커널기반의 통합 침입 탐지와 침입 차단 시스템에 관한 연구

정종근, 하추자, 김철원
호남대학교 컴퓨터공학과
e-mail : jkcom@hanmail.net

A study of Unioned Intrusion Detection System & Intrusion Prevention System based on Kernel

Jong-Geun Jeong, Chu-Ja Ha, Chul-Won Kim
Dept of Computer Engineering, Honam University

요 약

일반적인 침입탐지 시스템의 원리를 보면 공격자가 공격 패킷을 보내면 침입탐지서버에 IDS 프로그램으로 공격자의 패킷을 기존의 공격패턴과 비교하여 탐지한다. 공격자가 일반적인 공격 패킷이 아닌 패킷을 가짜 패킷과 공격 패킷을 겸용한 진보된 방법을 사용할 경우 IDS는 이를 탐지하지 못하고 로그 파일에 기록하지 않는다. 이는 패턴 검사에 있어 공격자가 IDS를 속였기 때문이다. 따라서 공격자는 추적 당하지 않고서 안전하게 공격을 진행할 수 있다. 본 논문에서는 이러한 탐지를 응용프로그램 단계가 아닌 커널 단계에서 탐지함으로서 침입탐지뿐만 아니라 침입 방지까지 할 수 있도록 하였다.

1.서론

방화벽은 능동적인 방어 시스템이 아니다. 이에 반해 침입 탐지 시스템(IDS)은 방화벽이 감지하지 못하는 공격에 대해 인식할 수 있으며 더 나아가서는 이전에 경험하지 못한 공격에 대해서도 이를 감지, 방어할 수 있는 능동적인 시스템이다[1]. 기존 방화벽의 결함에 대해 설명할 때 Cold Fusion을 들 수 있다. 침입 탐지 시스템의 핵심 기술은 행위 판별(Behavior Classification)과 자료축약(Data Reduction) 기술이다. 행위 판별은 주어진 일련의 행위들에 대해 침입인지 아닌지를 판별하는 것이고, 자료축약는 시스템에서 발생하는 거대한 양의 각종 로그 데이터(log data)를 의미있는 데이터로 추출하여 변환하는 작업이다. 일반적으로 침입 탐지 규칙에서는 규칙기반 시스템(Rule-based system)과 신경망 또는 통계적 분류 시스템을 사용한다. 기존에 사용된 규칙기반 시스템, 신경망, 통계적 분류 시스템은 많은 양의 데이터가 초기 학습을 위해 필요하며, 계속적으로 시스템을 유지하는데 많은 시간과 비용을 초래하며 새로운 공격 대응 능력이 약하다는 취약점을 가지고 있다. 또한, 일련의 정상적인 패턴이 아닌 우회하기 위해 또는 오동작을 고의로 유도하기 위한 방법으로 공격을 시도한다면 프로그램으로서는 오판단을 할 확률이 높다[2,3].

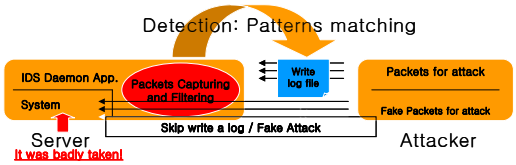
따라서, 본 논문에서는 크게 커널 단계에서의 IDS와 IPS의 두 단계의 침입 검증 과정을 거쳐 침입 시도를 원천적으로 봉쇄한다.

2. 기존 탐지 방법들의 문제점

2.1 IDS의 문제점

IDS는 일반적인 애플리케이션으로 구현되어 있으며 보안상 취약한 함수들이 존재하고 있다[6]. 예를들어, bound checking, format string, race condition, ptrace 와 같은 취약한 함수들이 존재한다든가, SetUID, SetEUID, SetRUID, SetRESUID 같은 퍼미션획득 등이다. 공격자는 단순히 하나의 SetUID 퍼미션이 아닌 여러 가지의 퍼미션을 획득할 수 있다. 또 잘못된 데이터나 패킷등 일련의 정상적인 패턴이 아닌 우회하기 위해 또는 오동작을 고의로 유도하기 위한 방법으로 공격을 시도한다면 응용프로그램 단계에서 탐지할 때 오판단을 할 확률이 높다[9].

[그림 1]은 일반적인 IDS의 구성도로서 공격자가 일반적인 공격 패킷을 보내면 서버측에서는 IDS 프로그램으로 공격자의 패킷을 기존의 패턴과 비교한다. 만약 일치한다면 로그 파일에 기록을 남긴다.



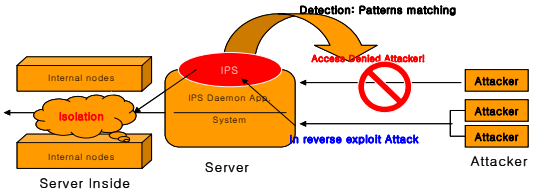
[그림 1] 일반적인 IDS 구성도

하지만, 우회 공격의 경우, 공격자가 일반적인 공격 패킷이 아닌 가짜 패킷과 공격 패킷을 겸용한 진보된 방법을 사용할 경우 IDS는 이를 탐지하지 못하고 로그 파일에 기록 또

한 하지 않는다. 이는 패턴 검사에 있어 공격자가 IDS를 속였기 때문이다. 따라서 공격자는 추적 당하지 않고서도 안전하게 공격을 진행할 수 있다. 따라서, 우회할 수 있는 local 과 remote 해킹으로부터 언제나 취약하다[7,8].

2.2 IPS의 문제점

프로그램 상에서 구현되는 IPS도 역시 불법적인 접속과 많은 시도로부터 내부망이 오히려 고립될 수 있다. 이는 서비스 거부 공격(DOS)이나 DDOS(Distribute Denial of Service) 공격과는 조금 다른 개념으로 서버쪽에서 IPS 우회로써 고립되는 상황과 IPS의 주요 내부 함수에 의해 취약점을 이용한 역공격이 취해질 가능성이 있다. 이는 실제 서버에 접속은 하지 않지만 외부 IPS를 우회해서 즉, 취약점을 이용해 reverse telnet과 같이 서버측에서 접속을 하도록 유도하는 것이다.

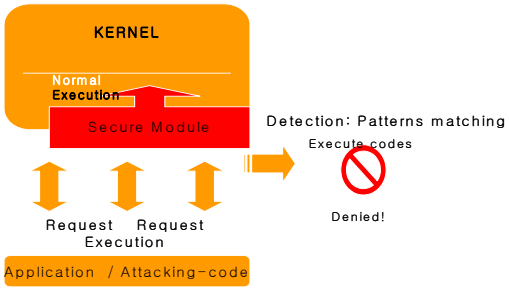


[그림 2] IPS의 기본 구조

일반적인 IPS의 원리는 공격자가 공격 패킷을 보내면 서버 측의 IPS가 패킷을 검사하고
패턴이 맞을 경우 이에 바로 대응한다. 즉, 공격자의 정확한 정보를 얻는다든지, 역 공격을 하는 방법이 있다.
공격자가 IPS를 우회할 경우 IPS를 역으로 이용해 원하는 작업을 수행 할 수 있다. 즉, IPS는 공격자의 공격에 대해 대응하게 설계된 만큼 그 대응방법을 구현해놓은 내부 함수를 이용하는 것이다. 이는 remote overflow/fsb 등과 비슷한 개념으로 IPS를 가짜 패킷 혹은 공격 패킷으로 대응 코드를 실행하도록 유도한 뒤 IPS로 하여금 내부망을 공격자 리스트에 올려 공격대상을 알려주는 것이다. 따라서 공격은 다른 사람이 하고 피해는 공격자가 아닌 내부망이 당하는 것이다.

3. 커널 기반의 IDnPS 기본 원리

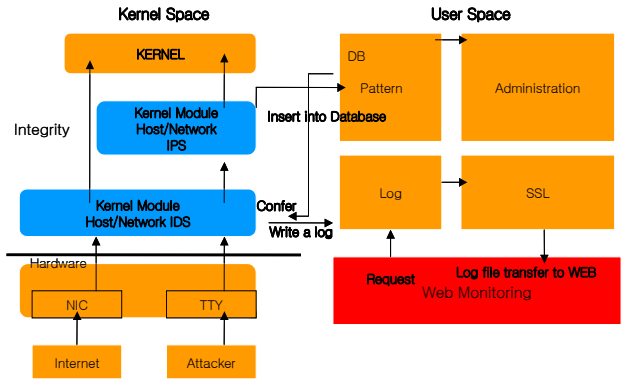
기존 IDS 와 IPS는 OS상의 일반 Application이지만, Kernel 기반의 IDnPS는 OS의 High-level의 권한를 가지고서 IDS와 IPS 역할을 한다. 따라서, 서버에 보안상 문제가 발생시 Kernel level에서 이를 즉각 처리하며 원천적(source)으로 대응한다. 여기에서 원천적이란, 해당 취약점(vulnerability) / Attack이 수행되기 전 여러 알고리즘을 먼저 수행한다. 이는 Kernel에서 High-level privilege모드로 동작하기에 가능하다. 다시 말해서 공격자가 공격을 수행하려고 한다면 IDnPS는 그 수행 권한을 가지고 있다. 즉, 공격자의 공격을 실행하는 것도 IDnPS이고 실행을 막는 것도 IDnPS 라는 것이다. 또한 최상의 level에서 높은 우선 순위(High-Priority)로 동작함으로써 공격용 Application이 쉽게 접근 할 수 없다. 일반 정상적인 애플리케이션이 실행을 요청했을 시에 커널에서의 secure 모듈이 이를 검사한 뒤 정상이라면 커널에 넘겨 실행을 하도록 한다. 하지만, 공격자의 코드가 실행을 요청 했을시에 마찬가지로 secure 모듈이 이를 검사한 뒤 공격 패턴과 맞을 경우 실행을 거부한다.



[그림 3] 커널기반 IDnPS의 기본 구조

4. 커널 기반 IDnPS의 구조

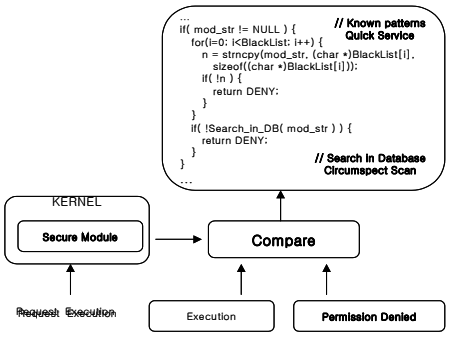
[그림 4]와 같이 일단 랜 카드를 통해 들어온 일반 패킷에 대해서 커널내의 IDS에서 검사하여 문제가 발견되지 않았다면 정상으로 처리하고 이상이 발견되면 일단 로그파일을 작성하고 IPS 모듈로 넘긴다. 2차 검증에서 큰 문제가 발견되지 않았다면 정상으로 처리한다. 하지만 IPS에서 문제가 발견되면 패턴 데이터 베이스에 IDS가 참고 할 수 있도록 로그를 추가하고 그에 해당하는 대응을 시작한다. 관리자가 web 에서 로그 메시지를 요청한다면 SSL을 통해 web으로 출력한다. 또한 관리입장에서 관리자에게 데이터 베이스를 제공한다. 본 시스템의 특징은 커널 레벨에서 파일 시스템에 직접 접근하여 네트워크 상의 패킷 캡쳐하며 침입 탐지 기능과 침입 방지 기능을 함께 가지고 있어서 자기 자신의 시스템을 방어할 수 있다. 가로챈 시스템 콜을 다른 용도로 개량하고 시스템 내의 장치와 제어를 커널 단계에서 모두 가로챌 수 있다.



[그림 4] 커널기반 IDS & IPS 구조

모듈이 종료되거나 시스템 종료시 원상태로 돌려놓으며 재시작 시에는 다시 모듈을 작동한다. 커널 레벨 내에서의 데이터 베이스 직접 접근하여 분석된 공격자의 패턴을 데이터 베이스에 추가하고 커널 모듈과 에이전트인 일반 애플리케이션과의 data 통신이 가능하다. 또한 완벽한 시스템 콜 제어와 root 보다 높은 특권을 가진다.

[그림 5]의 알고리즘 구성도는 커널 기반 IDnPS에서의 탐지를 위한 보안 모듈(Secure Module)에서의 동작원리를 표현한 것이다.
[그림 5]의 알고리즘에서 알 수 있듯이 알려진 패턴을 미리 정해놓은 데이터베이스의 리스트와 비교하고 분석한 패킷의 내용을 Black List와 비교한다. 이때 정상이라면 실행하고 공격 패턴과 일치한다면 실행권한을 주지 않고 커널 레벨에서 파일 시스템에 직접접근하여 네트워크 상의 패킷 캡쳐한다



[그림 5] 모듈의 기능

```
...
if( mod_str != NULL ) { // 알려진 패턴을 미리 지정해놓
      은 리스트와 비교, 빠른 검색
   for(i=0; i<BlackList; i++) {
      n = strncpy(mod_str, (char *)BlackList[i], // 분석한
      패킷내용을 Black List와 비교
         sizeof((char *)BlackList[i]));
      if( !n ) {
         return DENY;
      }
   }
   if( !Search_in_DB( mod_str ) ) { // 데이터 베이스에서
      의 검색
      return DENY;
   }
}
...
```

5. 시스템 비교 분석

다음의 표는 어플리케이션 단계에서의 IDS/IPS와 커널기반의 IDS/IPS를 비교한 것이다

[표 2] 시스템 비교분석

비교	IDS/IPS	커널기반의 IDnPS
Interface	Application	Kernel Module
Privilege	General as ROOT	Kernel
Priority	Middle or low	High
Known vulnerability	Intercept owner privilege by SetUID	Impossible yet, nothing but it has a serious problems
Exploitable	High	Impossible
Report	Hacking possible	Impossible yet

[표 1]에서 알 수 있듯 기존 일반 애플리케이션 기반의 IDS/IPS 프로그램은 우선 순위가 선점될 수 있는 가능성이 많으며 애플리케이션인 만큼 취약점이 존재. 따라서 취약점을 이용한 정보획득이나 root권한 획득이 이루어질 수 있다. 반면에 커널기반의 IDnPS는 커널에서 동작하는 만큼 어떤 특정 작업에 대해 우선 순위를 선점 할 수 있어서 사용자가 어떠한 작업을 수행했을 때 바로 수행하지 않고 이를 제어할 수 있다. 이 때문에 커널기반의 IDnPS가 일반 애플리케이션인 IDS/IPS에 비해 우수한 것을 알수 있다. 따라서 커널 기반의 IDnPS는 취약점 또한 낮으며 아직 까진 일반 애플리케이션처럼 해킹이 가능하지는 않다.

6. 결론 및 향후 연구

커널 기반의 IDnPS는 커널을 모두 건드리지 않고 주요한 부분만 우리가 원하는 함수로 replace 함으로써 코드의 간결화 및 빠른 실행(plug-in)을 유지할 수 있다. 이 모듈의 가장 큰 특징은 root라는 최고 권한을 가진 사용자 역시 관련 룰(rule)에 따라 시스템에 위협을 주는 행위는 용납되지 않는다. 즉, 탐지 모드(Detecting Mode)시 여러 행동에 대해 즉각 처리한다. 데이터베이스 연동으로 인해 검색시간 단축 및 많은 rule을 추가 할 수 있으며, 차후 여러 프로그램(Web, GUI Application)과 연동이 가능하다. 이 과정에서는 Web에서의 컨텐츠, Linux 와 Windows에서의 일반 애플리케이션으로 연동이 가능하며, 모듈의 시스템 로그에 관한 모든 내용은 Web interface로 출력해 관리 입장에서 편리하도록 구성하였다. Web 과 일반 애플리케이션으로, Web에서는 단순 결과를 통보 받고, 전용 애플리케이션에서는 제어와 결과에 대한 대응 및 rule-set을 포함한 시스템상의 추가/변경/수정을 가능케 하였다. 앞으로 더 연구해야 할 과제는 System Call Hooking을 이용해 다른 루틴으로 replace가 아닌 Kernel 소스 자체를 수정해 Secure OS를 만들어보는 것이다.
기대효과로서는 무엇보다도 시스템에 대한 안전한 환경을 제공한다. 더 나이가 secure OS의 기술발전에 기여하리라 기대된다.

참고문헌

[1] R. Buschkes, M. Borning, and D. Kesdogan, "Transaction based Anomaly Detection" Proc.of the Workshop on Intrusion Detection and Network monitoring, USENIX, Apr., 1999.
[2] Anup K. Ghosh, "Learning Program Behavior Profiles for Intrusion Detection", Proc. of the Workshop on Intrusion Detection and Network Monitoring, April., 1999.
[3] Samuel I. Schaen, "Network Auditing: Issues and Recommendations", IEEE 7th Computer Security Applications Conference, pp.66-79, Dec., 1991.
[4] T. Lane, "Filtering technique for rapid user

classification", In Proceedings of the AAAI98/ICML98 Joint Workshop on AI Approaches to Time series Analysis, 1998.
[5] U. Fayyad, G. Piatetsky-Shapiro and P.Smyth, "The KDD process of extracting useful knowledge from volumes of data", Communications of the ACM, 39(11):27-34, Nov., 1996.
[6] Abdelaziz Mounji, Baudouin Le Charlier, Denis Zampunieris and Naji Habra, "Distributed Audit Trail Analysis", Proc. 2000.
[7] P. Proctor, "Audit Reduction and Misuse Detection in Heterogeneous Environment; Framework and Application", Proc. 10th Annual Computer Security Applications Conference, Dec., 1994.
[8] Cheri Dowell and Paul Ramstedt. "The Computer Watch data reduction tool", In Proceedings of the 13th National Computer Security Conference, PP.99-108, Washington DC, Oct., 1990.
[9] Massimo Bernaschi, Emanuele Gabrielli and Luigi V.Mancini, "Operating System Enhancements to prevent the Misuse of System Calls", CCS'00 of ACM,pp.174-183, 2000.