보안 시스템의 성능 향상을 위한 SVDB 개발

이원영, 조대호

경기도 수원시 장안구 천천동 300 성균관대학교 정보통신공학부

Development of SVDB for performance improvement of security

Lee Won Young , Cho Tae Ho
Syungkyunkwan University

요 약

네트워크 보안의 중요성과 필요성이 증대됨에 따라 많은 조직들이 다양한 보안 시스템을 네트워크에 적용하고 있다. 침입 차단 시스템, 침입 탐지 시스템, 취약점 스캐너와 같은 보안 시스템들이 취약성 정보를 공유하게 되면 일관된 통합 보안 환경을 구축할 수 있다. 본 연구진은 통합 보안 시뮬레이션 환경의 구축을 위해 여러 보안 시스템 모델들이 사용할 수 있는 취약성 정보들을 집약시킴으로써 보안 시스템간의 정보 공유를 쉽게 할 수 있는 SVDB (Simulation based Vulnerability Data Base)를 구축하였다. 네트워크의 한 구성요소인 SVDB는 보안 시스템 모델의 구성에 필요한 다양한 정보를 담고 있어 한 호스트나 네트워크가 갖는 취약성을 조기에 발견할 수 있다. 또한 SVDB는 침입 탐지 시스템과 같은 보안 시스템이 존재하는 네트워크를 시뮬레이션 하는데 필요한 보안 정보를 제공한다. 보안 시스템을 위한 시뮬레이션 모델은 DEVS (Discrete EVent system Specification) 방법론을 사용하여 구성하였다. 또한 이렇게 구축된 시뮬레이션 모델들이 SVDB와 연동하기 위한 인터페이스 모듈을 구현하였다. 취약성 스캐너, 침입 탐지 시스템, 침입 차단 시스템이 정보를 공유함으로써 공격에 효과적인 대응하는 것을 시뮬레이션을 통해 보인다.

1. 서론

네트워크를 통한 통신 활발해지면서 정보자산의 가치가 올라가고 그에 따라 보안 문제도 커지고 있다. 많은 조직들이 이러한 보안 문제를 해결하기 위해 침입 차단 시스템, 침입 탐지 시스템(IDS), 취약점 스캐너와 같은 보안 시스템들을 사용하여 내부 조직을 침입으로부터 방어하고 있다. 실제 네트워크 환경에서 보안 시스템을 직접 사용해 성능을 평가하는 것은 많은 비용과 노력을 요구하므로 이를 효과적으로 해결하기 위한 대안으로 시뮬레이션 모델을 통해 보안 시스템 평가하는 방법이 소개되고 있다[1]. 본 연구의 목표는 여러 보안 시스템 모델들이 사용할 수 있는 취약성 정보들을 집약시킴으로써 보안 시스템간의 정보 공유를 쉽게 할 수 있는 SVDB (Simulation based Vulnerability Data Base)를 구축하고, 이를 사용해 공격에 효과적으로 대응하는 보안 시스템 모델을

디자인하는 것이다. 새로운 취약성이 발견되면 SVDB에 추가하여 여러 보안 시스템 모델이 새로운 보안 규칙을 생성할 수 있다. 또한 SVDB는 새로운 보안 시스템 모델을 추가할 경우 보안 시스템 특성에 맞게 구조와 내용을 변경함으로써 새로운 시뮬레이션을 수행 할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 통합 보안 시뮬레이션 환경 구축을 위한 시뮬레이션 방법론과 대상 시스템을 정의한다. 3장에서 취약성 데이터 베이스와 SVDB의 구성을 알아보고 보안 시스템 시뮬레이션을 위해 필요한 SVDB 인터페이스의 구성을 알아본다. 4장에서는 결론과 향후 연구 방향을 제시한다.

2. 네트워크 보안 모델

2.1 DEVS 방법론

Zeigler에 의해 정립된 DEVS 방법론은 연속적인 시간상에서 발생되는 이산 사건을 처리하는 시스템을 시뮬레이션 하기 위해 이론적으로 정립된 모델링 방법론이다[2]. 이는 모델의 구조와 행동을 시뮬레이션 수행으로부터 추상화시키기 위해 모델을 집합 이론적 방법으로 이용한 것으로, 시스템을 계층적(hierarchical)이고 모듈화(modular)된 형식으로 기술한다.

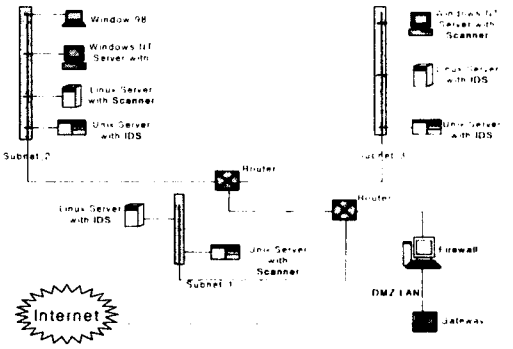
DEVS에서는 기본(Basic) 모델과 결합(Coupled) 모델을 정의한다. 기본 모델은 시스템의 동적인 특성을 표현하기 위한 모델이고, 결합 모델은 시스템의 구성 요소간의 상호작용을 표현하기 위한 모델이다. 이 모델들은 다음의 항들로 명세 할 수 있다.

$M = \langle X, S, Y, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, t_a \rangle$
- X : 입력 사건의 집합
- S : 상태들의 집합
- Y : 출력 사건의 집합
- δ_{int} : 내부 상태 변이 함수
- δ_{ext} : 외부 상태 변이 함수
- λ : 출력 함수
- t_a : 시간 갱신 함수

$DN = \langle D, \{M_i\}, \{I_i\}, \{Z_{i,j}\}, select \rangle$
- D : 구성 요소 이름의 집합
- M_i : 구성 모델
- I_i : 모델 i와 연관된 모델의 집합
- $Z_{i,j}$: 모델 i와 j모델간의 연결 함수
- $select$: $tie-breaking$ $selection$ 함수

2.2 대상 네트워크의 구성

<그림 1>은 3개의 서브넷을 갖는 대상 네트워크의 구성도이다. 대상 네트워크는 침입 차단 시스템(패킷 필터, 프락시), IDS, 취약점 스캐너 등의 시스템으로 구성되어 있다. 시뮬레이션을 위한 보안 시스템 모델은 IDS, 패킷 필터, 프락시, 취약점 스캐너 모델을 구현하였다.



<그림 1> 네트워크 구성도

2.3 보안 시스템 모델

모델링의 대상이 되는 대표적인 보안 시스템의 기능과 특성을 정의하면 다음과 같다.

필터: 패킷의 헤더 및 데이터 정보를 분석, 규칙 테이블을 적용하여 패킷 흐름을 제한한다. OSI 7 계층 모델에서 3, 4 계층에서 처리된다[3,4].

프락시: 동작 방식에 따라 회로 계층과 응용 계층 프락시로 나눌 수 있다. 모델링 대상은 응용 계층 프락시중 HTTP, FTP, SMTP 프락시를 대상으로 한다. IP주소, 포트 등에 의한 접근 통제뿐만 아니라 해당 내용(Contents)에 따른 접근 통제를 지원한다[3,4].

IDS: 오용 침입 탐지 기법(Misuse Detection Technique)을 사용하여 공격의 형태나 침입이라

규정해 놓은 규칙과 일치하는 경우를 탐지한다[5].

취약점 스캐너: 내부의 취약성 정보 리스트와 대상 시스템을 조사하여 얻은 소프트웨어 버전 정보와 설정 정보를 비교하여 취약성을 판단한다[6].

3. SVDB의 구성

3.1 취약성 데이터 베이스

취약성이란 위협 요소에 의해 침해될 수 있는 보안 절차, 기술적 통제, 물리적 통제, 기타 다른 통제들 내의 어떤 조건이나 결점이다[7].

취약성 분석의 목적은 분류의 방법이나 분류의 집합을 마련하는 것이다. 또한 취약성의 집합으로부터 원하는 정보를 추상화하는 것을 가능하게 한다. 이러한 정보들은 침입 탐지 시스템의 시그너처(signature), 공격자가 다른 취약성들을 이용하기 위한 시스템 환경 등으로 이루어진다[8].

각 국의 CERT (Computer Emergency Response Team), 보안 회사의 게시판 및 운영체제와 응용 프로그램의 개발 회사에서는 이러한 취약성들을 분석, 보고하여 취약성들로 인한 피해를 최소화하려는 노력을 한다. ISS (Internet Security Systems, Inc.), SecurityFocus.com에서는 데이터베이스를 운영하고 있으며, NIST (National Institute of Standards and Technology)에서는 산재해있는 취약점 데이터 베이스를 일괄적으로 참조할 수 있는 취약점 metabase를 운영하고 있다[9]. 이러한 취약성 정보들은 같은 취약성이라 할지라도 서로 다른 이름과 가지고 있다. 보안 시스템 모델들이 취약성 정보를 공유하기 위해, 취약성 정보의 유일성을 보장하기 위해 여러 보안 관련 기관들이 참여해 만든 CVE (Common Vulnerabilites and Exposures) 이름을 사용한다[10].

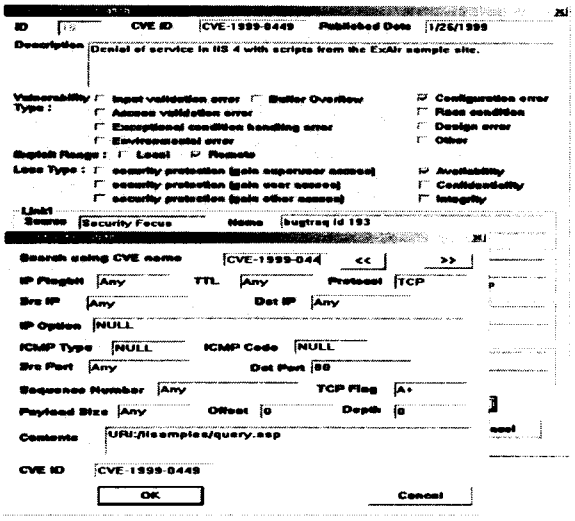
3.2 SVDB의 구성

보안 시스템 시뮬레이션 환경에서는 일반적인 취약성 정보뿐만 아니라 보안 시스템 모델이 사용할 수 있는 보안 툴이 가지고 있는 패킷 수준의 상세한 정보까지 포함해야 된다. 우선 CVE 이름, 취약성에 대한 요약 기술, 공격의 범위, 손실의 유형 등의 일반적인 취약성 정보를 기술한다[11]. 그리고 취약점 스캐너와 같이 내부의 취약성 정보 리스트를 가지고 대상 시스템을 점검하는 도구를 위해 취약한 시스템과 소프트웨어 및 버전을 기술한다[10]. 그리고 침입 차단 시스템과 침입 탐지 시스템이 사용할 수 있는 패킷 정보를 기술한다. 일반적인 패킷내의 정보뿐만 아니라 취약성 중 빈도가 많은 웹은 URL의 내용(Contents)은 따로 기술하고, 전체 패킷 크기, 패킷 내에서의 위치 등 패킷에 대한 규칙을 적용할 때 정확성과 효율성을 높일 수 있는 정보를 기술한다. 보안 시스템 모델의 성능을 평가하는 시뮬레이션 환경을 위해 구성한 SVDB의 내용은 <표 1>과 같다.

<표 1> SVDB의 구성

테이블(Table)	필드(Field)
취약성 정보	CVE 이름, 취약성에 대한 요약 기술, 취약성 공개 날짜, 취약성 유형, 공격의 범위, 손실의 유형, 취약한 소프트웨어 및 버전
패킷 정보	IP 프래그 비트, TTL 값, 프로토콜, 발신지 IP 주소, 목적지 IP 주소, IP 옵션, ICMP 코드, ICMP 타입, 발신지 포트 번호, 목적지 포트 번호, 순서 번호, TCP 프래그, 오프셋, Payload 크기, URL 내용(Contents), 내용, CVE 이름
시스템 정보	소프트웨어 개발 업체, 소프트웨어 이름, 소프트웨어 버전
참고 정보	제공자, 타입, 이름, 링크 주소

<그림 2>는 구현된 SVDB 내용 중 취약성 정보와 패킷 정보를 나타낸다.



<그림 2> 취약성 정보와 패킷 정보

3.3 SVDB와 시뮬레이션 모델의 연동

보안 시스템 모델과 SVDB의 연동에 있어서 보안 시스템 모델에서 데이터 베이스를 사용할 기준을 설정해야 한다. 이 기준으로 보안 정책에 기준을 두고 모델에서 사용할 수 있는 보안 시스템 별로 SVDB의 정보를 활용하게 된다. 보안정책의 수립은 정의된 보안 목적으로부터 특정 시스템에 적용되는 구체적인 보안규칙을 도출하는 것인데, 프로그램 정책, 개별쟁점 정책, 개별 시스템 정책에 대하여 이뤄진다. 보호자원의 비밀성(비인가된 노출 방지), 무결성(비인가된 변조 방지), 가용성(접근권한을 가지는 사용자수 제한) 등의 보안 요구사항을 근간으로, 자원에 대한 적절한 또는 부적당한 보안 행위를 기술한다. 즉 누가, 어떠한 조건하에서 어떤 자원에, 무엇을 할 수 있는가를 정의하는 것이다[1]. 패킷 필터와 응용 프락시에서 사용될 수 있는 세부 정책 사례들과 SVDB내의 관련 필드를 <표 2>와 같이 정리하였다.

<표 2> 보안 정책과 SVDB의 관계

보안 시스템	세부 시행 정책	SVDB 관련 항목
패킷 필터	내부 IP 주소로 위장한 패킷 차단	SrcIP, DstIP
	신뢰된 도메인으로부터 패킷 통과	SrcIP, DstIP
	ICMP echo/direct broadcast 차단	ICMPType, ICMPCode
	TTL 값이 1인 패킷 차단	TTL
HTTP Proxy	유해 사이트 URL, 스크립트 통제	SrcIP, DstPort, Offset, Contents
	URL 및 메시지 키워드 제한	DstPort, URLCodtent, Contents
SMTP Proxy	메시지 송신자, 크기, 키워드 제한	DstPort, Contents
	악성 첨부 파일 차단	DstPort, Contents
FTP Proxy	악성 파일의 다운 로드 감지	DstPort, Contents

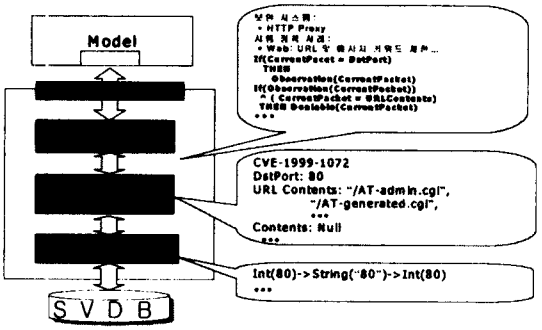
이러한 정책에 따른 보안 시스템 모델 정의를 위해 SVDB와 모델간의 인터페이스 모듈을 구성한다. 인터페이스는 ODBC API, Data Integrator, Rule Generator 세 부분으로 계층적으로 구성한다. 이러한 구성은 데이터 베이스 종속적인 부분과 정책에 종속적인 부분이 나누어져 있으므로 데이터 베이스의 변경이나 정책의 변경 시 다른 부분의 수정 없이 인터페이스를 유지할 수 있다. 하부 네트워크 접속과 데이터 변환과 데이터 베이스의 기본 기능들로 이루어진 ODBC API모듈을 구현한다. 다음 계층에서는 ODBC API모듈을 사용하여 상위 계층에서 원하는 데이터를 집합할 수 있는 Data Integrator모듈을 구현한다. 최상위 모듈은 정책과 보안 모델이 필요로 하는 필드의 매핑과 Data Integrator의 정보를 이용해 규칙 생성을 한다. 각 구성요소의 주요 기능은 다음과 같다.

- Rule Generator: 시행 정책에 따른 규칙을 정의하기 위한 정책과 필요한 필드의 관계 테이블과 Data Integrator의 정보를 이용해 규칙을 생성한다.
- Data Integrator: Rule Generator에서 필요로 하

는 필드의 정보를 DB에서 가져와 통합한다

● ODBC API: DB 접속과 데이터형 변환을 한다.

<그림 3>는 인터페이스 모듈의 구성과 기능을 그림으로 표현한 것이다.



<그림 3> 인터페이스의 구성과 기능

4. 결론 및 향후 연구과제

본 논문에서는 여러 보안 시스템 모델들이 사용할 수 있는 취약성 정보들을 집약시킴으로써 보안 시스템간의 정보 공유를 쉽게 할 수 있는 SVDB를 구축하였다. 또한 보안 시스템 모델이 SVDB와의 연동을 위해 보안 정책을 기준으로 규칙을 생성할 수 있는 인터페이스를 구축하였다.

향후 과제로는 구현된 SVDB와 보안 시스템 모델을 사용한 시뮬레이션을 위한 실제 패킷을 생성할 수 있는 시뮬레이션 입력 생성(Generator) 모델과 시뮬레이션을 통한 개선된 성능을 제공하는 보안 시스템 모델을 디자인하는 것이다. 또한 보안 시스템 모델간 연동 방법의 개발이 이루어져야 할 것이다.

참고문헌

[1] 고종영, 이미라, 김형종, 김홍근, 조대호, "보안 정책을 표현하는 침입차단시스템의 지식기반 모델링 및 시뮬레이션," 시뮬레이션학회 논문지, Vol 10, No 4, 2001.

[2] B. P. Zeigler, H. Praehofer, T. G. Kim, "Theory of Modeling and Simulation," 2nd Ed., Academic Press, 2000.

[3] 김태헌, 이원영, 김형종, 김홍근, 조대호, "네트워크 보안을 위한 침입차단 시스템과 운영체제 보안 기능 모델링 및 시뮬레이션," 시뮬레이션학회 논문지, Vol 11, No 2, 2002.

[4] E. D. Zwicky, "Building Internet Firewalls," 2nd Ed., O'Relly & Associates, 2000.

[5] H.S. Seo and T.H. Cho, "Simulation of Network Security with Collaboration among IDS Models," Lecture Notes on Artificial Intelligence, Springer Verlag, Dec. 2001.

[6] S. Garfinkel, G. Spafford, "Practical UNIX and Internet security, 2nd Ed.," O'Reilly, 1996.

[7] NIST, "An Introduction to Computer Security : The NIST Handbook," Technology Administration, U.S.A, 1995.

[8] M. Bishop, "Vulnerablities Analysis," Proceedings of the Recent Advances in Intrusion Dectection pp. 125-136 Sep. 1999.

[9] 김동현, 송주석, "이용 난이도에 따른 취약점 평가 방법," 한국정보처리학회 추계학술발표논문집, 제8권 제2호, pp947-950, 2001.

[10] Robert A. Martin, "Managing Vulnerabilities in Networked Systems," IEEE Computer, Vol. 34, No.11, pp. 32-38, Nov. 2001.

[11] http://icat.nist.gov, ICAT Metabase