보안 요구사항 기반의 보안 위험도 분석 기법

이동현, 이명락, 인호
고려대학교 정보통신대학
e-mail : tellmehenry@korea.ac.kr, lmr2010@kroea.ac.kr, hoh_in@korea.ac.kr

From Security Requirements to a Security Risk Analysis Method

Dong-hyun Lee, Myoung-rak Lee, Hoh In
College of Information and Communication, Korea University

요 약

실제 소프트웨어 개발에서 지속적으로 보안관련 문제들이 발생하고 있으므로 이를 해결하기 위하여 소프트웨어 개발 주기의 초기 단계인 요구사항 분석단계에서 보안 요구사항을 추출하는 것이 필요하다. 이는 요구사항 분석 단계에 대한 투자가 소프트웨어 개발의 성공률을 높일 수 있기 때문이다. 보안 요구사항을 추출하는 기법에 대해서는 여러 방면으로 연구가 시작되었으나, 보안 요구사항을 토대로 향후 소프트웨어 개발 과정에서의 보안 관련 위험도를 산정하여 보안 투자의 우선순위를 정하는 기법은 아직 연구되어 있지 않다. 그러므로 본 논문에서는 추출된 보안 요구사항을 가지고 소프트웨어 보안에 대한 위험도를 산정하여 투자 비용의 우선순위를 산정하는 절차에 대해 제안한다.

1. 서론

소프트웨어를 개발하는 동안에 발생하는 버그나 보안문제와 같은 소프트웨어 결함은 피할 수 없는 문제이다. 이를 해결하기 위하여, 소프트웨어 요구사항 명세에서부터 테스트에 이르는 전체 소프트웨어 개발 주기에 걸쳐 여러 가지 개발 방법론들이 연구되었다. 그러나 기존의 소프트웨어 개발 방법론들은 소프트웨어를 잘 만들어내고 생산성을 높이는 데 초점을 맞추어왔다. 또한 소프트웨어공학의 전통적인 관점에서 살펴보면, 소프트웨어 결함은 소프트웨어 품질을 개선하기 위하여 제어될 수 있는 요소로 인식되어왔다.

따라서 기존의 소프트웨어 개발자들은 보안에 대한 관심을 자주 간과하게 되었으나, 최근 소프트웨어공학계에서는 소프트웨어의 주요 기능이 정보보안과 관계가 없다 하더라도 정보보호 또한 중요하다는 것을 점차 깨닫기 시작하였다. 예를 들어 일반적으로 소프트웨어의 유저인터페이스에서 보안관련 요소나 동작구조는 중요한 요소가 아니었으나 지속적으로 메뉴구조나 설정을 탐색하다 보면 대개 몇몇 보안관련 아이템들이 빠져있는 것을 볼 수가 있다[1].

이와 같이 실제 소프트웨어 개발에서 지속적으로 보안관련 문제들이 발생하고 있으므로 이를 해결하기 위하여 소프트웨어 개발 주기의 초기 단계인 요구사항 분석단계에서 보안 요구사항을 추출하는 것이 필요하다[ ]. 이는 스텐디쉬 그룹의 CHAOS 보고서[ ]에서도 잘 나타나 있듯이 초기 요구사항 분석에 대한 투자가 소프트웨어 프로젝트의 성공에 많은 영향을 끼치는 것을 통해서도 알 수 있는 것처럼, 소프트웨어 개발 초기에 보안 요구사항을 추출하여 이후의 소프트웨어 개발에 반영하는 것은 소프트웨어의 결함을 줄일 수 있는 솔루션이 될 수 있다.

보안 요구사항을 추출하는 기법에 대해서는 여러 방면으로 연구가 시작되었으나, 보안 요구사항을 토대로 향후 소프트웨어 개발 과정에서의 보안 관련 위험도를 산정하여 보안 투자의 우선순위를 정하는 기법은 아직 연구되어 있지 않다. 그러므로 본 논문에서는 추출된 보안 요구사항을 가지고 소프트웨어 보안에 대한 위험도를 산정하여 투자 비용의 우선순위를 산정하는 절차에 대해 제안한다. 본 논문의 구성은 다음과 같다. 2 장에서는 관련연구로서 본 논문에서 사용한 보안 요구사항 추출기법과 보안 투자 비용 분석 기법에 대해 소개하며 3 장에서는 제안하는 보안 요구사항 기반의 보안 위험도 분석 기법을 제시하고 4 장에서는 결론 및 향후 연구에 대해서 서술한다.

2. 관련연구

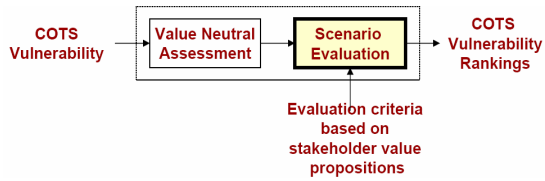
2.1 보안 요구사항 추출기법

보안 요구사항을 추출하는 기법에 대해서 여러 가지 기법들이 제안되었다[1]. 그 가운데 본 논문에서는 유즈 케이스(use case)를 가지고 보안 요구사항을 추출하는 방법에 대하여 살펴본다. Sindre 와 Opdahl 은 미스유즈 케이스(misuse case)라는 개념을 제안하여 보안 요구사항을 추출하는 기법을 제안하였다[2]. 유즈 케이스 간에는 일반적으로 include, extend, generalize 등의 관계를 가질 수 있는데, 미스유즈 케이스에서도 동일하게 정의되며, 일반적인 association 관계는 미스유저(misuser)와 미스 유즈케이스 사이에 정의된다. 그리고 일반 유즈 케이스와 달리 미스유즈 케이스에서는 다음과 같은 관계가 더 정의된다.

- *Mitigate*: 유즈 케이스가 미스유즈 케이스의 대책 수단이 되는 경우에 정의된다. 즉, 유즈 케이스가 미스유즈 케이스가 일어날 확률을 줄이는 수단이 되는 관계를 말한다.
- *Threaten*: mitigate 와는 반대로 미스유즈 케이스가 유즈 케이스를 방해하는 관계를 말한다.

2.2 보안 투자 비용 분석 기법

미국 남가주대학(University of Southern California) 시스템 및 소프트웨어 공학센터의 Chen 과 Bohem 교수는 소프트웨어공학 측면에서의 보안을 연구하였고 T-MAP 이라는 프레임워크를 제안하였다[3]. T-MAP 프레임워크는 보안 공격 경로를 분석하여 소프트웨어 시스템의 위험도를 모델링하는 프레임워크이다 (그림 1).



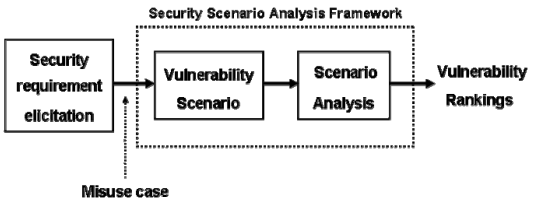
(그림 1) T-MAP Framework

T-MAP 프레임워크는 COTS(Commercial Off-The-Shelf) 소프트웨어 기반의 소프트웨어 시스템을 가정한다. 먼저 소프트웨어 시스템에 대한 이해당사자를 배제하고 COTS 소프트웨어의 취약점을 분석하고 (Value Neutral Assessment), 그 결과를 바탕으로 이해당사자들의 가치를 반영하여 시나리오를 만들어 (Scenario Evaluation) COTS 소프트웨어의 취약점 랭킹을 산정한다.

T-MAP 프레임워크는 보안 투자 비용 분석 기법에 있어서 선도적인 연구이기는 하나, COTS 소프트웨어를 기반으로 하고 있기 때문에 일반 소프트웨어 개발에 직접적으로 적용하기는 어렵다는 단점이 있다. 따라서 본 논문에서는 UML 을 이용하여 보안 요구사항 기반의 새로운 보안 투자 비용 분석 프로세스를 제안하고자 한다.

3. 보안 요구사항 기반의 보안 위험도 분석 프로세스

3 장에서는 본 논문에서 제안하는 보안 요구사항 기반의 보안 투자 비용 분석 프로세스에 대해 설명한다. 전체적인 프로세스는 그림 2 과 같다.



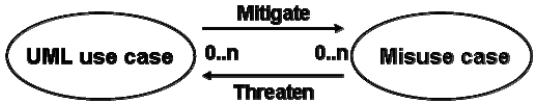
(그림 2) 전체 프로세스

보안 요구사항 추출(Security Requirement Elicitation)에서는 2.1 절에서 설명한 미스유즈 케이스를 통하여 보안 요구사항을 추출한다. 이 때 각각의 미스유즈 케이스가 보안 취약점이 되고, 추출한 미스유즈 케이스들을 보안 시나리오 분석 프레임워크(Security Scenario Analysis Framework)내의 보안 취약점 시나리오 생성(Vulnerability Scenario)으로 보낸다. 취약점 시나리오 생성단계에서는 미스유즈 케이스와 일반 유즈 케이스 간의 관계(Mitigate, Threaten)를 통해 유즈 케이스 별로 취약점 시나리오를 생성한다. 생성된 시나리오는 시나리오 분석과정(Scenario Analysis) 단계로 넘어가서 시나리오의 중요성을 분석하게 되고, 최종적으로 보안 취약성 랭킹(Vulnerability Ranking)을 산정하게 된다.

3.1 절과 3.2 절에서는 보안 시나리오 분석 프레임워크의 두 가지 컴포넌트인 보안 취약점 시나리오 생성과 시나리오 분석에 대해서 자세히 설명한다.

3.1 보안 취약점 시나리오 생성

2.1 절에서 설명한 것과 같이 일반 UML 유즈 케이스와 미스유즈 케이스 사이에는 Mitigate 와 Threaten 이라는 관계를 정의할 수 있다(그림 3). 또한 유즈 케이스와 미스유즈 케이스 사이에는 n:n 의 관계가 존재한다.



(그림 3) 유즈 케이스와 미스유즈 케이스 간의 관계

보안 취약점 시나리오 생성 단계는 다음과 같다.

1. 일반 유즈 케이스가 몇 개의 미스유즈 케이스와 관계를 가지고 있는지 찾는다.
2. 일반 유즈 케이스와 미스유즈 케이스 간의 Mitigate 또는 Threaten 관계를 찾아내고 유즈 케이스와 미스유즈 케이스에 따른 액터(actor)를 판별한다.

Scenario N: System administrator protects information mitigating stealing card information			
Actor	**Use case**	**Relation**	**Misuse case**
System Administrator	Protect information	Mitigate	Steal card information
Scenario M: A malicious man floods system threatening access of web page			
Actor	**Use case**	**Relation**	**Misuse case**
Malicious man	Access web page	Threaten	Flood system

(표 1) 보안 취약점 시나리오 테이블

3. Mitigate 관계이면, 액터-유즈 케이스-mitigate-유즈 케이스 순의 시나리오를 생성하고, Threaten 관계이면, 액터- 미스유즈 케이스-Threaten-유즈 케이스 순의 시나리오를 생성한다.

표 1 은 보안 취약점 시나리오 예제를 보여준다. 시나리오 N 은 Mitigate 관계를 보여주며 시나리오 M 은 Threaten 관계를 나타낸다.

3.2 시나리오 분석

시나리오 분석 단계에서는 생성된 보안 취약점 시나리오를 가지고 유즈 케이스 별 위험도를 산정하여 취약성 랭킹을 산정한다. 표 2 는 유즈 케이스 K 에 대한 위험도를 산정한 예제이다. 먼저 유즈 케이스 K 와 관련한 시나리오를 나열하고 관련 미스유즈 케이스와 relation, 시나리오 발생확률 P 와 시나리오 가중치 S 를 구하게 된다. 시나리오 발생확률과 시나리오 가중치는 [5]에서 제시한 방법을 따르고 위험도 계산법 역시 [5]에서 제시한 방법을 따르고 있다. 즉 위험도 = (시나리오 발생확률*시나리오 가중치)의 방법으로 산정하게 된다.

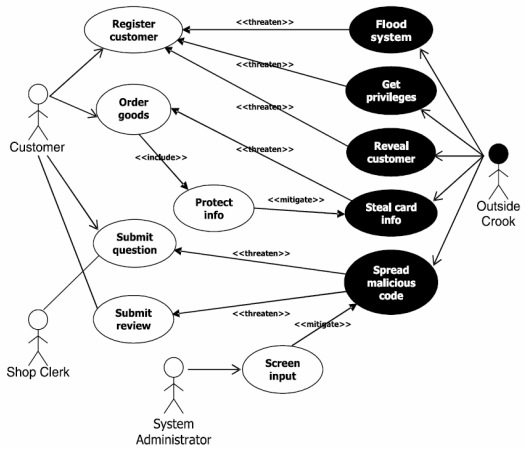
Use case K					
Scenario	Misuse case	Relation	Scenario Possibility (P)	Scenario weight (W)	Risk (P x W)
1	AAA	Mitigate	0.2	-0.5	-0.1
...	BBB	Threaten	0.8	+0.7	+0.56
K	CCC	Mitigate	0.4	-0.1	-0.04
...	...	...	...	...	...
N	ZZZ	Threaten	0.6	+0.9	+0.54

(표 2) 유즈 케이스 별 시나리오 분석

다만 본 논문에서는 Mitigate 와 Threaten 을 위험도 산정에 반영하기 위하여 Mitigate 의 경우 양의 값으로, Threaten 의 경우 음의 값으로 결과가 나오도록 하였다. 이는 Mitigate 의 경우 유즈 케이스 K 가 시나리오의 위험도를 낮출 수 있고, Threaten 의 경우 시나리오의 위험도를 증가시킬 수 있기 때문이다. 따라서, 유즈 케이스 K 의 총 위험도는 ∑(시나리오 별 위험도) = (- 0.1) + (+0.7) + (-0.4) + … + (+0.54) 의 계산값으로 구해진다. 그리고 유즈 케이스 별로 구한 총 위험도를 가지고 유즈 케이스 별 랭킹을 구하게 된다.

4. 사례연구

그림 4 는 관련 연구 [2]에서 제시하고 있는 미스유즈 케이스 예제로서, 인터넷 전자 상거래에 대한 예제이다. 검은색으로 표시 된 부분이 미스유즈 케이스들이고, 검은색 액터는 미스유즈 케이스를 사용하는 악의적 액터를 나타낸다. 표 3 은 그림 1 을 가지고 미스유즈 케이스의 시나리오를 추출한 결과의 일부를 나타내며, 표 4 는 Register customer 라는 유즈 케이스에 대하여 시나리오를 분석한 결과이다. 따라서 Register customer 유즈 케이스의 총 위험도는 0.72 + 0.42 + 0.16 = 1.3 이 된다. 표 4 와 같은 방법으로 모든 유즈 케이스에 대한 위험도를 산정하고 나면 표 5 와 같이 유즈 케이스의 위험도에 대한 랭킹을 산정하여 우선순위를 결정한다.



(그림 4) 미스유즈 케이스 예제[2]

Scenario 1: System administrator inputs screen to mitigate spread malicious code			
Actor	**Use case**	**Relation**	**Misuse case**
System Administrator	Screen input	Mitigate	Spread malicious code
Scenario 2: Outside crook reveals customer threatening register customer			
Actor	**Use case**	**Relation**	**Misuse case**
Outside crook	Reveal customer	Threaten	Register customer

(표 3) 전자 상거래 예제에서의 보안 취약점 시나리오

Use case Register customer					
Scenario	Misuse case	Relation	Scenario Possibility (P)	Scenario weight (W)	Risk (P x W)
1	Flood system	Threaten	0.8	+0.9	+0.72
2	Get privileged	Threaten	0.7	+0.6	+0.42
3	Reveal customer	Threaten	0.4	+0.4	+0.16
Total risk					**1.3**

(표 4) 전자 상거래 예제에서의 유즈 케이스 별 시나리오 분석

Use case	Total risk	Rank
Register customer	1.3	1
Order goods	0.84	3
Submit question	0.52	4
Submit review	0.33	5
Screen input	0.91	2

(표 5) 유즈 케이스 별 보안 위험도 랭킹

5. 결론 및 향후 연구 방향

일반적으로 보안 투자에 대한 ROI(Reteurn of Investment)는 가시적이지 않으므로, 많은 소프트웨어 개발자들은 보안 요구사항에 대해 소홀히 여기고 보안에 대한 투자를 부가적인 사항으로 생각하고 있다. 그러나 보안에 대한 투자는 점점 필수적으로 인식될 것이다.

본 논문에서는 보안 요구사항을 기반으로 하여 보안 투자 비용을 분석하는 기법에 대해서 제안하였다. 서론에서도 언급한 것처럼, 소프트웨어 개발의 초기 단계에서 보안에 대한 투자를 분석하고 가시화하는 것은 소프트웨어 프로젝트의 성공에 많은 영향을 끼칠 수 있다.

향후 연구로는 보안 요구사항 추출에 있어 보다 정량적인 방법을 사용하여 보안 투자 비용 분석을 보다 정량적으로 측정 가능하도록 하는 것이다.

Acknowledgement

"본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음" (IITA-2008-(C1090-0801-0032))

참고문헌

[1] Inger Anne Tøndel, Martin Gilje Jaatun, and Per Håkon Meland, "Security Requirements for the Rest of Us: A survey", IEEE SOFTWARE, vol. 25, num 1, 2008

[2] G. Sindre and A.L. Opdahl, "Eliciting Security Requirements with Misuse Cases," Requirements Eng., vol. 10, no. 1, 2005, pp. 34–44

[3] Yue Chen, Barry Boehm, Luke Sheppard, "Measuring Security Investment Benefit for COTS Based Systems - A Stakeholder Value Driven Approach", The 2007 Workshop on the Economics of Information Security (WEIS 2007), June 2007

[4] CHAOS Report, The Standish Group Report, 1995

[5] Hoh Peter In, Young-Gab Kim, Taek Lee, Chang-Joo Moon, Yoonjung Jung, and Injung Kim, "A Security Risk Analysis Model for Information Systems", AsiaSim 2004, LNAI 3398, pp. 505~513, 2005. Springer-Verlag Berlin Heidelberg 2005