특집 08

정보보증을 위한 시뮬레이션 접근: 취약성 분석을 중심으로

목 차

이장세 · 유용준 · 지승도
(한국해양대학교 · 한국항공대학교)

1. 서 론

전 세계적인 고도의 정보화와 인터넷의 보편화는 사회 전반에 걸쳐 다양한 순기능을 제공하고 있다. 그러나 정보시스템 및 정보통신 기반 환경에 대한 의존도가 증가함에 따라 이들에 대한 침해로 개인을 비롯한 국가적, 경제적, 사회적인 막대한 피해를 야기한다. 이에 미국에서는 2001년 9.11테러이후 사이버보안에 대한 연구의 필요성이 특히 강조되고 있으며[1], 국내에서도 2003년 12월 슬래머웜으로 인한 정보통신망의 마비사태를 계기로 정보보호에 대한 중요성이 새로이 부각되고 있다[2]. 한편, 정보보증(Information Assurance)이란 정보보안을 포괄하는 개념으로 정보와 정보시스템에 대한 침해에 관한 보호와 더불어 신뢰성, 가용성 및 무결성을 보장하기 위한 방어 및 정보시스템을 복원시키기 위한 조치로서[1,4], 이를 달성하기 위하여 정보 시스템 및 정보통신 기반환경에 대한 취약성 평가가 필수적인 것으로 인식되고 있다[4]. 이와같은 취약성 평가는 실제 시스템 및 환경에 대한 직접적인 시험을 통한 평가가 가장 효과적이나 그에 따른 비용, 시간, 피해의 책임문제, 다양한 시험의 어려움등으로 인하여 정보보증 관점에서의 IT시스템 모델링 및 시뮬레이션에 대한 연구의 중요성이 강조되고 있다[3]. 모델링과 시뮬레이션 기법은 시스템 설계 및 분석을 위하여 다양한 분야에서 적용되고 있으나, 정보 보증 분야의 경우 사이버 공격과 방어의 복잡성, 방대한 탐색 공간, 공격과 방어에 대한 데이타의 부족 등으로 다른 분야에 비하여 연구가 미흡한 실정이다[8]. 또한, 기존의 취약성 분석에 관한 연구는 주로 취약성을 탐지하고 제거하기 위하여 취약성을 분류하는 연구중심으로서 시뮬레이션에 적용하여 정량적으로 평가하기에는 어려움이 있다. 따라서, 본 연구에서는 보안 관점에서의 모델링 및 시뮬레이션을 기반으로 취약성을 분석하여 정량적으로 평가하는 방법에 대하여 기술하고자 한다. 시뮬레이션 기반의 정량적인 취약성 분석은 실제 시스템 및 환경에서의 시험이 갖는 한계를 극복할 수 있다. 또한 사이버 공격에 따른 구체적인 시스템의 변화를 분석할 수 있으며 이를 통하여 공격의 피해예측 및 방어 비용 분석 등 정보보증을 위한 정량적인 평가 및 응용에 적용될 수 있을 것으로 기대된다.

2. 관련 연구

2.1 취약성 분석 연구 동향

취약성이란 공격에 이용당하기 쉬운 약점을 말하는 것으로 최근까지 취약성을 정성적으로 분류하는 연구가 주류를 이루고 있다. Aslam[5]은 소프트웨어의 취약성을 코딩 취약성과 발생적 취약성으로 구분하고 특징에 따라 유사한 것을 그룹화함으로써 취약성 데이터를 데이터베이스화하는데 효과적으로 적용될 수 있는 분류기법을 개발한 바 있다. Landwehr[6]는 설계자 및 관리자의 보안 수행을 위한 분류법을 개발하였으나 분류기준에 대한 정보가 부족할 경우 분류에 모호함이 발행할 수 있는 단점이 있다. 또한, 이들 분류법을 토대로 Cohen, 퍼듀대학의 COAST, 미 표준연구소(NIST), 침해사고대응센터(CERT) 등에 의하여 침입탐지 및 대응을 위한 취약성 데이터베이스가 개발되었으며, 미 정부의 비영리 연합단체인 MITRE에서는 동일한 취약성에 대한 명칭의 표준화 연구를 통하여 CVE(Common Vulnerabilities and Exposures)를 개발한 바 있다[12]. 한편, 최근 Sample은 취약한 자산 및 각 자산의 방어 비용 분석, 공격의 예측 및 효율적인 정보보증을 위한 의사결정등을 위하여 정량적으로 취약성을 분석하는 것이 필요하다고 지적한 바 있다[7]. Hariri[14]는 시스템 및 네트워크의 상태로부터 측정 가능한 메트릭스를 정의하고 실제 시스템에 대한 온라인 분석을 통하여 정량적으로 취약성 분석을 시도하였으나, 실제 시스템을 대상으로 한 분석으로서 모델을 대상으로 취약성을 분석하고 예측하는 시뮬레이션 접근과 차이가 있다.

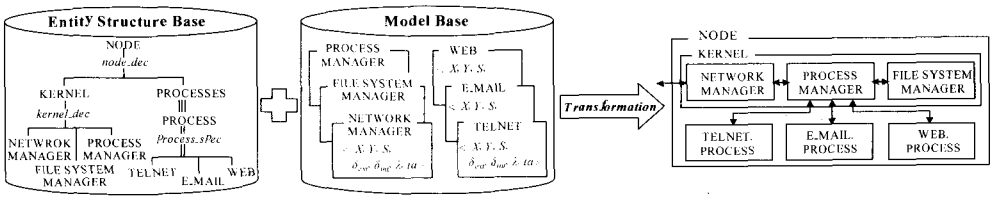
2.2 보안 모델링 및 시뮬레이션 연구 동향

본 연구와 관련된 정보보증 분야의 대표적인 연구로서 Cohen[8]의 원인-결과 모델(Cause-Effect Model)은 사이버 공격과 방어에 대한 확률론적인 모델링으로서 원인과 결과 사이의 지연시간 개념을 이용하여 특정 원인에 대한 결과가 출력될 수 있도록 시뮬레이션하였다. Cohen의 원인-결과 모델은 이미 알려진 공격과 방어에 대한 정의를 통하여 쉽게 시뮬레이션을 수행하여 공격과 방어로 인한 지연시간의 통계적인 분석을 제공할 수 있다는 장점이 있는 반면 공격과 방어에 있어서 대상 노드에서의 구체적인 상태의 변화나 동작 등을 알 수 없는 단점이 있다. 또한 Nong Ye[9]는 사이버 공격에 대한 방어를 위한 프로세스 제어 접근에 대한 연구를 통하여 복잡한 사이버 공격 모델의 추상화 단계를 제안하고 기능적 단계의 모델링의 필요성을 강조한 바 있다. 이와 같이 사이버 공격과 방어 모델 프로세스에 대한 추상화 단계를 정함으로써 범위가 방대하고 복잡성이 높은 보안 모델링에 대한 좋은 방향을 제시하고 있으나 이에 대한 모델링 및 시뮬레이션의 적용이 미흡하다. 최근에는 SSFNet, EASEL등과 같은 대규모 네트워크 모델링을 위한 프레임워크 및 언어가 개발되어 보안 분야에 적용되고 있으나 보안 시스템의 배치, 정책 적용 등에 따른 성능 분석 및 침해의 파급 분석 등을 위한 연구가 대부분으로 실제 네트워크에 대한 취약성 분석을 위하여 모델링과 시뮬레이션을 적용한 연구는 미흡한 실정이다.

3. 시뮬레이션을 이용한 정량적 취약성 분석 방법

3.1 SES/MB 프레임워크

Zeigler에 의해 제안된 SES/MB[16]는 기존의 동역학적 방법론과 AI의 기호적 방법론을 체계적으로 결합시킨 모델링 및 시뮬레이션 환경을 제공한다. SES/MB는 System Entity Structure와 Model Base의 두 구성원으로 이루어진다. SES는 시스템의 구성관계, 구성원의 종류, 구성원들의



(그림 1) SES/MB 프레임워크 개념

결합구조, 그리고 제약조건등의 구조적 지식을 표현할 수 있는 수단을 제공하며, MB는 동역학적으로 시스템의 행위를 표현할 수 있는 수단을 제공하는 모델들로 구성된다. 특히, 모델베이스에 저장되는 모델은 이산 사건 모델링을 위한 대표적인 형식론인 DEVS (Discrete Event System Specification)에 의하여 표현된다. DEVS모델[16]은 연속적인 시간상에서 이산적으로 발생하는 사건들에 대하여 시스템의 행위를 측정하는 것으로서 다음과 같은 형식론에 의해 모델을 표현한다.

$M = \langle X, Y, S, ta, \delta_{ext}, \delta_{int}, \lambda \rangle$

여기서, 3개의 집합과 4개의 함수로 구성된 7개의 구성요소는 입력집합 X, 출력집합 Y, 상태집합 S, 시간진행함수 ta, 외부상태전이함수 δ_{ext}, 내부상태전이함수 δ_{int}, 출력함수 λ로 이루어진다.

(그림 1)은 SES/MB개념에 따라 시뮬레이션 모델 구조가 생성되는 과정을 나타낸다. 시스템에 대한 모든 가능한 구조는 SES를 통하여 계층구조적으로 간결하게 표현되어 저장되며, 실세계의 대상에 대한 동역학적 모델은 모델베이스에 각각 저장된다. 이와같이 개별적으로 저장된 구조적 지식과 동역학적 모델로부터 변환(transformation)을 통하여 다양한 시뮬레이션 모델 구조를 생성함으로써 실세계의 시스템에 대한 모델링 복잡도를 줄일 수 있다[16]. 따라서, 구조적으로 복잡한 정보통신 기반 환경과 이산적으로 발생하는 사이버 공격 이벤트에 대한 모델링과 시뮬레이션이 요구되는 정보보증분야의 시뮬레이션을 위하여 계층구조적이고 모듈러한 특징을 갖는 SES/MB프레임워크는 효과적으로 적용될 수 있다.

3.2 정량적 취약성 매트릭스

시뮬레이션 결과를 토대로 취약성을 정량화하기 위하여 다음과 같이 노드 및 네트워크 취약성 매트릭스를 정의하였다[11,13].

3.2.1 노드 취약성

노드 취약성은 취약성 항목에 대한 네트워크 구성원의 구성설정 및 상태를 종합하여 정량화할 수 있다. 즉, 노드 취약성 〈표 1〉과 같이 시간에 의존적이지 않은 정적인 항목과 시간에 의존적인 동적인 항목에 의하여 분석되며 시간에 의존적인 동적인 취약성의 분석을 위하여 시뮬레이션을 이용한다.

〈표 1〉 취약성의 예

구분	취약성 항목	가중치(w)	관련 공격 시나리오	비고
정적 취약성	Vul_{Phf}	0.75	scenario-2	Phf CGI vulnerability
	Vul_{tmpfs}	1.0	scenario-5	SunOS4.1.4 tmpfs vulnerability
	Vul_{glance}	1.0	scenario-4	HP-UXB 1.0.2 glance vulnerability
	...	...	...	...
동적 취약성	$Vul_{Password}$	0.5	scenario-1 scenario-2	Password vulnerability
	$Vul_{Userfile}$	0.75	scenario-3	User file vulnerability
	$Vul_{Filesystem}$	0.75	scenario-3	Files system vulnerability
	...	...	...	...

노드 취약성은 식(1)과 같이 취약성 항목에 대한 취약성 값의 가중치 평균으로 정량화한다.

$$NV_i = \sum_{j=1}^{n}(w_j \times vul_j) / \sum_{j=1}^{n} w_j \quad (1)$$

여기서, n는 취약성 항목의 총 개수를 의미하며 w_j와 vul_j는 j번째 취약성 항목의 가중치와 취약성 값을 의미한다.

3.3.2 네트워크 취약성

네트워크 취약성은 평가 대상 네트워크에 대한 추상화된 취약성을 의미한다. 네트워크 취약성은 식(2)와 같이 네트워크를 구성하는 모든 노드에 대하여 각각의 중요도를 고려한 산술평균으로 정량화한다.

$$NetV_i = \sum_{j=1}^{n}(w_j \times NV_j) / \sum_{j=1}^{n} w_j \quad (2)$$

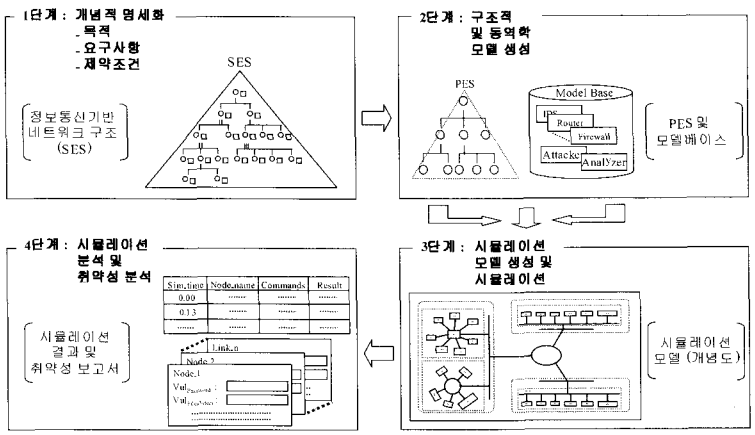
여기서, n는 평가 대상 네트워크를 구성하는 구성원의 총 개수를 의미하며, w_j는 평가 대상 네트워크에 대한 j번째 노드의 중요도를 나타낸다.

3.3 시뮬레이션 기반 취약성 분석 방법

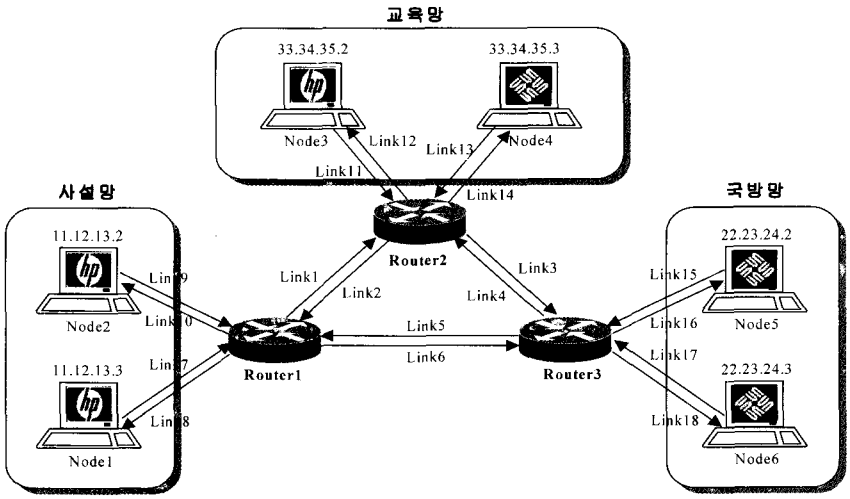
(그림 2)는 SES/MB 프레임워크를 기반으로 한 취약성 분석 방법을 나타낸다. 그림에서 1단계는 개념 명세화 단계로서, 취약성 분석의 목적, 요구사항 및 제약사항등을 고려하여 정보통신 기반 환경에 대한 전반적인 구조를 도식화한다. 이를 위하여 구조적 지식의 표현수단을 제공하는 SES를 이용한다. 2단계에서는 정보통신 기반 환경에 대한 SES에 가지치기(pruning)을 통하여 분석 대상에 대한 PES를 생성한다. 또한, PES의 최하위 노드들에 대한 동적 모델을 생성하여 모델베이스에 저장한다. 이를 위하여 DEVS형식론을 적용하여 연속적인 시간상에서 이산적으로 발생하는 보안 이벤트를 효과적으로 모델링할 수 있다. 3단계는 2단계에서 생성된 PES의 구조적 모델과 MB의 동적 모델을 합성함으로써 시뮬레이션 모델 구조를 생성하는 단계이며 끝으로 4단계에서는 다양한 공격시나리오를 적용하여 시뮬레이션을 수행하고 시뮬레이션 결과로부터 각 노드의 상태변화를 분석함과 더불어 취약성 매트릭스를 적용함으로써 노드 및 네트워크에 대한 취약성을 정량적으로 분석한다[13].

4. 사례연구

본 장에서는 시뮬레이션 기반 정량적 취약성 분석방법을 적용한 예에 대하여 설명한다.



(그림 2) 시뮬레이션 기반 취약성 분석 방법

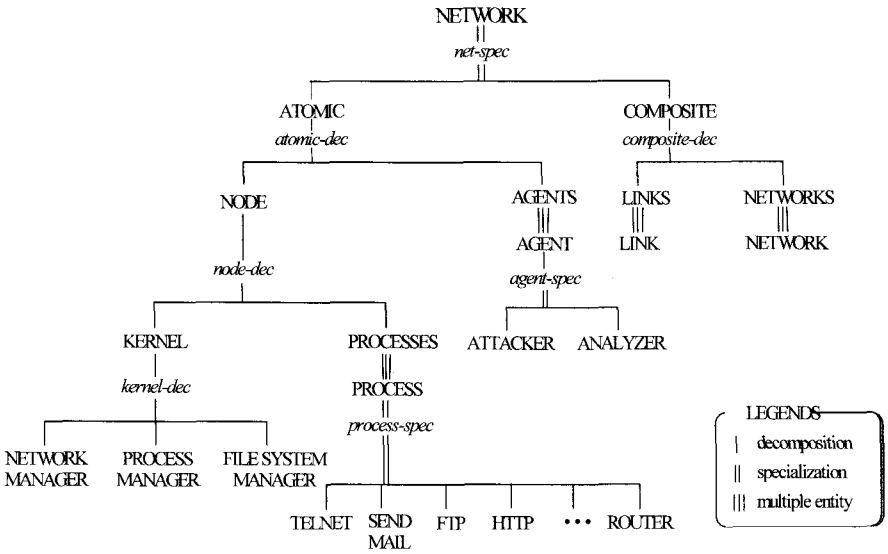


(그림 3) 샘플 네트워크

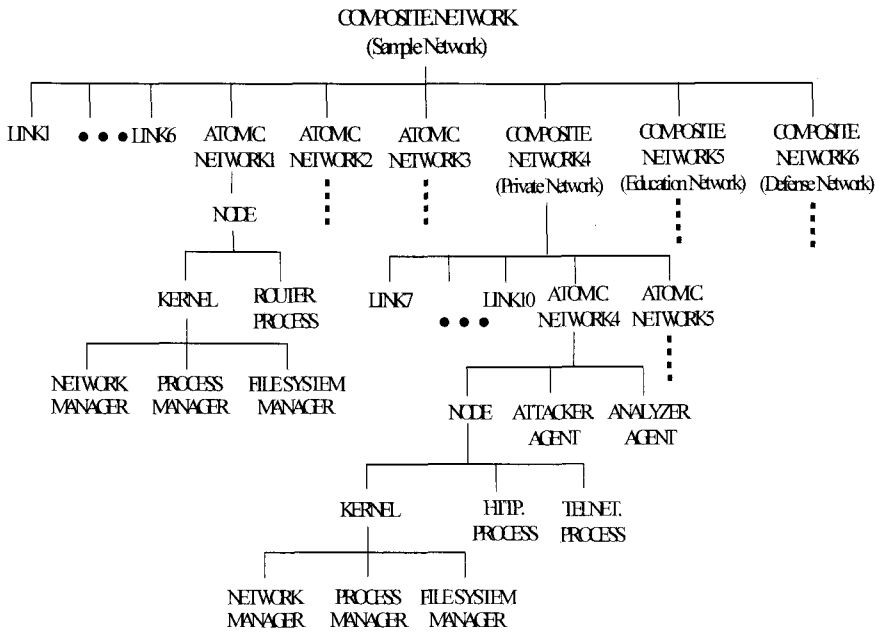
4.1 샘플 네트워크

(그림 3)은 SES/MB프레임워크를 이용한 취약성 분석을 위한 간단한 샘플 네트워크를 나타낸다. 샘플 네트워크는 사설망, 교육망, 국방망으로 구성되어 있으며 각각의 네트워크에는 두 개의 시스템이 존재하며 라우터를 통하여 외부와 연결된다.

(그림 4)는 SES/MB 프레임워크에 의하여 샘플 네트워크를 모델링한 예를 나타낸다. (그림 4(a))는 정보통신 기반 환경에 대한 보안 관점의 SES를 나타낸다. 최상위 엔티티인 NETWORK는 단일의 네트워크로 구성되는 ATOMIC과 여러 개의 네트워크로 구성될 수 있는 COMPOSITE로 분류된다. ATOMIC은 NODE와 AGENTS로 분할되며 NODE는 KERNEL과 여러개의 PROCESS로 나뉜다. 또한, AGENTS는 공격 시나리오에 따라 공격 이벤트를 발생하는 ATTACKER와 시뮬레이션 결과를 분석하는 ANALYZER로 구성된다. COMPOSITE는 다수의 NETWORK와 이들을 연결하기 위한 다수의 LINK로 구성된다. (그림 4(b))는 SES에 가지치기(pruning)를 수행하여 얻어진 샘플 네트워크에 대한 PES로서 라우터를 연결하는 6개의 LINK, 라우터를 의미하는 3개의 ATOMIC.NETWORK, 서브 네트워크를 의미하는 3개의 COMPOSITE.NETWORK로 구성된다. 3개의 ATOMIC.NETWORK는 NETWORK MANAGER, PROCESS MANAGER, FILE SYSTEM MANAGER로 구성된 KERNEL과 ROUTER.PROCESS로 구성된 NODE이며 3개의 COMPOSITE.NETWORK는 다시 4개의 LINK와 2개의 ATOMIC.NETWORK으로 구성된다. ATOMIC.NETWORK는 NODE, ATTACKER.AGENT 및 ANALYZER.AGENT로 구성되며 NODE는 KERNEL과 여러 개의 PROCESS(예: HTTP.PROCESS)로 구성된다. 이와같이 구성된 PES의 각 엔티티는 DEVS 기반의 동적 모델로 변환된다. (그림 4(c))는 노드, 공격자 및 분석자의 기능을 간략화하여 DEVS로 표현한 의사코드이다. 특히, 노드모델은 (그림 4(d))에 표현된 NODE의 서브모델들로 상세화하여 모델링하였다.



(a) 정보통신 기반 환경의 SES



(b) 샘플네트워크에 대한 PES

Node Model

state variables
Service_TYPe, HW_TYPe, OS_TYPe
Registered_User_list, Queue_size, etc.

external transition function
case inPut_Port
.in: case Phase
Passive: execute com_table(command)
hold_in busY Processing_time
busY: continue
else: continue

internal transition function
case Phase
busY: Passive

outPut function
case Phase
busY: send Packet(result) to Port .out

Attacker Model

state variable
scenario_tYPe, target_host

external transition function
case inPut_Port
.in: case Phase
Passive: next command := scenario_table
hold_in active attacking_time
active : continue
else: continue

internal transition function
case Phase
active: Passive

outPut function
case Phase
active: send Packet(command) to Port .out

Analyzer Model

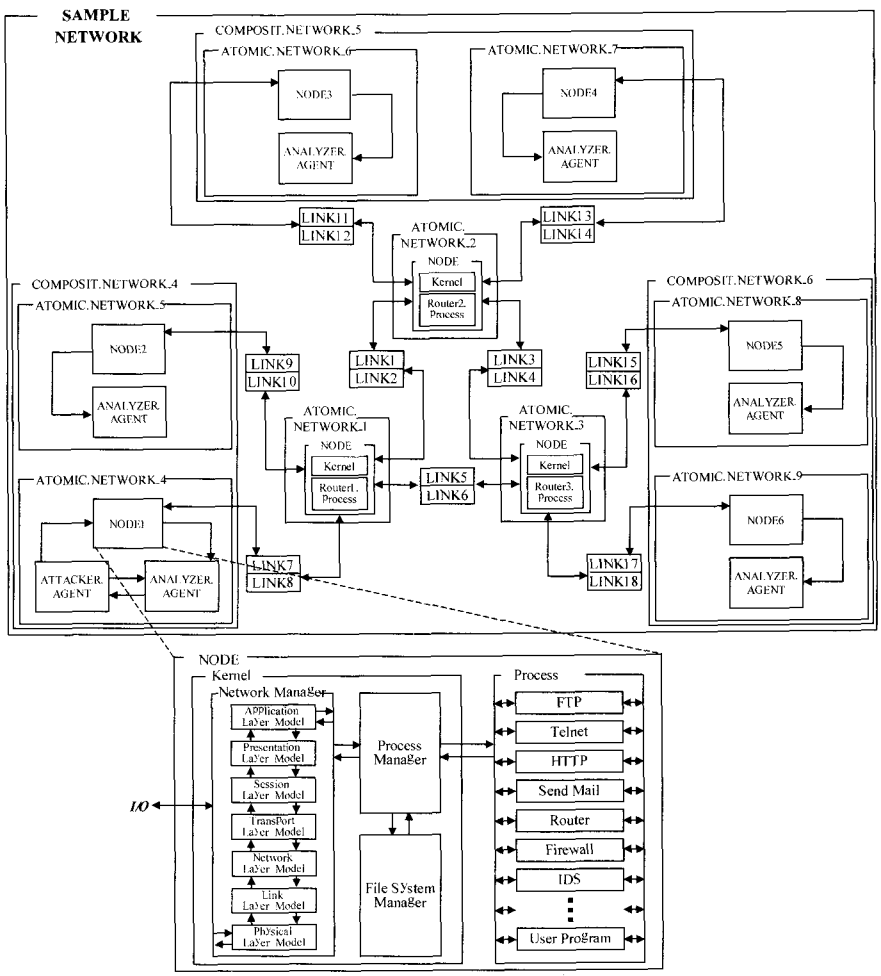
state variable
num_attack, num_success_attack, vulnerability

external transition function
case inPut_Port
.in: store result_table(command, states)
else: continue

internal transition function
case Phase
active: Passive

outPut function
case Phase
active: analYze result

(c) 노드, 공격자, 분석자 모델에 대한 DEVS 표현의 예(Pseudo code)



(d) 시뮬레이션 모델 구조

(그림 4) SES/MB를 이용한 샘플 네트워크 모델링의 예

4.2 시뮬레이션 테스트

(그림 4(d))는 샘플 네트워크에 대한 시뮬레이션 모델을 나타낸다. 시뮬레이션을 위하여 〈표 2〉와 같이 각 노드의 초기 설정과 중요도를 가정하였으며 〈표 1〉의 취약성 항목을 고려하였다.

〈표 2〉 시뮬레이션을 위한 노드의 구성설정 및 중요도

노드 이름	구성 설정	노드 중요도(w_i)
Node1	OS: HP-UNIX, Service: TELNET, HTTP	0.25
Node2	OS: HP-UNIX, Service: TELNET, HTTP	0.25
Node3	OS: HP-UNIX, Service: TELNET, HTTP	0.75
Node4	OS: SunOS, Service: TELNET	0.50
Node5	OS: SunOS, Service: TELNET	0.75
Node6	OS: SunOS, Service: TELNET	0.75

공격자 모델은 일련의 명령어들로 구성된 공격 시나리오(〈표 3〉 참조)를 가지고 있으며 해당 시나리오를 구성하고 있는 명령어를 각각의 노드에 전달한다. 각각의 노드에 전달된 명령어는 노드의 상태에 따라서 KERNEL 모델 및 해당 PROCESS 모델을 통하여 수행 또는 실패가 결정되며 명령어의 수행에 의하여 해당 노드의 상태가 변경됨으로써 동적 취약성이 변하게 된다.

〈표 3〉 공격 시나리오의 예

일련번호	명령어	이용취약성	영향취약성
1	telnet target.host Brute force passwords	$Vul_{Password}$	-
2	http://Node-3/cgi-bin/phf.cgi Brute force passwords	Vul_{Phf} $Vul_{Password}$	-
3	showmount -e target.host mount target.host:/usr/tmp cd /tmp echo abcxyz:1234:10001:1::: >> passwd su abcxyz echo attacker >> .rhosts rlogin attacker	$Vul_{Filesystem}$ $Vul_{Userfile}$	$Vul_{Filesystem}$ $Vul_{Password}$ $Vul_{Userfile}$
...	...	...	...

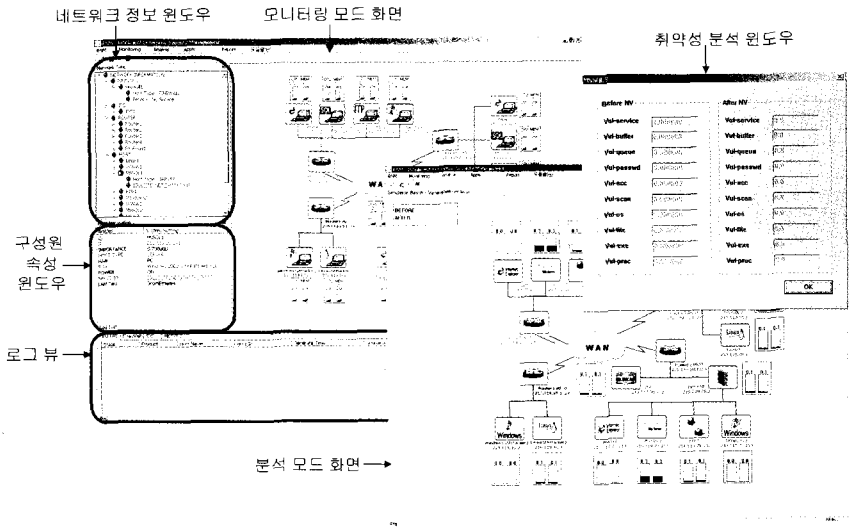
〈표 4〉는 시뮬레이션 결과에 취약성 매트릭스를 적용하여 분석된 각 노드 및 네트워크의 취약성 결과값을 나타낸다. 각 노드의 취약성은 정적 취약성 항목에 의한 취약성 값과 더불어 시뮬레이션을 통하여 분석된 각 노드의 동적 취약성 항목의 취약성 값에 의하여 분석할 수 있다(예: $NV_{Node1}=(0.75\times1.0+1.0\times0.0+1.0\times1.0+0.5\times0.75+0.75\times0.75+0.75\times0.75)/(0.75+1.0+1.0+0.5+0.75+0.75)=0.66$). 또한 각 노드의 취약성 값과 중요도를 토대로 네트워크에 대한 종합적인 취약성을 분석할 수 있다(예: $NetV_{사설망}=(0.25\times0.66+0.25\times0.66)/(0.25+0.25)=0.66$). 이와 같이 모델에 대한 시뮬레이션을 통하여 변화된 상태에 매트릭스를 적용함으로써 각각의 노드 및 네트워크에 대한 취약성을 정량적으로 분석할 수 있다.

〈표 4〉 분석된 노드 및 네트워크 취약성 결과값

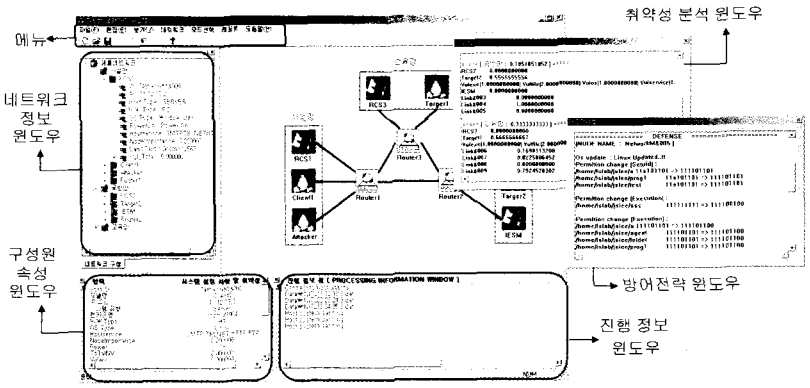
네트워크 이름	네트워크 취약성(NetV)	구성노드 이름	노드 취약성(NV_i)
사설망	0.66	Node1	0.66
		Node2	0.66
교육망	0.53	Node3	0.61
		Node4	0.42
국방망	0.39	Node5	0.39
		Node6	0.39

4.3 응용 연구

(그림 5)와 (그림 6)은 앞서 설명한 방법을 적용하여 개발된 응용 시스템의 예로서 (그림 5)는 시뮬레이션을 이용한 정량적인 취약성 분석 방법을 적용한 취약성 분석 시스템(SIMVA : Simulation-based Vulnerability Analysis system)[15]의 주요 화면을 나타낸다. SIMVA는 네트워크의 상태를 모니터링하고 이를 토대로 취약성을 자동적으로 분석할 수 있다. SIMVA는 시뮬레이션 모델을 생성하여 시뮬레이션을 수행함으로써 대상 네트워크에 대한 정적 취약성 및 동적 취약성을 정량적으로 분석할 수 있다.



(그림 5) SIMVA의 주요화면 스크린샷



(그림 6) IESM의 주요화면 스크린샷

(그림 6)은 모델링 및 시뮬레이션 환경을 기반으로 한 취약성 분석 방법과 지능 시스템 개념을 적용한 지능형 통합 보안 관리 시스템(IESM : Intelligent Enterprised Security Management system)[10]의 주요화면을 나타낸다. IESM은 네트워크 상태를 모니터링하고 취약성을 정량적으로 평가할 수 있으며 취약성에 따른 다양한 대응 전략을 시뮬레이션 평가함으로써 최적의 전략을 수립하여 네트워크에 자동으로 적용할 수 있다.

5. 결 론

전 세계적인 정보화와 인터넷의 보편화로 인한 순기능에 반하여 정보시스템 및 정보통신 기반 환경에 대한 의존도가 증가함으로써 이들의 취약성을 이용한 침해에 따른 피해가 급증하고 있다. 이에 따라 정보시스템 및 정보통신 기반 환경을 능동적으로 보호하기 위한 정보보증에 대한 필요성이 대두되었으며 이를 달성하기 위하여 다양한 연구가 진행되고 있다. 특히,

실제 시스템에 대한 접근에 따른 부작용을 극복하기 위하여 모델링을 통한 시뮬레이션 접근에 대한 필요성이 강조되고 있다. 또한 정보보증을 위한 공격의 피해 및 방어 비용 분석, 효율적인 의사결정 지원등을 위하여 취약성에 대한 정량적인 평가가 요구되고 있다. 본 연구에서는 계층구조적이고 모듈화된 모델링 및 시뮬레이션 환경을 제공하는 SES/MB 프레임워크를 기반으로 취약성을 분석하고 정량화하는 방법을 제안하였다. 또한, 제안한 방법을 적용한 응용연구를 통하여 정보보증분야에 대한 적용 가능성을 살펴 보았다. 시스템 설계 및 분석을 위하여 다양한 분야에서 활용되고 있는 시뮬레이션 기법은 정보보증 분야에서도 필수적인 기술로서 본 연구에서 살펴본 취약성 분석을 비롯하여 정보보증을 위한 다양한 연구에 적용 가능할 것으로 기대된다.

참고문헌

[1] 김여라, 김홍근, "사이버보안을 위한 국가 프레임워크 개발", 정보보호 글로벌 동향, 정책개발 06-06, 한국정보보호진흥원, 2006.

[2] 송관호, "인터넷대란과 대응방안", 한국통신학회지(정보통신), 제21권, 1호, pp.60-69, 2004.

[3] 윤재석, 원순재, "미국 사이버보안 R&D 계획 및 시사점", IITA 주간기술동향, 통권 1253호, 2006.

[4] A. Jones, "The challenge of building survivable information-intensive systems", IEEE Computer, Vol. 33, No. 8, pp. 39-43, 2000.

[5] Aslam, T., A taxonomy of security faults in the UNIX operating system, M.S. thesis, Purdue University, 1995.

[6] C.E. Landwehr, et. al., "A Taxonomy of Computer Program Security Flaws", ACM Computing Surveys, Vol. 26(3), pp.211-254, 1994.

[7] Char Sample, Ian Poynter, "Quantifying vulnerabilities in the networked environment: Methods and Uses", The Internet Security Conference 2000, 2000.

[8] F. Cohen, "Simulating Cyber Attacks, Defenses, and Consequences", Computer & Security, Vol. 18, pp 479-518, 1999.

[9] N. Ye, and J. Giordano, "CACS - A Process Control Approach to Cyber Attack Detection", Communications of the ACM, Vol.44(8), pp.76-82, 2001.

[10] J.S. Lee, et. al., "Design of Intelligent Security Management System using Simulation-based Analysis", AI2005: Advances in Artificial Intelligence, LNAI 3809, pp.766-775, 2005.

[11] J.S. Lee, J.R. Jung and S.D. Chi, "Vulnerability Measures for Network Vulnerability Analysis System", Proc. of 2002 IRC International Conference on Internet Information Retrieval, 2002.

[12] R.A. Martin, "Managing Vulnerabilities in Networked Systems", IEEE Computer, Vol. 34(11), pp.32-38, 2001.

[13] S.D. Chi, J.S. Park, and J.S. Lee, "A Role of DEVS Simulation for Information Assurance", Information Security Applications, LNCS 2908, pp.27-41, 2004.

[14] S. Hariri, et. al., "A Framework for Network Vulnerability Analysis", Communications, Internet and Information Technology 2002, 2002.

[15] Y.J. You, J.S. Lee, and S.D. Chi, "SIMVA : A tool for the Network Vulnerability Analysis", PROCEEDING of International Conference on Internet Information Retrieval 2003, 2003.

[16] B.P. Zeigler, H. Praehofer, and T.G. Kim. Theory of Modeling and Simulation 2ed., Academic Press, 1999.

저자약력

이 장 세

1997년 한국항공대학교 컴퓨터공학과(학사)
1999년 한국항공대학교 컴퓨터공학과(석사)
2003년 한국항공대학교 컴퓨터공학과(박사)
2004년~현재 한국해양대학교 컴퓨터제어전자통신공학부 조교수
관심분야 : 정보보안, 지능시스템, 모델링 및 시뮬레이션
이 메 일 : jslee@hhu.ac.kr

유 용 준

2003년 한국항공대학교 컴퓨터공학과(학사)
2005년 한국항공대학교 컴퓨터공학과(석사)
2005년~현재 한국항공대학교 컴퓨터공학과 박사과정
관심분야 : 모델링 및 시뮬레이션, 네트워크 보안
이 메 일 : ilog21c@kau.ac.kr

지 승 도

1982년 연세대학교 전기공학과(학사)
1984년 연세대학교 전기공학과(석사)
1991년 미국 아리조나대학교 전기전산공학과(박사)
1985년~1986년 두산 컴퓨터(현 한국 디지털) 연구원
1991년~1992년 미국 SIMEX Systems and S/W 회사 S/W담당자
1992년~현재 한국항공대학교 컴퓨터공학과 교수
관심분야 : DEVS 모델링 및 시뮬레이션, 지능시스템 디자인 방법론, 컴퓨터 보안, 교통모델링
이 메 일 : sdchi@kau.ac.kr