碩士學位論文

이동에이전트 기술을 이용한 네트워크 패킷기반 침입탐지 시스템

濟州大學校 大學院
情報工學科

方 聖 民

2000年 12月

이동에이전트 기술을 이용한 네트워크 패킷기반 침입탐지 시스템

指導敎授 宋 旺 暾

方 聖 民

이 論文을 工學 碩士學位 論文으로 提出함

2000年 12月

方聖民의 工學 碩士學位 論文을 認准함

審査委員長 안 기 강

審 査 委 員 송 왕 철

審 査 委 員 이 상 준

濟州大學校 大學院

2000年 12月

Network Packet based Intrusion Detection System using Mobile Agent Technology

SeongMin Bang

(Supervised by professor WangCheol Song)

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING

Department of Information Engineering
GRADUATE SCHOOL
CHEJU NATIONAL UNIVERSITY

2000. 12.

목 차

SUMMARY

The networked environment has the possibility of the illegal access to the private information and the threat of a network attack. Therefore, security management technologies to protect networks from intrusions and attacks have been given much attention. Intrusion detection system is an important component of the security management system for a large network. It is hard to adapt to the heterogeneous network because of the conventional intrusion detection system depends on the specific system environment.

The new intrusion detection architecture with mobile agent technology is proposed in the paper. The mobile agent is an itinerary software program that autonomously runs on behalf of its operator in a network environment. It helps the adaptability in the intrusion detection system and decrease the network traffic. For the purpose, I analyze the attack patterns and design the new intrusion detection architecture.

The prototype of suggested system is implemented and it operates correctly. The system helps the integration of the security management systems and the adaptability as the new attack patterns.

Ⅰ. 서 론

초기에 특정 그룹을 중심으로 이용되던 네트워크 환경은 최근 정보통신 기술의 발전과 현대의 정보화 지향적인 흐름에 힘입어 대규모의 인터넷 환경으로 변화하고 있다. 이러한 변화는 일반 사용자들의 네트워크 이용을 증가시키고, 네트워크를 통한 업무처리를 더욱 일반화시키고 있으며, 이질화된 네트워크 환경에서 적용이 가능한 효율적인 관리기술을 요구하고 있다(William, 1993. Morris, 1996).

사용자들에게 유용한 서비스를 지속적으로 제공하기 위한 관리기능으로는 초기화 등에 관련된 구성관리 기능, 장비들의 결함 탐지를 위한 장애관리 기능, 네트워크의 지속적인 성능유지를 위한 성능관리 기능, 사용자 관리를 위한 계정관리 기능, 그리고 개별적인 사용자의 정보보호 및 접근제어를 위한 보안관리 기능이 있다(Morris, 1996). 이질적인 네트워크 환경에서 네트워크 장비들간의 효율적인 관리정보 교환을 위해서 이러한 네트워크 관리기능들은 표준화된 관리기술로서 통합될 필요가 있다.

표준화된 네트워크 관리프로토콜로는 IETF(Internet Engineering Task Force)의 SNMP(Simple Network Management Protocol)와 ISO(International Standard Organization)의 CMIP / CMIS(Common Management Information Protocol / Common Management Information Service)등이 제안되었으며, 이러한 표준화된 관리 프로토콜을 이용하여 효율적인 관리 기능을 수행할 수 있는 토대를 마련하게 되었다(Morris, 1996. William, 1993. William, 1993).

인터넷의 성장에 따른 문제점들 중에서 주목할 만한 것으로 개인정보

에 대한 불법적인 접근과 네트워크를 통한 공격을 들 수 있으며, 이에 따라 보안 관리 기술의 중요성이 더욱 높아지고 있다. 보안 관리 기술에 대한 연구는 초기 네트워크 환경의 이용과 함께 이미 시작되었다. 초창기에는 주로 시스템 및 사용자의 정보를 보호하기 위한 접근제어, 인증체계 및 호스트내부의 보안체계 등에서 많은 연구가 이루어 졌으나 이러한 보안 관리 기술들은 시스템에 의존적인 측면이 강하고 빠른 속도로 변화하는 네트워크 공격 기술에 대응하기에는 어려움이 있다. 효율적인 네트워크의 보호를 위해서는 네트워크를 경유한 공격에 대한 빠른 탐지와 대응이 중요하다. 그러므로, 공격수행의 탐지 및 적절한 대응을 할 수 있는 침입탐지 시스템은 보안관리 시스템을 구축하는데 중요한 역할을 담당하게 된다. 새로운 네트워크 공격기법에 대한 적응력과 자동성은 침입탐지 시스템에서 중요한 성능척도가 될 수 있으며, 이를 효과적으로 지원할 수 있는 관리기술의 도입을 통한 통합 보안관리 시스템의 구축이 필요하다.

인공지능 분야에서 연구가 시작된 이동에이전트 기술은 에이전트 기술에 이동성을 부여한 것으로 이질화된 네트워크 환경에서 관리기능의 이동을 가능하게 하며 자율적인 작업이 가능하게 한다. 이동 에이전트 기술은 기존의 관리체계에서 네트워크의 효율을 떨어뜨리는 원인이 되었던 관리데이터의 전송을 최소로 하고, 처리된 결과만을 전송한다 (Bieszczad 등 1998. Aridor 등 1998). 이러한 기능은 대규모의 네트워크에 대해서 효율적인 관리를 가능하게 하며, 이를 침입탐지 시스템에 적용할 경우 대규모의 네트워크에 대해 대역폭의 낭비를 줄이면서 효율적인 보안관리가 가능하다. 더불어 새로운 공격기법의 발견에 대한 탐지규칙에 대한 관리를 중앙의 관리시스템이 수행하므로 써, 이를 적용한 탐지시스템의 관리면 에서 유연성을 부여할 수 있다.

따라서, 본 논문에서는 최근에 네트워크 관리기술로 많은 연구가 이루어지고 있는 이동에이전트 기술을 적용한 새로운 침입탐지 시스템 구

조를 제안하였다. 이를 위해서 본 논문의 Ⅱ장에서는 네트워크 보안관리 및 침입탐지 시스템의 개념, 그리고 이동에이전트 기술에 대해서 살펴보고, Ⅲ장에서는 네트워크를 통한 공격의 일반적인 분석을 기본 바탕으로 시스템 설계를 하며, Ⅳ장에서는 제안된 설계구조의 프로토타입 시스템을 구현하고, Ⅴ장에서 결론을 맺는다.

Ⅱ. 네트워크 보안관리

최근에 네트워크 사용자 및 네트워크를 통한 서비스의 증가와 함께 네트워크를 경유한 공격에 의한 피해 역시 증가하고 있다. 이에 따라 보안관리에 대한 중요성이 더욱 강조되고 있으며, 새로운 공격기법의 등장에 대한 유연성을 부여하고, 이질적인 인터넷환경에서 동작이 가능한 통합 보안관리 시스템을 필요로 하고 있다(Kim 등 1999).

1. 보안관리의 개념

보안관리란 "자산의 본래 가치를 손상되지 않도록 적절한 방법으로 보호하는 제반 행위"로 정의되며, 네트워크 정보들에 대한 접근제어, 네트워크 상에서 사용자 또는 호스트에 대한 인증, 사용자의 감시, 시스템 침입탐지 및 보고 등 네트워크 환경에서의 중요한 정보 및 시스템 보호를 목적으로 수행되는 활동을 말한다(Hughes, 1995). 네트워크상의 보안 위협과 공격유형은 매우 다양하나, ITU-T와 ISO에서 일반적으로 언급되는 공격기법을 살펴보면 다음과 같다.

- 위장(Masquerade) : 불법 사용자가 적법한 사용자로 위장하여 공격하는 형태로 가장 일반적인 방법은 다른 사용자의 인증에 관한 정보를 도용하는 것이다.
- 재사용(Replay) : 적법하게 사용된 패킷을 불법적인 공격을 위하

여 재사용 하는 형태이며 Timestamp를 이용한 대응이 가능하다.

- 가로채기 및 불법수정(Intercept/Modification) : 네트워크 상의 패킷을 변형, 삭제하여 거짓정보를 전달하는 형태로 브로드캐스팅 방식이나 멀티캐스팅 방식을 이용하는 LAN 환경에서 방지하기 어렵다.
- 서비스거부(Denial of Service) : 적법한 서비스를 방해, 중단시키는 형태로 공격자가 시스템을 독점하거나, 공격당하는 시스템의 적법성을 도용하기 위해서 사용되며, 일반적으로 사전공격이 수행된 후 후속공격으로 수행된다.
- 경로수정(Bypassing controls) : 공격자가 시스템의 약점이나 취약점을 이용하여 공격하는 형태로서 전자메일, HTTP, UNIX 시스템들의 취약점을 주로 이용한다.
- 거절(Repudiation) : 정당한 서비스를 거부하도록 만드는 공격의 형태로 매우 탐지하기 어려우며 각 시스템 상황에 맞는 대응방법이 개발되어야 한다.

위에서 언급했던 기법이외에도 현재 알려져 있는 공격방식은 매우 다양하며 빠르게 새로운 공격기법이 출현하고 있다(Siyank, 1995). 초창기의 공격유형은 특정 네트워크 또는 시스템을 목표로 하는 형태이었으나 최근 들어 인터넷의 대중화 및 관련 서비스의 증가와 더불어 전통적인 공격기법과는 다른 메일바이러스, 악성에이전트와 같은 새로운 유형의 공격기법들이 등장하고 있으며, 이러한 공격기법들은 불특정 다수를 대상으로 한 지능화, 자동화의 특성을 보이고 있다(CERT-kr). 이러한 경향에 따라서 최근에 개발되고 있는 보안관리시스템인 경우 개인용 PC를 대상으로 한 보안관리에도 많은 연구가 이루어지고 있다. 현재 알려지고 있는 공격기법들은 매우 다양하며, CERT를 비롯한 보안관련 단체에서는 새로운 공격기법의 등장과 함께 관련된 정보를 제공하고 있

다. 이 정보에 근거하여 일반적으로 알려진 공격기법들을 여러 가지 기준에 따라서 분류하면 Fig. 1과 같다.

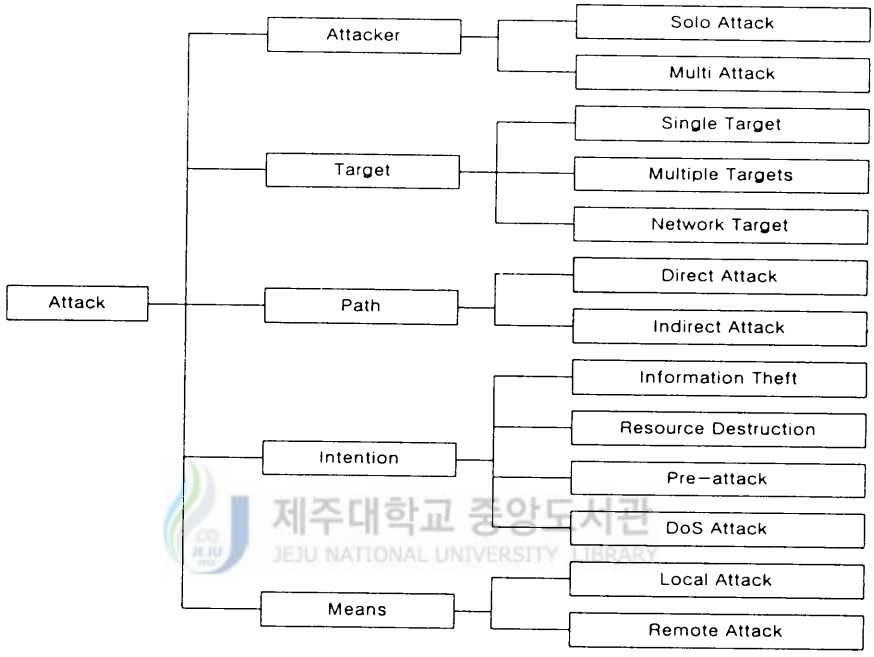

Fig. 1 Classification of Attacks

이러한 보안 위협과 공격으로부터 네트워크와 정보를 보호하기 위해서는 적절한 보안 서비스가 네트워크 환경 및 운영체제에서 제공이 되어야 하며, 주요 보안 서비스에는 다음과 같은 것들이 있다(Stallings, 1995. Siyank, 1995).

- 접근제어(Access Control) : 사용자의 식별에 관한 서비스 및 허가된 사용자가 허가된 범위 내에서 정보나 자원에 접근할 수 있도록 허용하는 기술적 방법
- 인증(Authentication) : 시스템 내에 있는 객체와 자원을 접근하려는 요구에 대해서 사용자의 신분확인을 수행

· 비밀유지(Confidentiality) : 시스템이 인증되지 않은 사용자에게 노출될 수 있는 위협에 대해 정보 혹은 자원 보호를 위해 비밀성이 유지되도록 하는 서비스
· 무결성(Integrity) : 정보가 우발적이건 고의적이건 간에 허가 없이 변경이 되지 않도록 하는 서비스
· 부인봉쇄(Non-repudiation) : 수신인이 정보를 받지 못했다고 부인하는 것을 방지하는 방법

보안관리 시스템에서는 암호화 기법, 인증체계, 접근제어 리스트(ACL, Access Control List)등의 기술을 이용하여 보안관리 서비스를 구현하게 된다. 보안관리 시스템에서 요구되는 서비스를 제공하기 위해서는 공격수행여부에 대한 빠르고 정확한 탐지 및 대응이 필요하며, 이를 위해서 침입탐지 시스템이 이용된다.

2. 침입탐지 시스템

침입탐지는 "컴퓨터 시스템 또는 네트워크에서 이벤트의 발생을 감시하고 여기에서 보안에 관련된 문제의 분석 및 침입여부를 결정하는 과정"을 말한다(Bace, 2000). 침입탐지 시스템은 보안관리 시스템에서 중요한 구성요소이며 침입차단 시스템(Firewall)등의 다른 보안도구와 함께 운용되어 통합 보안관리 시스템의 구축이 가능하도록 한다.

침입탐지 시스템에 대한 개념은 James Anderson에 의해 1980년에 처음 제안되었으며(Anderson, 1980), 1994년에서 1986년까지 Dorothy Denning과 Peter Neumann에 의해서 연구 개발된 실시간 침입탐지 시스템 IDES에서 구체화되었다(Denning, 1986). 침입탐지 시스템과 연관

된 최근의 주목할 만한 프로젝트로는 Purdue University의 COAST lab.(COAST)의 AAFID(Autonomous Agent for Intrusion Detection)와 SRI International의 NIDES, EMERALD등이 있다(SRI International).

1)침입탐지 시스템의 분류

침입탐지시스템(IDS, Intrusion Detection System)은 탐지를 위해 사용하는 감사자료, 탐지방법, 대응방식등에 따라서 다양하게 분류가 가능하다. Fig. 2는 다양한 기준에 따른 침입탐지 시스템의 분류를 보여주고 있다.

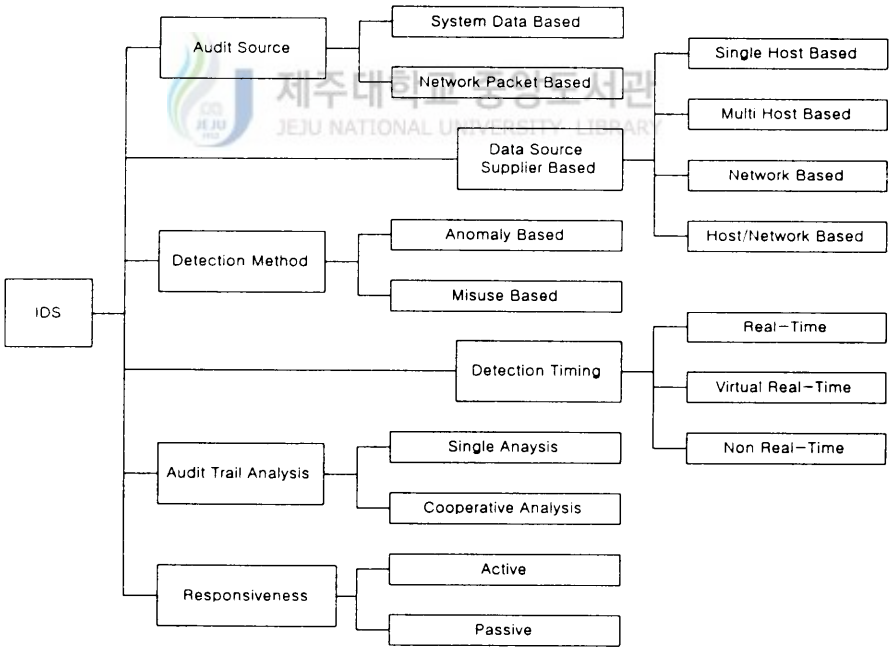

Fig. 2 Classification of Intrusion Detection Systems

이러한 분류방식중에서 가장 많이 이용되는 것은 탐지를 위해 분석하는 감사자료를 기준으로 하는 방식으로 네트워크 기반 침입탐지 시스템

과 호스트 기반 침입탐지 시스템으로 분류할 수 있다.

호스트 기반 침입탐지 시스템에서는 주로 시스템 로그파일, 프로세스 상태 및 시스템 호출 명령어 등의 실행과 같은 호스트 내부의 정보를 침입탐지에 이용한다. 이 경우 침입의 성공, 실패여부도 탐지가 가능하고 시스템 침입에 성공한 침입자의 활동에 대한 추적도 용이하지만, 시스템에 의존적이라는 단점이 있다.

네트워크 기반 침입탐지 시스템은 네트워크 패킷을 기본자료로 이용하여 탐지하게 된다. 네트워크 패킷을 이용할 경우 실시간으로 공격의 수행을 탐지할 수 있으며, 운영체제에 관계없이 동작이 가능하지만, 침입자가 공격의 증거를 없앨 경우에 탐지가 어려우며, 암호화된 패킷에 대해서는 탐지가 불가능하다는 단점이 있다(Vigna 등 1998). Fig. 3은 네트워크 기반 침입탐지 시스템의 기본적인 구조를 보여주고 있다.

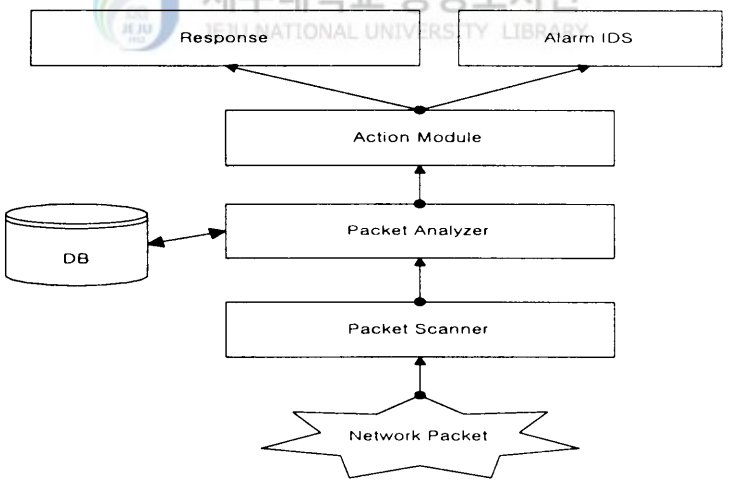

Fig. 3 Architecture of Network-based IDS

Packet Scanner에서는 네트워크 패킷을 잡는 역할을 하며, 수집된 패킷에서 분석에 이용되는 부분만을 필터링 하여 Packet Analyzer로 넘기게 된다. Packet Analyzer에서는 수집된 패킷데이터를 데이터베이

스로 저장된 규칙에 기인하여 분석을 수행하며, 탐지된 결과에 따른 실제의 대응은 Action Module에 의해서 Response Module 이나 Alarm을 이용하여 수행하게 된다.

2)침입탐지 시스템 모델

침입탐지 시스템 모델의 대표적인 경우로 비정상 탐지 모델과 오용 탐지 모델을 들 수 있다. 오용 탐지 모델은 알려진 공격패턴 및 비정상 동작에 대한 패턴을 이용하여 탐지를 하는 방식으로, 공격에 대해서 알려진 패턴과 시스템 활동의 내용이 일치하는 경우에 침입으로 탐지를 한다. 오용 탐지 모델은 알려진 공격에 대해서는 정확하게 탐지가 가능하지만 새로운 침입유형에 대해서 탐지를 할 수 없다. 따라서, 탐지규칙에 대한 지속적인 갱신이 요구되며, 감사 데이터에 대한 의존도가 높다는 단점이 있다(Bace, 2000).

비정상 탐지 모델은 통계적인 방법을 통하여 비정상적인 행위나 컴퓨터 자원의 사용을 감시하여, 이 값이 정해진 모델을 벗어나는 경우에 침입으로 판단하는 방식으로 알려지지 않은 공격방식에 대해서도 탐지가 가능하다는 장점이 있다. 하지만, 침입으로 판단하기 위한 임계값의 설정이 어려우며 경우에 따라서는 잘못된 결과를 출력할 가능성이 있고 구현 비용이 크다는 단점이 있다(Bace, 2000).

3. 이동에이전트 기술과 침입탐지 시스템

본 논문에서는 침입탐지를 위하여 이동에이전트 기술을 이용한다. 이동에이전트 기술은 인공지능분야에서 연구가 시작되어, 현재 다양한 응용분야에 대해서 연구되고 있는 기술로서 네트워크상의 이동성과 자율

적인 작업을 가능하게 한다. 코드의 이동성은 분산환경에서의 작업량 분배 및 이에 따른 대역폭절약, 고장에서의 회복, 시스템 관리 측면에서 고려가 되어 왔으며, 현재 인터넷에서 제공되는 서비스에 대한 적용방안이 다양하게 연구되고 있다(Bieszczad, 1998. CORBA, 1999. Dasqupta 등 1999.).

1)이동 에이전트 기술

이동에이전트는 사용자의 작업을 대신 수행하는 소프트웨어 프로그램을 의미하는 에이전트에 이동성을 부여한 것을 말하며, 인터넷을 비롯한 네트워크 환경에서 일정한 임무를 부여받아 자율적으로 시스템 사이를 이동하면서 필요한 정보의 수집 및 처리를 수행하는 실행 가능한 프로그램이다(Oshima, 1998). 이동에이전트 기술을 이용할 경우 네트워크상의 데이터 이동이 최소화됨으로써 네트워크의 효율적인 이용이 가능해진다. 개념적인 이동에이전트 시스템의 모델이 Fig. 4에서 보여지고 있다.

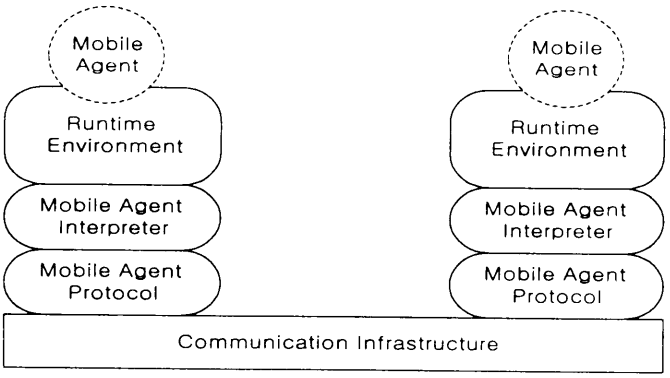

Fig. 4 Model of Mobile Agent System

이러한 이동에이전트 시스템을 구현하기 위해서는 이를 구현할 수 있는 프로그래밍 언어, 이동에이전트 코드를 실행할 수 있는 인터프리터,

그리고 이동에이전트간의 통신을 위한 프로토콜 등이 필요하다.

이동에이전트를 구현하기 위한 프로그래밍언어는 호스트간을 이동하면서 실행되는 이동에이전트의 특성 때문에, 플랫폼에 관계없이 실행이 가능해야 하므로 JAVA가 주로 이용되며, 그 외에 Perl, Lisp, TCL 등의 스크립트 언어가 고려될 수 있다. 인터프리터는 이동에이전트가 시스템의 환경에 관계없이 동작이 가능하도록 지원하며, 네트워크 하부구조에 상관없이 이동에이전트간의 통신이 가능하도록 이동에이전트 프로토콜이 정의되어야 한다.

현재 다양한 연구 단체와 업계에서는 이동에이전트 시스템을 위한 플랫폼을 개발하였으며, Java를 기반으로 하는 IBM의 Aglets Workbench, 미츠비시의 Concordia, General Magic의 Odyssey, Objectspace Voyager등과 OMG(Object Management Group)의 MASIF 표준에 따라 개발된 IKV++의 Grasshopper, TCL을 확장한 Agent-TCL을 이용하는 Dartmouth 대학의 D'Agent등이 있다.

2)이동에이전트기술의 장점

이동에이전트 기술은 관리자 입장이나 시스템 관점에서 다양한 장점을 제공하며, Fig. 5에서 이러한 장점을 개략적으로 보여주고 있다.

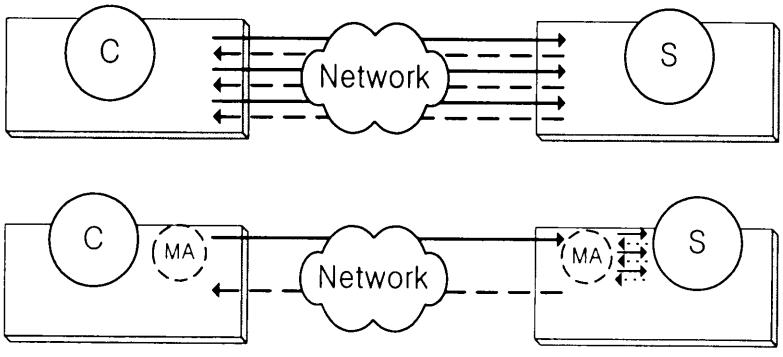

Fig. 5 Client/Server Model vs Mobile Agent Model

이동에이전트 기술은 기존의 RPC(Remote Procedure Call)를 이용한 클라이언트/서버 기술과 비교하여 향상된 효율과 적응력을 보여주며, 전송되는 자료가 원시자료가 아닌 처리된 결과만을 전송하므로 네트워크의 부하를 줄일 수 있으며, 관리대상과의 실시간 지역처리를 통해서 네트워크의 지연을 극복할 수 있다. 더불어 프로토콜 및 시스템 하부구조와 무관하게 동작이 가능하며, 비동기 적이고 자율적인 실행 및 적응이 가능해 지므로 결함허용성이 높은 견고한 프로그램이 가능해 진다.

전형적인 클라이언트/서버 모델에 대해서 이동에이전트 모델이 가지는 이러한 장점들로 인해서 이동에이전트기술은 전자상거래, 정보검색, 네트워크 관리 등의 다양한 분야에서 응용되고 있다.

3)이동에이전트기술을 적용한 침입탐지 시스템 연구동향

이동에이전트기술을 침입탐지 시스템에 적용하려는 시도는 침입탐지 및 보안관리와 연관하여 최근의 새로이 등장한 연구동향중의 하나이며, 많은 대학과 기업에서 현재 연구가 진행중이다. 주요 연구성과로는 일본의 IPA(Information-technology Promotion Agency)에서 개발한 IDA(Intrusion Detection Agent system)(Asaka, 1999), 미국 Iowa 주립대학의 Intelligent Agents for Intrusion Detection 프로젝트(Helmer, 1998), 미국의 NIST(National Institute of Standard and Technology)에서 현재수행중인 Mobile agent for Intrusion Detection System 프로젝트 등이 있다(Wayne 등, 1999. Wayne, 1999. Wayne, 2000).

IPA의 IDA는 다중 호스트 기반의 침입탐지 시스템으로, 침입과 연관된 전체 정보의 분석대신에 MLSI(Mark Left by Suspected Intruder)라는 시스템로그파일 및 주요 시스템 파일의 이상을 감지하여 침입여부를 판단하는 시스템이다. MLSI에 의한 이벤트가 발생할 경우 이 정보를 이용하여 공격자에 대한 추적 및 공격관련 정보의 수집을 이동에이

전트를 이용하여 수행한다.

NIST에서는 이동에이전트를 이용하여 침입탐지를 수행하는 프로젝트를 수행 중에 있으며, 1999년 중간보고서가 발표된 바 있다. NIST 중간보고서에서는 이동에이전트를 침입탐지 시스템에 적용할 경우 발생하는 다음과 같은 장점들을 제시하고 있다(Wayne, 1999).

- 네트워크 지연의 극복
- 네트워크 부하의 감소
- 비동기 적인 실행과 자율성
- 구조 및 조합의 편이성
- 동적인 적응력
- 이질화 된 환경에서의 운용가능
- 고장방지능력
- 확장용이성

이동에이전트를 이용하는 침입탐지 시스템에서는 관리대상 시스템으로 이동 후 임무를 수행하므로, 네트워크 대역폭의 효율적인 사용이 가능하다. 더불어 탐지에 관련된 규칙에 대한 관리를 중앙의 시스템에서 수행하게 되므로 새로운 공격기법의 등장에 대해서 유연성을 부여할 수 있다.

Ⅲ. 시스템 설계

시스템 설계를 위하여 CERT에서 발표된 자료를 기반으로 다양한 공격유형에 대한 조사 및 공격유형을 분석하고 침입탐지를 위한 보안관리영역을 결정하였으며, 보안관리영역을 구성하는 시스템 구성요소의 설계 및 동작을 정의하였다.

1. 공격유형의 분석

시스템을 설계하기 위해서는 공격유형 및 특징에 대한 분석작업이 선행되어야 한다. 이에 대한 기본 자료는 CERT에서 발표한 네트워크 공격유형에 대한 분석자료 및 통계를 활용하였다(CERT). Table 1은 한국정보보호센터에서 발표한 1999년 네트워크 공격의 유형 및 빈도에 대한 자료이다(CERT-kr).

공격유형을 살펴보면, 취약점 정보수집을 위한 공격과 운영체제의 버그를 이용한 버퍼오버플로우 공격이 많은 비중을 차지함을 알 수 있다. 취약점 정보수집인 경우 시스템에 대한 실제공격이전에 사전공격의 의미로 사용이 되며, 이러한 공격에 대한 탐지는 후속공격을 차단하는 효과를 거둘 수 있다. 공격자는 목표 시스템에 대한 정보를 수집 후 이를 바탕으로 버퍼오버플로우 취약점 공격 등의 후속공격을 수행하게 된다. 현재 알려져 있는 다양한 공격유형은 유형별 특징을 기준으로 몇 가지의 패턴을 구분할 수 있으며, 이에 따른 적절한 데이터 소스의 선정 및

설계상의 기능분산을 통하여 좀 더 효율적인 시스템의 설계가 가능하게 된다.

전형적인 공격의 유형은 공격근원지에 따라 두 가지로 분류할 수 있다. 외부 공격은 공격자가 목표시스템에 직접적으로 접근할 수 없는 경우에 공격수행초기에 수행이 된다. 내부 공격은 공격자가 시스템에서 허가된 권한을 초과하는 사용권한을 취득하기 위해서 외부공격에 이어 수행되는 공격이다.

Table 1 Attack Types in year 1999

분 류	횟수	비 고
사용자 도용	68	sniffer, brute force, crack, 계정 도용, 기타
S/W 보안오류 이용	3	phf-CGI 등
버퍼오버플로우 취약점 이용	214	popd, imapd, mountd, named, amd, ftpd, rpc.statd & automoutd, rpc.ttdbserver, rpc.cmsd, ufsrestore, 기타
구성·설정 오류	2	신뢰관계 이용, 기타
악성 프로그램	58	Back Orifice, NetBus, rootkit, 백도어, 인터넷웜, 기타
서비스거부공격	16	smurf, trin00, ping flooding, SYN flooding, nuke, 기타
E-mail 관련공격	20	spam mail, mail bomb
취약점 정보수집	272	mscan, sscan, imapd scan(143 port), popd scan(110 port), named scan(53 port), ftpd scan(21 port)

일반적으로 공격자는 목표시스템에 대해 허가되지 않은 접근을 가능

하게 하기 위해 외부 공격을 수행한 후 Super-user권한과 같은 상위의 권한을 취득하기 위해 내부 공격을 수행한다. 이러한 과정에서 공격자는 목표시스템이나 네트워크 패킷에 비정상적인 흔적을 남기게 되며, 침입탐지 시스템은 공격수행 여부를 판단하기 위해 이를 감시한다. 제안 시스템의 설계를 위해 공격의 수행과정을 검색단계, 수행단계, 그리고 위장단계로 구분을 하였으며 다음과 같다.

- 검색단계 : 공격자는 목표시스템을 검색하고, 운영체제의 유형, 버전 등의 목표시스템과 관련된 정보를 수집한 후 해당 시스템의 취약점을 검색한다. 광범위한 취약점 검색 공격이 이 과정에서 수행이 되며, 패킷감시, 시스템의 로그파일 조사, UNIX 운영체제의 netstat등의 명령을 이용하여 탐지할 수 있다.
- 수행단계 : 공격자는 목표시스템의 침투를 위하여 공개된 공격용 프로그램을 이용한 일련의 명령 또는 작업을 수행한다. 이러한 공격은 파일시스템 및 시스템 로그파일의 변화를 수반한다. 파일시스템의 수정여부, 시스템의 프로세스 상태 등의 검사를 통하여 공격을 탐지하지 할 수 있다.
- 위장단계 : Super-user권한을 취득한 공격자에 의해 공격수행에 대한 증거를 숨기고, 목표시스템의 접근을 위해 사용할 수 있는 뒷문(backdoor)이 설치된다. 이 과정까지 수행이 된 후에는 일반 사용자가 이용과정에서 공격의 여부를 인식하는 것은 매우 어렵게 된다.

이러한 공격단계의 구분과 가상적인 공격 시나리오의 활용은 적절한 보안정책을 수립하고, 보안관리시스템을 설계하는 과정에 도움을 주며, 결과적으로 개발된 보안관리시스템의 신뢰성을 확보하는데 도움을 줄 수 있다(Siyank, 1995. Stallings 1995).

다음 Table 2는 위에서 구분한 공격단계를 기준으로 현재 알려진 공격유형 및 각 단계별 특징에 대해서 간략하게 정리한 것이다.

Table 2 Characteristics of the Attack Stages

공 격 단 계	주요공격유형	특 징
검색단계 (Search Stage)	호스트 스캔 포트 스캔 서비스 접속시도 DNS 정보 검색	패킷감시를 통한 탐지 접속요청의 증가 네트워크 트래픽 증가
수행단계 (Action Stage)	버퍼오버플로우 exploit 코드 수행 IFS Attack	공개된 취약점 이용 주요 시스템 파일의 변경 setuid 파일의 이용
위장단계 (Masquerade Stage)	로그파일 삭제 시스템파일의 변조 백도어 설치	비정상적인 파일의 존재 허가되지 않은 데몬 비정상 포트의 존재

제안시스템에서는 각 단계별 공격의 특성에 따라서 탐지에 관한 임무를 분리하여 설계하였다. 검색단계의 공격은 네트워크 패킷을 대상으로 탐지가 용이하며, 신속하고 정확한 탐지가 가능하기 때문에 네트워크상의 패킷을 대상으로 탐지임무를 부여받은 시스템을 두었으며, 수행단계 및 위장단계의 공격에 대한 탐지는 실제 시스템에 대한 조사량이 많기 때문에 이동에이전트 기술을 적용하여 임무를 부여하였다.

대규모에 대한 네트워크에 대해 통합 관리를 위한 공격유형의 분석과 공격기법별 특성에 따른 임무의 분담을 통하여 보안관리 시스템 자체를 운용하는데 드는 부하를 줄일 수 있으며, 변화하는 네트워크 환경과 새로운 공격기법의 등장에 대한 유연성을 부여할 수 있다.

2. 보안관리영역

제안 시스템에서 고려하고 있는 보안 관리 영역은 보안관리 시스템 (SMS, Security Management System), 각 네트워크 세그먼트별로 하나의 탐지시스템 (DS, Detection System)과 SMA(Security Mission Agent), 그리고 다수의 관리대상시스템 들로 구성이 되며, Fig. 6 에서 이러한 구성이 보여지고 있다.

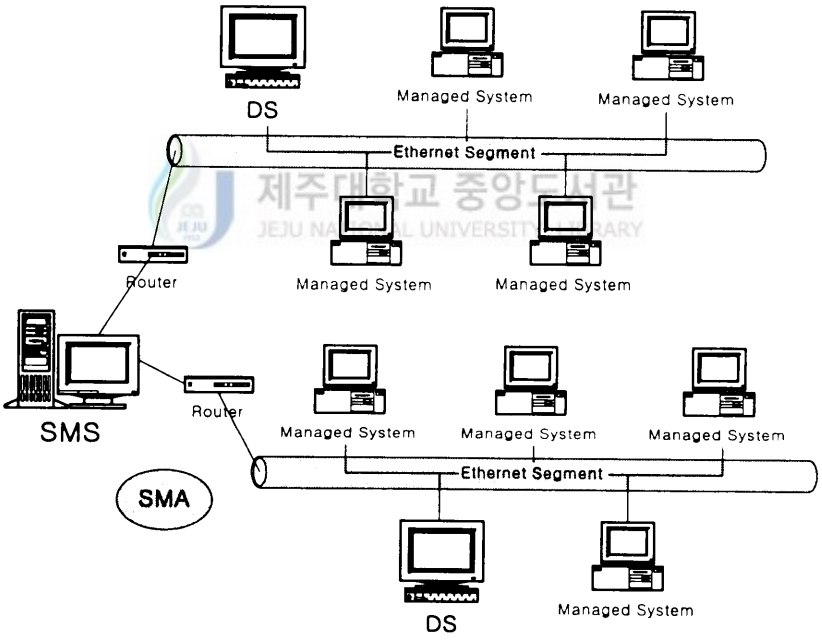

Fig. 6 Security Management Domain

전체 보안관리영역내의 관리대상 시스템 및 침입탐지시스템에 대한 관리를 담당하는 SMS를 두었으며, 각 네트워크 세그먼트별로 1대의 DS를 두었다. DS는 네트워크 패킷에 대한 감시를 통하여 검색단계의 공격을 탐지하고 공격의 수행이 탐지될 경우 이에 관한 정보를 SMS로

전달하는 역할을 담당한다. 수행단계 및 위장단계의 공격탐지를 위한 임무는 이동에이전트에 보안관련 임무를 부여한 SMA를 두어 담당하도록 하였다.

3. 시스템 구성요소

전체 보안관리 영역의 각 시스템에 대해서 각각의 동작환경 및 부여된 임무에 따라서, 각 시스템별로 요구되는 기능에 대한 세부 설계를 하였다.

1)SMS(Security Management System)

SMS는 전체 보안관리영역에서 발생하는 보안관련 사건들에 대한 감시 및 관리를 수행한다. 이를 위해서 DS에 의해 각 세그먼트별로 탐지된 검색단계공격에 대한 탐지정보를 기반으로 적절한 임무가 부여된 SMA를 선정 및 파견하며, 이에 대한 결과를 기반으로 보안관리를 수행하게 된다. 정형화된 공격방식에 대한 자동화된 관리업무의 수행이 가능하도록 하기 위해서 보안관리영역에 대한 관리기능과 SMA에 대한 관리기능을 별도의 모듈로 분리하여 구성할 필요가 있으며, 발생하는 사건 및 관리 기록을 유지하여 이후의 단계에 대한 감사자료로 활용할 수 있도록 한다.

2)DS(Detection System)

검색단계에서 DS는 각 네트워크 세그먼트내의 시스템들로 스캔공격을 비롯한 검색단계의 공격이 수행되고 있는지를 탐지한다. 취약점 검색 공격은 실제의 공격이 수행되기 전에 사전공격의 유형으로 수행되는

경우가 많으므로, 이에 대한 신속한 탐지는 전체 시스템의 성능에 크게 영향을 미치게 된다. 실시간 탐지를 위하여 DS에서는 네트워크 패킷을 기본 데이터로 이용하며, 해당 네트워크 세그먼트 내의 모든 패킷을 받아들일 수 있도록 promiscuous 모드로 동작을 한다. 따라서 하나의 DS를 이용하여 해당 네트워크 세그먼트 전체 시스템에 대하여 감시가 가능하다. 설계하는 과정에서는 일반적으로 소규모의 네트워크에서 많이 이용되는 Ethernet 환경을 가정으로 하였지만, 다른 하부구조를 갖는 네트워크 세그먼트인 경우에는 라우터 또는 게이트웨이 시스템에 설치될 경우 동일한 효과를 거둘 수 있을 것이다.

DS에는 기본적으로 네트워크 패킷의 수집 및 분석에 대한 기능이 필요하다. 이를 지원할 수 있는 도구들은 현재 상당수가 개발이 되어 있으며, 이러한 도구들은 다른 도구들과 연계하여 사용되는 경우가 많다. Table 3에는 패킷을 잡아서 분석할 수 있는 도구 및 특징에 대해서 소개되어 있다.

Table 3 Network Packet Monitoring Tools

도구	필터기능	출력양식	비고
etherfind	○	Text	
tcpdump	○	Text	패킷 필터링 기능이 우수
nfswatch	△		통계적 출력 기능
nnstat	○	Text	
traffic	×	GUI(Histogram)	
etherman	○	GUI	트래픽 상황 판단 용이
interman	○	GUI	
packetman	○	GUI	패킷의 내용분석 용이

검색단계의 공격은 일반적으로 목표시스템 외부에서 침투를 위한 기본정보의 수집을 목적으로 수행되며, 검색단계의 공격이 수행될 경우 네트워크 패킷을 관찰하여 탐지가 가능하다. 예를 들어 TCP SYN scan 공격인 경우, SYN 플래그가 설정된 TCP패킷의 수가 급격히 증가하게 되므로 이를 감시하여 공격의 수행여부를 알아낼 수 있다.

일반적인 취약점 검색 공격인 경우 이러한 네트워크 패킷에서 보이는 특징을 이용하여 탐지가 가능하지만, 네트워크의 상황에 따라서 임계값을 설정하는 문제는 어려움이 있을 수가 있다. 시스템이 올바르게 동작하기 위해서는 실제 공격이 아닌 것을 탐지하는 "거짓탐지(false-positive)"와 공격의 수행을 탐지하지 못하는 "탐지실패(false-negative)"를 최소화하는 방법을 고려해야 한다. 제안 시스템에서는 탐지실패를 줄이는데 중점을 두며, 거짓탐지를 줄이기 위하여 보안 관리자가 네트워크 환경에 따라서 DS의 환경설정이 가능하도록 설계한다.

DS는 네트워크 패킷을 패킷 캡쳐 모듈을 이용하여 추출한 필터링된 IP와 관련정보를 열려진 접속 또는 포트를 검색하는 용도로 많이 이용되는 해쉬테이블을 이용하여 저장한 후, 이 값의 변화를 이용하여 공격의 수행여부를 알아내게 된다. 제안 시스템의 프로토타입에서는 "동일한 IP에서 일정한 시간간격으로 임계 값 이상의 연결요청이 있는 경우에 이를 취약점 검색공격으로 판단"하도록 설계하였으나 다른 형태의 공격방식에 대해서는 해당공격에 대한 규칙을 추가함으로써 확장이 가능할 것이다.

3)SMA(Security Mission Agent)

수행단계와 위장단계의 공격은 시스템에 파일시스템 변경, 주요 시스템 파일의 수정, 프로세스의 변화와 같은 다양한 흔적을 남기게 된다. 이 단계들의 공격을 탐지하기 위해서는 호스트내의 정보를 검사하여 탐

지하는 것이 용이하며, 제안시스템에서는 이동에이전트에 보안관련 임무를 부여한 SMA가 탐지임무를 담당한다. 검색단계에서 공격이 탐지되면, DS는 SMS로 관련정보를 전달하게 된다. SMS의 보안관리자는 전달된 정보를 바탕으로 적절한 임무를 부여한 SMA를 파견하게 되며, SMA에 의해서 해당 시스템들에 대한 조사를 수행하기 된다. Fig. 7은 SMA의 동작과정을 보여주고 있다

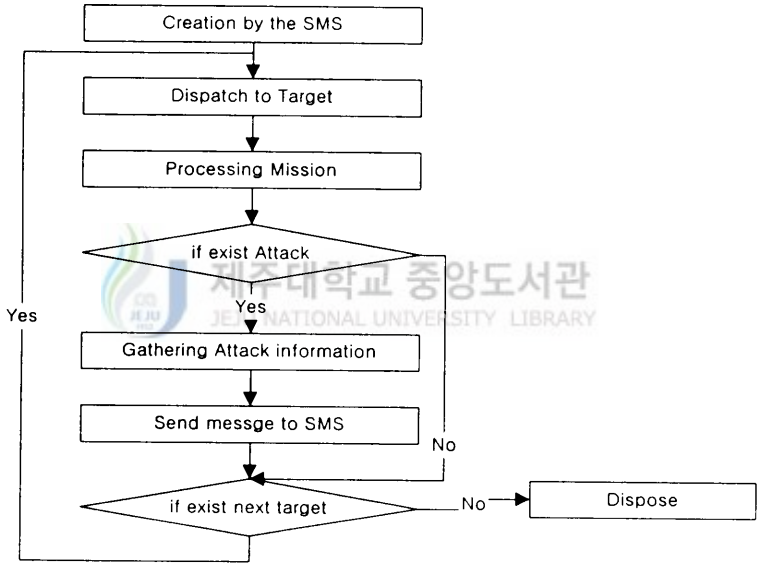

Fig. 7 Work flow of SMA

SMA에 부여되는 임무는 CERT의 권고안을 바탕으로 정의가 되었으며, 새로운 공격기법에 대해서는 관련 임무를 정의하여 추가함으로써 유연성을 부여할 수 있다. 이 경우에 탐지규칙에 대한 정형화에 대해서도 고려할 필요가 있다. 최근에 알려져 있는 공격기법들과 CERT의 권고안을 바탕으로 제안 시스템에서 정의한 SMA의 임무는 Table 4와 같다.

Table 4 Missions of SMA

임 무	설 명
SysInfoSMA	시스템의 운용환경에 관한 정보의 수집
LogInfoSMA	시스템 로그파일의 수정여부 및 공격여부 점검
SysFileInfoSMA	시스템 파일의 변경여부
SysBinSMA	/etc/inetd.conf파일이 참조하는 파일의 수정여부
UserInfoSMA	사용자 정보의 변화
GroupInfoSMA	그룹 정보의 변화
IllProcInfoSMA	인가되지 않은 프로세스의 존재여부 점검
HidFileInfoSMA	새로이 추가된 Hidden파일의 존재여부 점검
BackdoorInfoSMA	백도어의 존재여부 점검

SMA에 의한 임무의 수행은 데이터의 전송 량을 최소화시킴으로써 네트워크 대역폭을 절약하는 데 도움을 줄 수 있으며, 더불어 새로운 공격기법의 등장에 대해서 유연성을 부여할 수 있고, 환경의 변화에 대한 적응력을 부여할 수 있다.

4. 시스템 모듈설계

MIDS(Mobile Intrusion Detection System)로 명명된 제안 시스템은 이질화 된 인터넷 환경에서 동작이 가능하고, 대규모의 네트워크에 대한 침입탐지를 고려하여 설계하였다. 각각의 시스템 구성요소들은 이후 연구에서의 확장성에 대한 고려로 세부적인 모듈에 대한 기본적인 기능에 대해서만 정의가 되었다. 제안시스템의 전체적인 모듈구조는 Fig. 8

과 같다.

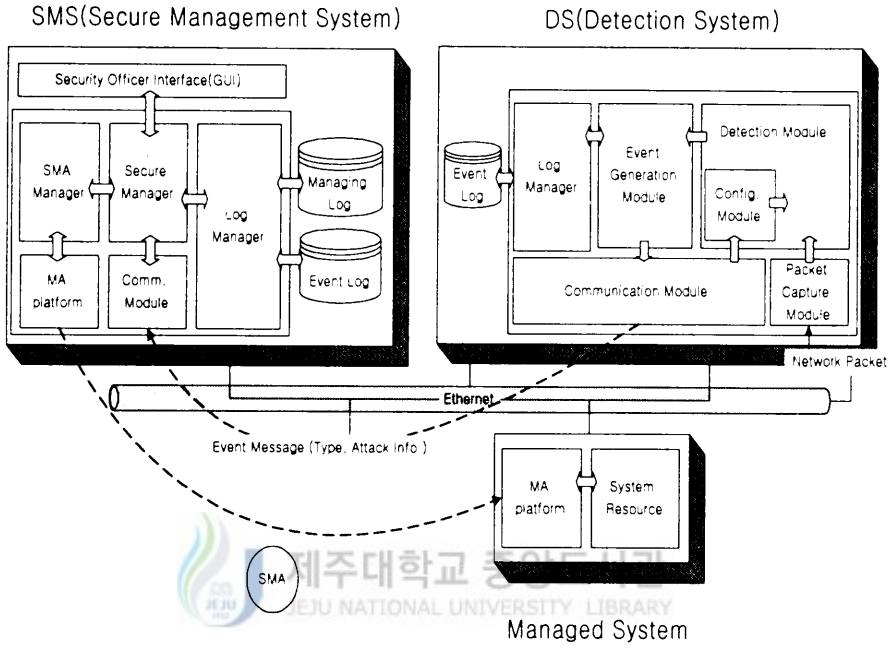

Fig. 8 Architecture of MIDS

SMS는 관리대상시스템의 침입여부를 점검하기 위해 사용하는 SMA 및 전체 보안관리 영역에 대한 관리를 하며 보안관리자와의 인터페이스를 제공한다. SMS를 구성하는 각 모듈의 기능은 다음과 같다.

- Security Officer Interface : 보안 관리자와 시스템의 인터페이스
- Secure Manager : 전체시스템의 관리
- Communication module : DS와 SMS간의 통신처리
- SMA Manager : SMA임무에 대한 관리
- SMA : 침입탐지 및 정보수집 에이전트
- Log Manager : 관리기록 및 이벤트 기록의 관리
- Managing Log : Secure Manager의 수행내용기록

· Event Log : DS에서 보고된 이벤트의 기록

DS는 각 네트워크 세그먼트별로 설치되어 네트워크 패킷을 기반으로 취약점 검색 공격의 수행을 감시한다. 이를 위해서 네트워크 패킷에 대한 수집 및 필터링, 탐지모듈, 그리고 SMS와 DS의 통신을 위한 통신모듈등이 포함되며, 각각의 모듈들은 다음과 같은 기능을 가진다.

· Packet capture module : 네트워크 패킷의 캡처 및 필터링
· Detection module : 공격의 수행여부 판단
· Configuration module : DS환경조절을 위한 SMS의 제어모듈
· Event Generation module : SMS로 전달되는 이벤트 처리
· Log manager : 이벤트 로그에 대한 관리
· Event log : 침입탐지 이벤트에 대한 기록

SMS와 관리대상 시스템들에는 SMA의 동작을 지원하기 위해서 이동에이전트 플랫폼이 설치된다. 이동에이전트 플랫폼은 SMA의 이동성을 지원하고, 각 시스템에서 자원을 할당받아 시스템의 대한 정보 및 자원을 접근할 수 있도록 한다. 제안 시스템에서는 이를 지원하기 위한 이동에이전트 플랫폼으로 IBM의 Aglet workbench를 이용하였다.

5. 시스템 동작과정

시스템의 전반적인 동작과정은 다음과 같으며 Fig. 9에서 전체 동작과정에 대해 그림으로 설명하고 있다.

① 공격자에 의한 검색단계의 공격이 시작된다.

② DS에 의한 검색단계의 탐지되고, 관련정보가 SMS로 전송된다.

③ SMS로 전송된 정보를 기반으로 보안관리자는 적절한 임무가 부여된 SMA가 해당하는 목표시스템으로 파견된다.

④ SMA는 목표시스템에서 수행단계 및 위장단계의 공격수행여부를 탐지한다.

⑤ 탐지 결과를 SMS로 전송한 후 다음 목표시스템으로 이동한다.

⑥ 임무를 마친 SMA는 소멸된다.

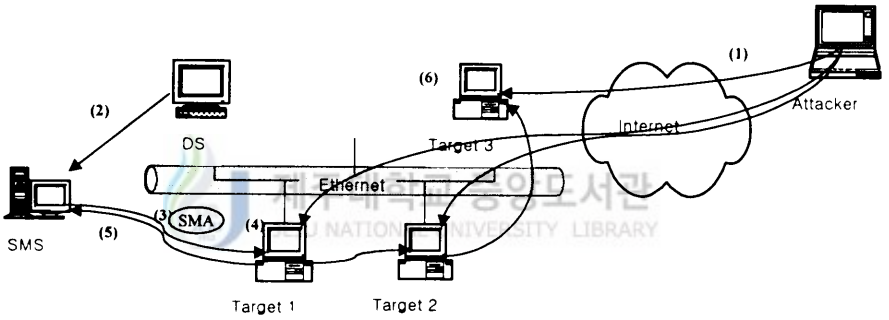

Fig. 9 Operations of The Suggested System

이 전체 과정에서 탐지에 관련된 처리과정이 해당 시스템 내에서 이루어지므로, 필요한 정보 및 데이터의 이동을 최소화할 수 있어 네트워크 대역폭의 효율적인 이용이 가능하게 된다.

Ⅳ. 시스템 구현 및 고찰

제안 시스템은 SMS, DS 및 일부의 임무를 부여한 SMA를 포함하는 프로토타입의 형태로 구현이 되었다. 본 장에서는 기본적인 구현환경 및 프로토타입 시스템 구현 및 결과에 대해서 서술한다.

1. 프로토타입 시스템 구현 환경

제안된 설계구조의 프로토타입 시스템 구현을 위해 사용된 구현환경은 다음과 같다.

- SMS : Solaris 2.6 on Sun Ultra spark 2
- DS : Linux Kernel 2.2.5-22 on IBM PC
- Managed system : Solaris 2.6 on Sun Ultra spark 2
  Windows 95 on IBM PC
- Languages : SUN's Java Development Kit 1.1.8
  GNU's gcc-2.95.2
- Mobile agent platform : IBM Aglet workbench 1.1b3
- Packet capture library : PCAP library 0.52

SMS의 전체 구성요소들은 자바언어를 이용하여 구현을 하였으며, 이동에이전트 플랫폼 역시 자바로 구현된 IBM Aglet workbench를 이용

하였다(Aglets. Lange 등 1998. Lange 등 1999. Karjoth 등 1997. Aridor, 1998). 이러한 구성은 SMS가 운영체제에 관계없이 동작이 가능하도록 한다.

DS의 Packet capture module의 구현에는 tcpdump를 비롯한 네트워크 패킷에 대한 감시를 위해 많이 이용되는 PCAP 라이브러리와 GNU의 gcc를 이용하였으며, 데몬 프로세스의 형태로 동작되도록 구현되었다. SMA는 이동에이전트에 보안관련임무를 부여한 것으로 자바언어를 이용하여 시스템에 독립적으로 수행 가능하도록 구현하였다.

2. 시스템 구성요소의 구현

제안된 설계구조의 동작 및 가능성을 확인하기 위해 프로토타입 시스템의 기본구조 및 몇 가지의 탐지 규칙에 대해서 구현을 하였다. 공격탐지를 위해서 사용되는 공격 및 탐지규칙은 다양한 공격유형 중에서 빈도수가 높은 공격의 유형에 대해서 고려했으며, 필요할 경우에는 탐지규칙의 추가만을 통하여 동작이 가능하도록 고려하였다.

1)SMS의 구현

SMS는 전체시스템을 관리하는 역할을 한다. 구현상에서는 DS에서 전송된 탐지정보를 감시할 수 있는 인터페이스 및 SMA Manager 모듈에 대한 사용자 인터페이스 및 관리를 수행할 수 있도록 하였다. 이동에이전트의 동작을 지원하는 플랫폼으로는 IBM의 Aglets Workbench를 이용하였다. Fig. 10은 SMS의 사용자 인터페이스 중에서 DS에서 보고된 검색단계의 공격에 대한 정보를 보여주는 EventWin 의 사용자 인터페이스 화면을 보여주고 있다. DS에 탐지된 공격은 EventWin의

사용자 인터페이스를 통하여 관련된 정보를 감시할 수 있으며, 보고된 정보를 eventlog에 기록하여 이후의 관리작업에서 참고를 할 수 있다.

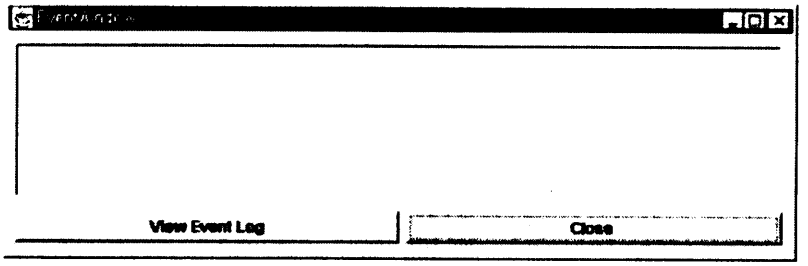

Fig. 10 Interface of the EventWin on SMS

SMA의 이동성을 지원하기 위한 플랫폼으로 IBM Aglet workbench를 이용하였기 때문에, SMA Manager의 기능을 stationary agent 형태의 aglet 프로그램으로 구현을 하였다. Fig. 11은 SMA Manager의 인터페이스 화면을 보여주고 있다.

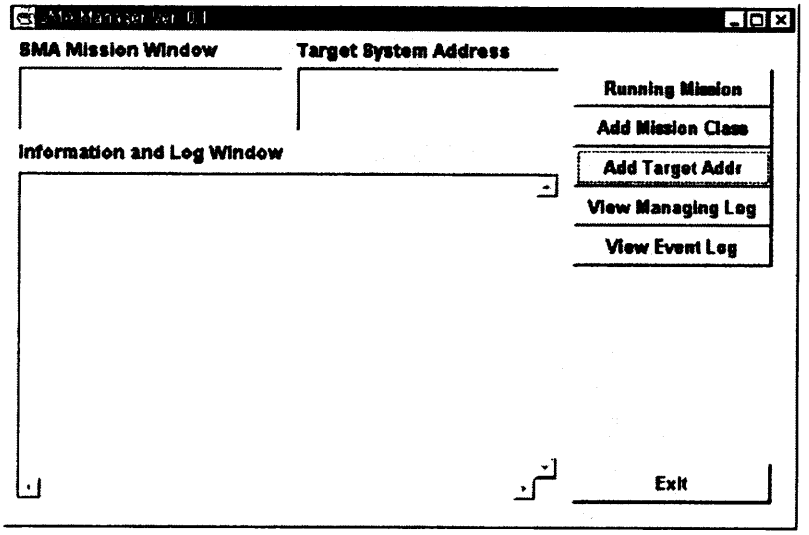

Fig. 11 Interface of the SMA Manager on SMS

이동에이전트에 부여되는 SMA Mission은 자바언어를 이용하여 구현되어 컴파일된 클래스 파일들로 구성된다. 구현된 SMA 클래스 파일은 Add Mission Class 메뉴를 이용하여 추가되며, Add Target Addr 메뉴에 의해서 설정된 목표시스템으로 이동한 후 결과를 전송해 오게 된다. 결과 및 관리기록에 대한 log는 Information and log Window를 통하여 전시된다.

SMS의 구현언어로 자바를 이용했으며, 시스템 의존적인 메소드의 사용을 하지 않았으므로, 자바가상머신(JVM, Java Virtual Machine)이 동작하는 시스템에서는 동작환경에 무관하게 수행이 가능하다.

2)DS의 구현

DS는 각 네트워크 세그먼트 별로 하나의 시스템에 설치되어 데몬 프로세스(daemon process)의 형태로 동작을 하며, 검색단계의 공격에 대한 탐지를 수행한다.

프로토타입 시스템의 DS에서 탐지하는 검색단계의 공격유형으로 TCP SYN scan과 Stealth Scan 공격을 선정하였다. TCP SYN Scan은 일반적으로 가장 많이 이용되는 취약점 검색 공격의 유형이며, 공격수행시 동일한 소스IP에서 연결요청이 급속하게 증가하는 특성이 있다. Stealth Scan 공격은 최근에 등장한 공격유형으로 정상적인 연결요청을 하지 않고, TCP TIN 패킷을 이용하는 방식으로 기존의 시스템에서 탐지기능이 구현되지 않은 경우가 많았기 때문에 선정을 하였다.

검색단계의 공격은 다수의 소스 IP에 대한 정보를 저장해야 하기 때문에 탐지를 위한 기본정보의 저장을 위하여 Hash Table을 이용하였으며, 시스템 동작상의 효율을 높이기 위해 탐지에 필요한 정보만을 필터링 하여 이 정보를 Hash Table에 저장하여 이용하였다. 공격으로 탐지를 하는 과정에서 필요한 시간측정을 위하여 기본 구조에 저장된 시각을 기록할 수 있는 필드를 두었다. 다음은 DS에서 기본정보의 저장

을 위해서 정의한 구조체(struct)이다.

```
struct syn_host{
    struct syn_host *next;
    clock_t timestamp;    // 저장된 시간
    time_t start;
    struct in_addr saddr;  // 소스 IP
    struct in_addr daddr[SYNCOUNT - 1];  // 목적지 IP의 배열
    unsigned short sport[SYNCOUNT - 1];  // 소스 Port의 배열
    unsigned short ports[SYNCOUNT - 1];  // 목적지 Port의 배열
    int count;                           // 저장된 순서
};
```

필터링을 하는 과정에서 정상적인 서비스에 대해서 공격으로 탐지하는 것을 막기 위하여 Scan 공격과 유사한 동작을 보이는 http(80), ftp-data(20), web-cache(8080)등의 정상 서비스 포트에 대해서는 제외하였다. 구현상에서 DS 자체의 무력화를 위한 서비스거부공격을 막기 위해 탐지횟수를 제한할 수 있도록 하였으며, 네트워크의 상태에 따라서 탐지를 위한 기준값 및 제외하는 포트에 대한 조절이 가능하도록 구현하였다. 다음은 DS를 구현한 원시 프로그램 중 필터링을 위한 기본정보 및 임계값의 설정부분이다.

```
#define LOGCOUNT 3    // DS에 대한 DoS공격을 막기 위한 제한 값
#define LOGDELAY (CLK_TCK * 20)

/* Your Local Net Number */      // 공격탐지에서 제외한 네트워크
#define IGNET "and not src net 0.0.0"
/* Default Ignore Port */          // 제외할 포트
#define IGPORT "and not (port 80 or 113 or 20 or 8080 or 143)"
```

```
#define SYNCOUNT 6    // TCP SYN Scan 탐지를 위한 임계값
#define SYNDELAY 50   // 50msec 동안에 6회 이상의 연결요청을 탐지
#define FINCOUNT 6    // Stealth Scan 탐지를 위한 임계값
#define FINDELAY 80   // 80msec 동안에 6개 이상의 TCP TIN 패킷
```

DS는 동작하는 과정에서 패킷필터링을 수행하여, 필요한 정보와 수집한 시간을 기본정보로 Hash Table에 저장하게 된다. 저장된 정보에 대해서 임계값의 초과여부를 검사한 후 공격으로 탐지가 될 경우 해당하는 공격자IP 및 관련정보를 SMS로 전송하게 된다. Fig. 12는 DS의 동작과정에 대해서 설명하고 있다.

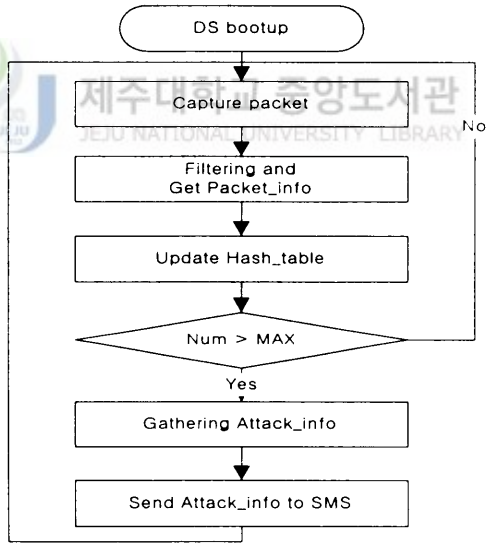

Fig. 12 Work flow of DS

위의 동작과정에서 Num은 Hash table에 저장된 동일한 소스 IP별 패킷수, MAX는 공격으로 탐지하기 위한 임계값, Packet_info는 공격유형에 따라서 패킷에서 추출된 관련정보, 그리고 Attack_info는 SMS에 보고되는 기본정보를 의미한다.

3)SMA의 구현

SMA는 이동에이전트에 탐지와 관련된 임무를 부여한 것으로, 프로토타입 시스템에서는 정보수집 및 공격탐지라는 SMA의 기본적인 임무를 반영할 수 있도록 SysInfoSMA와 SysFileInfoSMA를 구현하였다.

SysInfoSMA는 지정된 관리대상 시스템들을 이동하면서 시스템에 대한 정보수집 및 IFS를 이용한 내부공격에 대한 탐지를 수행하며, 수집되는 기본정보는 다음과 같다.

- os.name : 운영체제의 종류
- os.arch : CPU type
- os.version : 운영체제의 버전
- file.separator : IFS attack 탐지를 위한 파일 분리자
- path.separator : 경로 분리자
- line.separator : 라인 분리자

SysFileInfoSMA는 주요 시스템 파일의 변경여부를 통하여 호스트 공격이 수행되었는지를 탐지한다. SysFileInfoSMA에서 고려되는 주요 시스템 파일의 목록은 다음과 같다.

- /etc/system
- /etc/dfs/dfstab
- /etc/remote
- /etc/profile
- /etc/vfstab
- /etc/inetd.conf
- /etc/ttydefs
- /etc/inet/protocols

· /etc/ttysrch

· /etc/inet/services

· /etc/rpc

이러한 파일들은 운영체제의 설치 이후에 변경이 되지 않는 특성이 있다. 따라서 검색단계의 공격이 수행된 이후에 파일의 변경이 있었을 경우에 침입이 발생한 것으로 탐지할 수 있다. 최종 파일의 변경시간이 검색단계의 공격수행이 이전이더라도 경고 메시지를 SMS로 전송하도록 구현하였으며, Fig. 13은 SysFileInfoSMA의 기본 동작을 보여주고 있다.

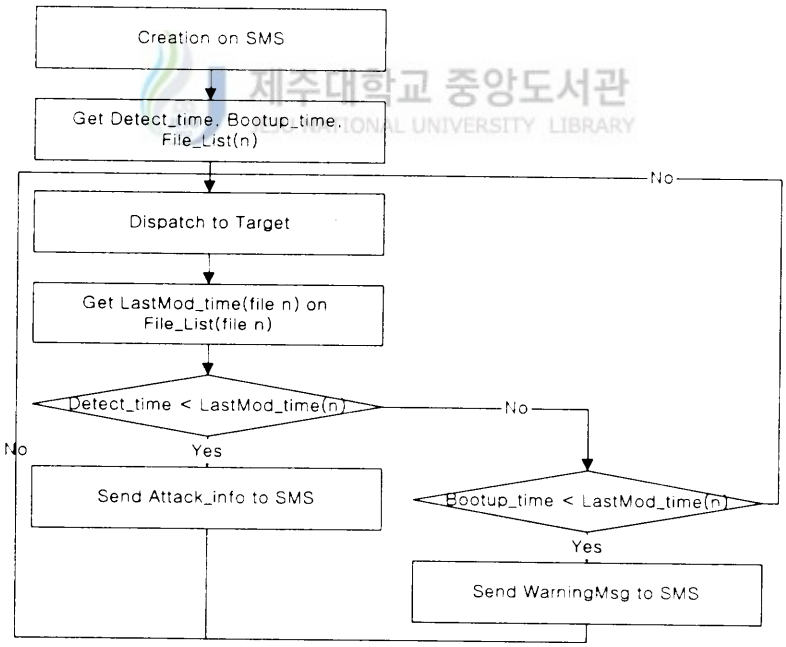

Fig. 13 Work flow of SysFileInfoSMA

위의 그림에서 File_List(file n)는 조사를 위한 시스템 파일의 목록,

Detect_time은 DS에서 검색단계 공격을 탐지한 시각, Bootup_time은 목표시스템의 설치시각, 그리고 LastMod_time(n)은 조사 대상 파일들의 최종 변경시각을 의미한다.

3. 구현결과 및 고찰

구현된 프로토타입 시스템은 두개의 이더넷 세그먼트와 다수의 관리 대상 시스템을 포함한 가상적인 보안관리 영역에서 실제의 공격도구를 활용한 공격을 수행하고, 전달된 결과를 바탕으로 SMA의 수행을 확인하는 형태로 시험되었으며, 최종적으로 정상적인 동작이 확인되었다. Fig. 14는 제안시스템의 프로토타입을 시험한 환경을 보여준다.

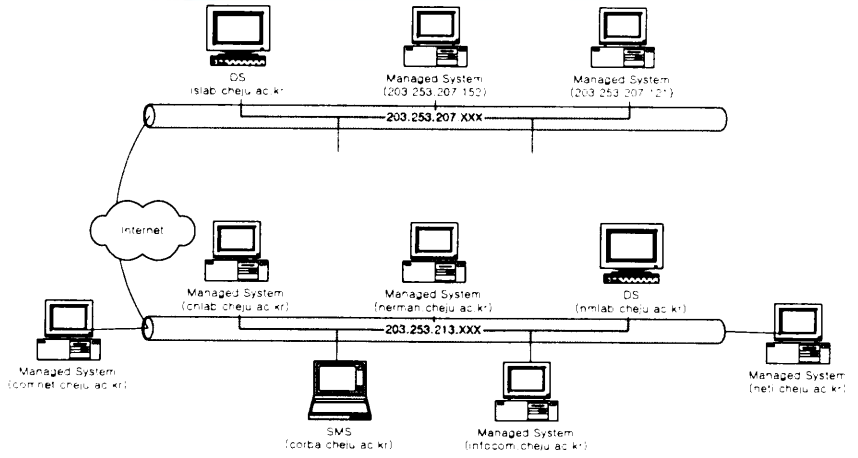

Fig. 14 Test Environment of the Prototype System

Fig. 15는 구현된 프로토타입 시스템의 동작을 확인하기 위해 시험하는 과정을 보여주고 있다. 가상적으로 설정한 nmlab이라는 이름의

공격시스템에서 nmap 이라는 취약점 검색 공격 도구를 이용하여 스캔공격을 시도하고(Nmap), islab이라는 시스템에 설치된 DS에서 탐지되어 SMS의 EventWin으로 탐지된 결과가 전달되는 과정을 보여주고 있다.

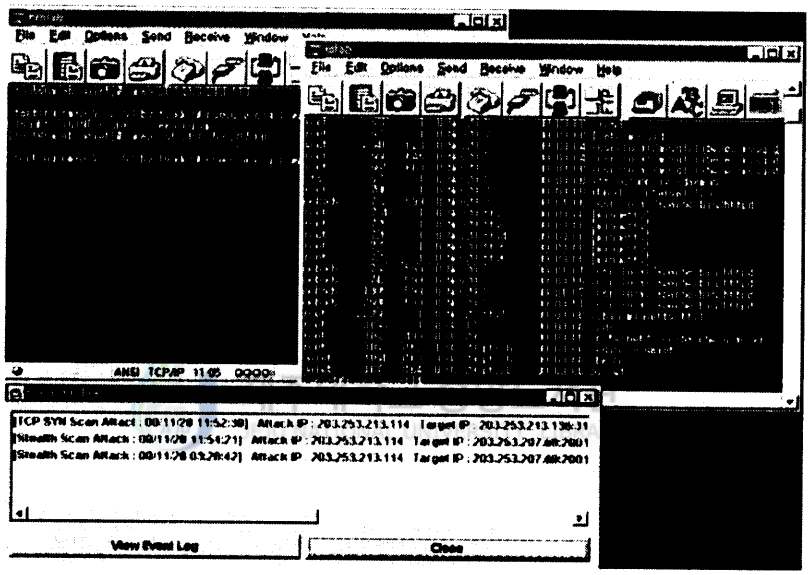

Fig. 15 Snapshot of the Prototype System Testing

위의 그림에서 스캔도구를 이용한 가상적인 공격수행에 대해 정상적으로 탐지가 되고, 공격과 관련된 정보가 SMS의 EventWin으로 전달되어 온 것을 확인할 수 있다.

1)DS의 구현결과

DS는 학내의 네트워크에서 213 과 207 네트워크 세그먼트의 각 1대의 시스템에 설치되었으며, Fig. 15에서 ds라는 이름의 프로세스로 해당 프로세스가 동작하고 있음을 확인할 수 있다. Fig. 16은 가상적

으로 DS의 동작을 시험하기 위하여 최근에 주로 이용되는 스캔 공격 도구인 nmap을 이용하여 가상적인 공격을 수행하고, 이를 DS에서 탐지하여 SMS에 보고된 결과를 보여주고 있다.

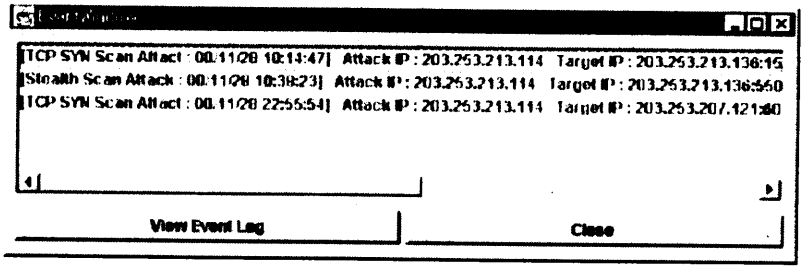

Fig. 16 Snapshot of the Detection Result

SMS로 보고되는 공격과 연관된 정보로는 탐지된 공격의 유형, 공격자와 목표시스템의 IP주소 및 port 번호 등이 포함된다. 이러한 정보를 기반으로 보안관리자는 적절한 임무를 부여한 SMA를 목표시스템으로 파견하여 호스트의 공격여부를 확인하게 된다.

2)SMA의 구현결과

SMA는 기본적인 시스템 정보 수집 및 IFS를 이용한 내부공격을 탐지하기 위한 SysInfoSMA와 주요 시스템 파일의 조사를 통한 시스템 공격여부를 탐지하는 SysFileInfoSMA가 구현이 되었다.

SysInfoSMA인 경우 주된 기능이 정보수집이므로 지정한 시스템들을 이동하면서 수집된 정보를 최종적으로 임무를 마치고 SMS로 돌아온 후 결과를 보여주도록 하였다. 동일한 세그먼트에 위치한 5대의 시스템을 이용한 시험적인 운용에서는 네트워크의 상태에 따라서 조금씩의 편차가 있지만, 약 900~1700msec 정도의 응답시간이 소요됨이 확인되었다. Fig. 17은 SysInfoSMA의 수행결과를 보여주고 있다. 그

림에서 SysInfoSMA는 지정한 시스템을 이동하면서, 각 시스템의 시스템 정보 및 IFS를 이용한 공격의 수행을 검사한 후 SMS로 돌아와서 결과를 보여주고 있다.

SysFileInfoSMA는 Solaris 운영체제가 설치된 시스템을 대상으로 동작을 수행하였으며, 정상적인 동작을 확인할 수 있었다. Fig. 18은 SysFileInfoSMA의 실행결과를 보여주고 있다. 그림에서 지정한 시스템 파일별로 파일의 수정여부를 통한 침입탐지임무 수행결과와 수행시간에 대한 정보를 알려준다. 이 과정에서 파일점검임무의 수행이전에 운영체제의 종류에 대한 검사를 먼저 수행하도록 구현하였다.

Fig. 19는 SMA의 처리된 결과가 SMA Manager의 Information and Log Window를 통하여 전시되고 있는 그림이다. SMA Manager의 View Managing Log 와 View Event Log 메뉴를 이용하여 침입탐지와 관련된 기록을 하나의 인터페이스에서 확인이 가능하도록 구현하였다.

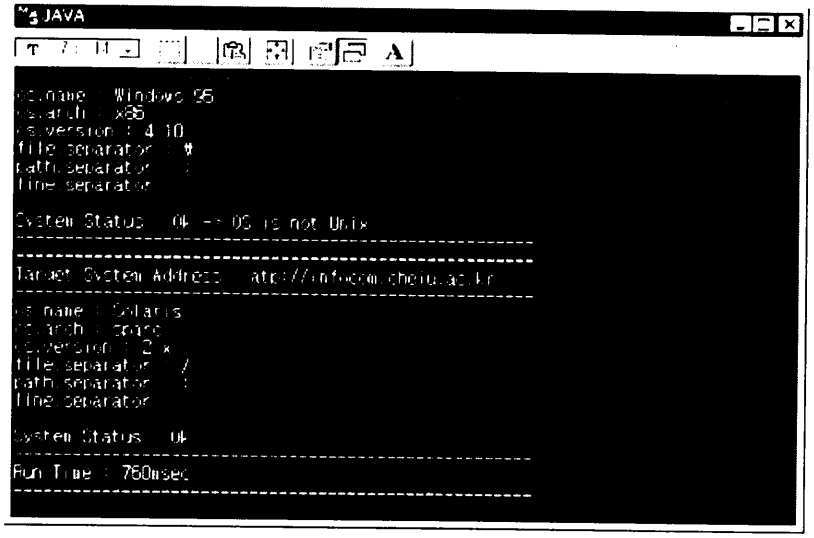

Fig. 17 The Result of the SysInfoSMA

Fig. 18 The Result of the SysFileInfoSMA

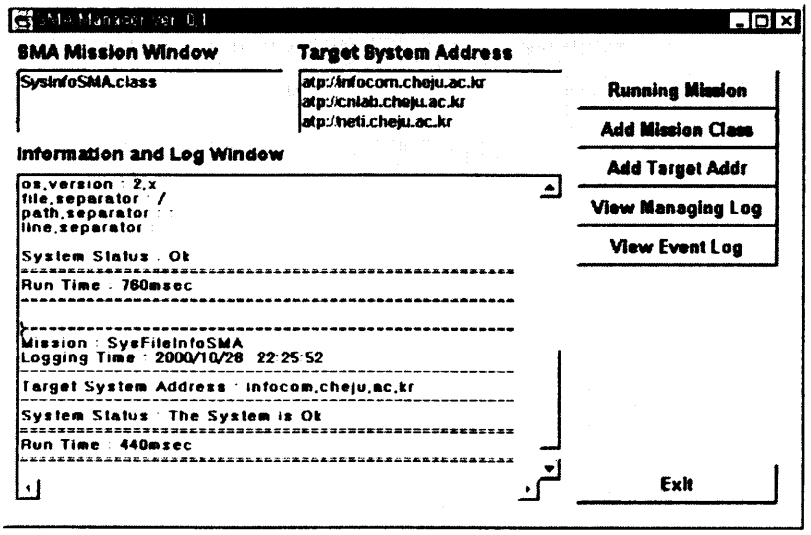

Fig. 19 The Result of the SMA Manager

지금까지 프로토타입 시스템을 통하여 설계시스템의 동작과정을 확인할 수 있었다. 일반적으로 보안관련 임무들인 경우에 시스템 의존적인 정보를 이용하는 경우가 많으므로 이를 대규모의 네트워크에 대한 통합관리 시스템으로 확장할 경우 많은 어려움이 있게된다. 이를 공격의 특성에 기인한 임무의 분담 및 이동에이전트를 이용하여 극복이 가능하다는 사실을 확인해 보았다.

V. 결 론

본 논문에서는 이동에이전트 기술을 이용한 새로운 형태의 침입탐지 시스템의 구조를 제안하였다. 시스템을 설계하기 위해서 현재 알려진 공격기법들을 바탕으로 전체 공격단계를 검색단계, 수행단계, 그리고 위장단계로 구분하였으며, 단계별 특징에 따라서 임무를 분담한 형태의 시스템을 설계하였다. 제안 시스템은 네트워크 세그먼트별로 설치된 DS에서 검색단계의 공격을 탐지하며, 수행단계와 위장단계의 공격에 대한 탐지기능은 이동에이전트에 보안관련 임무를 부여한 SMA에 부여하여 침입탐지 시스템 자체의 부하를 줄일 수 있다. 이동에이전트 기술을 이용하였으므로, 효율적으로 네트워크 대역폭을 활용할 수 있으며, 새로운 공격기법이 등장할 경우 해당 공격기법에 대한 탐지기능을 임무로 정의하여 추가하는 형식으로 침입탐지 시스템에 유연성을 부여할 수 있다.

DS를 제외한 전 시스템을 JAVA언어를 이용하여 구현하였으므로, 이질적인 네트워크 환경에서 효율적으로 운용이 가능하다. 하지만, 시스템 의존적인 특성을 갖는 보안에 관련된 정보를 JAVA언어 자체의 보안모델이라는 장벽 때문에 접근하기가 어려운 측면이 있으며, 이를 해결하기 위한 추가적인 연구가 필요할 것이다. 더불어 현재 프로토타입 시스템에서는 대표적인 공격기법 몇 가지를 선정하여 구현하였으나, 실제의 다양한 공격기법을 반영하기 위한 탐지규칙의 정형화 및 데이터베이스화에 대한 고려가 필요하다.

제안 시스템은 두개의 네트워크 세그먼트와 다수의 관리대상시스템으로 구성된 가상적인 보안관리영역에서 실제의 공격용 도구들을 이용한

환경에서 시험되었으며, 시험적인 공격수행을 통하여 정상적으로 탐지가 가능함을 확인하였다.

현재 제안 시스템의 구성요소들은 상호인증이 되었다는 가정 하에서 구현되었으며, 이동에이전트 자체에 대한 보안문제에 대해서는 고려되지 않은 상태이다. 따라서 SMA로 위장한 악성에이전트의 공격에 대해 고려할 필요가 있으며, 좀 더 완전한 시스템을 위하여 이동에이전트 및 시스템 구성요소 간의 인증 메커니즘에 대한 연구가 필요할 것이다. 더불어 SMS와 DS간의 효율적인 정보교환을 위한 프로토콜에 대한 정의도 고려할 필요가 있을 것이다.

참고문헌

Aglets Workbench, http://www.trl.ibm.co.jp/aglets

Anderson, James P. 1980, Computer Security Threat Monitoring and Surveillance Technical Report, James P. Anderson Co., Fort Washington, PA, April 1980

Y. Aridor, Danny B. Lange, 1998, Agent Design Patterns:Elements of Agent Application Design, Proceedings of Autonomous Agents'98, ACM Press, 1998

M. Asaka, S. Okazawa, A. Taguchi, and S. Goto, 1999, A Method of Tracing Intruders by Use of Mobile Agent, Proceedings of the 9th Annual Internetworking Conference (INET'99), San Jose, California, June 1999.

A. Bieszczad, B. Pagurek, T. White, 1998, Mobile Agents for Network Management, IEEE Communications 1998 Vol. No. 1

CERT, http://www.cert.org/

CERT-kr, http://www.certcc.or.kr/

COAST, http://www.cerias.purdue.edu/coast/

P. Dasqupta, N. Narasimhan, Louise E. Moser, P. M. Melliar-Smith, 1999, MAgNET:Mobile Agents for Networked Electronic Trading, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, Vol. 11, No. 4, July/august 1999

D. Denning, 1986, An Intrusion Detection Model. Proceedings of the Seventh IEEE Symposium on Security and Privacy, May 1986

G. Helmer, Johnny S. K. Wong, V. Honavar, and L. Miller. 1998, Intelligent Agents for Intrusion Detection. Proceedings, IEEE Information Technology Conference, Syracuse, NY, pp 121-124, September 1998

Hughes, 1995, Actually Useful Internet Security Techniques. New Riders

G. Karjoth, Danny B. Lange, M. Oshima, 1997, A Security Model for Aglets, IEEE Internet Computing Vol. 1, No. 4

S. C. Kim, Y. S. Choi, J. W. Chung, 1999, Study of Security Management System based on Client/Server model, Proceedings of the 1999 IEEE International Conference on Communications, Vol. 2, 1403-1408, 1999

L. Korba, 1999, Towards Securing Network Management Agent Distribution and Communication, INM VI

Danny B. Lange, M. Oshima, 1998, Programming and Deploying Mobile Aglets with Java, Addison-Wesley Pub.

Danny B. Lange, M. Oshima, 1999. Mobile Agents with Java:The Aglet API

Morris Sloman, 1996, Network and Distributed Systems Management, ADDISON-WESLEY

Nmap, http://www.insecure.org/nmap/

R. G. Bace, 2000, Intrusion Detection. MACMILLAN TECHNICAL PUBLISHING

SRI International, http://www.sdl.sri.com/intrusion/index.html

Wayne Jansen, Peter Mell, Tom Karygiannis, Don Marks, 1999, Applying Mobile Agents to Intrusion Detection and Response. NIST Interim Report(IR)-6416, October 1999

Wayne Jansen, 1999, Countermeasures for Mobile Agent Security, NIST

W. Jansen, P. Mell, T. Karygiannis, D. Marks, 2000, Mobile Agents in Intrusion Detection and Response, NIST

K. Siyank, C. Hare, 1995, Internet Firewalls and Network

Security, New Riders Pub.

William Stallings, 1995, Network and Internetwork Security Principles and Practice, Prentice-Hall Inc.

William Stallings, 1993, SNMP SNMPv2 and CMIP, ADDISON-WESLEY

William Stallings, 1996, SNMP SNMPv2 and RMON, ADDISON-WESLEY

G. Vigna, R. A. Kemmerer, 1998, NetSTAT:A Network-based Intrusion Detection Approach, Proceedings of the Fourteenth Annual Computer Security Applications Conference, P25-34, 1998

방성민, 송왕철, 2000, 이동에이전트 기술을 적용한 침입탐지 시스템, 한국통신학회 추계종합학술발표회 논문집, pp.1495-1498

J. S. Park, W. C. Song, 1998, CORBA Traffic Monitoring, Proceedngs of the APCC'98/ICSS'98

G. S. Kim, W. C. Song, 1999, CORBA-based OSI Network Management, Proceedings of the ICT'99, Vol. 2, pp.319-322