공개 보안 도구를 이용한 리눅스 기반 통합 보안 시스템의 설계 및 구현

전 용 희[†] · 김 민 수[††] · 장 정 숙[†††]

요 약

광범위한 인터넷의 보급에 따라 컴퓨터 및 네트워크 시스템의 취약점을 이용한 해커들의 통신 웹을 경유한 공격이 쉽게 되었다. 본 논문에서는 통합 보안관리가 가능한 LISS(Linux-based Integrated Security System)라고 이름 붙여진 통합 보안 시스템을 설계하고 구현한다. 이 시스템은 공개 운영 체제인 Linux에 기반하고 있으며, Linux 기반 서버의 보안관리에 효과적인 공개 보안 도구들로 구성되어 있다. LISS의 성능을 시험하기 위하여 테스트베드를 구성하여 시험한 결과, 구현된 시스템이 네트워크 매퍼로부터 발생되는 모든 공격 패턴들을 포획하는 것으로 나타났다.

Design and Implementation of Linux-based Integrated Security System(LISS) Using Open Security Tools

Yong-Hee Jeon[†] · Min-Soo Kim[††] · Jung-Sook Jang[†††]

ABSTRACT

The wide spread of Internet makes susceptible to the attacks via communication Web from hackers using the vulnerability of both computer and network systems. In this paper, we design and implement an integrated security system, named as LISS(Linux-based Integrated Security System) in which an integrated security management is possible. This system is based on the open operating system, Linux and consists of open security tools, which is effective in security management of Linux based-servers. We also construct a test-bed in order to testify the performance of the LISS. It is revealed that the implemented system captures all the attack patterns generated from Network Mapper.

키워드 : ESM(Enterprise Security System), 통합보안시스템(Integrated Security System), 공개보안도구(Open Security Tool), 리눅스(Linux), 보안 관리(Security Management)

1. 서 론

인터넷의 급속한 확산으로 컴퓨터와 네트워크 시스템의 취약점을 이용한 통신망을 통한 해커들의 공격에 무방비 상태에 놓이게 되어, 엄격한 보안이 요구되는 비밀 자료나 시스템에 대해서 불법침해 사례가 급증하고 있으며 이들에 대한 보안 문제를 해결하기 위한 방안이 시급히 요구되고 있다[1]. 특히, 지난 3년간 국가·공공기관 전산망의 해킹과 바이러스 침해사고 중 대부분이 교육기관에 집중돼 보안기능 강화가 절실하다. 국가정보원이 국정감사 자료로 제출한 '2002 해킹사고 사례분석'을 보면 국가·공공기관 전산망의 해킹과 바이러스 침해사고가 지난 2000년 102건, 2001년 277건, 2002년 539건으로 3년간 매년 2배 이상 증가한 것을 알 수 있다. 2002년 기준으로, 기관별로 보면 교육청·국공립 대학 등 교육기관이 369건(68%), 지방자치단체가 81건(15%), 정부산하기관이 31건(6%), 중앙행정부처가 22건(4%), 기타 36건(7%) 등으로 분류된다. 국공립 대학 등 교육기관은 전산망 규모에 비해 보안관련 예산 및 시스템 보안 담당자가 부족하고 운영 미숙과 그에 따른 보안 대책 수립이 미흡한 것이 문제라고 볼 수 있다[2]. 또한 관리자가 보안도구를 설치하였다 하더라도 보안도구에서 나오는 엄청난 양의 감사 자료를 분석하는 것이 쉬운 것이 아니다. 결과적으로, 침해사고를 미연에 방지할 수 있음에도 불구하고 감사 자료에서의 중요한 정보를 대부분 놓치는 경우가 허다하다. 그리고 보안 도구가 여러 종류로 나눠짐에 따라 관리자는 시간을 더 많이 투자해야 된다. 따라서 감사 자료를 한곳에서 필요한 정보만을 보여줄 수 있게 하는 도구의 필요성이

† 종신회원 : 대구가톨릭대학교 컴퓨터·정보통신공학부 교수
†† 정 회 원 : 경북직업전문학교 교수
††† 준 회 원 : 대구가톨릭대학교 컴퓨터·정보통신공학부 IT 교수
논문접수 : 2004년 4월 7일, 심사완료 : 2004년 6월 24일

대두 되었다.

이에 따라 본 논문에서는 전문적이고 세분화되고 있는 보안 제품 개발과 공공기관의 공개 소프트웨어 본격 도입 추세에 따라 공개용 운영체제인 리눅스를 기반으로 공개용 보안 도구 툴을 통합한 통합보안 시스템인 LISS(Linux-based Integrated Security System)를 설계 및 구현함으로서 리눅스 기반 서버의 보안관리에 있어서 효과적임을 보인다. 최근 국내의 교육행정정보시스템으로 NEIS(National Education Information System)의 도입 계획에 따라 개별 학교 전산망의 보안 문제가 제기되고 있다. 현재 중등학교 학사관리시스템은 대부분 C/S(클라이언트/서버) 체제이며 보안 관리자를 학교별로 배정해야 하기 때문에 예산이 많이 들고 보안이 취약한 점이 있다. 그러나 현재 실정으로 중등학교 환경에서 상용제품의 전사적 보안 관리(ESM : Enterprise Security Management) 시스템을 도입하는 것은 경제적인 부담으로 작용하고, 고성능의 시스템이 굳이 아니더라도 상당 부분 보안 문제를 해결할 수 있다고 판단되기 때문에, 본 논문에서는 학교의 전산 환경을 염두에 두고, 공개 보안 도구를 이용한 통합보안시스템을 개발하고자 한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 국내 해킹 동향, 학교 정보보안의 문제점을 기술하고, ESM의 정의와 역할, 그리고 특징과 분류에 대해서 기술한다. 제 3장에서는 리눅스 기반 공개용 보안 도구에 대하여 분석, 기술한다. 제 4장에서는 각 보안 도구들의 감사 파일을 기반으로 한 LISS를 설계 및 구현하고, 제 5장에서는 구현된 LISS에 대하여 다양한 공격방법을 통하여 성능 분석 및 평가를 한다. 마지막으로 제 6장에서 결론을 맺는다.

2. 관련 연구

2.1 피해 운영체제와 공격 수법

2001년 우리나라에서 운영하고 있는 운영체제별 피해현황을 보면 전체 324건 중에서 리눅스와 Windows '95/98이 모두 95건으로 상대적으로 많은 것으로 나타났다. 일반적으로 Windows '95/98의 사용자수에 비해 리눅스는 월등히 피해가 많은 것으로 나타났는데, 그 이유로는 리눅스의 시스템 운영미숙에 의한 것과, 해킹 대응 보안 전문 인력의 부족이라고 분석된다[3].

2003년 7월까지 기준으로, 공격 수법에 따른 건수를 보면 전체 25,836건의 침해 건수 중에서 구성설정오류가 8,575건으로 가장 많고, 그 다음에 E-mail 관련 침해(5,786건), 악성프로그램(4,814건), 취약점 정보수집(4,301건) 등이 있다. 같은 해 월별 침해 유형에 따른 건수를 보면, 전체 16,782건 중에서 불법자원사용(8,145건), 침입시도(4,295건), 불법침입(4,289건)이 대부분을 차지하고 있다[4]. 일반적으로 이러한 피해 통계는 보고 된 사례만을 근거로 하기 때문에 실제로는 훨씬 많은 침해가 있다고 볼 수 있다. 그리고 하나의 사고에 여러 가지 해킹수법이 사용되는 경우나 또는 공격자가 관련 로그 등을 모두 삭제하여 정확한 해킹수법을 파악할 수 없는 경우도 있다. 이러한 해킹의 결과로 해당 시스템의 파괴, 자료유출 및 변조 등뿐만 아니라 소속 네트워크 전체까지 영향을 미치고 있다.

2.2 학교 환경에서 보안 문제점 및 연구배경

학교현장에서 인터넷의 연결과 개인용 컴퓨터 및 학사업무 관련 서버가 매우 많이 보급되어 있지만, <표 1>에서 보여주는 바와 같이 그에 따른 활용 및 보안 관리체계가 미흡한 것이 심각한 문제로 대두되고 있다[4].

〈표 1〉 학교 환경에서 보안 문제점

관리 측면	◦ 교육정보화를 위한 총체적인 정책 부재 ◦ 교육정보시스템 보안 운영에 필요한 보안 지침 부재 ◦ 보안 문제 발생시 대응조치 지침 부재
기술적 측면	◦ 네트워크 및 시스템의 각 요소별 보안 대책 미비 ◦ 침입 차단 시스템 등 보안 시스템의 운영미숙 ◦ 별도 인터넷망 사용에 대한 보안 대책 미비
인적 측면	◦ 해킹 대응 보안 전문인력 부족

이와 같은 문제점을 개선하기 위하여 관리 측면으로서 교육정보시스템 보안 운영에 필요한 보안 지침과 그에 따른 문제 발생시 대응 조치에 대한 지침이 있어야 하고 담당 교사들과 정보 보안 관련 기관과의 정보교환이 활성화가 되어야 할 것이다. 또한 기술적인 측면에서 보면 시스템 운영에 대한 교육이 별도로 필요하며 부가적으로 보안 관련 툴의 운용에 대한 전문적인 교육이 필요하다. 그리고 보안 관련 툴의 설정과 실행을 한 곳에서 관리하고 사용이 편리한 중앙 집중적인 통합보안관리 시스템을 도입한다면 보안관련 인력들의 대부분의 시간을 로그 분석하는데 보내지 않고 보안 대책과 대응에 따른 보안 정책을 마련하는데 많은 시간을 투자할 수 있을 것이고 그에 따른 기술적 측면과 인적측면의 보안 문제점을 해결 할 수가 있을 것이다.

2.3 ESM

ESM(Enterprise Security Management)이란 전사적 통합보안관리 시스템으로서 광의적 정의로 IT자산 인프라에 대한 가용성, 무결성, 보안성을 보장하기 위한 위험관리이며 협의적 정의로는 이기종 보안 시스템의 효율적 운영(정책관리, 침해대응, 이벤트 통합관리), 분석 등을 통한 사전적 사고예방을 하기 위한 시스템이다.

ESM은 다음과 같이 크게 2가지 범주로 분류할 수 있다.

2.3.1 사용자 및 정책 관리

보안 또는 관리정책에 따른 사용자 및 접근 관리에 무게 중심을 가지고 있는 범주이다. 이 범주에는 인증이나 Single Sign-On의 기능을 포함하는 경우가 많다. 또한 이러한 유형에는 초기 ESM 모습이 많이 반영되어 보안적 측면보다는 시스템 관리적 측면의 성격이 강하다.

2.3.2 취약성 및 위협 평가

네트워크 및 시스템의 취약점, 위험요소들을 분석하고 모니터링 하는 관리도구의 형태를 취하며 제품에 따라 분석 또는 정책관리, 모니터링 및 경보(alert) 등 어느 쪽에 무게를 두느냐에 따라 특성을 약간씩 달리하고 있다. 이와 같은 형태가 최근의 ESM 주류를 이루고 있으며 기존 보안 제품들과의 통합(integration)이 활발히 진행되는 범주이다.

ESM은 침입차단시스템(즉, 방화벽), 침입탐지시스템(IDS : Intrusion Detection System), 가상사설망(VPN : Virtual Private Networks) 등 다양한 종류의 솔루션을 하나로 모은 통합 보안 관리 시스템이다. 최근 한국 정보보호 산업협회 산하의 인터넷보안기술포럼에서 추진하고 있는 정보보호 표준화 활동 중에도 보안 제품 간 인터페이스 표준을 준비하고 있고 업체들간의 컨소시엄인 SAINT에서도 표준 인터페이스를 개발하는 등 국내 ESM 환경은 급속도로 발전하고 있다[5].

3. 공개용 보안 도구 분석

3.1 네트워크 침입탐지시스템

일반적으로 IDS는 감사 자료를 어디로부터 얻는가에 따라서 NIDS(Network-based IDS)와 HIDS(Host-based IDS)로 나뉜다. NIDS는 네트워크 패킷을 캡처하여 감사 자료로 사용하며 침입탐지 여부를 분석한다. 구현이 쉽고 하나의 시스템으로 여러 시스템을 보호할 수 있는 등의 많은 장점 때문에 현재 상업용 제품이 가장 많이 나와 있는 침입탐지 모델이다[6-8].

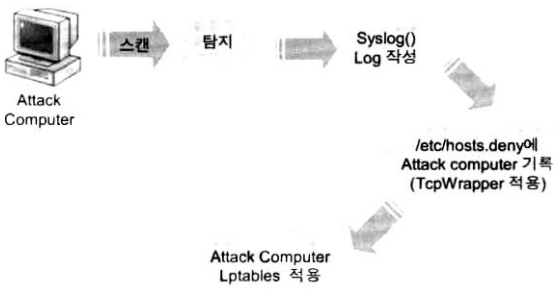
3.1.1 Snort

Snort는 네트워크에서 실시간 트래픽 분석과 패킷 로깅을 뛰어나게 수행하는 네트워크 침입 탐지 시스템(NIDS)이다. Snort의 시스템 요구사항으로는 우선 Snort의 탐지 엔진이 규칙 위반 검사에 사용하는 모든 데이터를 저장해야 되기 때문에 용량이 큰 하드 디스크가 필요하다. 운영체제로는 거의 모든 OS에 실행이 가능하나 Linux에 가장 적합하다. Snort를 Linux에서 설치하기 전에 automake, gcc, lex와 yacc, libpcap 등이 설치 되어있어야 가능하다[9, 10].

트래픽을 잡아내기 위해 유용한 모듈화된 플러그인 구조를 가진 탐지 엔진과 매우 유연한 규칙 언어를 사용한다. Snort는 시스템로그, 사용자가 지정한 특정 파일, 유닉스의 소켓이나 삼바 클라이언트를 이용한 윈도우 팝업 메시지와 일체된 경보 메커니즘과 같은 실시간 경보가 가능하다.

3.1.2 PortSentry

PortSentry는 Abacus 프로젝트 도구프로그램중의 하나이다. Abacus 프로젝트는 인터넷 공동체에 유지비용이 적게 들며, 포괄적이고 믿을 만한 호스트 기반 침입탐지 소프트웨어를 배포한 선구자이다[11, 12]. PortSentry는 실시간으로 특정 서버에 대한 포트스캔을 감지하고 이에 대응하기 위해 설계된 프로그램으로서 포트스캔을 감지하기 위해 많은 옵션을 제공한다. 포트스캔이 감지되면 (그림 1)과 같은 방식으로 응답한다.



(그림 1) PortSentry 동작 과정

① syslog()가 로그에 경고를 출력한다.
② 목적지 호스트는 자동으로 누락된다.
③ 로컬호스트를 자동으로 재설정하고 모든 트래픽을 실제 라우팅되지 않는 호스트로 보 내어 공격지 시스템이 보이지 않도록 한다.
④ 로컬 호스트는 공격지로부터 오는 모든 패킷을 로컬 패킷 필터링을 통해 누락 시킨다.

PortSentry는 크게 3가지 부분, 여섯 개의 모드로 동작하며, 하나의 프로토콜 모드만이 실행가능하다. 즉 하나의 TCP 모드와 하나의 UDP 모드로 실행한다. 스텔스 모드와 어드벤스 모드는 리눅스 시스템에서만 사용이 가능하다.

PortSentry의 로그 메시지는 3가지로 나타나며, 관리경고(Admin alert)는 PortSentry의 상태를 나타내는 메시지이며, 보안경고(Security alert)는 보안 관련 사건들이 일어났음을 나타내는 메시지, 그리고 공격경고(Attack alert)는 스캔을 감지했다는 것과 그에 따른 동작이 실행되었다는 메시지다.

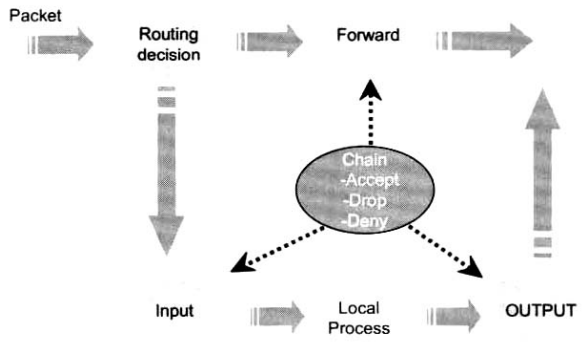
3.1.3 iptables

리눅스 커널 2.4에서는 이전의 커널 버전에 비해 방화벽 구축과 네트워크 패킷 필터링(netfilter)을 하기 위해 새로운

기법의 방화벽 프로그램을 제공한다. iptables라는 툴로 제어되는 이 새로운 기법은 이전의 ipchains에 비해 세련되고 보안 측면에서도 더욱 효율적이다. iptables는 설정하기 쉬울 뿐만 아니라 리눅스에서는 처음으로 상태 추적기법이 도입된 방화벽인데, 이것은 이전의 리눅스 기반의 방화벽에서는 탐지가 불가능했던 스텔스 스캔을 탐지 및 차단하는 등 상당한 기술적 진보를 이룬 지능형 방화벽을 뜻한다. 또한 iptables는 방화벽을 통과하는 각각의 연결을 메모리에 저장하므로 율 제한(rating limiting)을 이용하여 대부분의 서비스 거부 공격도 차단할 수 있다[13, 14].

iptables는 테이블 형식으로 관리를 한다. 그리고 먼저 등록된 것이 효력을 발생하기 때문에 등록하는 순서가 중요하다. 모든 것을 거부하는 설정이 먼저 오게 되면 그 이후에 포트를 열어주는 설정이 와도 효과가 없다. 그러므로 허용하는 정책이 먼저오고 나서 거부하는 정책이 와야 한다.

기본적으로 iptables에는 세 가지 chain이 있고 모든 패킷은 이 세 가지 chain중 하나를 통과하게 된다. 이 세 가지 chain은 INPUT, OUTPUT, FORWARD chain인데 우선 관리자가 관리하는 컴퓨터로 들어가는 모든 패킷은 INPUT chain을 통과한다. 그리고 관리자가 관리하는 컴퓨터에서 나가는 모든 패킷은 OUTPUT chain을 통과한다. 그리고 하나의 네트워크에서 다른 곳으로 보내는 모든 패킷은 FORWARD chain을 통과한다. iptables가 작동하는 방식은 이들 각각의 INPUT, OUTPUT, FORWARD chain에 관리자가 어떠한 rule을 세우는 지에 따라 달라진다. (그림 2)는 iptables의 동작방식이다.



(그림 2) iptables의 동작 방식

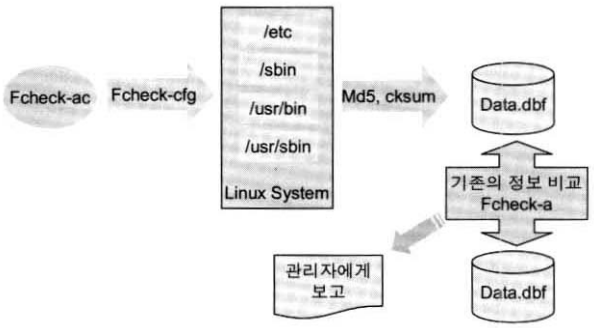
3.2 호스트 침입탐지시스템

HIDS는 호스트의 내부에서 얻을 수 있는 감사 자료를 수집해서 침입 탐지를 수행하는 IDS를 말한다. HIDS는 호스트에서 감사 자료를 얻으므로 자기 자신 이외의 컴퓨터의 침입을 탐지해 낼 수 없다. 일반적으로 보호하고자 하는 호스트 내부에서 동작을 하기 때문에 NIDS와는 다른 장단점을 가지고 있다. HIDS의 경우는 많은 감사 자료가 있으므로 제품들 또한 각기 다른 기능을 가지고 있다. 예를 들면, 백신 프로그램이나 무결성 검사 도구인 Tripwire같은 것을 들 수 있는데 이 두 제품이 하는 일은 다르지만 호스트의 정보를 바탕으로 침입을 탐지해 내기 때문에 HIDS로 부를 수 있다.

3.2.1 Fcheck

일반적으로 파일 무결성 체크 프로그램으로 Tripwire가 알려져 있지만, 많은 기능을 제공하는 만큼 설정이 너무 복잡하다. 간단한 설정과 강력한 기능을 가진 Fcheck는 특정 디렉터리나 파일 또는 전체 파일 시스템에 대해 파일의 추가, 변경, 삭제여부를 체크하여 그 결과를 알려 준다[15, 16]. (그림 3)은 Fcheck의 동작 방식이다.

Fcheck.cfg에 정의되어있는 디렉터리를 체크섬(checksum)이나 MD5(Message Digest 5)를 이용하여 Data.dbf란 파일에 해당 정보를 기록한 후 일정 시간 이후에 다시 정의되어있는 디렉터리를 체크하여 기존의 Data.dbf의 파일과 비교하여 무결성을 검사한다.



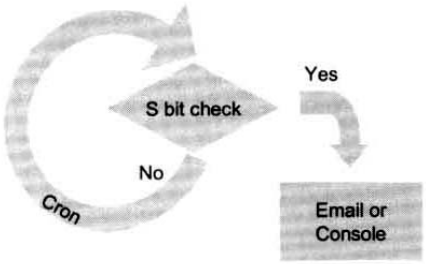
(그림 3) Fcheck 동작 방식

3.2.2 sXid

SUID/SGID 파일은 보안에 위험한 파일이다. 이러한 보안상의 위험을 줄이기 위하여 대부분의 관리자들은 루트 소유의 's'비트가 설정된 프로그램 중 's'비트가 필요하지 않은 파일을 삭제한다. 그러나 그 이후에도 서버 관리자가 인식하지 못하는 사이에 기존 파일이나 새로운 파일에 's' 비트가 설정될 수도 있다[13, 17].

(그림 4)와 같이 sXid는 cron을 이용하여 정기적으로 시스템의 모든 SUID/SGID가 설정된 파일이나 디렉터리를 모니터링 하는 프로그램으로서 기본적으로 시스템 내에서 SUID나 SGID가 설정된 디렉터리 또는 파일에 어떤 변화가 있는지 체크한다. 이전에 's'비트가 설정되지 않았던 파일이나 디렉터리에 새롭게 's'비트가 설정되거나 또는 비트나 다른 모드로 변경되었을 경우 sXid는 그 내용에 대해 쉽게 알 수 있도록 이메일이나 다른 명령 라인으로 이를 보

고한다. sXid는 시스템에서 SUID/SGID가 있는 모든 파일이나 디렉터리를 자동으로 찾아서 보고해 준다. 일단 설치가 되면 이후에는 프로그램이 자동으로 모든 일을 처리하므로 일일이 신경 쓰지 않아도 된다.



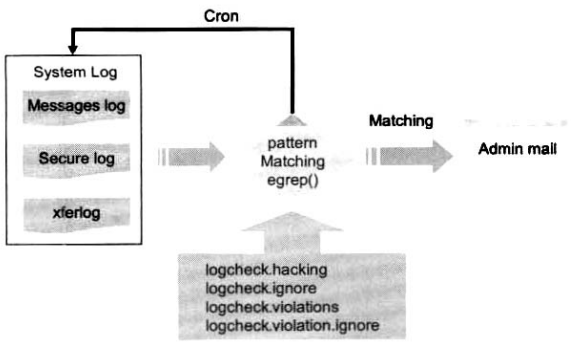
(그림 4) sXid 동작과정

3.3 기타 보안도구

3.3.1 Logcheck

Logcheck는 보안 위반 사항이나 비정상적인 행위를 발견하기 위해 시스템 로그 파일들을 자동으로 점검해주는 패키지이다. 이 패키지는 로그 파일 중 마지막에 읽은 위치를 기억해 주는 logtail이라는 프로그램을 사용하는데, 다음번 점검 시 중복된 검사를 피하기 위해 사용되어진다[11, 18].

Logcheck는 자동으로 프로세스를 감시하고 로그에 저장된 정보를 기반으로 현재의 문제점과 이후 발생 가능한 문제점을 한 눈에 볼 수 있도록 메일로 발송한다. Logcheck는 시스템의 침입과 비정상적인 활동에 대해 시스템의 로그 파일을 자동으로 체크하고 작동하기 위한 간단한 셸 스크립트로써 cron에 의해서 최소한 몇 시간마다 실행하여 로그에서 공격 패턴을 찾아내는 것이 안전하다. 이 스크립트는 grep이라는 명령어를 사용하여 공격이나 비정상적인 행위에 의해 남겨지는 키워드를 매칭시켜 시스템 관리자에게 발견된 사항들을 메일로 보고 한다.



(그림 5) Logcheck 동작과정

(그림 5)와 같이 주기적으로 검사를 해서 보안에 대한 이상 징후를 발견하게 되면 관리자에게 메일을 보내 경고를 하지만, 검사 시에 특별한 이상이 없을 경우 메일을 보내지 않고, logtail 프로그램에 의해 이전에 검사한 로그는 다시 검사하지 않으므로 검사주기를 짧게 한다고 해서 메일이 폭주하지는 않는다.

3.3.2 OpenSSL

IMAP(Internet Mail Access Protocol)과 POP(Post Office Protocol), SSH(Secure Shell), Samba, Sendmail, OpenLDAP, FTP, Apache 같은 소프트웨어와 서버는 서비스를 제공하기 전에 사용자에 대한 인증을 요구하고 기본적으로 입력받은 사용자의 아이디와 암호를 평문형태로 전달한다. 이 과정의 대안으로서 SSL(Secure Socket Layer)같은 암호화 메커니즘은 안전하고 보안성 있는 전송을 보장한다. 이런 기술은 네트워크를 통해 전달되는 데이터를 각각의 종단에서 암호화 한다. 일단 리눅스 서버에 OpenSSL(Open Secure Socket Layer)이 설치되면 다른 응용프로그램에서 SSL기능을 부여 할 수 있는 부가 프로그램으로 이용할 수 있다[19, 20].

OpenSSL 프로젝트는 완전한 기능을 갖춘 강력한 상용수준의 개발을 위해 공동으로 연구하는 프로젝트이며 강력한 암호를 작성하기 위해 Secure Sockets Layer와 Transport Layer Security를 사용한다.

4. 시스템 설계 및 구현

4.1 개발 환경

리눅스기반 공개용 보안 도구의 통합관리 시스템을 구현하기 위한 개발 환경은 <표 2>와 같다.

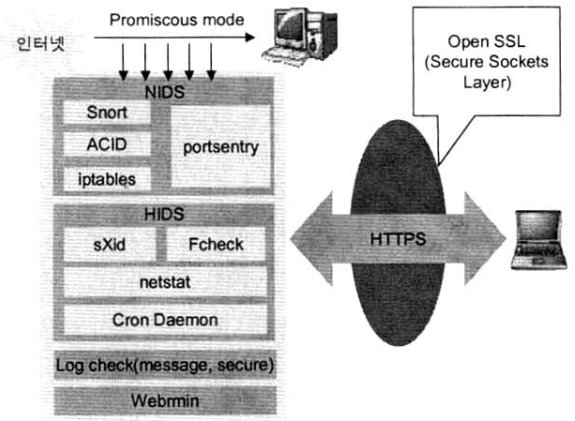
〈표 2〉 LISS 구현 환경

구 분	프 로 그 램
운영체제	Linux Kernel 2.4.7-10 on i686
개발언어	Bash Shell, PHP-4.0.6, GNU's gcc 2.96-98, Perl-5.6.0
데이터베이스	MySQL-3.23.38
웹 서버	Apache_1.3.22,OpenSSL-0.96g, mod_ssl-2.8.5-1.3.22
침입탐지도구	snort-2.0.2, portsentry-1.1, oinkmaster-0.8
감사도구	logcheck-1.1.1
방화벽	iptables-1.2.3-1
무결성 체크	Fcheck-2.7.59, sXid_4.0.4
시스템 설정 도구	webmin-1.110

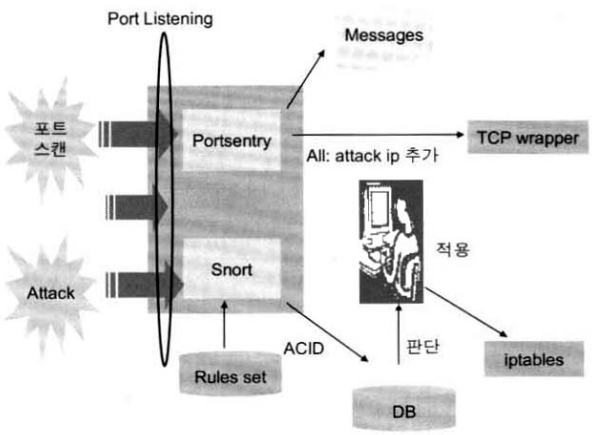
4.2 LISS 구조도

최근 침입탐지시스템이 침입탐지 효율을 좋게 하기위하여 네트워크 기반 침입탐지시스템과 호스트 기반 침입탐지시스템의 혼합형을 사용하는 추세에 따라, LISS에서도 HIDS

와 NIDS의 모듈들을 통합하여 (그림 6)과 같이 네트워크 보안과 호스트보안을 동시에 관리할 수 있도록 하였다. 위 부분에 NIDS 기능 수행을 위한 모듈들이 있고, 그 아래에 HIDS 모듈들을 탑재하였다. 먼저 NIDS에서 네트워크 상의 패킷을 분석하여 침입 탐지 및 대응을 수행한다. 특정 호스트에 대한 침입 탐지를 위하여 HIDS에서는 파일의 추가, 삭제, 변조 등의 파악과 시스템로그를 분석한다. 또한 Webmin으로 시스템의 사용 환경 및 웹 서버와 데이터베이스, Logcheck, PortSentry등의 도구들을 설정할 수 있도록 하였다.



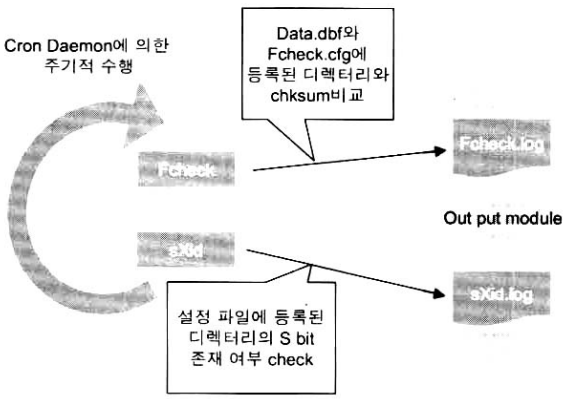
(그림 6) LISS의 구조



(그림 7) Snort와 PortSentry 및 FireWall 실행 과정

해커의 침입시도에 따라 Scan일 경우 PortSentry는 탐지뿐만 아니라 그에 따른 대응 즉, TcpWrapper로 접근제어를 하는 등의 방화벽 기능을 수행한다. Snort는 정해진 룰셋(rule set)에 의하여 프로토콜 분석과 패킷의 내용을 조사하고 패턴 매칭을 통하여, 버퍼 오버플로우나 스텔스 포트스캔, CGI(Common Gateway Interface) 공격, SMB(Server Message Block) 탐지, OS(Operating System) finger-printing 시도 등 다양한 공격의 탐지를 수행한다. Snort가 침입탐지를 하면 관리자가 LISS를 통하여 iptables에 해당 공격자 호스트에 대한 추가여부를 판단하여 등록하게 된다((그림 7) 참조). 또한 iptables는 방화벽을 통과하는 각각의 연결을 메모리에 저장하므로 율 제한(rating limiting)을 이용하여 대부분의 서비스 거부 공격도 차단할 수 있다.

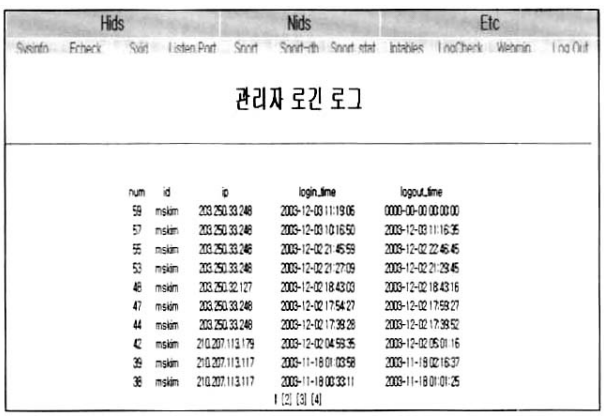
(그림 8)은 Fcheck와 sXid의 실행 과정을 나타내고 Fcheck와 sXid는 Cron Daemon에 의하여 주기적으로 수행되며, 이를 위하여 /etc/cron.daily에 sXid와 fcheck.cgi를 스크립트로 작성 후 추가한다((그림 8) 참조, Fcheck와 sXid의 세부적인 동작 설명을 위해서는 각각 (그림 3), (그림 4)를 참조할 수 있다). Data.dbf와 Fcheck.cfg에 등록된 디렉터리와 Checksum 비교를 통하여 Fcheck는 파일의 추가, 변경, 삭제여부를 체크하여 그 결과를 알려 주고, sXid는 설정 파일에 등록된 디렉터리의 s 비트 존재 여부와, SUID나 SGID가 설정된 디렉터리 또는 파일에 어떤 변화가 있는지 체크하여 보고한다.



(그림 8) fcheck와 sXid 실행 과정

4.3 LISS 구현

4.3.1 메인화면

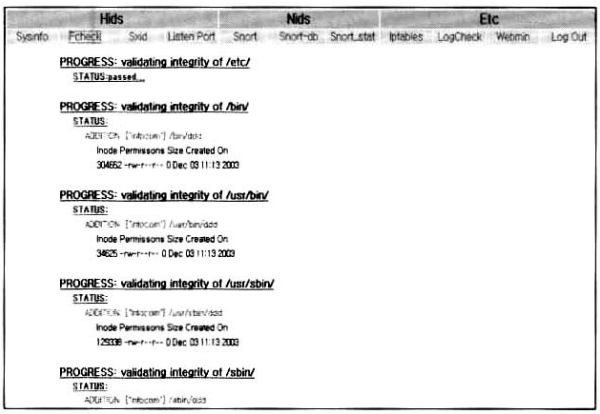


(그림 9) LISS에 접속한 메인 화면

로그인을 하면 (그림 9)와 같이 그 동안의 관리자가 언제 로그인을 하고 로그아웃을 하였는지를 접속한 ID와 접속한 컴퓨터의 IP, 로그인 시각, 로그아웃시각을 세션에 저장하여 그 세션의 내용을 로그인과 로그아웃 할 때에 데이터베이스에 자동으로 입력되게 하여 다른 사람이 로그인을 하였는지를 로그인, 로그아웃 시간을 통하여 알 수 있도록 하였다. 관리자가 로그인한 시각이 아니라면 바로 비밀번호를 변경할 수 있도록 하여 제 2차 접속을 막기 위한 방안으로 설계하였다.

4.3.2 Fcheck 실행 예

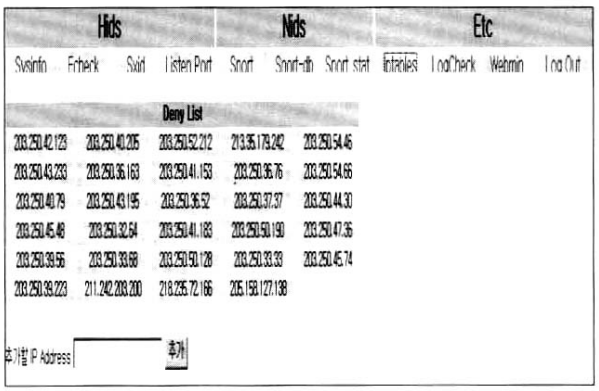
(그림 10)에서는 LISS가 설치된 컴퓨터에서의 Fcheck의 설정되어 있는 디렉터리의 파일의 추가, 변경, 삭제 등 변경이 있을 경우에 그 변경사항을 각 설정된 디렉터리와 해당 파일이름, 변경시간, 파일의 접근 권한, 파일의 크기를 파악할 수 있도록 설계하였다. 관리자가 해당 설정된 디렉터리에서 파일을 추가, 변경, 삭제 한 것이 아니라면 침입자에 의해 변경되어있을 수 있기 때문에 관리자는 확인 후 그 시각에 접속 또는 접속을 시도한 것을 logcheck를 통하여 확인하고 확인된 IP를 iptables에 등록하면 2차 침입을 막을 수 있게 설계하였다. 그리고 변경사항이 없을 경우에는 필요한 정보만을 표시하기 위해서 표시되지 않게 하였다.



(그림 10) Fcheck 로그화면

4.3.3 sXid 실행 예

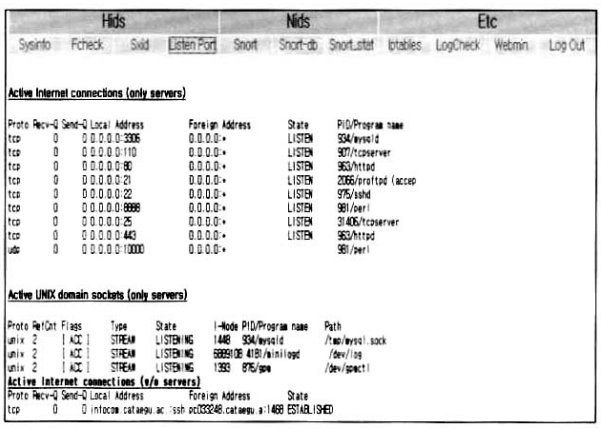
(그림 11)은 sXid가 Cron데몬에 의해 설정된 디렉터리 내의 파일의 권한 중에서 SUID/SGID를 체크한 결과로 Webmin을 설치하고 나서 확인한 화면이다. /usr/local/src/ESM/openssl-engine-0.9.6g/의 하위 디렉터리에 접근모드가 SUID/SGID로 되어있는 파일 또는 디렉터리를 한눈에 볼 수가 있다. SUID/SGID를 체크한 시간과 체크한 호스트 이름, 디렉터리를 보여주며, 해당파일의 접근모드와 소유자와 소유그룹, 그리고 위치를 표현하였으며 관리자가 판단하여 해당 파일의 접근모드에 대한 변경을 유도하도록 하였다. SUID/SGID는 보통 프로그램을 설치할 경우 생길 수 있기 때문에 관리자가 유의하지 않으면 불법침입자가 쉽게 SUID/SGID가 설정되어 있는 파일을 찾아 실행을 통해서 루트의 권한을 획득할 수 있다. 이러한 상황을 미연에 방지하기 위해서 웹에서 쉽게 확인할 수 있도록 구현하였다.



(그림 11) sXid 화면

4.3.4 Netstat 실행 예

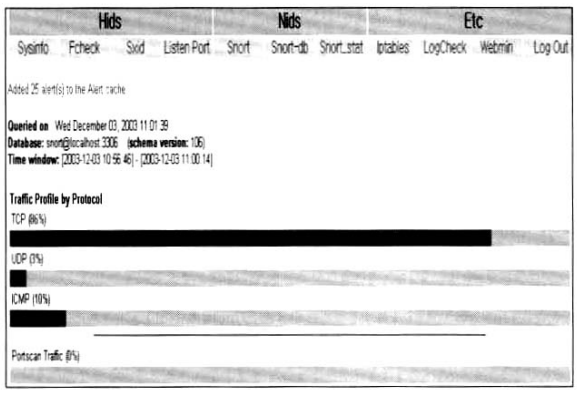
(그림 12)는 현재 호스트 시스템에서 /etc/services에 명시되어있는 포트에 대해 서비스의 요청을 대기하고 있는 데몬 및 포트정보, 소켓들을 보여주고 있는 화면으로 3306번 포트에서 서비스요청을 대기하고 있는 mysqld, 80번 포트에서 서비스요청을 대기하고 있는 httpd, 등 데몬들을 볼 수 있고 그 아래는 소켓들이 보여주고 있고, 다음에 ssh로 어느 컴퓨터가 서버에 접속되었는지를 보여준다. 특히 DoS (Denial of Service) 공격일 경우에는 Netstat 화면을 통해 소켓들이 급격히 증가하는 것을 볼 수 있으므로 관리자는 화면을 통해 확인하여 해당 프로세스 또는 데몬들을 중지시키게 하여 DOS 공격에 따른 시스템의 과부하를 조절할 수 있다.



(그림 12) netstat 화면

4.3.5 Snort 실행 예

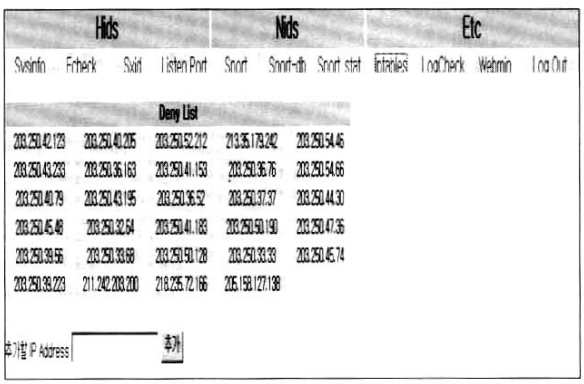
(그림 13)은 IDS인 Snort가 패킷들을 캡처하여 그 결과를 데이터베이스인 MySQL에 저장하고 그 저장된 결과를 웹을 통해서 TCP, UDP, ICMP의 전체 통계 현황을 보여주고 있는 것으로 TCP에 해당하는 패킷이 총 패킷의 86%라는 것을 알 수 있고 UDP에 해당하는 패킷이 3%, ICMP에 해당하는 패킷이 10%라는 것을 보여주고 있다. 총 합이 100%가 되지 않는 것은 소수점을 감안하지 않았기 때문이고, 해당 프로토콜을 선택하면 해당 프로토콜을 이용하여 시스템에 침입 또는 접근한 패킷의 내용을 상세하게 보여줄 수 있도록 하였고 침입, 접근에 대한 통계를 해당 IP별로 통계처리 한 화면은 모의실험결과에서 볼 수 있다.



(그림 13) Snort main 화면

4.3.6 iptables 실행예

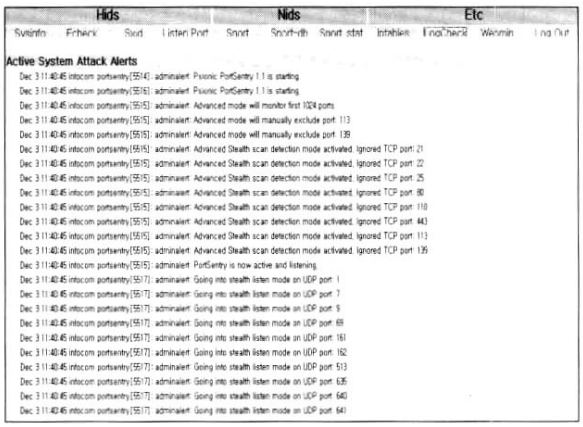
(그림 14)는 거부 리스트에 등록된 IP들을 볼 수가 있는데, 이것은 관리자가 침입 탐지된 로그를 LISS을 통하여 불법침입 흔적이 보이는 IP를 확인, 해당 IP를 iptables에 추가된 것이다. 여기에 추가되면 패킷 필터링에 의해 접근한 호스트는 시스템의 라우팅에 의해 패킷이 Drop되므로 접근이 불가능하게 된다.



(그림 14) iptables 화면

4.3.7 logcheck 실행 예

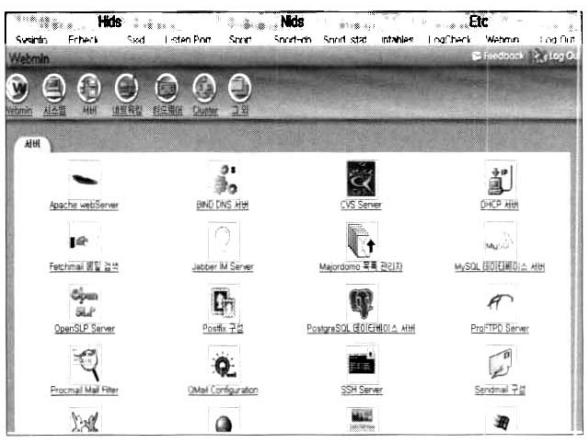
(그림 15)는 시스템 로그를 logcheck를 통하여 웹에서 로그를 보여주는 화면이다. 화면을 통하여 시스템에 대한 로그와 로그인 시도를 몇 번하였는지 그리고 어떠한 파일을 어느 계정이 업로드하고 다운로드를 하였는지, 데몬들의 종료와 재시작의 여부를 볼 수 있도록 하였고 PortSentry에 의한 탐지 여부 또한 볼 수 있도록 설계 하였다. 그림에 나와 있는 메시지는 infocom인 서버에서 PorSentry가 advanced 모드로 2개가 시작되었다는 것을 보여주고 있고, Stealth scan이 TCP와 UDP의 해당 포트에서 탐지되었다는 것을 알 수 있다.



(그림 15) logcheck 화면

4.3.8 webmin 화면

(그림 16)은 웹에서 웹서버인 Apache와 네임서버인 DNS, 데이터베이스인 MySQL 등 해당 설정을 수정하거나 설정할 수 있는 화면이다. webmin을 통하여 데몬들의 설정을 수정할 수 있고, 종료 및 재시작할 수 있도록 되어있고, 시스템의 설정 사항이나 변경을 할 수 있게 되어 있다.



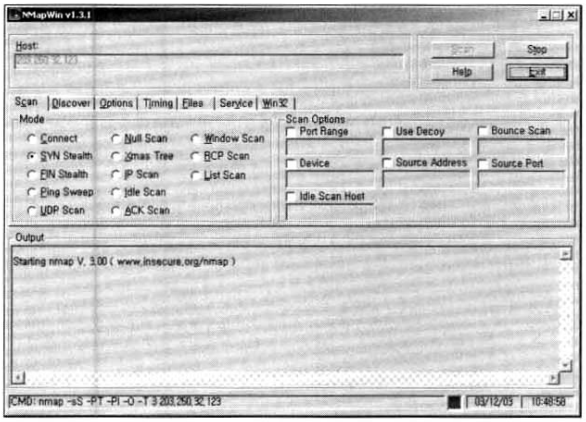
(그림 16) webmin 화면

5. 모의실험 및 평가

5.1 가상 공격 시도

5.1.1 NMAP를 이용한 가상공격

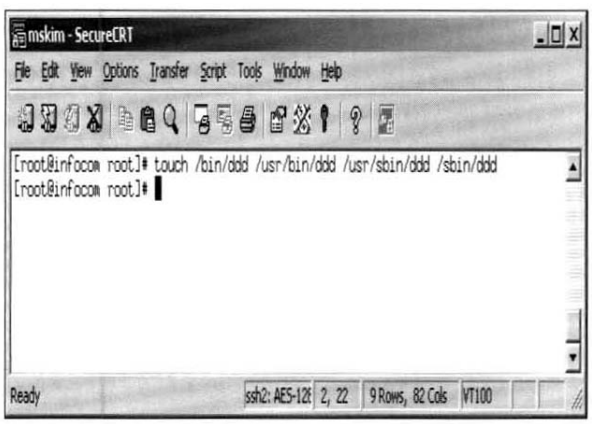
(그림 17)은 윈도우용 NMAP을 이용하여 LISS가 동작하는 서버에 SYN Stealth 스캔으로 완전한 TCP 연결을 열지 않는 방법으로 즉, 하나의 패킷을 보내어 SYN/ACK 응답을 받으면 그 즉시 RST 패킷을 보내어 완전한 연결 설정을 끊어 버리는 스캔방식으로 명령어와 옵션을 보면 nmap -sS -PT -PI -O -T 3 203.250.32.123이며 TCP와 ICMP를 이용하여 3초 간격으로 Stealth로 203.250.32.123에 스캔하고 있는 화면이다.



(그림 17) NMapWin v1.3.1을 이용한 공격 실행 예

5.1.2 LISS에서 파일 변조

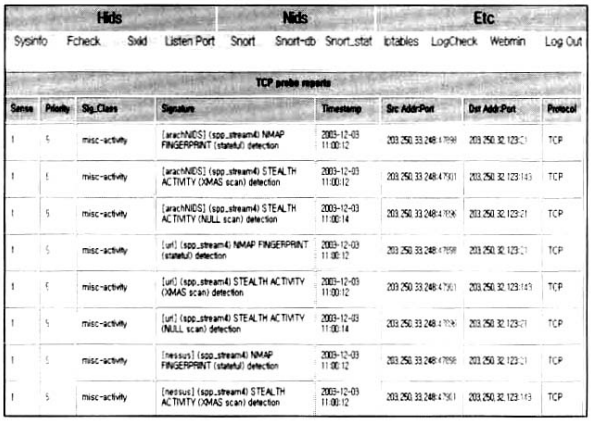
(그림 18)은 해커가 대상 서버의 침입에 성공하였을 경우 백도어를 심어놓았을 경우가 일반적이기 때문에 LISS가 설치되어 있는 서버에 접속하여 임의의 파일인 ddd를 touch라는 명령어를 통해서 /bin, /usr/bin, /usr/sbin, /sbin 디렉터리에 생성하고 있는 화면이다.



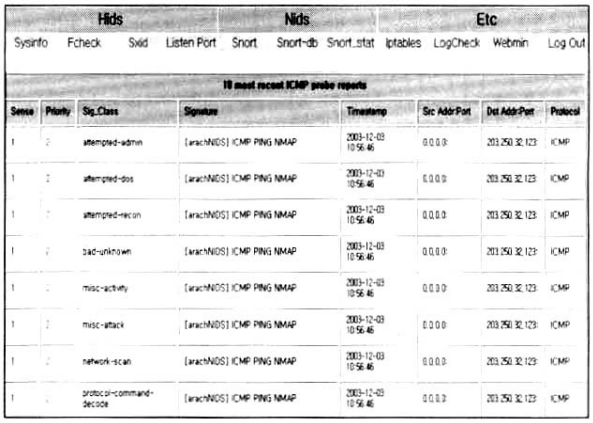
(그림 18) 시스템에 파일을 생성하고 있는 화면

5.2 가상 공격 결과

Nmap을 이용한 가상공격의 결과로 (그림 19)와 (그림 20)에서 보면 TCP/ICMP를 이용한 공격이 탐지되었다는 것을 알 수가 있다. (그림 19)를 보면 위험도에 관한 우선순위를 알 수가 있고 1부터 5까지 우선순위가 있고 5가 위험도가 가장 높다. Sig_Class는 사용자가 공격을 쉽게 이해하고 우선순위를 결정할 수 있도록, Snort에서 공격을 몇 가지로 분류해 놓은 것이다. 예를 들면, misc-activity는 사소한 행위, attempted-dos는 Dos시도, attempted-recon는 정보유출시도, misc-attack는 사소한 공격 등으로 분류되어 있다. 또한 Signature는 공격규칙에 대한 내용이며 패킷을 보낸 시각과 패킷을 보낸 송신지 컴퓨터의 주소와 목적지 컴퓨터의 주소와 포트를 보여주고 있다. NIDS가 탐지한 로그를 토대로 위험도가 높은 공격일 경우 관리자의 판단에 의해 해당 IP를 iptables에 적용시켜 탈락하게 한다.

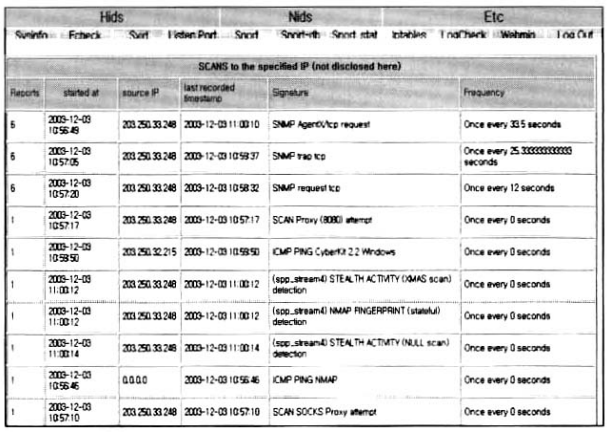


(그림 19) LISS가 탐지한 로그화면-TCP



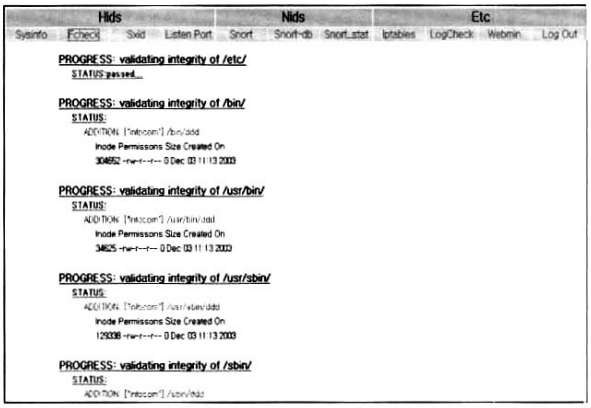
(그림 20) LISS가 탐지한 로그화면-ICMP

(그림 21)은 각 호스트 별 Signature에 따른 서버의 공격 횟수를 분석한 화면이다. 따라서 Signature에 따라 서버에 어떠한 공격들이 몇 번씩 일어났는지를 알 수 있다.



(그림 21) 호스트별 Signature에 따른 공격횟수

해당 서버의 침입에 성공했을 경우 파일을 생성한 이후의 변경상황을 (그림 22)를 보면 DDD란 파일이 /bin/, /usr/bin/, /usr/sbin/, /sbin/ 디렉터리에 2003년 12월 3일 11시 13분에 추가되었음을 확인할 수가 있다. /etc/ 디렉터리는 변경사항이 없기 때문에 PASSED로 표현되어있다. 변경되었을 경우는 WARNING, 삭제되었을 경우는 DELETION으로 표기 된다.



(그림 22) Fcheck의 무결성 체크 보고서

5.3 평 가

<표 3>은 기존의 ESM을 제공하는 툴과 본 논문에서 구현된 LISS를 상호 비교 평가한 결과를 보여준다.

기존의 ESM툴과 LISS를 비교해 보았을 때 기존의 ESM 개발 툴은 대부분의 사용자 인터페이스가 복잡하고, 각각의 보안 툴을 자회사의 제품을 이용하여 ESM을 구현하였다. 본 논문에서 제시한 LISS는 공개용 프로그램을 이용하여 구현하였고 침입탐지시스템의 대부분이 규칙기반에 근거하여 탐지하기 때문에 규칙의 업데이트를 관리자가 일일이 하여야하는 번거로움을 해결하였다. 또한 시스템 설정은 Webmin을 연동함으로써 웹에서 직접 시스템에 대한 설정을 할 수 있으며 SUID/SGID를 주기적으로 검사하게 구현하였다. 또한 사용자 인터페이스를 쉽게 설계함으로서 사용자의 추가적인 교육이 필요 없도록 하였다.

〈표 3〉 기존 ESM 제품과의 비교평가

제 품	spider-1	스나이퍼 IDS	Active ESM	Secureworks ESM	NeoWatcher@ESM	LISS
침입차단시스템연동	○	×	○	○	○	○
침입탐지시스템연동	○	○	○	○	○	○
시스템 무결성 관리	○	○	○	×	×	○
로그 Report	○	○	○	○	○	○
auto rule update	×	×	×	×	×	○
SSL	○	○	○	○	×	○
시스템 모니터링	○	○	×	○	×	○
시스템 설정	×	×	×	×	×	○
보안 정책 관리	○	○	○	○	○	○
SUID/SGID check	×	×	×	×	×	○
사용자 인터페이스	복 잡	복 잡	복 잡	복 잡	간 단	간 단

따라서 전산망의 해킹과 바이러스 침해사고 증가의 이유인 서버의 운영 미숙과 그에 따른 보안 대책 수립이 미흡한 것을 LISS를 사용함으로써 상당 부분 해결할 수 있을 것으로 판단한다.

6. 결론 및 향후 계획

인터넷의 급속한 확산으로 보안 침해 사고가 증가하고 있어 이에 대한 해결 방안이 시급히 요구되고 있다. 서론에서 기술하였듯이, 전산망의 해킹과 바이러스 침해사고 중 대부분이 교육기관에 집중돼 학교 전산망 환경에서의 보안기능 강화가 절실하다. 특히 중등학교에서는 각 보안 솔루션들을 운영하면서 상호연관성을 분석하여 이상 징후를 찾아 낼 수 있는 수준의 정보보호 전담조직 및 전문 인력이 턱없이 부족한 형편이고, NEIS와 같은 교육행정정보시스템의 도입에 따라 학사 정보보호정책 및 지침 수립 등 보안 준수 현황에 대한 사전 이상 징후탐지·예방, 경보를 위한 통합보안관리 분석시스템의 필요성이 절실히 요구되고 있다.

이에 따라 본 논문에서는 중등학교 전산망 환경을 염두에

두고, 아울러 최근 침입탐지시스템의 추세와 맞추어 네트워크 기반 시스템과 호스트 기반 시스템을 합친 혼합형(hybrid) 침입탐지 및 차단 시스템인 리눅스 기반 통합 보안 시스템(LISS)을 설계하고 구현하였다. 주요한 구현 특징으로 공개용 보안 도구의 각기 다른 출력 인터페이스를 한곳으로 통합하여 그 결과를 암호화된 웹을 통하여 보다 편리한 사용자 인터페이스가 되게 하였고, 보안 관리 정책 및 절차의 정립과 예방적 보안관리 체계를 수립할 수 있다는 점이 있다.

경제적인 측면에서 보면, 공개용 운영체제인 리눅스 서버에 통합 보안 관리 시스템을 설치 및 활용함으로서 비용 절감과 단순 반복 업무 해소에 의한 비용 및 시간 절감효과를 가져올 수 있다. 이에 따른 절감된 시간 및 비용으로 교육기관의 서버에 대한 보안 정책 수립 및 취약점 분석 등을 통하여 좀더 안전한 서버를 운영하는데 투자할 수 있고, 보안관련 인력들이 대부분의 시간을 로그를 분석하는데 보내지 않고 보안대책과 대응에 따른 보안정책을 마련하는데 시간을 투자할 수 있을 것으로 생각된다. 따라서 기술적 측면과 인적측면의 보안 문제점을 해결할 수 있을 것으로 기대된다. 또한 학교 교사들의 보안에 관련된 전문 지식에 대한 부족을 LISS를 활용함으로서 학교 정보 보안관리체계의 미흡함을 채워 줄 수 있을 것으로 또한 기대한다.

각 공개용 보안 도구들의 설정에 관한 부분이 없는데 이에 대한 설정을 웹에서 설정할 수 있도록 하여 보안 정책을 수립하는데 있어서 편리하도록 하고, Snort에서 출력된 이진 파일을 사람이 볼 수 있게 문서를 변환하여 보여주는 향후 연구가 필요하다. 또한 로그 분석에 필요한 로그들은 엄청난 양의 로그가 매일 매일 쌓이게 되는데 이는 곧 서버가 다운되는 현상을 초래할 수 있다. 그에 따라 로그들을 압축하고 로그서버에 압축된 로그파일을 전송하는 백업 자동화에 대한 연구가 수반되어야 하겠다.

참 고 문 헌

[1] 이윤성, 인터넷 해킹을 막기 위한 보안 방법 연구, 석사학위 논문, 동아대학교, Dec., 2000.

[2] 한국 교육 신문, 교육기관 해킹에 무방비, 사회면 기사, 2003. 9. 18.

[3] 한국 침해사고 대응팀, 국내외 해킹현황, http://cc.or.kr/statistics/hack/hack.htm, July, 2003.

[4] 박종오, "학내전산망의 안전성 확보를 위한 보안진단 에이전트 개발", 컴퓨터교육학회 논문지, 제4권 제3호, 2001.

[5] Market Trend, "보안 컨설팅 시장 조사", http://www.itdata.co.kr/column/200305/market/part3.asp, 시사컴퓨터, May, 2003.

[6] 제2회 리눅스공동체세미나, 리눅스 침입 탐지 시스템(NIDS) 구축, Apr., 2001.

[7] Carl Endorf, Eugene Schultz, Jim Mellander, Intrusion Detection and Prevention, McGraw-Hill, 2004.

[8] John McHugh, "Intrusion and intrusion detection," IJIS (2001) 1, pp.14-35, 2001.

[9] M. Roesch, Snort-lightweight intrusion detection for networks, Proc. of USENIX, pp.229-238, 1999.

[10] V. Jacobson, C. Leres, S. McCanne, libpcap, Lawrence Berkeley National Laboratory, http://www-nrg.ee.lbl.gov, 1994.

[11] 정병호, "정보화 환경에서 효율적인 전산 보안 시스템의 구축", 석사학위 논문, 전북대학교, Aug., 2001.

[12] Unix and Network Security, http://cbbrowne.com/info/secunix.html.

[13] 반장호, 홍석범 역, 제러드무라니 저, 리눅스보안과 최적화 완벽 솔루션, 한빛미디어, 2002.

[14] Iptables Tutorial 1.1.19, http://iptables-tutorial.frozentux.net/iptables-tutorial.html.

[15] 홍석범, 간단하면서도 막강한 파일 무결성 체크 프로그램 Fcheck, http://tt.co.kr/~antihong/Fcheck.doc, Jan., 2002.

[16] Fcheck, http://www.brandonhutchinson.com/fcheck.html

[17] suid/guid tool, http://linux.cudeso.be/linuxdoc/sxid.php.

[18] Intrusion Detection Systems, Part IV : Logcheck, http://www.freeos.com/articles/3540/.

[19] KLDP, "SSL Certificates HOWTO," http://doc.kldp.org/wiki.php/DocbookSgml/ SSL-Cezrtificates-HOWTO, Mar., 2002.

[20] http://www.openssl.org.

전 용 희

e-mail : yhjeon@cu.ac.kr

1971년~1978년 고려대학교 전기공학과

1985년~1987년 미국 플로리다공대 대학원 컴퓨터공학과

1987년~1992년 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 석사, 박사

1978년~1978년 삼성중공업(주)

1978년~1985년 한국전력기술(주)

1989년~1989년 미국 노스캐롤라이나주립대 Dept of Elec. and Comp. Eng. TA

1989년~1992년 미국 노스캐롤라이나주립대 부설 CCSP(Center For Comm. & Signal Processing) RA

1992년~1994년 한국전자통신연구원 광대역통신망연구부 선임연구원

2001년~2003년 대구가톨릭대학교 공과대학장 역임

1994년~현재 대구가톨릭대학교 컴퓨터·정보통신공학부 교수

2000년~현재 한국통신학회 학회지 편집위원

2004년~현재 한국전자통신연구원 정보보호연구단 초빙연구원

관심분야 : 네트워크 보안, 통신망 성능분석, QoS 보장 기술

김 민 수

e-mail : net94@korea.com

2004년 대구가톨릭대학교 교육대학원 (교육학 석사)

현재 영남IT캠퍼스 교수

관심분야 : 네트워크 보안, 고속 통신망 응용 서비스, 리눅스 보안

장 정 숙

e-mail : jsukjj@cu.ac.kr

1991년 경일대학교 공과대학 컴퓨터공학과 (학사)

1992년~1995년 대구가톨릭대학교 교육대학원 전자계산교육전공(석사)

2004년 대구가톨릭대학교 대학원 컴퓨터·정보통신공학부 네트워크보안전공 (박사)

2004년 현재 대구가톨릭대학교 컴퓨터·정보통신공학부 IT교수

관심분야 : 네트워크 보안, Active Network, 통신망 성능분석, 고속 통신망 응용 서비스