LSM 기반 시스템 보안 모니터링 구현 방법

조성목*
*동명대학교 정보보호학과
e-mail:smcho@tu.ac.kr

An Implementation Method of LSM Based System Security Monitoring

Sung-Mok Cho*
*Dept of Information Security, Tongmyong University

요 약

본 논문에서는 리눅스 운영체제를 기반으로 오픈 소스 침입탐지 시스템인 Snort 등을 LSM(Linux Security Module) 구조의 hooking 부분에 구현시켜 보안 관리자가 의도하는 포트 스캔, DDOS 공격, ARP/IP 위장, 서명 탐지 등이 가능하도록 시스템을 구현하는 방법을 제시하고자 한다.

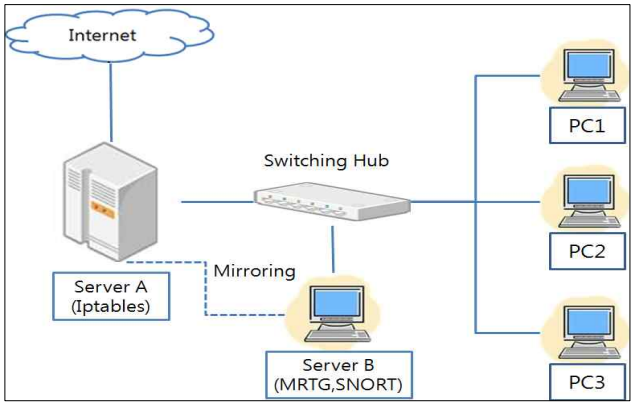
1. 서론

인터넷을 통한 정보교환의 양적 팽창과 정보의 가치가 증대되면서 서버 보안에 대한 관심이 더욱 높아지고 있다. 향후에도 다양한 IT 기술 발달과 인터넷으로 인해 생활의 편의성이 제고되는 반면, 누구나 쉽게 접근할 수 있는 개방적 환경으로 인해 악의적인 해킹과 개인정보 침해, 사이버 범죄 등과 같은 부정적인 기능이 심각할 것으로 예측된다. 따라서 침입탐지 등 보안관제 시스템은 개방적 네트워크 환경에 있어서 매우 중요한 비중을 차지하게 될 것이다. 기존의 침입탐지시스템은 수동적이며, 사후조치라는 취약성으로 인하여 실제적인 정보보호에 도움을 주지 못하는 실정이기 때문에 현재 실정에 부합하는 자동화된 침입탐지시스템이 네트워크 환경에서 요구되는 정보보호의 실현을 위해 반드시 필요한 요소로 자리 잡아가고 있다.[1] 본 논문에서는 개인정보의 누출과 해킹의 위험이 상존하고 있는 상황에서 외부 침입자에 대한 침입탐지 정보를 확인할 수 있는 침입탐지 시스템을 운영체제의 커널레벨에서 구현함으로써 별도의 보안 시스템을 구축하지 않고서도 보안상의 위험을 해소하고자 한다. 리눅스 운영체제에서 접근제어 방식으로는 객체의 소유자에 의해서 접근정책이 결정되는 임의 접근제어(DAC: Discertionary Access Control), 객체의 소유자가 아니라 시스템에 의해 강제 접근제어(MAC: Manatory Access Control) 및 규칙기반 접근제어(RBAC: Rule Based Access Control)이 있다. 그러나 커널 레벨에서의 접근제어와 침입탐지 등 다중 보안 정책이나 보안 기능을 구현하기 위한 방법을 모색하여 LSM Hooking을 통하여 침입탐지 기능을 수행하는 Snort를 커널 레벨에서 수행될 수 있게 하는 방법을 살펴본다.

2. 보안 관제 시스템

침입탐지 시스템(IDS: Intrusion Detection System,)은 보안과 관련된 비밀성, 무결성, 가용성에 문제가 되는 비정상적인 상황을 탐지하여 즉각적으로 대응하거나 시스템 관리자에게 메시지를 전달하는 정보보호 시스템으로 접근 제어 기능뿐만 아니라 침입 패턴을 구성하는 데이터베이스와 전문가 시스템 등을 통하여 네트워크나 시스템을 실시간으로 모니터링할 수 있다. Snort는 실시간 트래픽 분석과 IP 네트워크상에서 패킷 로깅이 가능한 경량 네트워크 침입 탐지 시스템으로 패킷 스니핑 뿐만 아니라 규칙을 이용한 분석 기능 등 지속적인 기능 보완과 향상으로 매우 강력한 침입탐지 기능을 갖추고 있다.[2] 또한 시스템 관리자는 네트워크 트래픽을 모니터링하여 트래픽에 대한 문제를 확인할 수 있어야 한다. 네트워크 모니터링 분석에 사용되는 툴로는 MRTG(Multi Router Traffic Grapher), SNMP(Simple Network Manegement Protocol),

MRTG(Multi Router Traffic Grapher)등이 있다.



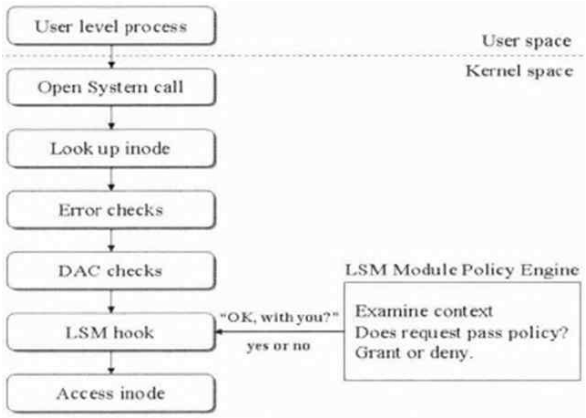
[그림 1] 침입 탐지 방지 시스템의 네트워킹 구성

이와 같은 네트워크 모니터링 방식을 사용하면 네트워크 모니터가 설치된 컴퓨터의 네트워크 어댑터를 통과하는 네트워크 트래픽에 대한 정보를 수집할 수 있고, 정보를 캡처한 후 네트워크 모니터를 사용하여 정보를 분석하고 트래픽 패턴 문제를 진단하며 향후 발생할 수 있는 네트워크 트래픽 문제를 예방할 전략을 세울 수 있다. 특히, 모니터링에 사용되는 대표적인 툴인 MRTG는 SNMP를 지원하는 네트워크 장비가 발생하는 트래픽을 모니터링해 주는 소스가 공개된 프로그램이다. 지정한 시간마다 모니터링한 결과 값을 GIF또는 PNG 이미지로 생성하여 HTML 페이지로 뿌려주기 때문에 누구나 쉽게 트래픽 현황을 볼 수 있는 특징이 있다. 또한, SNMP는 네트워크 장치 사이에 관리 정보 교환이 가능한 응용계층 프로토콜이다. SNMP는 네트워크 관리자가 성능을 관리할 수 있는 기능을 제공하고 네트워크에서 발생하는 문제점을 찾아 해결할 수 있도록 움을 제공할 뿐만 아니라 네트워크 증설에 따른 계획 수립을 위한 정보를 제공하기도 한다.[3] 한편, MRTG는 네트워크 링크 상의 트래픽 부하를 측정하는 도구로서, SNMP를 이용하여 라우터나 스위치 등으로부터 네트워크 트래픽 정보를 수집하여 트래픽 상황을 웹 상에서 실시간으로 보여 줄 수 있는 매우 편리한 기능을 제공하고 있다.[3] [그림 2]에서 Snort는 정해진 룰 셋(rule set)에 의하여 프로토콜 분석과 패킷의 내용을 조사하고 패턴매칭을 통하여 buffer overflows, stealth ports scans, CGI attacks, SMB(Server Message Block) probes, OS fingerprinting attempts 등 다양한 공격의 탐지를 수행한다. Snort가 침입탐지를 하면 관리자는 iptables에 해당 공격자 호스트에 대한 추가여부를 등록하게 된다. 또한 iptables는 방화벽을 통과하는 각각의 연결을 메모리에 저장하므로 Rating 제한을 이용하여 대부분의 서비스 거부 공격도 차단할 수 있다. MRTG는 SNMP를 지원하는 네트워크 장비가 발생하는 트래픽을 모니터링하는 프로그램이다. 지정한 시간마다 트래픽량을 세부적으로 그래프로 생성해서 보여주기 때문에 트래픽 상황을 알아보기 쉽고 관리하기 쉽다.

[그림 2] 시스템 보안 모니터링 시스템

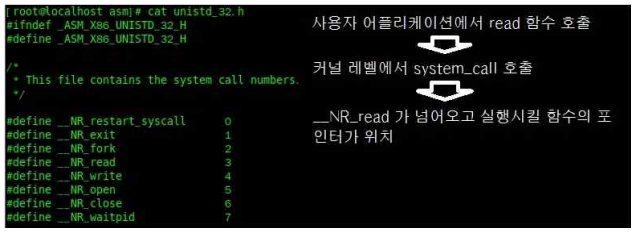
3. LSM Hook 구조

LSM은 리눅스 커널 2.5 서밋에서 NSA(National Security Agency)가 보안이 강화된 리눅스(Selinux)를 발표한 후 라이너스 토발즈(Linus Torvalds)가 리눅스 커널에 포함 시킬 수 있는 보안 모듈 프레임워크을 제시하였다. 이는 기존의 커널 패치 방식으로 구현되는 LKM(Loadable Kernel Module) 방식의 문제점인 다중접근제어 정책을 해결할 수 있도록 후킹 함수를 이용할 수 있도록 설계된 방식이다. [그림 3]의 LSM Hook 구조에서 나타나듯이 기존의 리눅스 시스템 호출 루틴에 LSM Hook이 추가되어 있으므로 시스템 보안 관리자는 LSM Hooking 방식을 통하여 보안에 필요한 접근제어나 정책을 구현할 수 있도록 설계되어 있다. 특히, LSM 프레임워크는 감사와 같은 보안 요구를 지향하는 쪽으로 발전해 나갈 것으로 전망되지만 원래는 접근 제어 모듈을 지원하는데 초점이 맞추어져 있었다. LSM 프레임워크 자체로는 부가적인 보안을 제공하는 것이 아니며, 단순히 보안 모듈을 지원하는 인프라를 지원하고 있다.[4-7]



[그림 3] LSM Hook 구조

[그림 3]의 LSM Hook 구조에서 더미 함수를 침입 탐지 등의 보안 기능을 수행할 후킹 함수로 대체함으로써 시스템의 보안 모니터링을 수행할 수 있다.



[그림 4] LSM Hook 구조

시스템 콜 식별번호는 [그림 4]에서와 같이 "/usr/include/asm/unistd.h"에 정의되어 있으므로 여기에 새로운 시스템 콜 번호를 하게 되고, 그 처리 함수는 "kernel/sys.c"에 정의되며, 시스템 호출 처리는 "arch/i386/kernel/entry.S"에 의해 이루어 진다. 시스템이 호출되면 [그림 5]의 sys_call을 호출하게 되고, sys_call_table에서 호출 처리함수의 주소를 eux에 전달하고, 함수를 호출하여 처리하게 된다.

```
ENTRY(system_call)
        pushl %eax                      # save orig_eax
        SAVE_ALL
        GET_THREAD_INFO(%ebp)
        cmpl $(nr_syscalls), %eax
        jae syscall_badsys
                                        # system call tracing in operation
        testb $_TIF_SYSCALL_TRACE,TI_FLAGS(%ebp)
        jnz syscall_trace_entry
syscall_call:
        call *sys_call_table(,%eax,4)
        movl %eax,EAX(%esp)             # store the return value
syscall_exit:
        cli                             # make sure we don't miss an interrupt
                                        # setting need_resched or sigpending
                                        # between sampling and the iret
        movl TI_FLAGS(%ebp), %ecx
        testw $_TIF_ALLWORK_MASK, %cx   # current->work
        jne syscall_exit_work
```

[그림 5] 시스템 호출 처리

4. 결론

본 논문에서 제안한 LSM Hooking 을 이용한 침입탐지 시스템의 구현은 커널 레벨에서 수행되므로 운영체제 상에서 설치된 침입탐지 시스템에 비하여 외부 공격에 견고한 장점을 가진다. 따라서 이와 같은 방식의 보안 접근제어 방식은 중소기업에 활용 가능한 침입탐지 시스템으로써의 기능을 수행 할 수 있을 것으로 기대되며, 향후 다양한 보안관련 오픈소스를 활용하여 Linux 기반 시스템을 구성함으로써 보안운영체제로써의 확장성과 유연성을 제고할 수 있을 것으로 판단된다.

참고문헌

[1] 박재홍 「리눅스 기반 통합보안시스템 설계 및 구현」, 전남대학교 컴퓨터공학 논문, 2005. 8.

[2] 한국정보보호진흥원 "snort를 이용한 IDS 구축", http://www.kisa.or.kr

[3] 김용진, 이승윤, 정재훈 "MRTG(Multi Router Traffic Grapher)의 설치 및 구성법", IPv6 포럼 코리아 기술문서, 2002. 2.

[4] C.Wright, C.Cowan, C.Smalley, J.Morris, J.Kroah-Hartman (PROCEEDINGS OF THE USENIX SECURITY SYMPOSIUM, Vol.11 [2002])
Linux Security Modules: General Security Support for the Linux Kernel.

[5] 김정순, 이재서, 이승용, 김민수, 노봉남, "리눅스 보안운영체제를 위한 접근통제 프레임워크", 정보보호학회, 제15권, 제2호, 2005.4.

[6] 이천희, "시스템 호출 후킹 모듈과 리눅스 보안 모듈의 분석 및 평가", 한서대학교 대학원 석사 학위논문, 2004. 2.

[7] 류정각, LSM 프레임워크 기반의 리눅스 보안 강화 방안에 대한 연구", 아주대학교 대학원 석사학위 논문, 2003. 2.