ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr

웹 응용 보안을 위한 객체지향 분석·설계 방법론

An Object-Oriented Analysis and Design Methodology for Security of Web Applications

주 경 수[1] 우 정 웅[1*]
Kyung-Soo Joo Jung-Woong Woo

요 약

요즘 웹을 이용하여 많은 일들이 처리되고 있다. 이에 따라 다양하고 복잡한 기능을 가진 웹 기반의 응용 시스템들이 요구되고 있다. 이러한 웹 기반의 응용 시스템들을 효율적으로 개발하기 위하여 객체지향 분석·설계 방법론을 사용하고 있으며, 그 구현을 위하여 Java EE(Java Platform, Enterprise Edition) 기반의 기술들이 사용되기도 한다.

이렇게 개발된 웹 기반의 응용 시스템을 통해 많은 일들을 처리하면서 점차 보안과 관련된 이슈들이 중요해졌다. 이를 위하여 Java EE는 보안과 관련된 메커니즘을 제공하고 있지만, 효율적인 웹 응용 시스템을 개발하기 위한 객체지향 분석·설계 방법론과의 상호 연관성은 제공하지 못하고 있다. 이에 따라 Java EE 메커니즘에 따른 보안 방안은 개발 마지막 단계에서 비로소 구현되기 때문에, 요구사항 분석부터 구현에 이르기까지 시스템 개발 전 주기에 따른 일관된 보안 적용은 어려운 실정이다.

따라서 본 논문에서는 요구사항 분석부터 구현에 이르기까지, 보안이 강조된 '안전한 웹 응용 시스템을 위한 객체지향 분석·설계 방법론'을 제안한다. 제안한 객체지향 분석·설계 방법론은 보안에 관한 요구사항 분석과 시스템 분석 및 설계를 위하여 보안이 강조된 모델링 언어인 UMLsec을 사용하고, 그 구현을 위해서 Java EE 기반 기술 중 서블릿의 역할기반 접근제어(RBAC: Role Based Access Control)를 이용한다. 아울러 본 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'을 온라인 뱅킹 시스템 개발에 적용하여 그 효율성을 확인하였다.

주제어 : 객체지향 분석·설계, 웹 응용, 보안, RBAC, Java EE

ABSTRACT

Nowadays many tasks are performed using the Web. Accordingly, many web-based application systems with various and complicated functions are being requested. In order to develop such web-based application systems efficiently, object-oriented analysis and design methodology is used, and Java EE(Java Platform, Enterprise Edition) technologies are used for its implementation.

The security issues have become increasingly important. For such reasons, Java EE provides mechanism related to security but it does not provide interconnections with object-oriented analysis and design methodology for developing web application system. Consequently, since the security method by Java EE mechanism is implemented at the last step only, it is difficult to apply constant security during the whole process of system development from the requirement analysis to implementation.

Therefore, this paper suggests an object-oriented analysis and design methodology emphasized in the security for secure web application systems from the requirement analysis to implementation. The object-oriented analysis and design methodology adopts UMLsec, the modeling language with an emphasis on security for the requirement analysis and system analysis & design with regard to security. And for its implementation, RBAC (Role Based Access Control) of servlet from Java EE technologies is used. Also, the object-oriented analysis and design methodology for the secure web application is applied to online banking system in order to prove its effectiveness.

☞ keyword : Object-Oriented Analysis and Design, Web Application, Security, RBAC, Java EE

1. 서 론

요즘 웹을 이용하여 많은 일들이 처리되고 있다. 이에 따라 다양하고 복잡한 기능을 가진 웹 기반의 응용 시스템들이 요구되고 있다. 이러한 웹 기반의 응용 시스템들을 효율적으로 개발하기 위하여 객체지향 분석·설계 방법론을 사용하고 있으며, 그 구현을 위하여 Java EE 기반

1 Dept. of Computer Software Engineering, SoonChunHyang University, Asan, 336-745, Korea)
* Corresponding author (jyone0715@gmail.com)
[Recived 8 April 2013, Reviewed 15 April 2013(R2 11 June 2014), Accepted 11 July 2013]

의 기술들이 사용되기도 한다[1,2,3].

보안과 관련된 요구사항들이 증가되면서 보안에 대한 중요성 역시 점차 증가되고 있다. 따라서 모델링 과정부터 보안 대책을 도출하고 이를 시행하는 것이 필수이다[3-5]. 이를 위하여, Java EE는 기반 기술 중 서블릿을 통해 역할기반 접근제어와 같은 보안 방안을 지원하고 있지만, 이러한 기술들이 대부분 분석·설계의 결과로 사용된 것이 아니기 때문에 일관성이 없어, 보안에 취약한 웹 응용 시스템으로 개발될 가능성이 매우 높다[3,6-9]. 아울러 보안 취약에 따른 사례로, 국제 웹 보안 표준 기구(OWASP; The Open Web Application Security Project)에서 웹 어플리케이션 취약점 중 빈도가 많이 발생하고, 보안상 영향을 크게 줄 수 있는 것들을 10가지 선정하고 발표하였다[10]. 그 중, 본 논문에서 다루는 인증에 대한 내용과 보안상 잘못된 시스템 구성 등을 확인할 수 있으며, 본 논문에서 적용된 온라인 뱅킹 시스템과 보안에 대한 응용사례는 다음과 같다[11].

본 논문에서는 객체지향 분석·설계 방법론 중의 하나인 CBD(Component Based Development) 방법론을 기반으로, 요구사항 분석부터 구현에 이르기까지, 시스템 개발 전 주기에 걸쳐 보안에 대한 일관성을 제공하는, 보안이 강조된 객체지향 분석·설계 방법론을 제안한다. 아울러 보안에 대한 구현은 Java EE의 기반 기술 중 서블릿의 역할기반 접근제어를 이용한다.

본 논문에서는 Java EE 기반 기술 중 접근성을 위해 JSP와 서블릿을 대상으로 하였고 EJB(Enterprise Javabeans)는 엔터프라이즈급에서 사용되는 기술이기 때문에 제외하였다.

본 논문의 구성은 다음과 같다. 2장에서는 제안한 방법론의 이해를 돕기 위한 관련 연구들을 소개하고, 3장에서는 제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'을 설명하며, 4장에서는 제안한 방법론의 평가를 서술한 후, 마지막으로 5장에서 결론을 제시한다.

2. 관련연구

2.1 객체지향 분석·설계 방법론

객체지향 분석·설계 방법론 중 대표적인 UP(Unified-Process)의 특징은 유스케이스 기반, 아키텍처 중심, 반복 및 점증적이며, 유스케이스 모델, 분석 모델, 설계 모델 그리고 구현 모델 등으로 작성된다는 것이다[8].

또한 CBD 방법론은 컴포넌트를 기반으로 소프트웨어 시스템을 개발함으로써 사용자의 요구사항 변화에 신속하고 유연하게 대처하고자 하는 것을 목표로 한다[12].

한편 기존의 객체지향 분석·설계 방법론으로 도출된 개념적 모델은 클래스 다이어그램을 바탕으로 객체지향 프로그래밍 코드를 생성할 수 있지만, 보안에 대한 일관된 분석·설계 방법론은 제시하지 못하고 있다[8]. 따라서 보안과 일관된 분석·설계 방법이 고려되어야 한다.

2.2 UMLsec을 이용한 보안 유스케이스 모델링

UMLsec은 UML(Unified Modeling Language)에서 보안과 관련된 정보를 통합하여 확장된 모델링 언어이다. 아울러 사용자 데이터의 기밀성, 무결성 등의 보안 요구사항을 분석·설계에 반영하는 표준화된 기호를 제공하고 설계의 검증을 지원하며, 프로파일(Profile)의 속성 및 제약에 대한 명세를 통해 정의된다[13-15].

보안과 관련한 분석·설계 방법으로는, 기존의 객체지향 분석·설계 방법론과 보안 요구사항을 통합한 UML 기반의 개발방법론이 제시되었다[5]. 이 연구에서는 보안에 대해 확장된 UMLsec을 이용해서 보안이 중요한 응용 시스템 개발을 위한 일관된 객체지향 분석·설계 방법론을 제시하고는 있지만, Java EE와의 상호 연관성은 제공하지 못하고 있다. 따라서 보안과 Java EE와의 상호 연관성을 위한 방안이 제시되어야 한다.

2.3 Java EE 기반의 웹 보안

웹 응용 시스템들은 다양한 위험에 노출되어 있다. 이러한 위험을 막기 위해 Java EE에서 보안을 설정할 수 있으며, 서블릿 보안의 4요소는 인증, 인가, 비밀보장, 데이터 무결성으로 이뤄진다. 이에 서블릿에서의 인증은 BASIC, DIGEST, CLIENT-CERT, FORM과 같이 4가지 인증 방법이 존재한다[3,16].

3. 안전한 웹 응용을 위한 객체지향 분석·설계 방법론

본 논문에서 제안한 '안전한 웹 응용을 위한 객체지향 분석·설계 방법론'은 (그림 1)과 같이 요구사항 분석 단계에서 비기능적 요구사항 중 하나인 보안에 대한 정의를 추가하였으며, 추가된 요구사항은 UMLsec을 이용하여 정의하였다. 아울러 시스템 분석 및 설계 단계에서도 UMLsec을 이용하여 보안이 강조된 분석·설계를 표현하

였다. 또한 마지막 구현 단계에서는 분석·설계의 결과를 바탕으로, 보안에 대한 요구사항을 Java EE의 역할기반 접근제어를 이용하여 구현한다. 한편 요구사항 중 기능적 요구사항 분석과 시스템 분석 및 설계는 기존의 CBD 방법론을 적용하여 수행한다.

(그림 1) 제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'의 과정

(Fig. 1) Process of Object-oriented analysis and design methodology for security of web application

3.1 요구사항 분석

3.1.1 요구사항 리스트 작성

요구사항 정의는 사용자들이 소프트웨어에 기대하는 기능 및 비기능적 요구를 도출하고 검증하는 활동을 뜻한다.[1,17]. (표 1)은 기능적 요구사항과 비기능적 요구사항 중 보안에 해당하는 요구사항 정의가 함께 포함되어 있는, 온라인 뱅킹 시스템의 일부분에 대한 요구사항 리스트이다.

(표 1) '온라인 뱅킹 시스템'을 위한 요구사항 리스트

(Table 1) Requirement list for On-line banking system

1. 사용자는 조회 서비스를 이용할 수 있다.
2. 조회 서비스는 잔액 확인, 거래 목록 확인, 지난 기록 확인 및 다운로드 기능이 있다.
3. 사용자는 요금 지불 서비스를 이용할 수 있으며, 각종 세금을 납부하는 기능이다.
4. 사용자는 거래 서비스를 이용할 수 있다.
5. 거래 서비스는 자금 이체와 같은 기능이 포함된 기능 이다.
6. 관리자는 관리 기능을 통해 시스템의 전체적인 접근권한을 가지고 있으며, 또한 새로운 계좌에 대한 생성 및 삭제, 잔액 수정, 거래 취소, 사용자 등급을 설정할 수 있다.
7. 특정 사용자에 대한 시스템 사용권한을 설정할 수 있다.
8. 해당 시스템을 사용하기 위해서는 로그인이 필요하다.
9. 데이터 관리 및 보호를 위한 기능이 필요하다.

(표 2)는 (표 1)의 내용 중 보안과 관련된 요구사항만을 정리한 내용이며, (표 2)의 1번은 관리자 권한에 대한, 2번은 인증에 대한, 3번은 인가에 대한 보안 요구사항에 해당된다. 그리고 4번은 비밀보장 및 데이터 무결성에 해당하는 보안 요구사항이다.

(표 2) 보안 요구사항 정의

(Table 2) Defining security requirements

유형	설 명
보안	1. 관리자는 관리 기능을 통해 시스템의 전체적인 접근권한을 가지고 있으며, 또한 새로운 계좌에 대한 생성 및 삭제, 잔액 수정, 거래 취소, 사용자 등급을 설정할 수 있다. 2. 해당 시스템을 사용하기 위해서는 로그인이 필요하다. 3. 관리자는 특정 사용자에 대한 시스템 사용권한을 설정 할 수 있다. 4. 데이터 관리 및 보호를 위한 기능이 필요하다.

3.1.2 유스케이스 작성

유스케이스는 시스템이 어떤 일을 수행하기 위해 거쳐야 하는 단계들을 말하며, 또한 새로 만들 시스템이나 소프트웨어 변경사항에 대한 요구사항을 찾아내는 방법이다[17].

(표 1)에서 정의된 사용자 요구사항 리스트를 기반으로, 유스케이스를 작성한다. 다만 보안 요구사항이 있는 유스케이스의 경우에는 UMLsec 방법론에 따라 유스케이스를 확장해야 한다[5]. (표 3)은 온라인 뱅킹 시스템에 대한 유스케이스 목록의 일부분이며, (표 4)는 보안을 위해 UMLsec 방법론에 따라 확장된 유스케이스를 보여준다.

(표 3) 유스케이스 목록

(Table 3) Use case list

유스케이스명	설 명
회원가입	각 사용자는 시스템을 사용하기 위해서 회원가입을 할 수 있다.
로그인	각 사용자는 시스템을 사용하기 위해서 로그인 할 수 있다.
계정확인	시스템이 사용자의 계정을 확인할 수 있다.
잔액확인	각 사용자는 계좌의 잔액을 확인할 수 있다.
거래목록확인	사용자의 거래 목록을 확인할 수 있다.
거래목록다운로드	사용자의 거래 목록을 다운로드 할 수 있다.
요금지불	시스템을 통해 각종 세금을 납부할 수 있다.
계좌생성	관리자는 새로운 계좌를 생성할 수 있다.
계좌삭제	관리자는 기존 계좌를 삭제할 수 있다.
잔액수정	관리자 및 직원은 모든 일반 사용자의 잔액정보를 수정할 수 있다.
거래취소	관리자는 사용자가 수행한 거래를 취소할 수 있다.
등급설정	관리자는 각 사용자에 대한 접근권한을 설정할 수 있다.

(표 4) 보안 요구사항이 있는 유스케이스 - 등급설정 유스케이스

(Table 4) Use case having security requirement; Use case for rating set-up

Use Case : 등급설정

※ 액터와 관련된 위험성
- 고객은 자신과 관련된 정보만 확인할 수 있어야 한다. 관리자는 모든 사용자의 정보를 확인 및 수정할 수 있다.

※ 보안이 요구되는 입출력 데이터와 보안이 요구되지 않는 입·출력 데이터

보안이 요구되는 I/O	보안이 요구되지 않는 I/O
ID	결과 출력
패스워드	-

※ 변경된 시스템의 행동
- 사용자는 회원가입을 해야 한다.
- 시스템은 로그인을 통해 인증 절차를 거쳐야 하며, 그렇지 않을 경우 사용자는 시스템을 사용할 수 없다.
- 특히 인증 과정에서 입력 정보가 틀릴 경우 시스템은 관련 오류 메시지를 출력해야 한다.
- 관리자는 사용자의 등급을 설정한다.
- 시스템은 사용자에게 결과를 출력해 준다.

3.1.3 유스케이스 모델 상세화

유스케이스 상세화 활동에서는 직전 활동에서 도출된 각 유스케이스 별로 개요, 관련 액터, 우선순위, 선행/후행 조건, 시나리오, 비기능적 요구사항 항목으로 구성된 유스케이스 명세서를 작성해야 한다[17]. 또한 보안이 요구되는 유스케이스의 경우에는 비기능적 항목에서 보안에 대한 정의를 간결하면서 명확하게 정의하기 위해, 표 2를 참조하여 작성한다.

(표 5)는 보안이 요구되는 '등급설정' 유스케이스 명세서이다. 또한 유스케이스 명세서를 통해 해당 유스케이스의 다양한 상황, 즉 시나리오를 작성한다[17]. (표 6)은 '등급설정' 유스케이스의 기본 시나리오에 해당한다.

(표 5) '등급설정' 유스케이스 명세서

(Table 5) Use case description for rating set-up

항 목	설 명		
이름	등급설정		
개요	관리자는 각 사용자에 대한 접근권한을 설정할 수 있다.		
관련 액터	주액터	관리자	
우선 순위	1	중요도	1(상)
		난이도	1(상)
선행 조건	- 관리자로 로그인이 되어 있어야 한다. - 설정하고자 하는 사용자가 회원가입이 이루어진 상태이어야 한다.		
후행 조건	- 로그인 상태가 유지 되어야한다. - 시스템은 관리자에게 변경된 사용자의 정보를 보여준다. - 시스템은 사용자의 등급을 기록한다.		
시나리오	기본 시나리오	액터와 시스템 간의 기본 시나리오	
비기능적 요구사항	보안 요구사항 - 관리자는 시스템의 전체적인 접근권한을 가지고 있다. - 관리자는 특정 사용자에 대한 시스템 사용권한을 설정할 수 있다.		

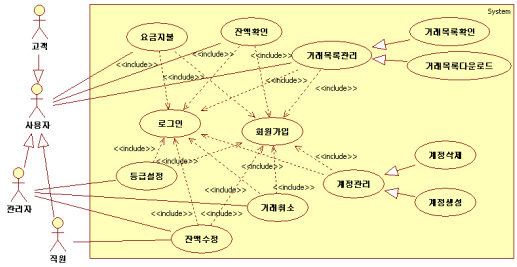
(표 6) '등급설정' 유스케이스의 기본 시나리오

(Table 6) Basic scenario of rating set-up use case

1. 사용자는 회원가입이 되어 있어야한다.
2. 사용자는 로그인 화면에서 ID와 패스워드를 입력하고 로그인 버튼을 누른다.
3. 시스템은 관리자화면을 보여준다. 관리자는 관리자화면에서 등급설정을 선택한다.
4. 등급설정화면에서 해당 사용자의 등급을 확인할 수 있으며, 등급을 수정하기 위해서는 등급설정버튼을 누른다.
5. 시스템은 상세한 등급정보화면을 보여준다.
   ※ 상세한 등급정보화면 : ID, 이름, 등급
6. 관리자는 등급을 수정한 후, 확인버튼을 누른다.
   시스템은 수정된 데이터를 기록하고 상세한 등급정보화면을 갱신한다.
7. 전 화면으로 되돌아가기 위해서는 취소버튼을 누른다.

3.1.4 유스케이스 모델 작성

유스케이스 모델 작성은 시스템이 제공할 개별 기능을 유스케이스로 표현하고, 유스케이스와 상호작용을 하는 시스템 외부의 존재를 액터로 표현한다. 그리고 시각적인 표현을 위해 UML의 유스케이스 다이어그램을 사용하여, 액터와 유스케이스 간의 연관 관계를 표현함으로써 어떤 액터가 어떤 유스케이스를 이용하는지를 기술한다[17]. (그림 2)는 온라인 뱅킹 시스템의 유스케이스 모델 작성을 보여준다.



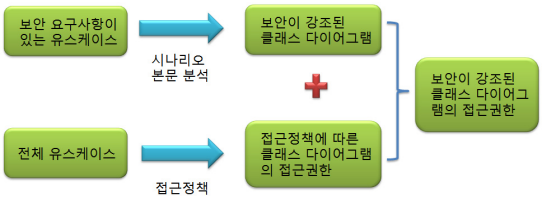
(그림 2) '온라인 뱅킹 시스템'을 위한 유스케이스 모델

(Fig. 2) Use case model for on-line banking system

3.2 시스템 분석 및 설계

시스템 분석 및 설계 단계는 사용자의 요구사항을 충족시킬 수 있도록 시스템의 구성 요소를 파악하는 것을 목표로 하며, 요구사항 모델을 바탕으로 수행되어야 한다[17].

제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'의 시스템 분석 및 설계 과정은 (그림 3)과 같다.



(그림 3) 접근정책에 따른 보안이 강조된 클래스 다이어그램의 생성과정

(Fig. 3) Creating process of security emphasized class diagram depending on access policy

3.2.1 유스케이스 본문 분석

유스케이스 본문 분석은 사용자로부터 얻은 요구사항 정보들을 토대로 작성된 유스케이스의 기본 시나리오 내용을 본문 분석하여, 시스템에 필요한 클래스들을 추출해 내는 작업을 말한다[1,17]. 본문 분석을 통해 추출할 수 있는 클래스는 경계 클래스(Boundary Class)와 제어 클래스(Control Class), 그리고 엔티티 클래스(Entity Class)가 있다.

3.2.2 접근정책 작성

다음은 접근정책 작성 활동으로, 각 액터들이 각각의 유스케이스에 대한 접근권한을 작성해야 한다[5]. 접근정책 작성 과정은 앞서 작성한 보안이 강조되어야 하는 유스케이스 명세서와 보안이 강조될 필요가 없는 일반 유스케이스 명세서를 토대로 유스케이스에 대한 접근권한을 명확하게 명시할 수 있다[3]. 작성된 접근정책은 이후 도출될 클래스 다이어그램에 대한 접근권한을 나타낸다. (표 7)은 온라인 뱅킹 시스템의 일부 유스케이스에 대한 접근정책을 정의한 것이다.

3.2.3 분석 클래스 다이어그램 작성

접근정책 작성 활동 이후, 분석 클래스 다이어그램의 작성은 유스케이스 시나리오를 본문 분석하여 클래스 다이어그램을 작성하는 활동이다[17]. 즉, 클래스들을 도출하고 클래스 간의 관계를 정의하는 활동이다.

보안 요구사항이 있는 유스케이스로부터 도출된 클래스들은 보안이 강조되는 클래스들이며, 각 클래스들은 (표 7)을 참고하여 접근정책에 따른 접근권한을 UMLsec 방법론에 따라 <<secrecy>> 스테레오 타입을 이용하여 작성한다.

(표 7) 액터에 따른 유스케이스 접근정책

(Table 7) Use case access policy according to an actor

	고객	직원	관리자
회원가입	X	X	X
로그인	X	X	X
계정확인	P	X	X
잔액확인	P	X	X
거래목록확인	P	X	X
거래목록 다운로드	P	X	X
요금지불	X	-	X
계좌생성	-	-	X
계좌삭제	-	-	X
잔액수정	-	X	X
거래취소	-	-	X
등급설정	-	-	X
범례 : 모든 권한(X), 일부 권한(P), 권한 없음(-)			

3.2.4 분석 클래스 다이어그램의 상세화

분석 클래스 다이어그램의 상세화에서는 직전 활동에서 도출된 보안이 강조된 클래스 다이어그램을 바탕으로 유스케이스 시나리오를 추가 본문 분석하여, 각 분석 클래스들의 속성들과 연산들을 정의한다[1,17].

3.2.5 Java EE 기반에서의 MVC 패턴 적용

상세화된 분석 클래스 다이어그램에 다음과 같이 MVC (Model-View-Controller)패턴을 적용한다.

① <<entity>> 스테레오 타입을 사용한 클래스는 Model로 대응 시킨다.
② <<boundary>> 스테레오 타입을 사용한 클래스는 View로서 JSP 등으로 구현한다.
③ <<control>> 스테레오 타입을 사용한 클래스는 Controller로서 서블릿 등으로 구현한다.
④ <<secrecy>> 스테레오 타입을 사용한 클래스는 보안이 강조되어야 하는 클래스이며, <<control>> 및 <<boundary>> 스테레오 타입과 같이 사용되었다면 Java EE의 역할기반 접근제어를 이용하여 구현한다.

3.3 구현

3.3.1 Java EE 기반의 역할기반 접근제어

'등급설정' 유스케이스와 관련된 '사용자관리' 클래스에 <<control>>과 <<secrecy>>가 사용되어 있으므로, Java

EE의 보안 메커니즘을 적용하기 위해 역할을 정의한다. (표 8)은 인증과 인가를 위한 역할을 정의한 것이다.

(표 8) 역할 정의
(Table 8) Role defining

```
- Tomcat-user.xml
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
    <role rolename="admin"/>
    <role rolename="customer"/>
    <user username="admin"
      password="admin1234"
      roles="admin"/>
    <user username="customer"
      password="customer1234"
      roles="customer"/>
</tomcat-users>
```

(표 9)는 인증에 대한 내용으로, 앞서 설명한 바와 같이 BASIC, DIGEST, CLIENT-CERT, FORM 으로 4가지 방식이 존재한다. 그 중, 본 논문에서는 FORM으로 인증을 구현하였으며, '<form-login-page>'와 '<form-error-page>'에서는 인증이 FORM 방식일 때 개발자가 임의로 작성한 페이지를 띄워주도록 정의할 수 있다. (그림 4)는 FORM 방식으로 구현된 인증에 대한 사항으로, 등록되지 않은 사용자가 로그인 할 경우 임의로 작성된 로그인 오류 페이지를 보여준다.

(표 9) 인증 구현
(Table 9) Implemented authentication

```
- web.xml
<login-config>
  <auth-method>FORM</auth-method>
  <form-login-config>
    <form-login-page>/login.jsp</form-login-page>
    <form-error-page>/loginerror.html</form-error-page>
  </form-login-config>
</login-config>
```

또한 (표 10)과 (표 11)은 특정 서블릿에 요청을 할 수 있도록 배포서술자에 그에 맞는 역할을 사상(Mapping) 해주어야 하며, 접근 가능한 자원 및 사용 가능한 HTTP 메소드를 지정해야 한다. 따라서 등급설정등과 같은 고객관리 페이지는 관리자만 접근 가능하기 때문에 해당 페이지에 대한 접근권한을 다음과 같이 설정하였다. 결과적으로, 관리자가 아닌 일반 사용자가 고객관리 페이지에 접근할 시 (그림 5)와 같은 오류 페이지를 보여주게 된다. 반면, (그림 6)은 관리자가 고객관리 페이지에 접근했을 때, 정상적으로 접근된 고객관리 페이지에 대한 그림이며, 아울러 비밀보장 및 데이터 무결성을 위해 https로 접근된 그림이다.

(표 10) 인가 1단계 : 역할 등록
(Table 10) Authorization step 1; Role registration

```
- web.xml
<security-role>
      <role-name>admin</role-name>
      <role-name>customer</role-name>
</security-role>
```

(표 11) 인가 2단계 : 자원 및 메소드 제약 정의
(Table 11) Authorization step 2; Defining resource and method restriction

```
- web.xml
<security-constraint>
  <web-resource-collection>
  <!-- 사용하는 이름 -->
      <web-resource-name>test web resource
      </web-resource-name>
  <!-- 해당 디렉토리에 접근가능 -->
      <url-pattern>/admin/Member.jsp</url-pattern>
  <!-- 제약을 걸 HTTP 메소드 -->
      <http-method>GET</http-method>
      <http-method>POST</http-method>
  </web-resource-collection>
  <!-- 정의된 자원을 호출할 수 있는 역할 -->
  <auth-constraint>
      <role-name>admin</role-name>
  </auth-constraint>
</security-constraint>
```

(그림 4) 로그인 오류 페이지
(Fig. 4) Login Error page

(그림 5) 접근권한에 따른 오류 페이지
(Fig. 5) Access rating Error page

(그림 6) 고객관리 페이지
(Fig. 6) Customer management page

4. '웹 응용 보안을 위한 객체지향 분석·설계 방법론' 평가

본 논문에서 제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'은 기존의 객체지향 분석·설계 방법론

이 제시하지 못했던 보안에 대한 연관성뿐만 아니라, UMLsec에서 제시하지 못했던 Java EE와의 상호 연관성을 역할기반 접근제어를 통해 지원하고 있다. 아울러 기존의 CBD 방법론을 기반으로 각 단계마다 UMLsec을 이용하여 보안에 대한 분석·설계를 진행했기 때문에 보안에 대한 일관된 분석·설계가 가능하다. 이에 따라 기존의 객체지향 분석·설계 방법론과 보안, 그리고 Java EE에 이르기까지, 안전한 웹 응용 시스템 개발을 위한 일관된 객체지향 분석·설계 방법론을 제시하였다.

인증을 통해 비 허가된 사용자가 허가된 사용자처럼 속일 수 있는 공격 유형을 (그림 4)와 같이 방어하며, 자신의 등급을 속이는 공격 유형은 인가를 통해 (그림 5)와 같이 방어한다. 그리고 사용자의 중요한 정보를 수정하거나 데이터를 훔쳐보는 공격 유형을 비밀보장 및 데이터 무결성을 통해 (그림 6)과 같이 방어함으로써, 본 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'의 효율성을 확인할 수 있다.

5. 결 론

본 논문에서는 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'을 제안하였다. 이를 위하여 기존의 CBD 방법론을 기반으로 각 단계마다 보안이 강조된 모델링 언어인 UMLsec을 사용하여 보안에 대한 분석·설계를 진행했기 때문에 보안에 대한 일관된 분석·설계가 가능하다. 보안이 강조된 모델링 언어인 UMLsec을 사용하고, 그 구현을 위해서 Java EE 기반 기술 중 서블릿의 역할기반 접근제어를 이용하였다. 이에 따라 보안 요구사항을 요구사항 분석부터 구현에 이르기까지 시스템 개발에 필요한 모든 단계에 일관되게 반영하였다.

따라서 본 논문에서 제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'은 기존의 객체지향 분석·설계 방법론이 제시하지 못했던, 보안에 대한 일관된 분석·설계 방법을 제공하고 있다. 아울러 UMLsec에서 제시하지 못했던 Java EE와의 연관성도 역할기반 접근제어를 통해 제공하고 있다. 이에 따라 기존의 객체지향 분석·설계 방법론과 보안 그리고 Java EE와의 상호 연관성을 제시하여 시스템 개발 전 주기에 대한 일관된 객체지향 분석·설계가 가능하다.

본 연구에서 제안한 '웹 응용 보안을 위한 객체지향 분석·설계 방법론'은 온라인 뱅킹 시스템 개발에 적용하여 그 효율성을 확인하였다.

참 고 문 헌(Reference)

[1] Brett D. McLaughlin, Gary Pollice, David West, "Head First Object Oriented Analysis & Design", pp.96-103, Hanbit Media. Inc, 2007.

[2] Han Jeong-Su, Kim Gwi-Jeong, Song Yeong-Jae, "Introduction to UML : Object-Oriented Design as in a friendly learning", Hanbit Media. Inc, pp. 58-66, 2009.

[3] Joo Kyung-Soo, Woo Jung-Woong, "A Development of the Unified Object-Oriented Analysis and Design Methodology for Security-Critical Web Applications Based on Object-Relational Data‐Forcusing on Oracle 11g-", Korea Society of Computer Infomation, Vol. 17, No. 12, pp. 169-177, 2012.

[4] Eduardo Fernández-Medinaa, Juan Trujillob, Rodolfo Villarroelc and Mario Piattinia, "Developing secure data warehouses with a UML extension", Journal Information Systems archive, vol. 32 No. 6, pp.826-856, 2007.

[5] G.Popp, J. Jurjens, G.Wimmel, R. Breu, "Security-Critical System Development with Extended Use Case", Asia-Pacific Software Engineering Conference, 5-1 self, 2003.

[6] Madan, s, "security Standards Perspective to Fortify Web Database Applications From Code Injection Attacks", International Conference on Intelligent Systems, Modelling and Simulation(ISMS), vol. 10, pp. 226-230, 2010.

[7] lqra Basharat, Farooque Anam, Abdul Wahab Muzaffar, "Database Security and Encryption: A Survey Study", International Journal of Computer Application, vol. 47, No. 12, pp28-34, 2012

[8] Cho Wan-Su, "UML 2 & UP Object-Oriented Analysis&design", pp.189-205, Hongrung Publishing Company, 2005.

[9] David Basin, Jürgen Doser and Torsten Lodderstedt, "Model Driven Security: from UML Models to Access Control Infrastructures", ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 15 No. 1, pp39‐91, 2006

[10] OWASP TOP 10, https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

[11] Certification : protect of on-line banking, http://www.tekbar.net/ko/network-knowledge/two-factor-authentication-the-protection-of.html
[12] Jeon Byeong-Seon, “CBD, WHAT&HOW”, Wowbooks, pp. 189-205, 2005.
[13] R. Matulevicius, M. Dumas, “Towards Model Transformation between SecureUML and UMLsec for Role-based Access Control”, IEEE, DB&IS, pp.339-352, 2010.
[14] Denis Hatebur, Maritta Heisel, Jan Jürjens, Holger Schmidt, “Systematic Development of UMLsec Design Models Based on Security Requirements”, Lecture Notes in computer Science, Vol. 6603, pp.232-246, 2011.
[15] Salim Chehida, Mustapha kamel Rahmouni, “Security Requirements Analysis of Web Applications using UML”, ICWIT, Vol. 867, pp.232-239, 2012.
[16] Kathy Sierra, Bert Bates, Bryan Basham, “Head First Servlet & JSP”, pp.683-721, Hanbit Media. Inc, 2009.
[17] Chae Heung-Seok, Object-oriented CDB Project for UML and Java as learning, Hanbit Media. Inc, pp. 84-112, 2009.

◐ 저 자 소 개 ◑

주 경 수
1993년 고려대학교 대학원 전산학과 졸업(박사)
1986년~현재 순천향대학교 컴퓨터소프트웨어공학과 교수
관심분야 : Database System, Object Oriented System, Cluud Databases, BigData Databases
E-mail : gsoojoo@sch.ac.kr

우 정 웅
2012년 순천향대학교 컴퓨터학과 졸업(학사)
2012년~현재 순천향대학교 컴퓨터소프트웨어공학과 석사과정
관심분야 : Database System, Object Oriented System, Cluud Databases, BigData Databases
E-mail : jyone0715@gmail.com