가상공격 시뮬레이션을 위한 공격자 및 리눅스 기반 호스트 모델링

정정례, 이장세, 박종서, 지승도
한국항공대학교 컴퓨터공학과
경기도 고양시 화전동 200-1

Attacker and Linux based Host Modeling For Cyber Attack Simulation

JungRae Jung, JangSe Lee, SongSou Park, SungDo Chi
Department of Computer Engineering, Hangkong University

요 약

본 논문은 가상 공격 시뮬레이션을 위한 공격자 및 리눅스 기반 호스트에 대한 모델링 방법의 제안을 주목적으로 한다. 최근, Amoroso는 보안 메커니즘 중심의 침입 모델을 제안하였으나, 시뮬레이션 접근이 분명치 않은 단점이 있다. 또한, Cohen은 원인-결과 모델을 이용하여 사이버 공격과 방어를 표현한 바 있으나, 개념적 단계의 추상화 모델링으로 인해 실제 적용이 어려운 실정이다. 이를 해결하고자 하는 시도로 항공대 지능시스템 연구실에서 SES/MB 프레임워크를 이용한 네트워크 보안 모델링 및 시뮬레이션 방법을 제안한 바 있으나, 공격에 따른 호스트의 복잡한 변화를 표현하기에는 부족하다. 이러한 문제점들을 해결하고자, 본 논문에서는 시스템의 구조를 표현하는 기존 SES에 합성용 규칙기반 전문가 시스템 방법론을 통합한 Rule-Based SES를 적용하여 공격자를 모델링하고, DEVS를 기반으로 하는 네트워크 구성원을 모델링한다. 제안된 모델링 방법의 타당성을 검증하기 위해 본 논문에서는 샘플 네트워크에 대한 사례연구를 수행한다.

I. 서 론

인터넷의 급속한 발전으로 인해 사회 주요 기반 구조 전반에 걸쳐 정보통신 기술이 이용되고 있으며, 이를 이용한 해킹 및 사이버 테러 등과 같은 불법적인 행위가 증가하는 추세에 있다. 이러한 불법적인 행위를 방지하기 위해서는 물리적인 기반 구조를 대상으로 취약 요소 평가, 피해 파급효과 분석, 보안 대책의 적절성 등의 평가를 위한 직접적인 실험을 시행해야 한다. 그러나, 실제의 기반 구조를 대상으로 실험을 시행할 경우 비용, 시간, 피해의 책임문제, 파해배상 등의 많은 문제를 내포함에 따라, 정보보호의 관점에서 시뮬레이션 접근은 정보기반구조에서의 보안 대책 및 위험 요소 분석을 위한 필수 불가결한 요소로 인식되고 있다. 모델링과 시뮬레이션 기법은 시스템 설계 및 분석을 위하여 여러 분야에서 적용되고 있으나, 정보보호분야의 경우 사이버 공격 및 방어의 복잡성, 방대한 탐색 공간, 공격과 방어에 대한 데이터의 부족 등으로 다른 분야에 비하여 연구가 미흡한 실

정이다[1]. 네트워크 보안 모델링 및 시뮬레이션 관련 연구로 Cohen[1], Amoroso[2], Waldlow[3], 그리고 Nong Ye[4] 등의 연구가 수행되었으나, 개념적인 수준의 모델에 그치거나, 실질적인 적용 사례에 대해 검증이 이루어지지 못한 단점이 있다. 최근에 항공대 지능시스템 연구실[5,6]에서는 네트워크 보안 모델링 및 시뮬레이션을 제안하고 각 Component 모델에 대한 명령어 수준의 상세한 모델링의 필요성을 제시하여 주목받은 바 있다. 그러나, 이 연구 또한 네트워크 구성원별로 security 관점에서의 상세한 모델링이 제시되지 않아 실질적 적용성에 문제를 안고 있다. 한편, 최근 CERT Coordination Center에 발표된 기술문서[7]에서는 정보보안과 생존성을 위한 AND/OR Tree를 이용한 공격자 모델링을 제안한 바 있다. 공격자 모델은 AND/OR Tree[7,8]를 토대로 하여 Attack Tree라는 형태로 공격자의 행위를 표현하고 있다. 이러한 Attack Tree는 다양한 공격 패턴을 표현하는 작은 단위의 Tree들의 합성으로 표현될 수 있다고 설명하고 있지만, 실질적인 적용이 가능한지에 대한 검증이 되어있지 않은 상태이다. 또한, 공격 행위에 대한 중복적인 표현이 어렵다는 단점을 가진다. 따라서, 본 논문에서는 기존의 연구들의 단점을 보완하여 실질적인 적용이 가능한 기능적 단계의 상세화 된 모델링 방법을 제안하고자 하며, 이러한 연구 목적을 수행하기 위해 현재 주요 공격 대상이 되고 있는 리눅스 기반의 호스트를 모델링 하고자 한다. 또한, 다양한 공격 시나리오를 생성하는 Rule-based SES를 이용한 지능형 공격자 모델을 제안하고자 한다. 제안된 공격자 및 리눅스 기반 호스트 모델링 방법을 통하여, 첫째 구성원의 상태에 따른 다양한 공격시나리오를 자동으로 생성할 수 있고, 둘째, 공격에 대한 구성원의 구체적인 행위를 시뮬레이션 할 수 있다. 셋째, 시뮬레이션 결과는 보안 취약성 분석 및 대응전략 생성에 적용될 수 있다.

Ⅱ. Rule-based SES 및 DEVS 개요

2.1. Rule-based System Entity Structure (Rule-based SES)

SES(System Entity Structure)는 선언적 특징을 가지며, 구성원들의 분할, 분류, 결합관계, 제약조건 등을 표현할 수 있는 구조체를 말한다[9,10,11]. 그림 1은 간단한 SES를 나타낸 것이다.

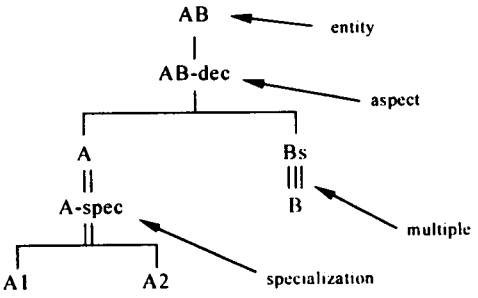

그림 1. 간단한 SES의 예

한편, SES로 표현되는 여러 가능한 구조 중에서 하나의 대상이 되는 구조를 선택하기 위한 Pruning과정을 적용할 수 있는데, Pruning과정은 설계대상 시스템에 필요로 하는 구성 요소들 및 결합관계의 선택 폭을 제한시켜 줄 수 있다[12]. 이를 통해, 구조적 설계문제를 합성문제로 전환시킬 수 있다. 여기서, 합성문제란 충분한 지식을 통해 표현된 모든 구성원들의 집합으로부터 하나의 시스템을 체계적으로 구성하는 것을 뜻한다. 즉, 합성문제에 있어서 우리는 설계 전문가의 지식과 경험으로부터 추출한 일련의 규칙들을 활용하여 자동화함으로써, 설계과정을 줄일 수 있다. Rule-based SES에서는 SES상의 각 Entity들이 선택 및 분할에 관련된 각종 속성 값과 이들을 처리하는 규칙들을 갖는다. 즉, Pruning과정은 요구사항과 제약조건에 상응하는 적절한 Entity를 선택하기 위하여 전문가 시스템을 활용한다. 이러한 방법으로 선택된 Entity들로 구성된 PES는 주어진 요구사항 및 제약 조건을 충족시키는 하나의 설계

구성대안이 될 수 있다[13]. 그림 2는 Rule-based SES의 예를 나타낸 것이다. 여기서 분할노드(예, Attack1-dec)를 가진 Entity(예,Attack1)는 합성에 관련된 규칙들을 가지며, 분류노드(예,Method-spec)를 가진 Entity(예, Cracking)는 종류별 선택에 관련된 규칙들을 가짐으로써, 해당 속성 값의 부여 시 최적의 설계 대안이 제시될 수 있다. 이를 위하여, 각 Entity별 속성 값과 규칙들을 체계적으로 표현하고 있는 Generic frame을 이용한다. Generic frame에 정의되어있는 다양한 규칙과 제약 조건에 따라, 모든 가능한 대안들을 표현하는 SES로부터 하나의 대안인 PES를 얻어낼 수 있다.

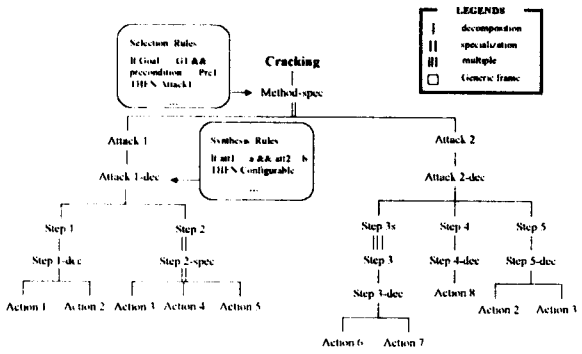

그림 2 Rule-based SES 접근방법의 예

2.2 Discrete Event System Specification (DEVS)

이산 사건 모델링을 위한 대표적인 형식론인 DEVS(Discrete Event System Specification)모델은 연속적인 시간상에서 이산적으로 발생하는 사건들에 대하여 시스템의 행위를 측정하는 것으로 다음과 같은 집합에 의해 표현된다.[9,10]

$$M=< X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta >$$

여기에서

X : 입력 집합

S : 상태 집합

Y : 출력 집합

δ_{int} : $S \rightarrow s$, 내부상태 전이함수

δ_{ext} : $Q \times X \rightarrow S$, 외부상태 전이함수

$Q = \{(s,e) | s \in S, 0 \leq e \leq ta(s)\}$

λ : $S \rightarrow Y$, 출력함수

ta : $S \rightarrow R'_{0,\infty}$, 시간 진행 함수,

단, $R'_{0,\infty}$는 음수를 제외한 실수집합

입력 집합 X는 시스템 외부에서 발생하는 사건들의 집합을 의미하고, 출력 집합 Y는 출력 변수들의 집합을 나타낸다. 상태 집합 S는 상태 변수들의 각 정의 구역들의 곱집합을 의미하며 상태 s는 시간 진행에 따른 시스템의 순차적인 snap shot 상태를 의미한다. 사건 진행 함수 ta(s)는 시스템이 외부 사건을 입력받지 않는 한 상태 s에 머물 수 있도록 허용한 시간으로 정의한다. 내부 상태 전이 함수 δ_{int}는 외부사건이 없는 경우 시간 진행에 따라 모델의 상태변화를 설명해주는 함수로 정의하고, 외부상태 전이 함수 δ_{ext}는 시스템 외부에서 발생한 사건에 의한 모델의 상태변화를 나타내는 함수로 정의한다. 출력 함수 λ는 상태 s에서 시스템의 출력을 정의한다[14].

Ⅲ. 보안 모델링 및 시뮬레이션 환경

네트워크 보안 시뮬레이션 환경구축에 있어서의 가장 큰 문제점은 공격자(Attacker)의 사고 및 인지 능력 등을 표현하기 힘들다는 점과 복잡한 네트워크 구성원에 대한 전문지식을 가지고 있어도 다양한 공격에 대한 일반화된 모델을 제시하기 힘들다는 점이다. 이러한 문제를 해결하기 위한 방법으로, 본 논문에서는 Rule-based SES를 이용하여 다양한 공격 시나리오를 생성해내는 지능형 공격자 모델링과 해킹의 주요 목표가 되고있는 Linux 기반의 호스트 모델링을 제안한다.

3.1. 공격자 모델링

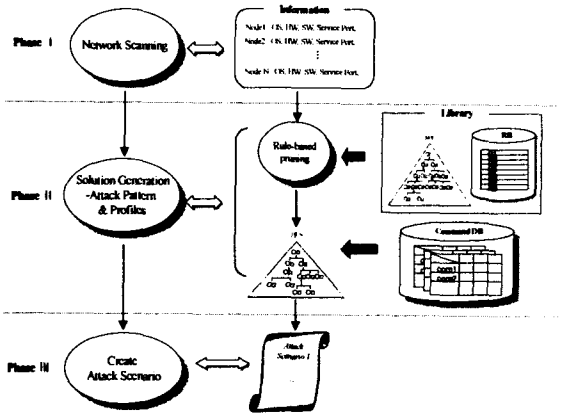

그림 3. 공격자 모델의 시나리오 생성 방법

공격자 모델은 대상 호스트에 공격을 수행하기 위해 공격 시나리오를 사용하게 되는데, 공격 시나리오는 Rule-based SES를 이용하여 생성된다. 그림 3은 공격자 모델이 공격 시나리오를 생성해내는 과정을 나타낸 것이다. 첫 번째 단계에서 네트워크 구성원들로부터 하드웨어 타입, 운영체제 타입, 파일 시스템 등과 같은 시스템 정보들을 얻어온다. 두 번째 단계에서는 불법행위를 표현하는 SES, 제약조건 및 Rule을 저장하고 있는 Library, 그리고 공격 대상 호스트에 대한 다양한 시스템 정보를 이용하여 하나의 공격 패턴 PES를 구한다. 마지막 단계에서는 얻어진 공격 패턴 PES를 바탕으로 하여 공격 명령어, 명령어가 수행되기 위한 선행조건, 그리고 명령어 수행 후의 상태 변화를 나타내는 후행조건 등을 저장하고 있는 Command DB로부터 공격단계에 따른 적절한 명령어들을 선택하여 하나의 공격 시나리오를 구성하게 된다. 그림 4는 Library에 저장되어 있는 불법행위 SES를 나타낸 것이다. 그림 4의 SES를 기반으로 하여 그림 5와 같은 공격패턴 PES를 생성하기 위해서는 Generic frame에 정의되어있는 pruning rule과 제약 조건들을 참조하게 된다. 그림 6은 공격패턴 PES를 생성하기 위한 Generic frame의 예를 나타낸 것이다.

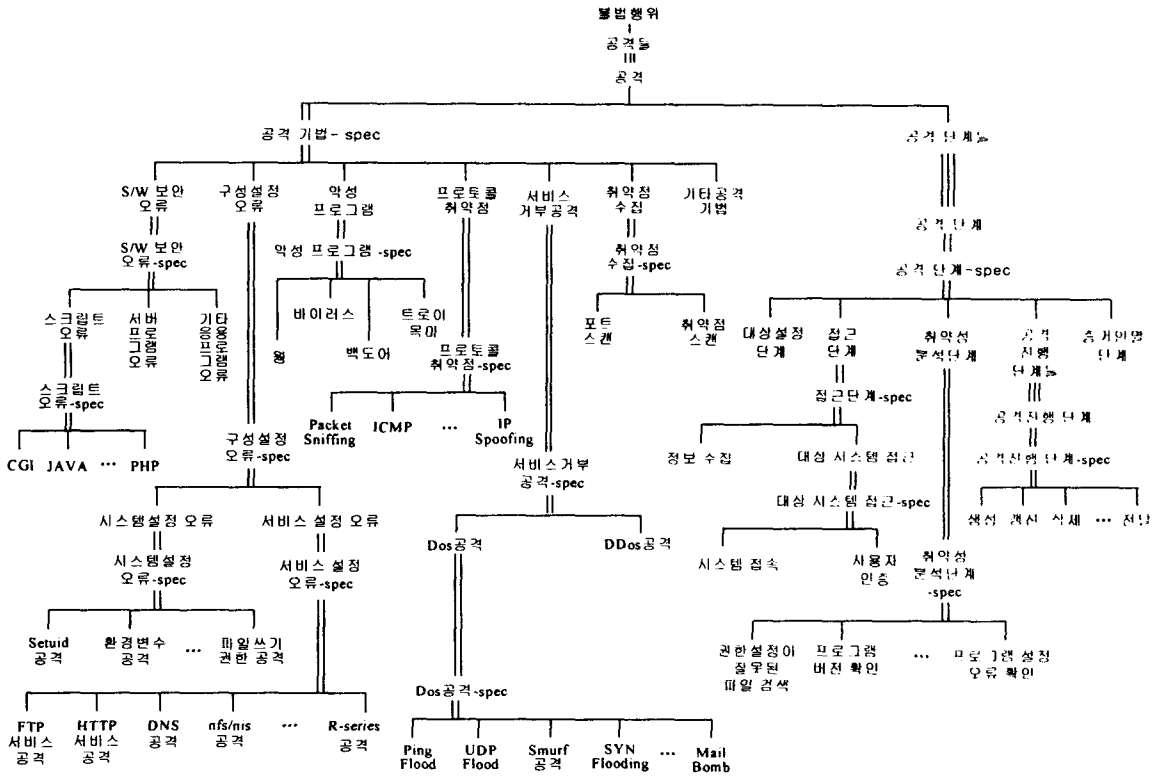

그림 4. 불법행위 SES

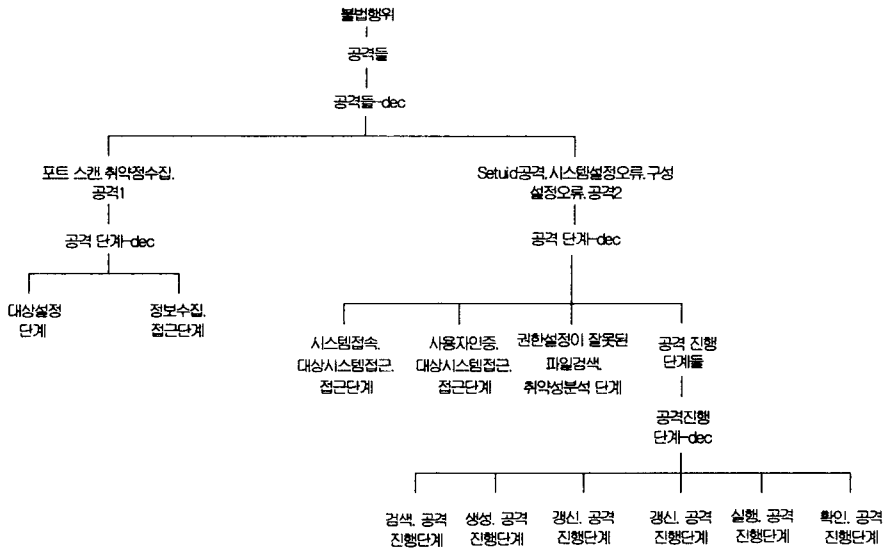

그림 5. PES(Pruned Entity Structure)의 예

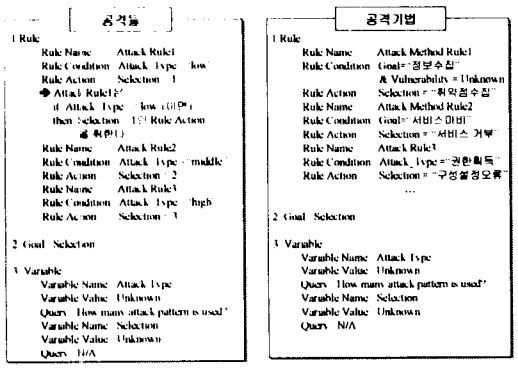

그림 6. Generic Fram의 예

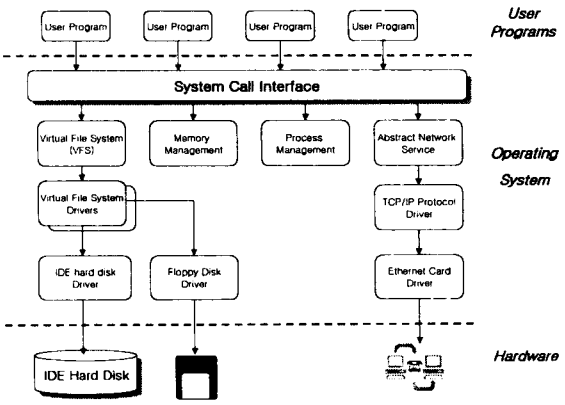

그림 7. Linux OS의 구조도

불법행위 SES와 Generic Frame, 그리고 공격 대상 호스트의 시스템 정보를 이용하여 그림 6과 같은 취약점 수집 공격과 Seuid 공격의 조합으로 이루어진 하나의 공격패턴 PES를 생성하고, 생성된 공격패턴 PES에 적절한 공격 명령을 선택하여 하나의 독립된 공격 시나리오를 생성하게 된다.

3.2. 구성원 모델링

Linux system은 그림 7에서 알 수 있듯이, 크게 User Program, Operating System, Hardware의 세 부분으로 나뉘어진다.

User Program으로부터 System call이 요청되면 Interface를 통해서 각각 해당하는 구성요소로 메시지를 전달하게 된다. Interface를 통해서 전달된 메시지는 Operating system이 갖는 각종 리소스를 이용하여 소프트웨어 또는 하드웨어와의 연결을 제공한다. 예를 들어, 네트워크 연결 서비스를 요청한 경우, Operating system에 있는 Abstract Network Service, TCP/IP Protocol Driver, Ethernet Card Driver를 통해 통신을 위한 초기화 및 LAN 카드 등의 하드웨어와 연결이 이루어짐으로써 네트워크 서비스를 제공하게 된다. 본 논문에

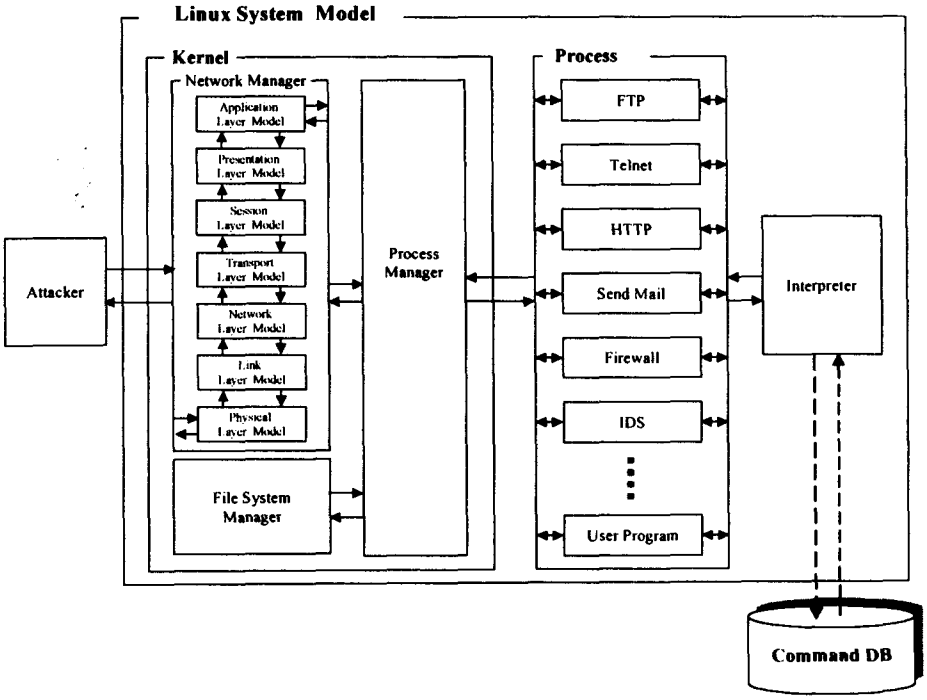

그림 8. 리눅스 기반 호스트 모델

서는 그림 7과 같은 리눅스 운영체제의 구조를 바탕으로, 리눅스 기반 호스트 모델을 제안하며, 그 구조는 그림 8과 같다. 호스트 모델을 구성하는 주요 모델들의 기능 및 속성에 관해 간단히 설명을 하면, Network Manager Model은 OSI 7 layer에 따라 패킷을 생성하여 외부로 전달하는 기능과 외부로부터 받은 패킷에 대한 전달 및 간단한 처리 기능을 수행한다. File System Manager Model은 중요 파일 및 디렉토리에 대한 정보를 가지며, 파일 읽기, 수정, 삭제 등과 같은 작업 요청에 대하여 파일 접근 권한에 따라 작업을 수행하고 처리 결과를 보내는 역할을 한다. 각각의 파일은 파일 이름, 파일 크기, 파일의 유형(정규파일, 디렉토리 등), 파일의 허가권, 소유권, 그룹 ID, 저장 위치, 현재 상태(작업 중, idle) 등의 속성 값을 갖는다. Process Manager Model은 프로세스에 대한 정보를 저장, 관리하며 스케줄링 기능을 제공한다. Process Manager Model은 Process ID(PID), 현재 사용자 ID, 제공되는 서비스명, 프로세스 상태 등과 같은 개별의 process정보를 가지고 있어서 요청이 들어올 때마다 해당하는 프로세스와 적절히 연결 시켜주는 역할을 한다. Process Model들은 서비스 요청에 따라 수행되는 다양한 서비스 모델들로 구성된다. 즉, Process Model들은 원격 접속 서비스를 제공하는 Telnet Model, 단말노드 간의 파일 전송 서비스를 제공하는 FTP Model, 웹 서비스를 제공하는 Http Model, 전자메일 서비스를 제공하는 Send-Mail Model, 다른 네트워크 망이나 대상 호스트 모델로 패킷을 전달해주는 Router Model, 불법적인 패킷을 차단하는 Firewall Model, 외부로부터 불법 침입을 탐지하는 서비스를 제공하는 IDS Model, 그리고 인증 서비스를 제공하는 SSL Model 등으로 구성될 수 있다. 마지막으로, Interpreter Model은 입력된 명령어를 토큰으로 분리하여 해당 명령어를 분석하고 Command DB를 통해 명령어 수행에 필요한 선행조건과 명령어 수행 후 발생되는 시스템의 변화를 표현하는 후행조건 등을 패킷에 실어서 Process 모델에 전달하게 된다.

Ⅳ. 사례연구

본 장에서는 공격자 모델과 리눅스 기반 호스트 모델의 검증을 위하여 그림 9와 같은 가상의 네트워크 망을 대상으로 시뮬레이션을 수행한다.

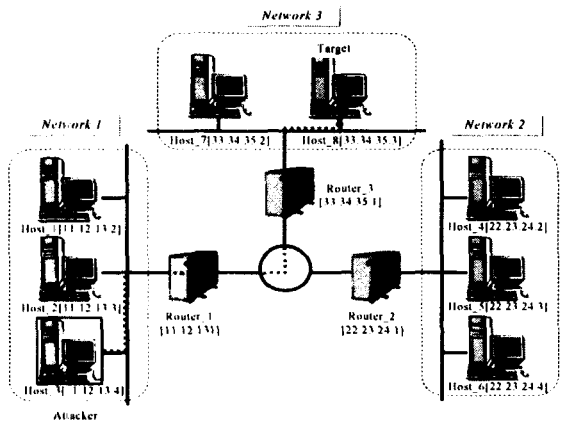

그림 9. Sample Network

공격자 모델은 공격을 수행하기 위해 공격 시나리오를 생성하게 되는데, 이때 공격 대상 호스트에 대한 정보(IP주소, 열린 포트 리스트 등)가 있는지 여부에 따라 공격순서를 정하게 된다. 만약 공격 대상에 대한 정보가 없다면, 다른 공격에 앞서서 대상에 대한 정보 수집공격을 수행한다. 공격 대상에 대한 어느 정도의 정보 수집이 이루어지면, 공격 목표 등과 같은 공격자 모델의 속성 값들의 조합으로 수행하고자 하는 공격의 범위가 좁혀진다. 표 1은 공격 대상 호스트가 제공하는 서비스의 종류, 공격 목적, 공격 취약점, 인증된 사용자 계정의 유무 등과 같은 여러 가지 제약조건과 규칙에 따라 pruning과정을 거치고, Command DB를 통해 얻어오는 명령어의 적용을 통해 얻어진 공격 시나리오를 나타낸 것이다. 생성된 공격 시나리오는 setuid가 설정된 파일을 이용한 공격으로, 공격 대상 시스템에 대한 허가된 사용자 ID와 password를 알고 있다는 사실과 일부 setuid설정이 되어있는 실행파일이 존재한다는 가정 하에 이루어지는 공격이다.

표 1. 생성된 공격 시나리오

Attack	Step	Command	Condition
1. 포트 스캔 -취약성수집	대상설정		
	정보수집	nmap -sS 33.34.35.3	
2. Setuid공격 -시스템 설정오류 공격	시스템 접속	telnet 33.34.35.3	Open_Port_List에 port 23이 있을 경우
	사용자 인증	login dayfly77 1230	대상 시스템에 계정이 존재
	권한설정이 잘못된 파일 검색	find / -perm -4000	
	검색	string /home/dayfly77/prog1	
	생성	cat > ls /bin/sh	Setuid로 인해 사용자 임의의 공격 패턴
	갱신	chmod 755 ls	
	갱신	export PATH=	
	실행	execute /home/dayfly77/prog1	실행권한이 존재
	확인	id	

생성된 공격 시나리오에 대해 시뮬레이션을 수행한 결과, 표 2의 결과를 얻을 수 있었다. 시뮬레이션 결과에 대해 간단히 설명을 하면, 우선 공격자 모델은 대상호스트 모델에 허가된 사용자 ID와 Password를 가지고 telnet 접속을 시도한다. 그런 다음 find명령을 통해서 seuid 설정 파일을 검색한다. 검색된 목록 중에 "ls -al" 이라는 telnet명령을 자동으로 수행하는 prog1이라는 임의의 setuid 파일을 선택한다. 그 다음 단계로 bin/sh을 실행시키는 ls라는 실행파일을 생성하고, ls실행파일에 대한 실행 권한 및 환경변수 PATH의 값을 변경한다. 이러한 과정을 수행한 후, prog1 파일을 수행하면 이전에 수행되던 telnet 명령 "ls-al"이 수행되는 것이 아니라 환경 변수 PATH에 설정되어있는 home/dayfly77/ls를 실행하게 된다. 이때 ls 파일은 bin/sh을 실행하도록 되어있기 때문에 root 권한 shell이 공격자에게 떨어지게 된다. 시뮬레이션 결과에서 알 수 있듯이 공격자 모델에 의해서 생성된 패킷이 대상 호스트 모델에 전달되면, Component Model 내부에서 Process Manager Model, Network Manager Model, 그리고 File system Manager Model 등에 전달되고, 수행되는 명령어에 따라 적절히 모델 상태와 속성을 변화시키게 된다.

표 2. Setuid 공격에 따른 시뮬레이션 결과

Time	Model	What	Remarks
0.0	Attacker (11.12.13.4)	nmap -sS 33.34.35.3	대상 호스트에 대한 열린 포트 검사
4.0	Target (33.34.33.3)	Processing OK!!! (Open_port_List= 80,23)	열린 포트 리스트 전달 (HTTP: 80, Telnet : 23전달)
8.2	Attacker	telnet 33.34.35.3	공격 대상노드에 telnet 접속
8.0	Target[NMM]	telnet 33.34.34.3 **[Physical -> Link]**	MAC Address를 검사
8.1	Target[NMM]	telnet 33.34.35.3 **[Link -> Network]**	IP Address 검사
8.2	Target[NMM]	telnet 33.34.35.3 **[Network -> Transport]**	Opent port List 와 packet.dest_Port와 비교
12.8	Target[PMM]	Processing OK!!! (로그인 요구)	프로세스 생성 **[11.12.13.4 [Telnet | Running | Anybody]**
15.1	Attacker	로그인	대상 호스트에 로그인
21.9	Target : PMM	Processing OK!!!	프로세스 정보 변경 **[11.12.13.4 [Telnet | Running | dayfly77]**
26.2	Attacker	Find / -perm -4000	적절한 setuid 파일 검색
33.0	Target : FMM	Processing OK!!!	Setuid가 설정된 파일의 목록을 전달
37.3	Attacker	string /home/dayflly77/prog1	Setuid가 설정된 파일 prog1을 분석
45.2	Target : FMM	Processing OK!!!	실행파일에 대한 명세 전달 **(ex) prog1 => ls al 수행**
49.5	Attacker	cat > ls /bin/sh	bin/sh을 실행시키는 ls라는 실행 파일 생성
57.4	Target	Processing OK!!!	cur_directory에 ls파일 생성 및 실행파일의 명세 **(ex) ls.content = /bin/sh**
61.5	Attacker	chmod 755 ls	ls의 권한 변경(실행가능)
69.6	Target : PMM	Processing OK!!!	ls파일의 접근 권한을 **rwxr-xr-x로 변경**
73.9	Attacker	export PATH=.	PATH를 설정
81.8	Target	Processing OK!!!	PATH 값 변경 **(ex) PATH = /home/dayfly77**
85.9	Attacker	execute /home/dayfly77/prog1	파일 실행
94.0	Target : PMM	Processing OK!!!	root권한 획득- 프로세스 맵의 정보 수정 **[11.12.13.4 | Telnet | Running | root]**
94.1	Attacker	id	현재 권한 정보 요청
106.2	Targe	Success!!!	권한획득 확인(Uid = 0 gid=508groups=508)

** NMM : Network Manager Model, PMM : Process Manager Model, FMM : File System Manager Model*

V. 결론

대부분의 정보 시스템들은 다양한 운영체제를 사용하고 있으며 특히, 리눅스의 경우, 매커니즘이 유닉스와 유사하고 source가 공개되어 있는 바, 사용량의 급격한 증가와 함께 다양한 취약점이 보고되고 있으며 이를 이용한 해킹사례가 증가추세에 있다. 따라서, 본 논문에서는 리눅스 시스템에 대한 가상공격 시뮬레이션을 위한 보안 관점의 모델링과 다양한 공격 시나리오를 생성시킬 수 있는 지능형 공격자 모델링을 주목적으로 하였다. 아직까지 개념적인 수준을 벗어나지 못하거나 또는 전통적인 통계적 시뮬레이션 기법을 위주로 하는 기존의 보안 시뮬레이션 연구와는 달리, 본 연구에서는 사이버 공격의 세부적인 행위를 재현해 낼 수 있도록 이산 사건 모델링 및 시뮬레이션 방법론을

도입함으로써, 취약성 분석 및 대응방안 검토 등 보다 실질적인 보안 시뮬레이션 응용 연구의 토대가 될 것으로 기대된다. 향후 연구로는 공격자 모델의 정확한 행동 표현을 위하여, Pruning과정에 필요한 다양한 규칙 및 제약조건들에 관한 연구와 Command DB의 확장에 따른 모델의 상세화 관련 연구가 진행되어야 할 것이다. 또한 다양한 운영체제를 기반으로 하는 호스트 모델에 대한 연구를 수행하여, 통합적인 호스트 모델에 관한 연구도 진행되어야 할 것이다.

Acknowledge

본 논문은 과학기술부. 한국과학재단 지정 경기도 지역협력연구센터(RRC)인 한국항공대학교 인터넷 정보검색연구센터의 지원에 의한 것임.

참고 문헌

[1] Fred Cohen, "simulating Cyber Attacks Defenses, and Consequences". 1999 IEEE Symposium on Security and Privacy Special 20th Anniversary Program, The Claremont Resort Berkeley, California, May 9-12. 1999

[2]Amoroso, E., Intrusion Detection, AT&T Laboratory, Intrusion Net Books, January, 1999

[3]Wadlow T. A., The Process of Network Security, Addison-Wesley, 2000.

[4]Nong Ye, Joseph Giordano, CACS - A Process Control Approach to Cyber Attack Detection, Communications of the ACM.

[5]Sung-Do chi, Jong-Seo Park, Ki-Chan Jung and Jang-Se Lee, "Network Security Modeling and Cyber Attack Simulation Methodology", 2001,

[6]"SECUSIM: A Tool for the Cyber-Attack Simulation", Lecture Notes on Computer Science Series, ICICS 2001 Third International Conference on Information and communications Security Xian, China, 13-16 November, 2001

[7]http://www.cert.org, "Attack Modeling for Information Security and Survivability", CMU, 2001

[8]브루스 슈나이어 지음, 채윤기 옮김,"디지털 보안의 비밀과 거짓", 나노미디어, 2001

[9] Zeigler, B.P. Object-oriented Simulation with Hierarchical, Modular Models: Intelligent Agents and Endomorphic systems, Academic Press, 1990

[10] Zeigler, B.P. Multifacetted Modeling and Discrete Event Simulation, Academic Press, 1984

[11] S.D. Chi, Modeling and Simulation for High Autonomy Systems, Ph.D. Dissertation, Dept, of Electrical and Computer Engineering, Univ. of Arizona, 1991

[12] Chi, S.D., Lee, J.S., Lee, J.K and Whang. J.H. "NETE: Campuse Network Design Tool", in Proc. IASTED International Conference, July, 1997.

[13] "정보시스템의 구성 및 성능 분석 자동화 방법론에 관한 연구", 과학재단, 1998.4

[14] 지승도, 박종서, 이장세, 김환국, 정기찬, 정정례, "SES/MB 프레임워크를 이용한 네트워크 보안 모델링 및 시뮬레이션", 한국통신정보보호학회 2001 제11회 제2호 p.15-16