수행 절차는 다음과 같다.

① Vulnerability Analysis Platform를 통하여 분석 대상이 되는 사이트 혹은 SW의 소스파일을 기본정보(SW 명세 등)와 함께 등록한다.

② Vulnerability Analysis Platform에 등록된 자료를 Static Analysis Engine에 등록하여 Static Analysis로 얻을 수 있는 취약점을 검출한다. 또한 Static Analysis을 통하여 외부에서 데이터가 유입되는 소스를 검출하고, 검출된 소스를 바탕으로 Dynamic Analysis를 수행하기 위한 입력 값 대비 취약점 의심 소스 검출을 한다.

③ Static Analysis을 통하여 얻은 데이터를 모델링하여 Data set DB에 저장하고, 분석데이터 저장 시 분석에 대한 편의성 및 분석대상의 정확성을 위하여 Vulnerability Analysis Platform의 결정에 의해 자동과 수동 두 가지 옵션중 하나로 저장한다. 첫째 자동 옵션은 ②에서 검출된 소스에 대하여 전체 항목을 자동으로 Data set DB에 저장하고 추후에 Dynamic Analysis으로 분석할 퍼징 데이터를 추출한다. 둘째 수동 옵션은 ②에서 검출된 소스 중에서 분석 담당자에 의해 체크된 소스만 Data set DB에 저장하고 Dynamic Analysis으로 분석할 퍼징 데이터를 추출한다.



[Fig. 3] Proposed Smart Fuzzing System

④ ①에서 Vulnerability Analysis Platform에 의해 Data set DB에 등록된 기본정보를 Dynamic Analysis Engine에 등록하여 블랙박스 테스트를 준비한다.

⑤ Vulnerability Analysis Platform에 의하여 Data set DB에 모델링화 되어 등록된 데이터를 기반으로 Dynamic Analysis Engine를 통하여 분석대상(Software)에 블랙박스 테스트를 수행한다.

⑥ Dynamic Analysis Engine의 블랙박스 테스트를 통하여 얻은 정보를 Vulnerability Analysis Platform을 통하여 Data set DB에 저장한다. 블랙박스 테스트를 통하여 얻은 자료에는 검출된 보안취약점과 대상 Software에 에서 추출된 정보를 저장한다.

⑦ Vulnerability Analysis Platform에 저장된 화이트박스 테스트(Static Analysis Engine)을 통하여 검출된 데이터와, 블랙박스 테스트(Dynamic Analysis Engine)를 통하여 검출된 데이터를 매칭하고, 테스트의 정보를 종합하여 스마트 퍼징을 하기 위한 SW 명세와 데이터셋을 Data set DB에 전송하여 등록한다. 또한 매칭 시에도 수동매칭 옵션을 더하여 데이터 정확성을 높인다. 그리고 테스트 대상이나 목적의 특성에 따라서 옵션을 두어 관리자가 Dynamic Analysis Engine에서 스마트 퍼징을 수행하고 Data set DB에 수행하였던 내역과 결과를 저장할 수 있도록 한다.

⑧ 최종적으로 등록된 데이터를 통하여 Dynamic Analysis Engine을 통하여 스마트 퍼징을 수행하고 Data set DB에 상태 및 결과값을 저장하여, 스마트 퍼징 수행 결과를 Vulnerability Analysis Platform를 통하여 report 한다.

5. 결론

IT 기술이 발달함에 따라 이제 SW는 사회 전반에 걸쳐 거의 모든 영역에 영향을 미치고 있다. 그러나 향상된 서비스 질과 증가하는 분야에 비례하여 악의적인 공격자의 위협 또한 크게 증가 하였으며, 특히 SW 보안 사고의 75%가 응용프로그램에 내재되어 있는 취약점이라고 발표한 가트너의 보고서처럼 SW 취약점은 큰 문제가 되었다. 그렇기 때문에 현재 SW의 취약점을 사전에 발견하여 방지 할 수 있는 방법이 많은 분야에서 연구 중이다.

본 논문에서는 이러한 노력의 일환으로 악의적인 공격자로부터 사전에 공격을 예방할 수 있도록 SW를 출시하기 전에 혹은 출시한 이후에라도 SW의 취약점을 미리 탐지하여 리포트 할 수 있는 블랙박스 테스트, 화이트박스 테스트를 연계한 스마트 퍼징 시스템을 설계하였으며, 블랙박스 테스트, 화이트박스 테스트의 연계를 통하여 기존의 퍼징 시스템이 가지고 있던 한계를 극복하고 자동화된 방안으로 효율적으로 취약점을 검출할 수 있도록 하였다.

ACKNOWLEDGMENTS

This work (Grants No. C0364522) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2016.

REFERENCES

[1] SH Lee, DW LEE,"A Study on u-Health Fusion Field based on Internet of Thing", Korea Convergence Society, Vol 7, No. 4, pp. 19-24, 2016

[2] LS Kim, "Convergence of Information Technology and Corporate Strategy", Korea Convergence Society, Vol. 6, No. 6, pp. 17-26, 2015

[3] SS Shin, GS Chae, TH Lee, "An Investigation Study to Reduce Security Threat in the Internet of Things Environment", Convergence Society for SMB, Vol. 5, No. 4, pp. 31-16, 2015

[4] Software security weaknesses diagnostic guide, KISA, 2012.

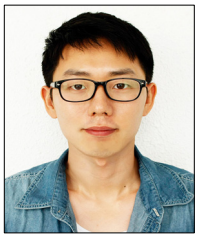
[5] MS Gu, YZ Li, "A Study of Countermeasures for Advanced Persistent Threats attacks by malicious code", Convergence Society for SMB, Vol. 5, No. 4, pp. 37-42, 2015

[6] Symantec, "2013 Internet Security Threat Report, Volume 18," 2013.

[7] Christey, S. M., and R. P. Glenn. Common weakness

enumeration. 2013.
[8] Robert C. Seacord, The CERT C Secure Coding Standard, Addison-Wesley, October 2008.
[9] Robert C. Seacord, Secure Coding in C and C++, Addison-Wesley, May 2010.
[10] Fred Long, Dhruv Mohindra, Robert C. Seacord, Dean F. Sutherland, David Svoboda The CERT Java Secure Coding Standard, Addison-Wesley, September 2011.
[11] Sutton, Michael, Adam Greene, and Pedram Amini. Fuzzing: brute force vulnerability discovery. Pearson Education, 2007.
[12] Patton, Ron. Software testing. Sams Pub., 2006.
[13] KHAN, Mohd Ehmer; KHAN, Farmeena. A Comparative Study of White Box, Black Box and Grey Box Testing Techniques. Editorial Preface, 2012.
[14] Bekrar, S., Bekrar, C., Groz, R., & Mounier, L. Finding software vulnerabilities by smart fuzzing. In Software Testing, Verification and Validation (ICST), IEEE Fourth International Conference, pp. 427-430. 2011.
[15] BALL, Thomas; RAJAMANI, Sriram K. The S LAM project: debugging system software via static analysis. In: ACM SIGPLAN Notices. ACM, pp. 1-3, 2002.
[16] OWASP, Top. Top 10 - 2013. The Ten Most Critical Web Application Security Risks, 2013.
[17] Ministry of Government Administration and Home Affairs, Software development security guide for developer and operator in E-government SW, 2012
[18] NEWSOME, James; SONG, Dawn. Dynamic taint analysis for automatic detection, analysis, and signature generation of exploits on commodity software. 2005.

김 만 식(Kim, Man Sik)

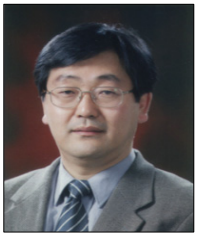


- 2010년 2월 : 안양대학교 컴퓨터공학과(공학사)
- 2012년 6월 : Towson University Computer Science(공학석사)
- 2014년 9월 ~ 현재 : 숭실대학교 컴퓨터학과 박사과정
- 관심분야 : 네트워크 보안, 시큐어코딩
- E-Mail : mansik@ssu.ac.kr

강 정 호(Kang, Jung Ho)

- 2000 2월 : 서울과학기술대학교 컴퓨터공학과(공학사)
- 2002년 2월 : 서울과학기술대학교 컴퓨터공학과(공학석사)
- 2013년 12월 : 숭실대학교 컴퓨터공학과 (박사)
- 관심분야 : NFC, 시큐어코딩
- E-Mail : kjh@naver.com

전 문 석(Jun, Mun Seog)



- 1989년 Univ. of Maryland Computer Science(공학박사)
- 1991년 2월 ~ 현재 : 숭실대학교 컴퓨터학과 정교수
- 관심분야 : RFID, PKI 컴퓨터통신
- E-Mail : mjun@ssu.ac.kr