

블랙보드구조를 활용한 보안 모델의 연동

Coordination among the Security Systems using the Blackboard Architecture

서희석, 조대호
(Hee-Suk Seo and Tae-Ho Cho)

Abstract : As the importance and the need for network security are increased, many organizations use the various security systems. They enable to construct the consistent integrated security environment by sharing the network vulnerable information among IDS (Intrusion Detection System), firewall and vulnerable scanner. The multiple IDSEs coordinate by sharing attacker's information for the effective detection of the intrusion is the effective method for improving the intrusion detection performance. The system which uses BBA (Blackboard Architecture) for the information sharing can be easily expanded by adding new agents and increasing the number of BB (Blackboard) levels. Moreover the subdivided levels of blackboard enhance the sensitivity of the intrusion detection. For the simulation, security models are constructed based on the DEVS (Discrete EVent system Specification) formalism. The intrusion detection agent uses the ES (Expert System). The intrusion detection system detects the intrusions using the blackboard and the firewall responses to these detection information.

Keywords : IDS, firewall, network security, coordination, DEVS formalism, BBA, simulation

I. 서론

급격한 정보화 추세와 함께 사회 전 분야에 걸쳐서 가속화 되는 컴퓨터와 정보통신 기술의 발달은 전자 메일, 파일 전송 등과 같은 기본적인 서비스뿐만 아니라 분산 환경을 바탕으로 하는 멀티미디어, 전자 결제, 전자 상거래 등과 같이 복잡한 네트워크 서비스들로 확장되고 있다. 이와 같은 발전은 전송 속도의 고속화, 대용량의 데이터 전송 등으로 업무 효율을 향상시키고 생활의 질을 높여 주며 국가 경쟁력을 강화시켜주는 긍정적인 효과를 거두고 있는 반면, 인터넷의 개방으로 인한 컴퓨터 바이러스의 확산, 정보 자원에 대한 불법 침입 및 정보 자원 자체의 사용을 불가능하게 하는 서비스 거부 공격은 날로 증대되고 있다[1]. 현재 인터넷의 규모는 날로 커지고 있으며, 다양한 서비스를 제공하고 있다. 이로 인해 각각의 호스트에 대한 보안을 수행하는 것보다 네트워크 전체에 대해 보안 서비스를 제공하는 것이 더 효과적인 방법으로 여겨지고 있다. 네트워크 보안 모델을 사용하게 되면 다양한 호스트들이 제공하는 서비스에 대한 접근 제어를 효과적으로 사용할 수 있는 장점이 존재한다[2].

보안 기술은 어플리케이션 또는 네트워크 부분에서 보안 서비스를 제공하기 위한 접근 방식을 취하고 있다. 네트워크 부분에서 보안을 제공하는 대표적인 기술로는 SSL (Secure Socket Layer) 및 TLS (Transport Layer Security), IPSec 프로토콜 등이 있으며 어플리케이션 부분은 S/MIME, PGP (Pretty Good Security)가 존재한다. 본 논문에서는 네트워크 보호를 위해

많이 사용되는 시스템인 침입 탐지 시스템과 침입 차단 시스템을 사용하여 시뮬레이션을 수행할 것이다[3].

침입 탐지 시스템은 컴퓨터나 네트워크 자원에 대한 악의적인 행위들에 대해 확인하고 이러한 행위에 대해서 응답하는 시스템이다[4]. 침입 탐지 시스템은 능동적으로 네트워크를 보호하는 방법으로 방화벽과 함께 네트워크 보호에 많이 사용되는 시스템이다. 침입 차단 시스템은 방어를 목적으로 하는 장치로서 내부와 외부 네트워크 사이에서 접근 제어 정책을 구현한 시스템이다. 네트워크의 속도가 급속하게 증가하고 발전하는 상황에서 많은 양의 데이터를 처리해야 하는 보안 시스템을 직접 사용하여 성능을 평가하는 것은 효율적이지 못하다[5]. 이러한 문제를 해결하기 위하여 DEVS 방법을 사용하여 시뮬레이션 모델을 구축하였고, 이러한 모델들을 사용하여 네트워크 보안 모델을 구축하였다.

현재 침입은 광범위해지고 복잡하게 되어 하나의 침입 탐지 시스템이 독립적으로 네트워크의 침입을 판단하기 어렵게 되었다. 이를 위해 네트워크에 여러 개의 침입 탐지 에이전트를 배치하였고, 이러한 다수의 에이전트가 서로 정보를 공유하며 침입을 탐지하도록 구성하였다. 침입 탐지 에이전트는 효과적으로 침입을 탐지하기 위하여 전문가 시스템을 내장하고 있으며, 에이전트들 간의 통신은 블랙보드구조(Blackboard Architecture)를 사용하여 서로 통신하도록 하였다. 본 연구진은 블랙보드구조를 사용하여 침입을 탐지하는 것이 효과적인 방법임을 이전의 연구에서 보였는데[6-7]. 이번 연구에서는 블랙보드 레벨의 세분화를 통하여 침입 탐지의 민감도를 높일 수 있음을 보인다. 또한 블랙보드의 세분화를 통하여 다양한 상황에 대한 대처가 용이함을 설명한다. 침입 탐지 시스템의 성능을 알아보기 위해 서비스 거부 공격(Denial of Service) 공격 중 mailbomb 공격과 jolt 공격을 사용

논문접수 : 2002. 9. 16., 채택확정 : 2002. 2. 18.

서희석, 조대호 : 성균관대학교

(histone@ece.skku.ac.kr/taecho@ece.skku.ac.kr)

※ 본 연구는 한국과학재단 목적기초연구 (R05 - 2002 - 000 - 00107 - 0) 지원으로 수행되었음.

하여 시뮬레이션을 수행하였다.

2장에서는 본 연구에서 사용된 이론 및 시스템에 관해 설명할 것이고, 3장에서는 보안 모델링을 위해 구성된 각 모델들에 대해 설명할 것이다. 4장에서는 이러한 모델들 간의 연동에 대해 설명할 것이고, 5장에서는 시뮬레이션 수행 및 결과에 대해 설명한다. 마지막으로 6장에서는 결론에 대하여 설명할 것이다.

II. 배경 이론

이 장에서는 논문의 배경 이론에 대해서 설명한다. 2.1에서는 침입 탐지 시스템의 구성 요소와 기능에 대해서 설명하고, 2.2에서는 침입 차단 시스템에 대해 설명할 것이다. 2.3에서는 분산 인공지능의 한 분야인 블랙보드구조에 대해서 설명하고, 2.4에서는 모델링 및 시뮬레이션 이론인 DEVS 형식론에 대해서 설명한다

1. 침입 탐지 시스템

침입 탐지 시스템은 컴퓨터 네트워크나 시스템이 정해진 보안 정책을 위반하는지를 주시하는 시스템이다[8-12]. 침입 탐지 시스템은 모니터링 모듈, 분석 및 조치 모듈과 관리 모듈로 나뉜다[2].

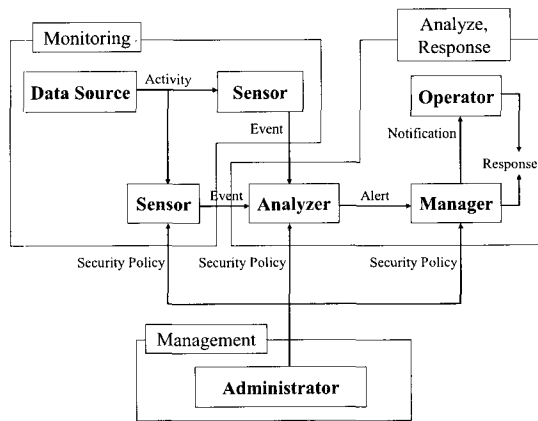


그림 1. 침입 탐지 시스템의 구성 요소.
Fig. 1. Intrusion Detection System Component Organization.

그림 1은 일반적인 침입 탐지 시스템이 갖추고 있는 구성요소들을 나타낸다. 모니터링 모듈은 정보의 수집 단계로서 호스트나 네트워크로부터 데이터를 수집하며, 수집된 데이터는 특정 패턴을 검출하기 위해 다음 단계로 넘겨진다. 분석 및 조치 모듈은 모니터링 모듈에서 넘겨진 정보를 가공 및 축약하고 분석하여 침입을 탐지하는 역할을 수행한다. 즉 네트워크의 많은 데이터 중 침입의 판정에 사용될 수 있는 정보만을 선별하고 이 데이터를 기반으로 침입을 판정하게 되는데, 침입을 탐지하게 되면 이에 대한 대응까지 담당하게 된다. 관리 모듈은 모니터링 모듈과 분석 및 조치 모듈에 대한 통제, 관리 기능과 보안 정책을 제공하는 역할을 담당한다.

2. 방화벽

인터넷 방화벽은 외부 네트워크와 내부 네트워크 혹은 네트워크 간에 설치되어 관리자의 정책에 의해 트래픽의 흐름을 막거나 허용하는데 사용된다[1,13,14]. 즉 방화벽은 외부에

서 내부로 들어오는 트래픽에 대해서 제약을 가할 수 있을 뿐만 아니라 내부 사용자가 외부 네트워크로 접속하여 기밀 정보를 외부로 유출하는 것을 막을 수 있다. 방화벽은 그 동작 계층에 따라 표 1과 같이 분류된다.

표 1. 방화벽의 종류 및 특징.

Table 1. Firewall Types and Characteristics.

방화벽의 종류	특징
패킷 필터링 시스템	<ul style="list-style-type: none"> • 네트워크 계층과 전송 계층에서 동작 • 패킷의 헤더정보에 의해 필터링 수행 • 발신지와 목적지 주소에 의해 데이터의 흐름을 통제 • 처리가 비교적 간단
응용 게이트웨이 방식	<ul style="list-style-type: none"> • 응용 계층에서 동작 • 내부와 외부의 연결을 위해 프락시 서비스 사용 • 상대적으로 강력하고 세밀한 제어 가능

이 중 패킷 필터링 시스템에 대해 자세히 설명한다. 패킷 필터링 시스템은 다시 정적 패킷 필터링과 동적 패킷 필터링으로 분류된다.

정적 패킷 필터링은 관리자가 미리 입력한 필터링 규칙만을 가지고 패킷의 허용 여부가 결정되는 방화벽이다. 네트워크로 유입되는 개개의 패킷 헤더를 보고 허용, 거부가 결정된다. 이 방법은 처리 속도가 빠르고, 응용 서비스에 쉽게 연동되며, 구현이 용이하다는 장점을 갖는다. 반면 데이터의 내용에 관한 세부적인 분석이 불가능하다. 또한 패킷의 헤더는 공격자에 의해 쉽게 조작될 수 있다는 단점과 공격당하게 되면 네트워크에 미치는 영향이 매우 크다는 단점을 갖는다.

동적 패킷 필터링은 입력되는 패킷에 의해 필터링 테이블이 갱신된다. 즉 네트워크로 유입되는 패킷에 의해 동적으로 DST (Dynamic State Table)이 갱신되고, 이렇게 갱신된 테이블을 참조하면서 필터링을 수행하게 된다. 이 방법은 정적 패킷 필터링에 비해 정확하며 상위 프로토콜의 정보도 활용할 수 있다.

3. BBA

분산 인공지능의 한 영역인 블랙보드구조는 분산된 에이전트들이 공동 작업을 통하여 문제를 해결하기 위한 방법을 제공한다[15-17]. 그림 2는 블랙보드구조의 구성요소를 나타낸다.

블랙보드구조의 한 요소인 블랙보드는 문제에 적합한 추상화된 몇 개의 레벨로 분할되어 있다. 특정한 레벨을 통해 통신을 수행하던 에이전트들은 상호작용을 통하여 인접한 레벨로 전이할 수 있다. 이러한 방법을 통해 에이전트들이 수집한 데이터는 한 레벨을 통해 공유되고, 이렇게 공유된 데이터들을 활용하여 목표로 하는 단계로의 전이를 할 수 있다. 일반적으로 목표 레벨은 바로 찾아가기 어려운 작업으로 여러 에이전트들이 서로 조금씩 일을 분담하여 처리하여 그 결과를 블랙보드를 통해 공유하여 최종적으로 목표에 이

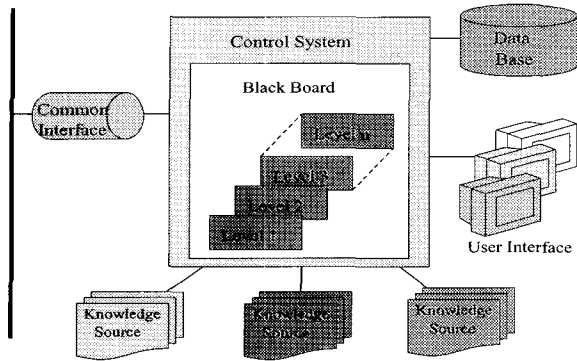


그림 2. 블랙보드구조의 구성 요소.
Fig. 2. Components of the BBA.

르는 방법이다. 블랙보드구조의 단순성으로 인해 분산 인공지능 분야에서 많이 사용되는 개념이다.

4. DEVS formalism

Zeigler에 의해 정립된 DEVS 방법론은 연속적인 시간상에서 발생하는 이산 사건을 처리하는 시스템을 시뮬레이션 하기 위해 이론적으로 정립된 모델링 방법론이다[18][20]. 이는 모델의 구조와 행동을 시뮬레이션 수행으로부터 추상화시키기 위해 모델을 집합 이론적 방법으로 이용한 것으로, 시스템을 계층적(hierarchical)이고 모듈화(modular)된 형식으로 기술한다.

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

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

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

III. 네트워크 구조

이번 장에서는 네트워크 보안 모델에 대해 설명한다[21]. 3.1에서는 네트워크 구조에 대해서 설명하고, 3.2에서는 침입 탐지 모델에 대해 설명하고, 3.3에서는 방화벽에 대해서 설명한다.

1. 네트워크 구조

본 논문에서는 복잡한 네트워크 구조를 계층적으로 표현 하기 위해서 SES (System Entity Structure)를 사용하였다[11]. SES는 시스템의 구조적인 지식을 효과적으로 표현할 수 있는 방법을 제공한다. SES는 분해 (decomposition), 분류 (taxonomy)와 연결 관계 (coupling relationship)가 결합된 지식 표현 방법이다. 각 개체 (entity)와 개체와의 관계는 분해와 세분화 (specialization)의 관계로 표현된다.

그림 3은 시뮬레이션을 위해 구성된 전체 네트워크 모델의 SES를 나타낸다. SES를 사용하여 모델 간의 관계와 계층을 파악하기 용이하도록 하였다. Network Simulation 모델은 크게 Network 모델과 Experimental Frame 모델로 구성된다.

Experimental Frame 모델은 다시 Generator 모델과 Transducer 모델로 구성된다. Generator 모델은 시뮬레이션의 입력으로 사용될 패킷을 생성하는 모델이고, Transducer 모델은 시뮬레이션 수행 후의 통계 자료를 처리하기 위한 모델이다. Network 모델은 다시 BBA, Gateway, Router, Subnet53, Subnet50 과 Subnet49 모델로 구성된다. Gateway 모델은 Route 모델과 Firewall 모델로 구성되고, Firewall 모델은 Outbound_Filter, Inbound_Filter 와 Controller 모델로 구성된다. Outbound_Filter 모델은 다시 Protocol, Address, Port 와 Data 모델로 세분화된다. 에이전트 간의 통신을 담당

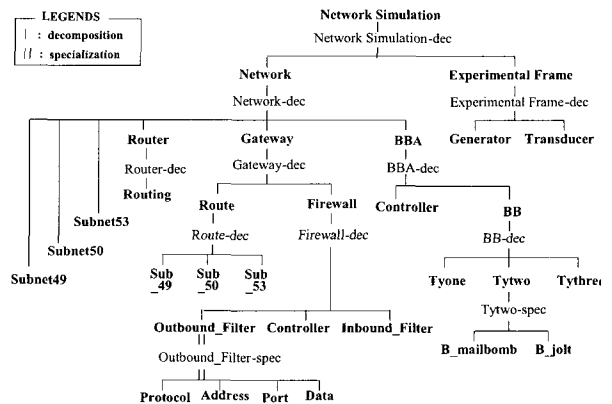


그림 3. 전체 네트워크의 구조.
Fig. 3. Overall Network Architecture.

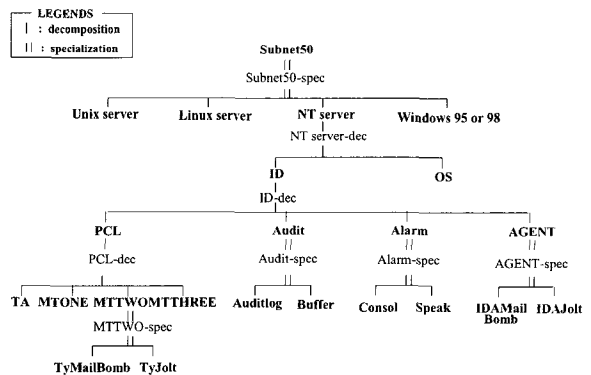


그림 4. 서버넷 50의 구조.
Fig. 4. Structure of Subnet 50.

하는 모델은 Controller 모델과 BB 모델로 구성된다. BB 모델은 다시 Tyone, Tytwo와 Tythree 모델로 구성되고, Tytwo 모델은 다시 B_mailbomb 모델과 B_jolt 모델로 세분화된다.

그림 4는 그림 4의 Subnet50을 구성을 자세히 표현한 것이다. Subnet50 모델은 Unix server, Linux server, NT server와 Windows 95/98 모델로 세분화된다. 각 서버 모델은 OS 모델과 ID 모델로 구성되는데, ID 모델은 다시 PCL (Packet Classify Library), Audit, Alarm과 AGENT 모델로 구성된다. PCL 모델은 TA, MTONE, MITTWO와 MTTTHREE 모델로 구성되고, 이중 MITTWO 모델은 다시 TyMailBomb 모델과 TyJolt 모델로 세분화된다. Audit 모델은 Auditlog 모델과 Buffer 모델로 세분화되고, Alarm 모델은 Consol 모델과 Speak 모델로 세분화된다. AGENT 모델은 IDAMailBomb 모델과 IDAJolt 모델로 세분화된다.

2. ID 모델

그림 5은 각 호스트에 탑재된 침입 탐지 모델의 구성도이다. 침입 탐지 모델은 크게 PCL, Audit, Alarm과 AGENT 모델로 구성된다. 각 모델의 세부 기능은 아래에서 설명한다.

2.1. PCL 모델

PCL 모델은 AGENT 모델에서 사용될 패킷을 분류하고, 필터링하는 역할을 수행하는 모델이다. 침입 탐지 시스템은 많은 양의 데이터를 처리해야 하므로 네트워크에서 수집된 모든 패킷을 검사하는 것은 비효율적이다. 그러므로 침입 탐지에 필요한 정보만을 추출할 필요가 있는데 이러한 역할을 하는 부분이 바로 PCL 모델이다. mailbomb 공격을 예로 들어 PCL 모델의 동작을 설명한다. mailbomb 공격은 메일 서버에 많은 양의 메일을 보내 메일 서버의 동작을 느리게 하거나 전복시키기 위한 DoS 공격의 일종이다. 일반적으로 한 사용자가 다른 사용자에게 전자 메일을 보내기 위해서는 TCP 프로토콜을 사용하고, 25번 포트를 사용한다. 그러므로 PCL 모델의 TyMailBomb 모델은 TCP 프로토콜을 사용하고, 25번 포트를 사용하는 패킷만을 통과시키고, 그 이외의 패킷은 소멸시킨다. 이렇게 함으로써 침입 탐지 시스템의 처리량을 줄이게 된다.

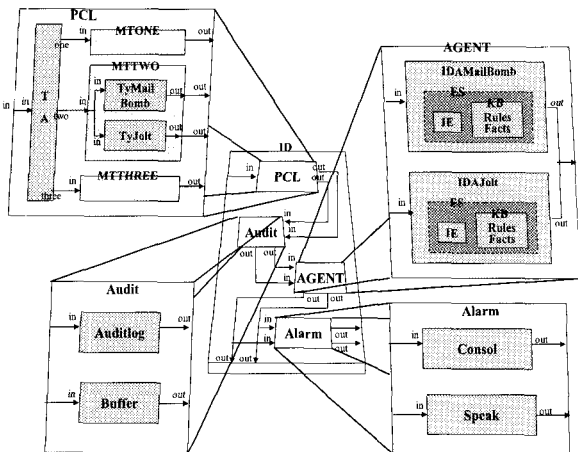


그림 5. 침입 탐지 모델의 구성.
Fig. 5. Structure of Intrusion Detection Model.

2.2. Audit 모델

컴퓨팅 환경이나 네트워크 환경에서와 같이 침입 탐지를 위한 처리 환경에서도 정보의 저장은 매우 중요한 역할을 한다. 정보의 저장은 시스템의 상태값, 공격 과정, 공격의 과거 자료들, 크래커들을 구분하기 위한 증거 자료나 다른 여러 곳에 사용될 중요한 정보원으로 활용된다. Audit 모델은 Auditlog 모델과 Buffer 모델로 구성된다. Auditlog 모델의 역할은 다음과 같다. 침입 탐지 시스템은 종종 자신이 사용한 감사 기록 정보나 네트워크에서 수집한 정보를 보관한다. 감사 정보는 일반적으로 보안상의 중요한 가치를 지니므로 안전한 저장소에 저장할 필요가 있다. Auditlog 모델은 이렇게 침입 탐지 시스템의 log 정보를 기억하는 저장소이다. 다음으로 Buffer 모델에 대해서 설명한다. 일반적으로 침입 탐지 시스템은 많은 양의 데이터를 처리해야하고, 이렇게 많은 양의 데이터 처리를 위해서 저장 공간이 필요하게 된다. 침입 탐지 시스템이 대상 시스템의 처리 용량이나 성능과 맞추기 위해서 하드웨어적이나 소프트웨어적으로 구현된 버퍼 공간이 필요하다. Buffer 모델은 이렇게 대상 시스템의 많은 트래픽을 잃지 않고 저장하면서 사용하기 위해서 구현된 모델이다.

2.3. AGENT 모델

AGENT 모델은 침입 탐지 모델의 핵심 모델로 침입을 판별하기 위해 규칙 기반 전문가 시스템을 내장하도록 하였다. AGENT 모델은 Audit 모델에서 전달받은 패킷을 전문가 시스템에서 사용하는 사실 (fact)의 형태로 전환하고, 이 사실을 전문가 시스템에게 넘겨준다. 전문가 시스템은 자신이 갖고 있는 규칙에 이 사실을 적용하여 침입을 판별하게 된다. 전문가 시스템의 지식 기반 (Knowledge Base)는 침입 탐지에 필요한 다양한 규칙을 가지고 있다. AGENT 모델이 침입을 탐지하게 되면 Alarm 모델에게 이 사실을 알린다.

2.4. Alarm 모델

침입 탐지 시스템이 호스트나 네트워크의 상황을 살펴면서 침입이나 의심스러운 행위 등을 탐지하게 되면 이러한 침입 상황을 알리는 모듈이 있어야 한다. 이러한 모듈은 많은 침입 탐지 시스템이 갖고 있고, 유용한 역할을 담당하게 된다. Alarm 모델의 역할은 단순한 텍스트 형태의 메시지를 화면에 내보내기도 하고, 특정 사용자에게 자동으로 메일을 보내거나 전화 연결을 시도한다. 또 설정된 특별한 곳으로 팩스를 보내게 할 수도 있으며, 원격지나 현재 사용 중인 컴퓨터의 특정한 프로그램을 실행하도록 좀 더 향상된 기능을 제공하기도 한다. 본 연구진은 화면에 경고를 보내는 consol 모델과 일정한 경보음을 내보내는 speak 모델을 구성하였다.

3. Firewall 모델

방화벽도 한계와 결점을 갖고 있지만 방화벽이 네트워크를 연결하고 그 네트워크를 보호할 수 있게 해주는 효과적인 방식이기 때문에 많이 설치하고 있다. 방화벽은 인터넷과 내부 네트워크 사이의 엄격한 접근 제어 수단을 제공하는 방법이다.

그림 6은 시뮬레이션을 위해서 사용될 방화벽 모델의 구성을 보이고 있다. 방화벽 모델은 Controller 모델, Inbound_Filter 모델과 Outbound_Filter 모델로 구성된다.

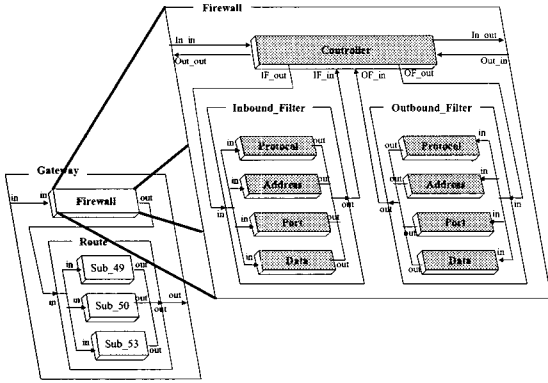


그림 6. 방화벽의 구성.
Fig. 6. Structure of Firewall Model.

Inbound_Filter 모델과 Outbound_Filter 모델은 Protocol 모델, Address 모델, Port 모델과 Data 모델로 구성된다. Controller 모델은 인터넷이나 내부 네트워크로부터 패킷을 받게 되는데 그 패킷을 방향에 의해서 Inbound_Filter 모델이나 Outbound_Filter 모델로 전달하게 된다. 각 Filter 모델은 받은 패킷을 각각 가지고 있는 보안 정책에 의해서 패킷을 처리하여 다음 모델로 전달하게 된다. Controller 모델은 각 Filter 모델로부터 받은 패킷을 보고, 다음 모델로 보내든지 버리게 된다. 예를 들어 mailbomb 공격의 경우 Firewall 모델은 인터넷으로부터 패킷을 받게 된다. 이 패킷을 Controller 모델에게 전달하게 되고, Controller 모델은 이 패킷을 Inbound_Filter 모델에게 전달하게 된다. Inbound_Filter 모델은 IP(Internet Protocol) 주소를 조사하여 자신이 갖고 있는 규칙 테이블 정보에 의하여 패킷을 버릴 것인지 내부 네트워크로 들여보낼 것인지를 판단하게 된다.

IV. 보안 모델 간의 연동

이번 장에서는 보안 모델 간의 연동이 어떻게 이루어지는가에 대해서 설명한다. 우선 침입 탐지 모델 간의 연동에 대해서 설명하고, 다음으로 침입 탐지 모델과 방화벽 모델 간의 연동을 통해서 공격자의 패킷이 내부로 들어오는 것을 차단하는 방법을 소개한다. 그림 7은 대상 네트워크의 블랙 보드 구조를 나타낸다. 침입 탐지 에이전트들은 침입을 탐지하기 위해서 관련된 정보를 블랙보드 상에 게재하거나 열람하기 위해서 블랙보드 구조를 사용한다.

1. 침입 탐지 모델의 연동

블랙보드의 레벨은 Joseph Barrus & Neil C. Rowe가 제안한 Danger value를 사용하였다[21]. Joseph과 Neil이 제안한 Danger value는 다섯 개의 다른 레벨로 나뉘어 진다. 본 연구진이 사용한 블랙보드의 레벨은 이러한 분류에 의하여 호스트 공격에 대한 다섯 가지 레벨과 네트워크 공격에 대한 다섯 가지 레벨로 각각 분류하였다. 이 다섯 가지 레벨은 Minimal, Cautionary, Noticeable, Serious와 Catastrophic이다.

각 에이전트는 두 가지 메시지에 의해서 통신을 수행한다. 하나는 제어 메시지고, 다른 하나는 데이터 메시지이다. 제어 메시지는 에이전트와 제어기 사이의 통신에 필요한 메시

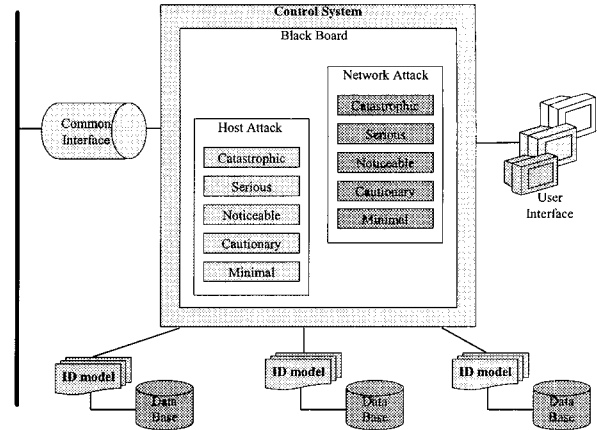


그림 7. 대상 네트워크의 블랙보드구조.
Fig. 7. Structure of blackboard within the target network.

지이고, 데이터 메시지는 에이전트와 블랙보드 간의 데이터 전송에 사용되는 메시지이다.

우선 호스트 공격이 발생한 경우 블랙보드의 상태를 살펴본다. 호스트 공격은 네트워크 상의 호스트 중 하나의 호스트만이 공격을 받고 있는 경우이다. 이러한 경우 공격을 받고 있는 호스트는 블랙보드 상의 Host-Attack에 해당 정보를 게재하게 된다. 블랙보드에 메시지를 게재하기 위해서 해당 에이전트는 BB_update_request 메시지를 제어기에 보낸다. 이러한 방법을 사용하는 이유는 통신상의 무결성과 에이전트 간의 메시지 전송 충돌을 방지하기 위해서이다. 블랙보드에 메시지를 게재할 수 있다면 제어기는 해당 에이전트에게 BB_update_permit 메시지를 전송한다. 이 메시지를 수신한 에이전트는 블랙보드에 침입에 관련된 정보를 게재 (BB_update action)하고 BB_update completion 메시지를 제어기에 보낸다. 제어기는 각 에이전트에게 BB broadcasting of_action request 메시지를 보내고 이 메시지를 수신한 각 에이전트는 블랙보드에서 침입 관련 정보를 열람 (BB_information_retrieval action)한다. 정보를 모두 열람한 에이전트는 제어기에 BB_information_acquisition_completion 메시지를 보내 통신을 마치게 된다. 이러한 과정을 거쳐 공격을 받고 있는 에이전트는 블랙보드 상에서 전이를 하게 된다. 블랙보드의 레벨이 Host-Attack의 Serious 레벨에 이르면 공격 IP (Internet Protocol)에서 에이전트로 전송되는 모든 패킷은 방화벽에 의해서 차단된다.

다음은 네트워크 공격이 발생한 경우 블랙보드의 상태를 살펴본다. 네트워크 공격은 네트워크 상의 여러 호스트들이 공격을 받는 경우이다. 이러한 경우 공격을 받고 있는 호스트는 블랙보드 상의 Network-Attack에 해당 정보를 게재하게 된다. 한 에이전트가 공격을 받게 되면 Host-Attack에서 레벨의 전이를 하게 된다. 이렇게 한 에이전트가 공격 정보를 블랙보드에 게재하고 있는 동안, 다른 에이전트 또한 공격을 받게 된다면 이는 네트워크 공격에 해당한다. Network-Attack의 각 레벨은 다음과 같이 정해졌다. Network-Attack의 Minimal, Cautionary, Noticeable, Serious와 Catastrophic 레벨은 2개 이상의 호스트가 해당 공격을 받는 경우에 해당된다. 예를 들어, Network-Attack의 Cautionary 레벨은 2개 이상의

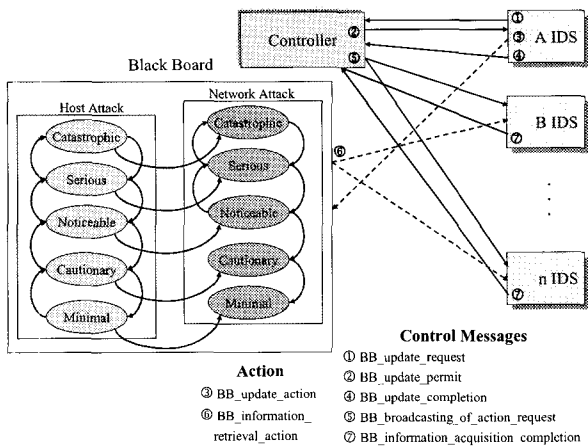


그림 8. 침입 탐지 시스템과 블랙 보드 구조 간의 메시지 교환.
Fig. 8. Message Exchange of IDS and BBA.

Host-Attack 레벨이 Cautionary 이상일 때를 의미한다. 하나의 호스트가 Cautionary 레벨이고, 하나의 호스트가 공격을 받아 Minimal에서 Cautionary로 전이를 하게 되면, 네트워크 전체는 Network-Attack의 Cautionary 레벨이 된다. 네트워크 공격 시 블랙보드 상의 메시지 전송 방법은 기본적으로 호스트 공격에서의 전송 방법과 동일한 방법으로 메시지를 전송한다. 구성된 시뮬레이션 환경에서는 네트워크 공격을 받는 경우 몇 번의 전이를 거쳐 Network-Attack의 Noticeable 레벨이 되면 공격지에서 전송되는 모든 패킷을 차단하여 네트워크가 공격자로부터 보호되도록 하였다. 공격이 지속되어 Network-Attack의 Serious 레벨에 이르면 네트워크로 유입되는 모든 패킷을 차단하여 네트워크 전체를 보호하였다. 이러한 조치를 통하여 관리자는 네트워크 전체나 일정 호스트에 보안 설정을 다시 할 수 있으며 해당 공격을 막을 수 있다. 이와 같이 블랙보드의 레벨을 세분화하여 관리함으로써 각 레벨에 대한 대처를 용이하게 하고, 침입 탐지의 민감도를 높일 수 있다. 그림 8은 mailbomb 공격이 발생한 경우, 블랙보드를 통해 침입 탐지 에이전트들이 서로 통신하는 것을 나타낸다. mailbomb 공격이 여러 호스트에 행해져 Network-Attack의 Cautionary 레벨에서 Noticeable 레벨로 전이하는 과정을 보여 준다.

2. 침입 탐지 시스템과 침입 차단 시스템의 연동

침입 탐지 시스템과 침입 차단 시스템간의 통신에 사용되는 메시지는 역시 두 가지이다. 제어 메시지는 침입 탐지 에이전트와 제어기, 제어기와 침입 차단 시스템이 통신하는데 사용되는 메시지이다. 데이터 메시지는 방화벽과 블랙보드 간의 데이터 전송에 사용되는 메시지이다.

호스트 공격이 발생한 경우 침입 탐지 시스템과 침입 차단 시스템의 연동에 대해 설명한다. 공격을 받고 있는 침입 탐지 시스템에 의해 BB_update_completion 메시지가 제어기에 전해지고 현재 블랙보드의 상태가 Serious 레벨인 경우, 제어기는 침입 차단 시스템에게 BB_unicasting_of_action_request 메시지를 보낸다. 이 메시지를 수신한 침입 차단 시스템은 블랙보드에서 침입 관련 정보를 열람 (BB_information_retrieval action)하게 된다. 정보를 모두 열람한 침입 차단 시

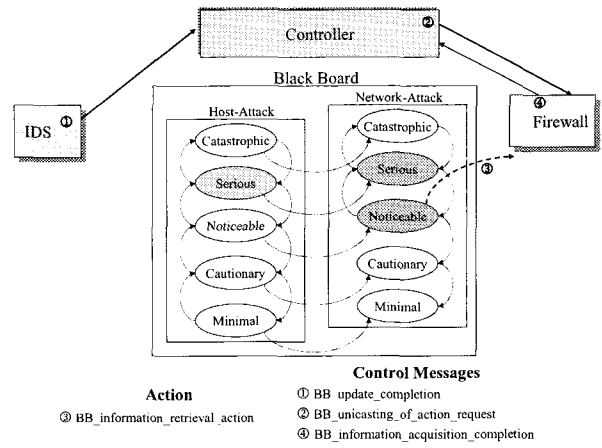


그림 9. 침입 탐지 시스템과 침입 차단 시스템의 메시지 교환.
Fig. 9. Message Exchange of the IDS and Firewall.

스템은 제어기에 BB_information_acquisition_completion 메시지를 전송하여 통신을 마치게 된다. 침입 차단 시스템은 이렇게 열람한 정보를 갖고 자신의 규칙 테이블을 수정하여 공격자의 패킷이 네트워크로 유입되는 것을 막는다.

다음은 네트워크 공격이 발생한 경우 보안 시스템의 연동에 대해 설명한다. 네트워크 공격은 여러 호스트가 공격을 받은 경우이므로 Network-Attack 상에서 레벨의 전이가 발생한다. Network-Attack의 Noticeable 레벨에서는 모든 공격자의 패킷을 차단하므로 네트워크 상에 유입되는 공격 패킷을 좀 더 빠르게 차단할 수 있다. 또한 Network-Attack의 Serious 레벨에서는 네트워크의 모든 패킷을 차단한다. 이러한 조치는 네트워크를 사용할 수 없음을 의미하지만 네트워크 전체에 행해지는 공격을 막기 위한 최후의 수단이 될 것이다. 이상태가 되면 관리자는 신속하게 보안 패치를 설치하고 공격 시스템을 파악하여 신고하는 등의 필요한 조치를 취해야 한다. 그림 9는 블랙보드의 레벨이 Network-Attack의 Noticeable이 된 경우 침입 탐지 시스템과 침입 차단 시스템이 연동되는 메커니즘을 나타낸 것이다.

V. 시뮬레이션 결과

본 논문에서는 두 가지의 경우에 대해서 시뮬레이션을 수행하였다. 첫 번째 경우는 호스트 공격이 발생한 경우 침입 탐지 시스템이 침입을 탐지하는 경우이고, 다른 경우는 네트워크 공격이 발생한 경우 침입을 탐지하는 경우이다. 시뮬레이션을 수행하기 위한 시뮬레이션 환경은 본 연구진이 개발한 DEVS-ObjC를 사용하였다. 내부 시스템을 공격하기 위해서 mailbomb 공격과 jolt 공격을 사용하였고, 이런 공격을 통해 침입 탐지의 성능을 측정하였다. mailbomb 공격은 서비스 거부 공격의 한 형태이다. mailbomb 공격은 많은 양의 메일을 메일 서버에 보냄으로써, 메일서버의 동작을 느리게 하거나 동작을 멈추게 하는 공격이다. mailbomb 공격의 패킷을 생성하기 위해 Kaboom version 3.0을 사용하였다. jolt 공격 역시 서비스 거부 공격의 한 형태이다. jolt 공격은 IP 데이터그램을 작은 조각으로 나누고, 그 나누어진 패킷을 공격 대상 시스템에 전송하는 공격이다. 이러한 패킷을 수신하는 대상

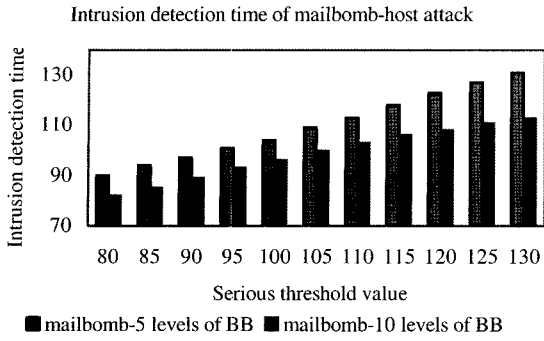


그림 10. mailbomb 공격의 침입 탐지 시간 - 호스트 공격인 경우.
Fig. 10. Intrusion Detection Time of mailbomb attack - a case of host attack.

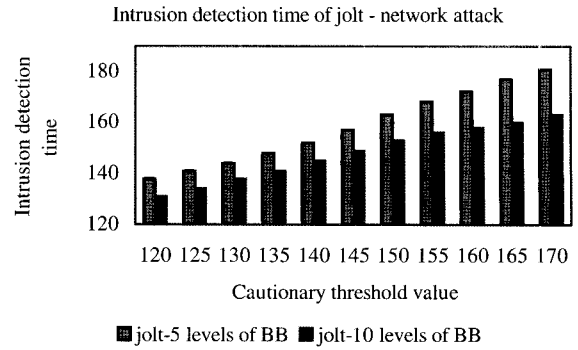


그림 13. jolt 공격의 침입 탐지 시간 - 네트워크 공격인 경우.
Fig. 13. Intrusion Detection Time of jolt attack - a case of network attack.

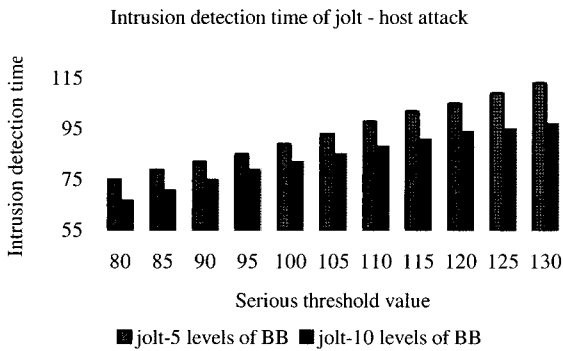


그림 11. jolt 공격의 침입 탐지 시간-호스트공격인 경우.
Fig. 11. Intrusion Detection Time of jolt attack - a case of host attack.

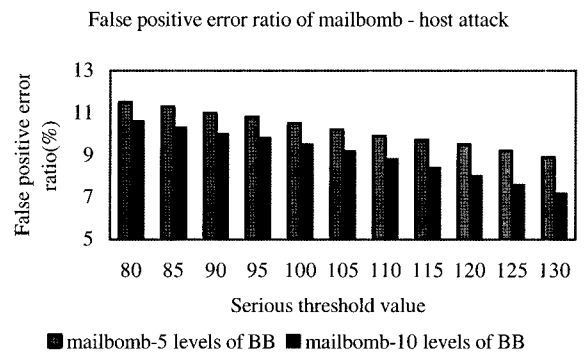


그림 14. mailbomb 공격의 FPER-호스트 공격인 경우.
Fig. 14. FPER of mailbomb attack - a case of host attack.

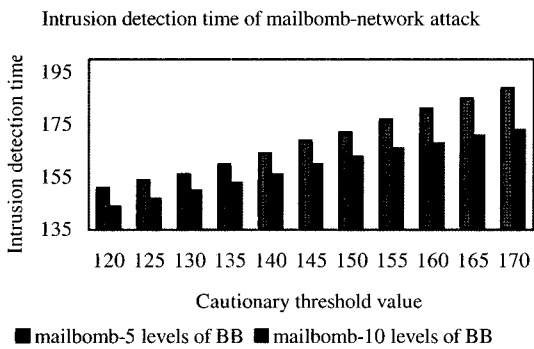


그림 12. mailbomb 공격의 침입 탐지 시간 - 네트워크 공격인 경우.
Fig. 12. Intrusion Detection Time of mailbomb attack - a case of network attack.

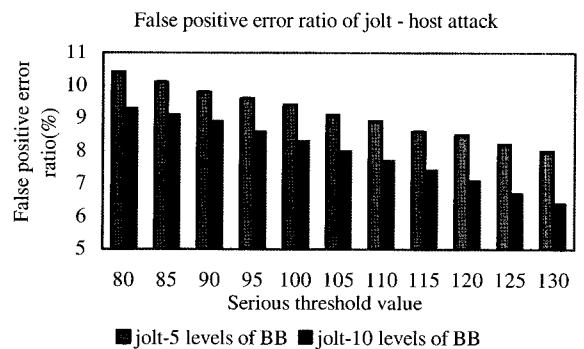


그림 15. jolt 공격의 FPER - 호스트 공격인 경우.
Fig. 15. FPER of jolt attack - a case of host attack.

시스템의 CPU 는 과부하가 걸리게 된다. 즉 작은 패킷을 저장하고 다시 재조립하는데 CPU 의 모든 시간이 소모되어 버린다. 결과적으로 CPU 의 사용량은 거의 100%에 이르게 되고, 다른 작업을 처리할 수 없게 된다.

시뮬레이션을 위한 성능 지표로는 침입 탐지 시간, false positive error ratio 와 false negative error ratio 를 선택하였다. 본 연구진은 이미 이전의 연구에서 하나의 침입 탐지 시스템과 여러 개의 침입 탐지 시스템을 사용한 경우에 여러 개의 침입 탐지 시스템을 활용하는 것이 효과적으로 침입을

탐지함을 보였다[6,7]. 이전의 연구에서는 블랙 보드의 레벨을 5가지 레벨-Minimal, Cautionary, Noticeable, Serious, Catastrophic-로 구분하였다.

그림 10, 11, 14, 15, 18, 19는 공격이 하나의 호스트에 대해서 발생한 경우이고, 그림 12, 13, 16, 17, 20, 21은 공격이 네트워크에 발생한 경우이다. 시뮬레이션은 이전의 연구와 비교함으로써 새로운 시스템의 성능을 보인다.

그림 10-13는 mailbomb 공격과 jolt 공격이 발생한 경우 블랙보드의 레벨이 5개로 분류된 기존의 시스템과 10개로 분

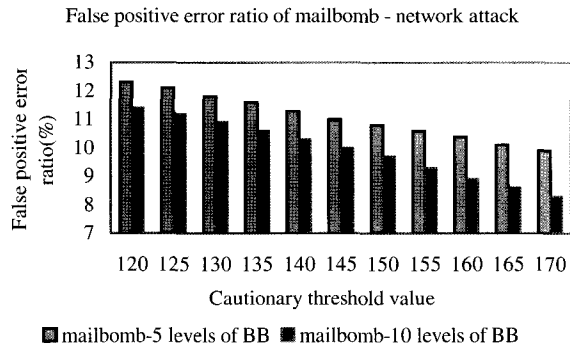


그림 16. mailbomb 공격의 FPER -네트워크 공격인 경우.
Fig. 16. FPER of mailbomb attack - a case of network attack.

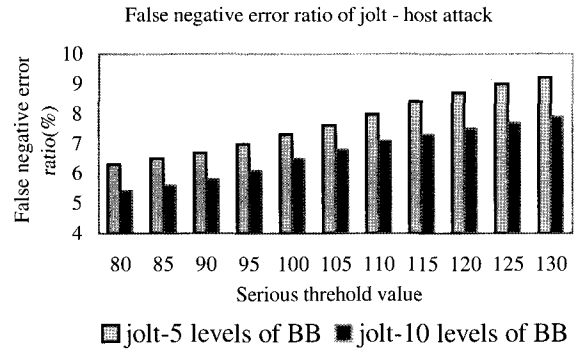


그림 19. jolt 공격의 FNER - 호스트 공격인 경우.
Fig. 19. FNER of jolt attack - a case of host attack.

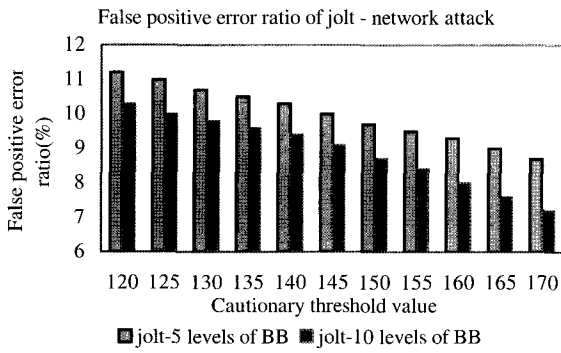


그림 17. jolt 공격의 FPER - 네트워크 공격인 경우.
Fig. 17. FPER of jolt attack - a case of network attack.

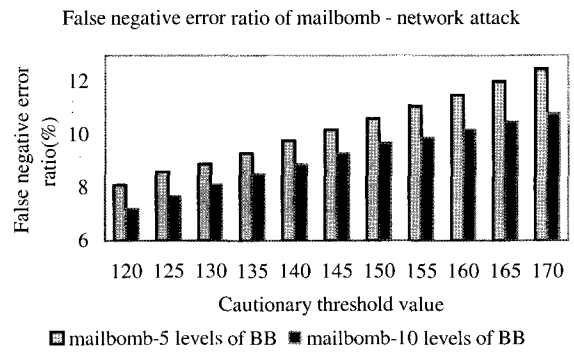


그림 20. mailbomb 공격의 FNER - 네트워크 공격인 경우.
Fig. 20. FNER of mailbomb attack - a case of network attack.

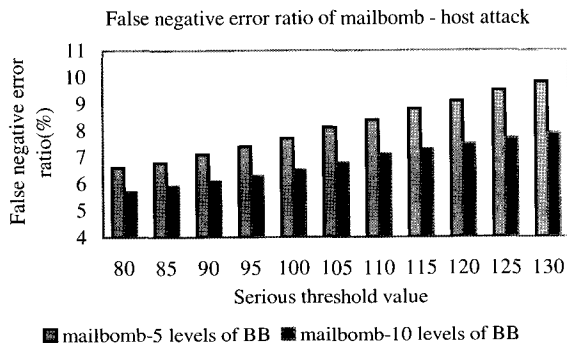


그림 18. mailbomb 공격의 FNER-호스트 공격인 경우.
Fig. 18. FNER of mailbomb attack - a case of host attack.

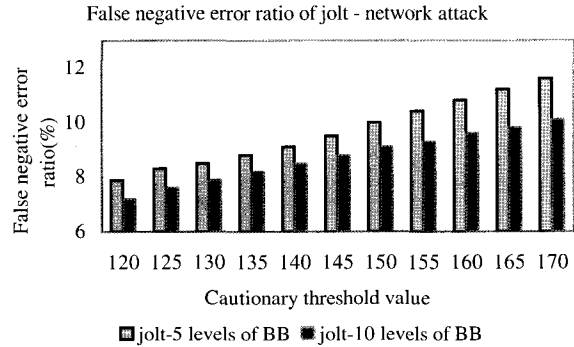


그림 21. jolt 공격의 FNER - 네트워크 공격인 경우.
Fig. 21. FNER of jolt attack - a case of network attack.

류된 제안된 시스템의 침입 탐지 시간을 측정하였다. 시뮬레이션을 위해서 선택된 블랙보드의 레벨은 Host-Attack은 Serious 레벨이고, Network-Attack은 Cautionary이다. 블랙보드 레벨의 임계값 (Serious, Cautionary)이 변함에 따라 다른 레벨의 임계값 역시 같은 비율로 변한다.

두 가지 공격에 대해서 기존의 시스템보다 제안된 시스템이 침입을 더 빠르게 탐지한다. 이러한 결과는 블랙보드의 레벨을 세분화하므로 각 에이전트가 침입의 상황을 좀 더 빠르게 파악했음을 보여준다. 침입을 빠르게 탐지하게 되면 관리자는 침입에 더 빠르게 대응할 수 있다. 네트워크 관리자

가 침입을 신속하게 파악하여 대처를 하게 되면 네트워크를 좀 더 안전하게 유지할 수 있다. 네트워크의 보안 레벨을 낮추게 되면 (본 시스템에서는 블랙보드의 임계값을 높게 되면), 기존의 시스템과 제안된 시스템의 성능의 차이가 더 커지게 된다. 보안 레벨을 낮추면 침입 탐지 시스템이 정보를 공유하는데 있어 민감도가 증가하기 때문이다. 민감도에 관련된 이러한 현상은 모든 다른 시뮬레이션 결과에도 적용이 된다.

그림 14-17은 mailbomb 공격과 jolt 공격이 발생한 경우 5개의 블랙보드 레벨을 갖는 기존의 시스템과 10개의 블랙보

드 레벨을 갖는 제안된 시스템의 false positive error ratio를 나타낸 것이다. 그림 14-17에서는 보안 수준이 강화됨에 따라 false positive error ratio가 증가함을 보인다. 이러한 어려움은 증가는 보안 수준이 증가함에 따라 침입 탐지 시스템이 두 가지 공격에 대해 더 많은 실수를 생성하기 때문이다. 그림 14-17에서 보이듯 10개의 블랙보드 레벨을 갖는 시스템의 어려움이 더 낮음을 알 수 있다. 이러한 결과는 침입 탐지 시스템이 블랙보드의 세분화를 통해 더 민감하게 침입에 반응하였기 때문이다.

그림 18-21은 mailbomb 공격과 jolt 공격이 발생한 경우 기존 시스템과 제안된 시스템의 false negative error ratio를 나타낸 것이다. 아래 그림과 같이 보안 레벨이 강화될수록 false negative error ratio가 감소함을 알 수 있다. 기존의 시스템에 비해 제안된 시스템의 오류가 적은데, 이것은 블랙보드 레벨의 세분화를 통해 에이전트간의 활발한 통신이 이루어졌음을 의미한다.

VI. 결론 및 향후 과제

현재 인터넷상에서 진행되고 있는 공격들은 과거에 비하여 더 심각하고, 기술적으로 더 복잡해 졌다. 따라서 침입을 탐지하기 위해 하나의 침입 탐지 시스템을 사용하는 것은 효과적이지 못하다. 다수의 침입 탐지 시스템을 사용하여 서로 정보를 공유하며 침입을 탐지하는 것이 침입 탐지의 성능을 높이는 좋은 방법이다. 네트워크 상이 침입에 관한 정보를 공유하기 위한 다양한 방법들이 소개되고 있으며, 본 논문에서는 블랙보드 구조를 통한 방법을 소개하였다. 블랙 보드 구조는 그 단순성으로 인하여 내용의 열람 및 개제가 용이한 장점을 갖고 있다. 또한 블랙보드 레벨의 세분화를 통해 에이전트들 간의 정보 교환을 충분히 함으로 침입 탐지의 성능을 높일 수 있다. 침입 탐지 시스템간의 연동뿐 아니라 침입 탐지 시스템과 침입 차단 시스템의 연동은 네트워크를 안전하게 보호하는 방법을 제공한다.

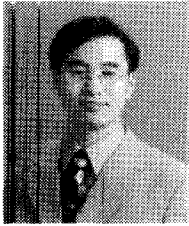
향후 과제로는 네트워크의 공격 패킷을 수집한 것과 같은 종류의 공격 생성기를 추가하여 시뮬레이션에 활용하며 취약점 데이터베이스에 관한 연구를 수행하여 보안 시스템들이 구성된 취약점 데이터베이스를 사용할 수 있도록 할 것이다.

참고문헌

- [1] E. D. Zwicky, S. Cooper and D. B. Chapman, Building Internet Firewalls second edition, O'reilly & Associates, 2000.
- [2] E. Amoroso, Intrusion Detection-An Introduction to Internet Surveillance, Correlation, Traps, Trace Back, and Response, Intrusion.Net Books, 1999.
- [3] S. McClure, J. Scambray and G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw-Hill, 1999.
- [4] R. Bace, Intrusion Detection, Macmillan Technical Publishing, 2000.
- [5] F. Cohen, "Simulating Cyber Attacks, Defences, and Consequences," *Computer & Security*, vol. 8, pp. 479-518, 1999.
- [6] H. S. Seo, and T. H. Cho, "Simulation of Network Security with Collaboration among IDS Models," *Lecture Notes on Artificial Intelligence*, Springer Verlag, LNAI 2256, pp. 438-448, Dec. 2001.
- [7] H. S. Seo, and T. H. Cho, "Modeling and Simulation of Network Security with the Coordination of IDSes and Firewall," *Proceedings of International Conference on Security and Management*, Las Vegas, Nevada, USA, pp. 207-212, Jun. 2002.
- [8] N. Puketza, M. Chung, R. Olsson and B. Mukherjee, "A Software Platform for Testing Intrusion Detection Systems," *IEEE Software*, pp. 43-51, Oct. 1997.
- [9] U. Lindqvist and P. A. Porras, "Detecting Computer and Network Misuse Through the Production-Based Expert System Toolset(P-BEST)," *Proceedings of the IEEE Symposium on Security and Privacy*, Oakland California, May. 1999.
- [10] P. Porras and P. Neumann, "EMERALD: Event Monitoring Enabling Responses to anomalous live disturbances," *Proceedings of the 20th National Information Systems Security Conference*, National Institute of Standards and Technology, 1997.
- [11] M. Crosbie and G. Spafford, "Active Defence of a Computer System using Autonomous Agents," Technical Report No. 95-008, COAST Group, Dept. of Computer Science, Purdue University, Feb. 1995.
- [12] P. Neumann and D. Parker, "A Summary of computer misuse techniques," *In Proceedings of the 12th National Computer Security Conference*, pp. 396-407, Oct. 1989.
- [13] D. Haixin, W. Jianping and L. Xing, "Policy based access control framework for large networks," *Proceedings of IEEE International Conference on ICON 2000*, Sept. 2000.
- [14] N. A. Noureldien and I. M. Osman, "On Firewalls Evaluation Criteria," *Proceeding of TENCON 2000*, pp. 104-110, Sept. 2000.
- [15] G. V. Zeir, J. P. Kruth and J. Detand, "A Conceptual Framework for Interactive and Blackboard Based CAPP," *International Journal of Production Research*, vol. 36(6), pp. 1453-1473, 1998.
- [16] K. Decker, A. Garvey, M. Humphrey and V. R. Lesser, "Control Heuristics for Scheduling in a Parallel Blackboard System," *International Journal of pattern Recognition and Artificial Intelligence*, vol. 7, no. 2, pp. 243-264, 1993.
- [17] F. Klassner, V. R. Lesser and S. H. Nawab, "The IPUS Blackboard Architecture as a Framework for Computational Auditory Scene Analysis," *IJCAI-95 Workshop on Computational Auditory Scene Analysis*, Montreal, Canada, Aug. 1995.
- [18] B. P. Zeigler, Object-Oriented Simulation with Hierarchical, Modular Models, USA:Academic Press, San Diego CA, 1990.
- [19] B. P. Zeigler, Theory of Modeling and Simulation, John Wiley, NY, USA, 1976, reissued by Krieger, Malabar, FL, USA. 1985.

[20] T. H. Cho and B. P. Zeigler, "Simulation of Intelligent Hierarchical Flexible Manufacturing: Batch Job Routing in Operation Overlapping," *IEEE trans. Syst. Man, Cyber. A*, vol. 27, pp. 116-126, Jan. 1997.

[21] J. Barrus and N. C. Rowe, "A Distributed Autonomous-Agent Network-Intrusion Detection and Response System," *Proceedings of Command and Control Research and Technology Symposium*, Monterey CA, pp. 577-586, Jun. 1998.



서희석

2000년 성균관대학교 산업공학과 졸업 (공학사). 2002년 성균관대학교 전기전자 및 컴퓨터공학부 졸업 (공학석사). 2002년~현재 성균관대학교 정보통신공학부 박사과정 재학중. 관심분야는 네트워크 보안 시뮬레이션, 지능형 시스템, 취약성 분석.

템, 취약성 분석.



조대호

1983년 성균관대학교 전자공학과 졸업 (공학사). 1987년 University of Alabama 전자공학과 졸업 (공학석사). 1993년 University of Arizona 전자 및 컴퓨터공학 졸업 (공학박사). 1993년~1995년 경남대학교 전자계산학과 전임강사. 1995년~1999년 성균관대학교 전기전자 및 컴퓨터공학부 조교수. 1999년~현재 성균관대학교 정보통신공학부 부교수. 관심분야는 모델링 및 시뮬레이션, 네트워크 보안, 지능 제어, ERP.