센서 개념을 적용한 침입 탐지 시스템

장철연°†, 김근래†, 조성제†, 김준모*
단국대 정보컴퓨터학부†
한국정보보호진흥원 기술단*
aceoftop@hanmail.net, lepffm@hotmail.com, sjcho@dankook.ac.kr, jmkim@kisa.or.kr

An Intrusion Detection System Based on Sensor

Chul-Yean Chang†, Keun-Rae Kim†, Seongje Cho†, Joonmo Kim*
Division of Information and Computer Science, Dankook University†
Information Security Technology Dept., Korea Information Security Agency*

요 약

본 논문에서는 센서(센서 파일, 센서 데이터 등)를 이용한 침입 탐지 시스템인 SbIDS(Sensor based Intrusion Detection System)를 제안한다. 리눅스 시스템에 구현된 SbIDS는 호스트 기반 침입탐지 기법과 네트워크 기반 침입탐지 기법이 통합된 시스템으로, 일차적으로 커널 수준에서 침입을 감지하고 대응하는 KMOD 모듈과 이차적으로 네트워크 수준에서 침입을 감지하고 대응하는 NetMOD 모듈로 구성되어 있어 호스트 내에서의 침입과 네트워크를 통한 침입을 동시에 탐지할 수 있다. SbIDS를 이용한 침입 탐지를 위해 먼저 주요 디렉토리에는 센서 파일을, 주요 파일에는 센서 데이터를 설치한다. 그 다음, 침입자에 의해 센서가 접근될 때마다 위기 상황으로 보고 커널 수준과 네트워크 수준에서 로그를 작성하며, 공격자를 식별하여 추적할 수 있고 침입으로 판단된 경우 해당 프로세스를 조기에 종료시킬 수 있도록 구현하였다.

1. 서론

침입 탐지 시스템(Intrusion Detection System, IDS)은 컴퓨터 시스템 또는 네트워크에서 일어나는 사건 및 사용자 행위들을 감시하고 침입 여부를 파악하기 위해 그 사건들을 분석하여 침입에 대응하는 소프트웨어 또는 하드웨어이다[1-3]. IDS는 두 종류로 분류되며, 네트워크 기반 IDS(NIDS)는 네트워크 상의 모든 호스트로 향하는 트래픽을 감시하여 수상한 활동을 보고한다. 호스트 기반 IDS(HIDS)는 각 호스트에 설치되어 운영체제 감사 로그 파일(audit logs)과 다른 지역 데이터를 조사하여 부적절한 활동을 찾는다. NIDS는 고수준 네트워크 프로토콜로부터 감추어진 하나의 탐지기를 이용하여 여러 호스트에 대한 트래픽을 감시하여 분산 공격을 인지할 수 있는 장점이 있으며 HIDS는 시스템 보안 상태 변화를 추적하여 실제 공격을 탐지할 수 있다는 장점이 있다.

그러나 NIDS는 암호화된 네트워크 트래픽을 취급할 수 없으며 공격이 실제 성공하였는지를 판단할 수 없고, HIDS는 네트워크 및 다른 호스트에 대한 공격을 인지할 수 없으며 대부분 설치 및 유지가 복잡하다. 이외에도, 새로운 공격이나 기존 공격의 변종을 탐지할 수 없으며 과도한 네트워크 및 처리 부하가 있을 때 공격의 탐지, 보고, 빠른 대응이 어렵다. 또한 IDS 자신에 대한 공격에는 취약하며, 공격을 상세 분석하기 위해서는 인간의 개입이 필요하다.

이러한 문제를 부분적으로 해결하기 위해, 본 논문에서는 센서 데이터(sensor data)나 센서 파일(sensor file)을 설정하여 주요 파일 및 디렉토리에 대한 접근을 감시하고 중요 데이터에 대한 임의접근 및 외부 유출을 방지하여 주는 시스템, SbIDS (Sensor based IDS)을 설계하고 구현하였다. SbIDS는 HIDS와 NIDS의 기능이 결합된 시스템으로, 먼저 커널 수준에서 주요 데이터의 접근을 감시하며 공격 프로세스를 식별·종료시킬 수 있다. 다음 단계로 정보를 불법적으로 외부로 유출하려는 네트워크 연결을 감시하며 불법 연결의 경우 커널과 연계하여 관련 프로세스를 종료시킨다. SbIDS는 새로운 공격도 탐지할 수 있으며 일부 모듈은 커널 내에 구현되어 있어 안전하다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 기술하며, 3장에서는 SbIDS의 개요 및 설계 고려 사항, 센서의 설정에 대해 설명한다. 4장에서는 커널 수준에서의 감지 및 대응, 네트워크 수준에서의 감지 및 대응 등의 구현에 대해 기술한다. 5장에서 결론을 짓고 향후 연구에 대해 언급한다.

2. 관련 연구

침입시도 기술로는 핑스윕(ping sweeps), 포트스캔(port scan), 운영체제 식별, firewalking 등을 이용하여 네트워크를 통해 어떤 시스템에 접근할 수 있는지, 제공되는 서비스는 무엇인지를 알아내는 스캐닝(scanning) 기법(sscan, mscan, ISS, cgichk, winscan 등)과, 시스템으로부터 노출되는 자원의 이름이나 유효한 계정을 추출하는 enumerating 기법이 있다[3, 4].

이러한 침입시도를 탐지하는 기술로는 syslog나 message 등의 시스템 로그 파일을 분석하여 침입을 탐지하는 chkwtmp, 알려진 침입시도에 대한 정보를 통해 탐지하는 오용탐지, SATAN과 같은 특정 공격을 탐지하기 위해 특정 포트만 감시

하는 gabriel, 포트들에 대한 접속을 감시 및 기록하는 detect-scans, 네트워크 트래픽과 패킷을 분석하여 호스트 접근을 차단하고 공격에 대응하는 RTSD, tcplogd, scanlogd, snort, libnids 등이 있다[3, 4]. 자동화된 탐지를 지원하는 기존의 도구들은 한 호스트에서 일정시간 간격으로 일정한 수 이상의 패킷이 전송될 경우, 이를 취약점 검색공격으로 인지한다.

그러나, 기존 IDS는 전반적으로 1장에서 언급한 한계점이 있으며, 어떤 IDS는 특정 시간 간격동안 전송된 패킷 수가 임계값 이상일 경우를 침입시도로 보고 있어, 이를 회피하는 공격시도에 약하다. 즉, 느린 스캔공격이나 협동공격에 약하다[4]. 또한 취약점 포트와 일반 포트를 모두 동일하게 감시하는 IDS 경우, 취약점 포트 위주의 스캔 공격을 효율적으로 탐지하지 못한다. 대부분의 IDS 는 침입을 탐지하는데 주력하며 자료의 보호라는 측면이 약한 것이 사실이다.

효율적으로 침입을 탐지하고 대응하기 위해서는 호스트 수준과 네트워크 수준에서 서로 긴밀한 연계를 통해 통합적으로 각종 사건 및 행위를 감시 분석하여 침입에 대응해야 한다. 이에 본 논문에서는 SbIDS를 제안하고 구현하였다.

3. 시스템 설계

3.1 시스템 개요

본 SbIDS은 침입시도를 종합적으로 감시하고 대응하기 위해 커널 모듈(KMOD)과 네트워크 모듈(NetMOD)로 구성된다. 시스템에 대한 개략적인 구조가 그림 1에 나타나 있다. 본 시스템에서는 가장 먼저 해야 할 일은 주요 디렉토리와 주요 데이터 파일에 센서를 설치하는 것이다. 센서가 설치되어 있으면, 첫 단계에서 호스트의 커널 모듈이 일차적으로 침입을 감지하여 대응하며, 두 번째로 네트워크 수준 모듈에서 침입을 감지하여 대응한다.

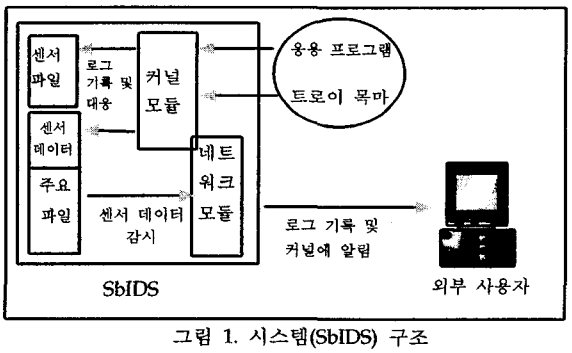

그림 1. 시스템(SbIDS) 구조

구동을 위해 먼저 주요 디렉토리에는 센서 파일을, 주요 파일에는 센서 데이터를 삽입한 다음, KMOD는 커널 수준에서 주요 디렉토리와 파일이 접근될 때마다 언제, 누가(어떤 프로세스)가 접근을 시도했는지 감시하여 로그를 작성하며 필요시 접근 프로세스를 종료시킨다. 즉, 커널의 *open*, *read* 루틴에서 센서를 감지하여 센서가 접근되면 그에 관련된 정보를 sids_klog란 로그 파일에 기록하고, 침입이라고 판단되면 공격 프로세스를 종료시킨다. 호스트 내부 침입은 KMOD를 우회할 수 없다.

다음 단계로 NetMOD가 외부로 향하는 네트워크 패킷 중에 센서 파일/데이터가 포함되어 있는지를 검사함으로써 주요 파일이 외부로 유출되는지 여부도 감시한다. 외부 유출 시, 언제, 어느 외부 호스트로 데이터가 유출되는지에 대한 로그를 sids_nlog에 작성하고, 해당 내부 호스트의 커널에게 실시간 대응을 지시하도록 한다. 본 논문에서는 센서 데이터(sensor_data와 같은 특정 문자열 패턴)와 센서 파일(특정 파일 이름)을 일반적으로 센서라고 부르며, 누군가가 이를 접근하면 그 접근이 감지된다.

제안된 SbIDS의 경우 트로이 목마(Trojan Horse)나 트랩도어 등의 숨겨진 모듈이 어떤 주요 디렉토리 내용 전체나 특정 파일을 외부로 송신하려고 할 때, 제안된 센서 개념을 적용하여 숨겨진 모듈의 존재 여부도 파악할 수 있으며 주요 데이터 접근 자체를 차단할 수 있다.

3.2 고려사항 및 센서의 설정

시스템을 설계할 때 고려 사항은 다음과 같다. 첫째, 센서를 설치할 디렉토리 및 파일의 선정이다. 예를 들어 Unix나 Linux의 경우 /etc 디렉토리에 센서를 설치해야 한다. 디렉토리 내의 한 파일만 중요하다면 그 파일에 센서 데이터를 설치한다. 둘째는 가장 중요한 것으로 주요 데이터에 대한 정상적인 접근과 공격 목적의 접근을 구별할 수 있어야 한다. 이를 위해 Unix 시스템에서 센서 파일은 점(.)으로 시작하는 유일한 이름(.sensor_file)을 가진 크기가 0인 파일이며, 최초 만들어질 때를 제외하고는 접근될 필요가 없도록 하였다. 또한 센서가 네트워크를 통해 외부로 유출될 때를 침입으로 판단한다. 셋째, 센서 데이터의 경우 주요 파일의 어디에 위치시킬 것인 가이다. 현재는 데이터 유출을 효과적으로 막기 위해 파일의 앞부분에 설치한다. 넷째, 네트워크 패킷 검사의 경우 네트워크 부하에 적응하기 위하여 효율적인 문자열 검색 알고리즘이 필요하다.

4. 구현

본 SbIDS는 IBM PC 펜티엄 III, 와우리눅스 커널 버전 2.4.2 상에 구현되었다. 네트워크 패킷 분석 도구로는 libpcap 라이브러리 버전 0.4 [7]를 사용하였다.

4.1 커널 수준에서의 감지 및 대응

먼저 sids_klog 로그 파일을 열고 닫기 위해 커널을 수정하여 sids_klog_open(), sids_klog_close() 시스템 호출을 추가하였으며 klog_open과 klog_close 명령을 작성하였다. 또한 커널의 sys_open()과 sys_read() 루틴에 센서 파일을 개방할 때를 감지하는 부분과 센서 데이터가 읽힐 때를 감지하는 부분을 추가하여 KMOD를 구현하였다. KMOD는 센서 파일이 접근될 때(open될 때)의 접근 프로세스 이름, 센서 파일의 경로, 접근 시간, 사용자 ID, PID 등을 sids_klog에 기록하고 해당 프로세스

를 즉시 중단시키도록 구현되었다. 센서 데이터가 읽혀질 때 (read될 때)에는 정상적 접근인지 공격적 접근인지 즉시 파악할 수 없으므로 접근에 관련된 로그만 기록한다. 단, 센서 데이터가 네트워크를 통해 외부로 유출되는 경우에는 NetMOD와 연계하여 침입이라고 판단하고 해당 프로세스를 종료시킨다. 그림 2는 센서 파일이 접근되었을 때 sids_klog에 기록된 내용을 보여준다.

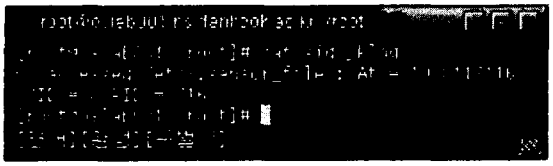

그림 2. sids_klog에 기록된 내용의 예

4.2 네트워크 수준에서의 감지 및 대응

NetMOD는 pcap 라이브러리를 사용하여 호스트 내부로부터 외부로 나가는 네트워크 패킷만을 분석하여 특정한 문자열 (즉 "sensor_data")이 접근되는지 감시한다. Pcap 라이브러리는 사용자 수준에서 패킷을 캡쳐할 수 있게 하여 준다. 패킷에서 해당 문자열이 발견되면 sids_nlog에 패킷에 관련된 정보, 즉 소스 노드의 주소, 목적지 노드의 네트워크 MAC 주소, 목적지 IP 주소, 포트 번호, 시간 및 날짜 등을 기록하게 된다 또한 beep 음이나 전자우편으로 시스템 관리자에게 알릴 수도 있다.

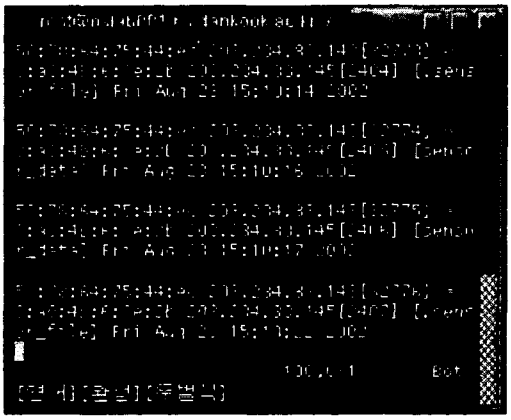

그림 3. sids_nlog에 기록된 내용의 예

네트워크를 통해 센서 파일 및 센서 데이터가 유출될 때 sids_nlog에 기록된 정보가 그림 3에 나타나 있다. 각 줄에는 sensor_data나 .sensor_file이 내부 노드 203.234.83.143으로부터 외부 노드 203.234.83.145로 빠져나갔다는 것을 나타낸다. 센서 데이터가 유출될 때 NetMOD는 커널에 이를 즉시 알리게 되며 KMOD와 연계하여 침입 프로세스를 파악하여 종료시켜 주요 데이터 유출을 즉각 통제할 수 있다. 또한 기록된 IP와 MAC 주소로 공격자를 추적할 수 있다.

효과적으로 침입을 감지하기 위해서는 네트워크 트래픽 부하 감소와 빠른 스트링 매칭 알고리즘이 요구된다. 이를 위해 현재 외부로 나가는 패킷만 분석하며 센서 유출 여부를 검사하는 스트링 매칭에 Boyer_Moore algorithm[6]을 이용하였다. 또 NetMOD를 방화벽(firewall) 출력 단에 설치한다면, NetMOD가 감시할 패킷 양이 많이 감소하게 되고 따라서 SbIDS 성능을 더 향상시킬 수 있다.

시스템 또는 사용자마다 보호하고자 하는 주요 디렉토리 및 주요 파일이 다를 수 있고 또 보안 유지를 위해, 센서 파일 이름이나 센서 데이터를 고정시키지 않고 임의로 동적으로 변경하여 설치 및 감시할 수 있도록 하였다.

5. 결론 및 향후 연구

본 논문에서는 센서 데이터나 센서 파일을 설정하여 주요 파일 및 디렉토리에 대한 접근을 감시하여 중요 데이터가 외부로 유출되지 않도록 유지시켜 주는 SbIDS를 제안하였다. SbIDS는 호스트 및 네트워크 기반의 통합된 시스템으로 주요 데이터 접근을 커널 수준에서 방지할 수 있고, 그 데이터가 외부로 유출되는 것도 네트워크 수준에서 통제할 수 있다.

향후, 가장 중요한 과제는 센서 개념을 시스템이나 네트워크 취약성(vulnerability)을 분석 및 검출하는데 적용하는 것이다. 취약성 분석 도구는 침입이 발생하기 전에 시스템 관리자에 적절한 보안 행동을 유도하여 IDS가 찾아야 하는 공격의 수를 현저히 감소시켜 주는 유용한 도구로 IDS와 상호 보완적으로 현재 SbIDS의 일부 기능은 취약성 검출에 이용될 수 있다. 또한 좀 더 완벽한 보안 정책을 구현하기 위해 SbIDS를 파일 일관성 체크 도구(file integrity checker)와 연계시키는 방안에 대해서도 연구할 계획이다.

참고문헌

[1] ISO/IEC WD 18043 (SC 27 N 3180): Guidelines for the implementation, operation and management of intrusion detection systems (IDS) (http://www.din.de/ni/sc27/doc7.html), 2002-04-26.

[2] R. Bace and P. Mell, "Intrusion Detection Systems," Special Publication SP800-31, National Institute of Standards and Technology, Gaithersburg, MD, Released by NIST in August 2001.

[3] SAMS, "Maximum Linux Security" pp 538~549, 1999.

[4] 유일선 "네트워크 취약점 검색공격에 대한 개선된 탐지시스템," 단국대 박사학위 논문, 2001.

[5] http://kldp.org

[6] C. Charras and T. Lecroq, "Handbook of Exact String-Matching Algorithms," (http://www-igm.univ-mlv.fr/~lecroq/biblio_en.html)

[7] ftp://ftp.ee.lbl.gov