情報保護學會論文誌
第13券 第3號, 2003. 6

# 감사로그 상관관계를 통한 호스트기반의 침입탐지시스템*

황 현 욱**, 김 민 수**, 노 봉 남**

# The host-based Intrusion Detection System with Audit Correlation

Hyun-Uk Hwang**, Min-Soo Kim**, Bong-Nam Noh**

요 약

침입탐지시스템의 침입 여부는 감사로그를 기반으로 판단되며, 그 성능은 감사로그를 바탕으로 침입 패턴에 대해 얼마나 정확하고 효율적으로 기술했느냐에 달려있다. 본 논문에서는 시스템 호출, 네트워크 패킷, Syslog의 정보를 통해 상관성을 도출하고, 상태전이 기반의 패턴과 이에 대한 규칙기반의 상관관계 패턴을 작성하였다. 이러한 상관관계를 이용하여 탐지율의 정확성을 높일 수 있었다. 특히, covert channel의 탐지 실험을 통해 상관관계 패턴을 통한 탐지가 가능함을 보였다.

ABSTRACT

The presence of the intrusion is judged by intrusion detection system based on the audit log and the performance of this system depends on how correctly and effectively it has been described about the intrusion pattern with audit log. In this paper, the relativity concerning intrusion is demonstrated among the information those are 'System call, Network packet and Syslog' and the related pattern of the state-transition-based method and those rule-based pattern is identified. By applying this correlation to them, the accuracy rate of detection was able to be improved. Especially, the availability of detection with correlation pattern through Covert Channel detection test has been substantiated.

**Keyword** : *IDS, correlation, audit log, pattern*

## Ⅰ. 서 론

침입탐지 시스템은 초기의 알려진 침입을 탐지하는 단순 모델에서, 점점 지능화되고 복잡한 침입을 판단하기 위한 다양한 모델이 제시되고 있으며, 네트워크 관점에서부터 호스트의 침입을 감시하는 보안의 기본요소가 되고 있다. 이러한 침입탐지 시스템은 네트워크의 부하(traffic), 시스템 감사기록, 네트워크 내의 다양한 호스트로부터 나오는 다양한 정보를 분석하여 침입을 감시하는 하나의 수단으로 사용된다[1,2].

침입탐지 시스템에서 침입 판단의 기본이 되는 것은 감사 자료이다. 감사 자료를 기반으로 침입탐지시스템을 분류하면, 단일 시스템에서 발생하는 사건(event), 사용자 활동(activity), 시스템 상황 등의 감사 기록 정보를 바탕으로 침입을 탐지하는 호스트 기반 침입 탐지 시스템이 있으며, 네트워크에 흐르는 패킷들의 정보를 바탕으로 침입을 탐지하는 네트워크 침입탐지 시스템이 있다.

* 본 연구는 대학 IT 연구센터 육성/지원사업의 연구결과로 수행되었습니다.
** 전남대학교 일반대학원 정보보호협동과정(LSRC) (hhu@lsrc.chonnam.ac.kr)

본 논문에서는 호스트 기반의 3가지 감사로그의 상관관계를 토대로 침입을 탐지하는 시스템을 제안한다. 감사로그는 정형화된 형태로 기록하여 탐지규칙에 쉽게 적용할 수 있도록 되어있다. 탐지규칙은 상태전이 방식으로 구성되고 패턴간의 상관관계를 규칙(rule)기반으로 표현하여 침입탐지 패턴이 완성된다.

기존의 시스템에서 단일 감사로그만을 토대로 탐지하는 경우는 있으나 복잡한 환경에서는 오탐율이 높아지고, 이상 탐지나 관리자 직관적인 측면에서의 탐지는 고려치 않은 것이 사실이다. 본 논문에서는 호스트기반의 3가지 감사로그의 상관성을 토대로 설계하였으며, Covert channel을 생성하는 Loki[3]를 탐지하도록 적용해 보았으며 탐지범위를 통해 다른 공격방식에 대한 탐지 방법도 기술하였다.

2장에서는 관련연구로서 로깅시스템들과 침입탐지시스템의 현황과 방법을 설명하고, 3장에서는 제안된 침입탐지시스템의 구조를 설명하고, 4장에서는 상관관계를 통해 Covert channel을 생성하는 Loki와 탐지범위를 설명하였고, 5장에서는 결론을 맺었다.

## II. 관련 연구

### 2.1 로깅(logging)시스템

감사로그를 기록하는 로깅시스템은 침입탐지 시스템의 가장 중요한 요소 중의 하나로 시스템 내에서 수행된 각종 응용 프로세스와 운영체제가 동작중에 중요한 정보와, 네트워크 통신 내역 등을 기록한다. 이러한 정보는 침입탐지 시스템의 근간으로 사용되어 침입여부를 판별하게 된다.

침입탐지 시스템에서는 분석 자료에 따라 네트워크기반 침입탐지 시스템, 호스트 기반 침입탐지 시스템, 또한 다수의 호스트로부터 수집된 로그를 기반으로 침입을 탐지하는 다중 호스트 기반의 침입탐지 시스템으로 구분한다.

네트워크 기반의 침입탐지 시스템은 네트워크에 흐르는 패킷들의 정보를 잡기 위해 네트워크 카드를 무차별모드(Promiscuous mode)로 설정한다[4]. 패킷 수집 방법은 운영체제마다 다르며, [표 1]과 같다.

호스트 기반의 침입탐지 시스템은 시스템의 사용내역, 시스템호출(system call) 정보를 사용한다. 일반적인 다중 호스트 기반의 침입탐지 시스템은 여러 호스트 기반의 침입탐지 시스템에서 얻은 정보를 토대로 종합 판정을 수행한다.

[표 1] 패킷필터의 종류

| 시스템 | 패킷 필터 |
|---|---|
| Windows 계열 | SMS(Systems Management Server) |
| BSD 계열 | BPF(Berkely Packet Filter) |
| SunOS 계열 | NIT(Network Interface Tap) |
| System V | DLPI(Data Link Provider Interface) |
| 시스템에 독립적 | Libpcap |

시스템 감사 자료로 Sun Solaris의 BSM(Basic Security Module)과 Windows NT 계열의 이벤트 로깅 메커니즘 등이 있다. 또한, 유닉스 계열의 운영체제에서는 syslogd가 제공해주는 풍부한 시스템 로그로서 wtmp, utmp, sulog, pacct, messages, secure 등이 있다.

### 2.2 침입탐지시스템의 기술 현황

근래의 침입탐지 시스템은 단일 환경, 단일 시스템을 넘어 다양한 형태의 침입을 탐지하기 위해 대규모 네트워크 기반에서의 감시 및 탐지, 침입여부에 대한 판정과 함께, 각 시스템이 제공하는 침입탐지 정보의 통합 분석을 통하여 광범위한 분석을 가능하게 하는 상호 협력의 침입탐지 시스템으로 발전하고 있다. 침입탐지 시스템의 발전 방향에서 보는 것처럼 하나의 탐지 센서로는 모든 공격 방법을 탐지하기는 힘들어 각각의 공격방법에 뛰어난 특성을 지닌 여러 개의 탐지센서를 설치하여 이를 종합 분석하여 신뢰적인 탐지결과를 가져오는 방향으로 발전하고 있다. 이러한 침입탐지 시스템들의 결과를 통합하고 판정하는 대표적인 연구로 DARPA/ITO에 의해 추진된 EMERALD, IBM의 TEC 그리고 MIRADOR의 CRIM이 있다[5,6].

이러한 상호협력의 침입탐지 시스템 연구와 함께 상관성(Correlation)에 대한 연구가 국외에서 진행되고 있다. 상관성은 명백한 상관성(explicit correlation of events)과 함축적인 상관성(implicit correlation of events)의 두 가지 접근 방식이 존재한다[7]. 명백한 상관성은 공격방식에 대한 정보를 충분히 보유하고 있다는 전제하에서 진행되며, 공격에 대한 정보는 공격과 관련된 감사데이터 하나에서부터 연속적으로 서로 연관된 감사데이터를 논리적 링크로 표현한다. 함축적 상관성은 각 감사 데이터내의 각 구성 정보들 간의 매핑(mapping)이나 관련성을 찾아내는 방식으로 분류, 데이터마이닝, 신경망 같은 학습기술이 사용된다[8].

### 2.3 침입방법

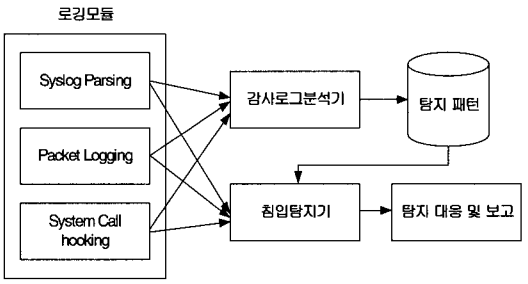
시스템을 위협하는 방법에는 위장(masquerading), 특성의 오용, 구현 결함, 시스템 설정 오류, 사회공학적인 방법 등으로 분류 할 수 있다[9].

위장 기법으로는 스니핑(sniffing), 스푸핑(spoofing) 등이 있다. 특성의 오용에는 서비스거부공격(DoS: Denial of Service)류의 공격방식을 포함한다. 구현 결함에는 대표적인 방법으로는 버퍼 오버플로우(buffer overflow) 공격과 포맷 스트링(format string) 공격과 free/malloc 공격이 대표적이다. 버퍼 오버플로우 공격이 1세대 공격이라면 포맷 스트링과 free/malloc 공격[10]은 2, 3세대 공격으로 떠오르고 있으며 성공하였을 때 관리자 권한을 얻게 된다. 또한 시스템에서 일어나는 CPU의 경쟁관계를 이용하여 임의의 파일을 생성하고 권한을 변경 할 수 있는 레이스컨디션 공격 방법이 있다. 설정 오류는 시스템의 권한이나 각종 응용프로그램의 접근정책 등으로 의도하지 않은 권한을 공격자에게 주는 경우를 말한다. 사회 공학적 방법은 다양한 물리적인 방법으로 공격자에게 시스템에 대한 유용한 정보가 유출되는 경우를 말한다.

위에서 본 것처럼 공격의 방법은 매우 다양하며, 갈수록 지능화 되고 있으며, 또한 현재 침입탐지 시스템을 우회하는 다양한 기법들이 개발되고 있다. 침입탐지 시스템을 우회하는 가장 널리 쓰이는 방법 중 하나는 가장 널리 쓰이는 패턴 매칭 기법의 우회하도록 패턴 규칙을 조금 벗어난 형태로 공격하는 것이다. 또한 TCP/IP 상의 패킷 단편화(fragment) 기법이나 Covert channel 형성을 통한 기법이 많이 이용되고 있다[11].

## Ⅲ. 제안된 침입탐지 시스템

본 논문에서 제안된 전체적인 침입탐지 시스템의 구조는 [그림 1]과 같다. 3가지 로깅모듈에서 수집된 감사로그를 토대로 감사로그 분석기에서 탐지패턴을 생성하고, 이 탐지패턴은 로깅시스템에서 생성된 감사로그의 상관관계를 토대로 침입탐지기에서 침입을 판단하는 패턴으로 사용되며 3가지 로그는 또한 실시간에서 침입을 판단하는 자료가 된다.




〔그림 1〕 침입 탐지 시스템 전체 구성도

본 논문에서 제안한 침입탐지 시스템은 Linux7.2 커널 2.4.18 버전에서 구현되었다.

### 3.1 감사로그의 통합

3가지 감사로그에서 수집된 자료는 침입탐지시스템이 해석하기 쉽게 정형화되는 과정을 거치게 된다. [표 2]는 감사로그의 형식을 6하 원칙에 근거하여 침입에 최적화된 감사로그를 얻기 위해 일정한 형식으로 도식화한 모습이다. 각 로깅시스템은 3.2절에서 한다.

[표 3]은 감사로그간의 상관성을 나타내고 있다. syslog parser와 LSM 그리고 패킷 로거(packet logger)가 시간 축을 중심으로 사건에 대하여 발생되는 감사로그의 일치되는 부분을 나타내었다. IP정보나 포트정보는 LSM과 패킷 로거 두 부분에서 상관성을 연결지을 수 있는 부분이며, LSM과 Syslog parser 는

〔표 2〕 6하 원칙에 근거하여 정형화 된 감사로그 형식

| | how | who | where | what | when | why | 기타정보 |
|---|---|---|---|---|---|---|---|
| 로깅시스템 | 메시지 유형 | 침입자ID | 위치 | 프로그램의미 | 발생시간 | 수행내용 | |
| packet logger | 연결프로토콜 | Source IP and Port | Destination IP and Port | Packet Flags | Time | Data | 시퀀스 번호 등 |
| LSM | System Call Number | User ID | Source Host | Process and Session ID | Time | Return Value | 소켓이나 경로정보 등 |
| syslog parser | Message Type | User ID | Host Name | Daemon program and Process ID | Time | Message | 메시지 카운트 등 |

프로세스 측면에서 PID나 UID 부분에서 서로 상관성을 연결지을 수 있는 부분임을 보여주고 있다.

〔표 3〕 감사 로깅 모듈에 따른 상관성 연결 정보

| 연결정보 | Src IP | Src Port | Dst IP | Dst Port | Time | protocol | sid | pid | uid | daemon |
|---|---|---|---|---|---|---|---|---|---|---|
| Syslog Parser | | | | | ○ | △ | | ○ | △ | ○ |
| LSM | ○ | ○ | △ | △ | ○ | | ○ | ○ | ○ | △ |
| Packet Logger | ○ | ○ | ○ | ○ | ○ | ○ | | | | △ |

## 3.2 감사자료 수집

본 논문에서 제안한 감사로깅 모듈은 크게 3가지 측면에서 동작한다.

- 네트워크에 흐르는 패킷을 기반으로 한 패킷 로거
- 시스템에서 동작하는 시스템호출을 기록하는 LSM (Linux Security Module)
- syslogd에서 발생시키는 메시지를 기반으로 하는 syslog parser

### 3.2.1 패킷 로거(packet logger)

호스트로 들어오거나 나가는 패킷에 대한 정보를 수집하여 네트워크에서 발생하는 침입패턴을 기술할 수 있는 로그를 제공하게 된다. IP, TCP, UDP, ICMP 프로토콜을 표현할 수 있으며 이에 따른 응용프로그램의 프로토콜을 표현하게 된다. 이 모듈은 tcpdump처럼 libpcap 라이브러리를 사용하여 패킷을 수집한다. 다른 점은 패킷에 대한 필터링 규칙, 기록하는 정보의 형식, 패킷의 세션별 분리 기능이다. 일반적인 네트워크 IDS와 달리 패킷 로거는 호스트 내로 출입하는 네트워크 패킷 정보만을 분석함에 따라 스위치 네트워크에서도 장애 없이 사용할 수 있으며 evasion attack이나 insertion attack에 대한 취약점도 나타나지 않는다[12].

패킷 로거에서 기록하는 정보는 필터링 규칙에 따라 다르며 네트워크 전송 데이터 내용만을 제외한 모든 내용을 기록 할 수 있다. 패킷 로거에서 기록되는 정보는 다음과 같은 모든 내용을 포함하여 정형화된 형식의 로그를 기록하게 된다.

- 기본정보 (시간, 장치명, 발신지 MAC 주소, 목적지 MAC 주소, 이더넷 종류)
- IP (발신지 IP 주소, 목적지 IP 주소, 버전, 헤더 길이, 서비스 종류, 전체 길이, identification, fragment offset, TTL, checksum)
- TCP (발신지 포트, 목적지 포트, 시퀀스 번호, 응답번호, window, 코드 비트, checksum, urgent pointer)
- UDP (발신지 포트, 목적지 포트, 길이, checksum)
- ICMP (ICMP 종류, 코드, checksum, ID, 시퀀스)

실제 기록되는 로그의 형태는 [표 4] (a)와 같다.

### 3.2.2 LSM(Linux Security Module)

LSM은 리눅스 시스템에서 솔라리스 BSM에서 처럼 시스템 상에서 일어나는 모든 시스템 호출(system call)을 기록하게 된다. 이는 호스트 내에서 발생하는 모든 시스템 호출을 기록하고 네트워크를 통해 일어나는 시스템 호출 또한 기록하게 된다[13].

LSM은 리눅스 시스템에서 커널을 사용할 때 발생하는 전체 시스템 호출을 커널에서 가져와 로그파일에 기록한다. 또한, 적재가능 모듈(loadable module)로 구현되어 쉽게 커널에 적재하거나 삭제할 수 있다. 이때 커널 수준의 필터링 규칙을 재설정 가능하도록 하여 여러 가지 필터링 규칙을 적용할 수 있다. 커널에 적재된 LSM 모듈에서는 시스템호출 정보를 커널 디바이스(/dev/audit)에 기록하고 LSM 데몬에서 그 정보를 사용자가 읽을 수 있는 로그 파일에 기록한다. 일반적으로 리눅스 시스템에서 제공하는 로그 정보는 syslogd나 klogd에서 기록하는 내용이

〔표 4〕 packet logger(a), LSM(b)과 syslog parser(c)의 로그 형태

| | |
|---|---|
| (a) | p:1::.A:.:.:.::005004C3D583:168.131.34.196:00402B1B0815:168.131.34.163:20:3201:20021121203543:2525978041:64<br>p:1::.A:P:.:.:.::00402B1B0815:168.131.34.163:005004C3D583:168.131.34.196:3201:20:20021121203543:2025756972:128<br>p:1::.A:.:.:.:.::005004C3D583:168.131.34.196:00402B1B0815:168.131.34.163:20:3201:20021121203543:2525978041:64 |
| (b) | l:setpgid :2145,1902,1902,hhu,hhu,hhu,hhu,hhu :0 : 20020109152721 :<br>l:execve :2145,1902, 1902,hhu,hhu,hhu,hhu,hhu :0 : 20020109152721 : exec ? , path./sendmail_attack<br>l:open :2145,1902, 1902,hhu,hhu,hhu,hhu,hhu : 4294967294 : 20020109152721: path /etc/ld.so.preload |
| (c) | s:10100002: 7361, nooree:210.103.122.2: ipop3d:20021121103639:<br>s:10010012: 7363, :168.131.33.33: sendmail:20021121103746:<br>s:10010001: 7364,hhu@athena.chonnam.ac.kr:127.0.0.1: sendmail:20021121103746: |

전부이나, LSM은 syslogd나 klogd에서 제공하지 않는 시스템 호출 정보를 기록한다.

LSM에서 기록되는 정보는 다음과 같다.

- 시스템 호출 이름
- 수행 시간
- 수행 프로세스 정보(pid, ppid, pgid, sid, uid, gid, euid, egid)
- 시스템 호출 결과 반환 값
- 파일 접근시 파일 관련정보(owner, group, permission)
- 소켓 연결시 연결 호스트 정보(ip address, port)

실제 기록되는 로그의 형태는 [표 4] (b)와 같다.

### 3.2.3 Syslog parser

syslog parser는 리눅스 시스템이 제공하는 syslogd과 klogd에서 제공하는 여러 로그파일에서 보안과 관련된 정보를 수집하게 된다. 이러한 정보는 리눅스 시스템의 '/var/log' 밑에 쌓이게 되며 messages, secure, maillog 등의 로그 메시지를 재해석하여 기록한다. syslog parser는 flex와 yacc을 이용하여 수행된다.

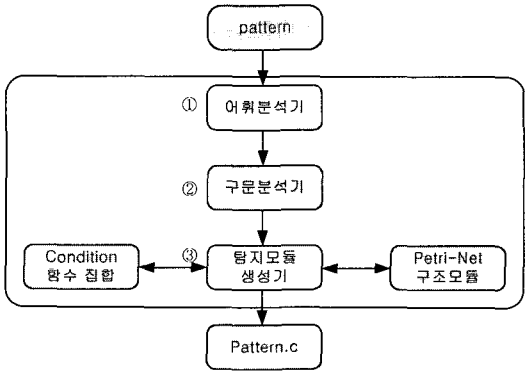
syslog parser에서 기록하는 내용은 아래와 같다.

- 발생 이벤트 타입
- 이벤트 발생 데몬 프로세스, 프로세스 ID
- 발생 시간
- 발생 호스트
- 관계 정보 (uid, ip address, terminal type, file name, etc.)

실제 기록되는 로그의 형태는 [표 4] (c)와 같다.

### 3.3 감사로그 상관관계

공격을 탐지하기 위해서는 탐지패턴이 정의되어야 한다[14]. 탐지패턴이 정의되면 탐지패턴의 형식에 따라 패턴 파일이 생성되어 탐지모듈 생성기로 들어가 처리되는 구조도로 [그림 2]와 같은 단계로 수행된다.

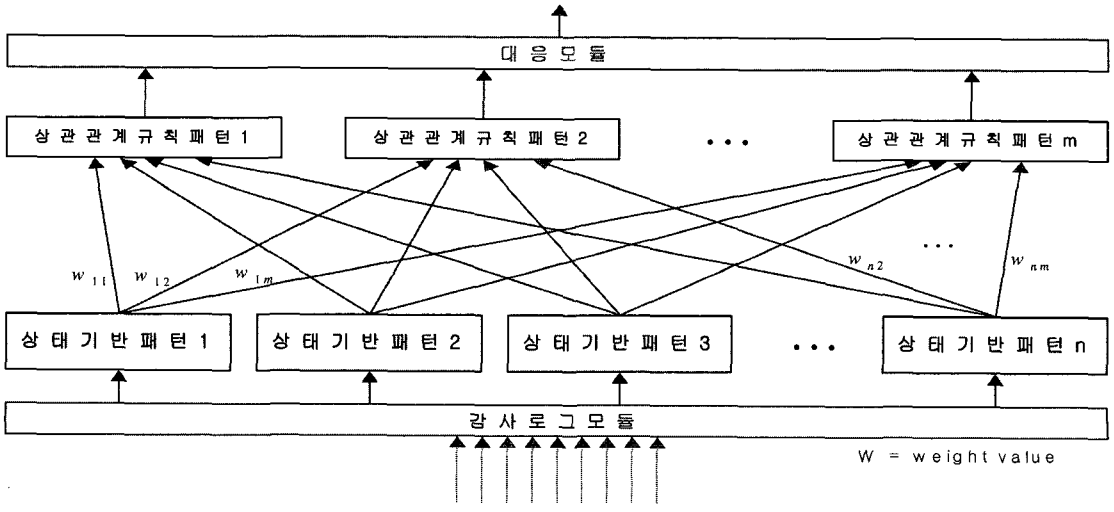



〔그림 2〕 탐지 모듈 생성기 구조

① 어휘분석기에서 탐지패턴을 읽어 들이고 정해진 토큰을 추출한다.

② 구문분석기에서 탐지패턴의 문법오류를 검사하고, petrinet의 플레이스(place)에 확률 값을 적용하는 PPN(Probability Petri Net)[14] 형태의 상태전이 패턴으로 변환한다.

③ 탐지패턴을 탐지모듈 생성기에 적용하여 만들어지는 파일(pattern. c)로 각 사건마다 함수가 만들어




〔그림 3〕 상관관계 패턴 구성도

지며, 트렌지션(transition)[14]과 플레이스에 대한 확률 값과 상태 값이 기록된다. pattern.c는 침입탐지 기로서의 역할을 한다.

감사로그 상관관계를 나타내는 구성도는 [그림 3]과 같다. 시스템에서 발생하는 감사로그들이 감사로그 모듈을 통해 전달되고, 3가지 감사로그에 따라 기술된 상태기반 패턴들이 상관성을 토대로 규칙기반 패턴으로 기술되어 침입을 판단하게 된다. 이 침입을 판별하는 패턴들은 하나의 상태기반 패턴으로 판단할 수도 있으며, 두 개나 세 개로 상관성이 결합된 상관성 패턴을 만들어 침입을 판단 할 수도 있다.

### 3.4 상관관계 패턴을 이용한 탐지 예

탐지패턴은 [표 5]와 같은 구조로 기술된다[14]. [표 5]에서 일반 패턴구조는 3가지 로깅 모듈의 패턴을 기술하는 구조이며, 상관관계 패턴 구조는 3가지 로깅 모듈의 상관관계를 기술하는 패턴 구조를 나타낸다. 각 로깅 모듈에 따라서 각각의 패턴을 추가 할 경우 각 일반 패턴이 추가되며, 상관관계에 따른 패턴이 추가될 경우 역시 상관관계 패턴을 추가하면 된다.

〔표 5〕 패턴 구조

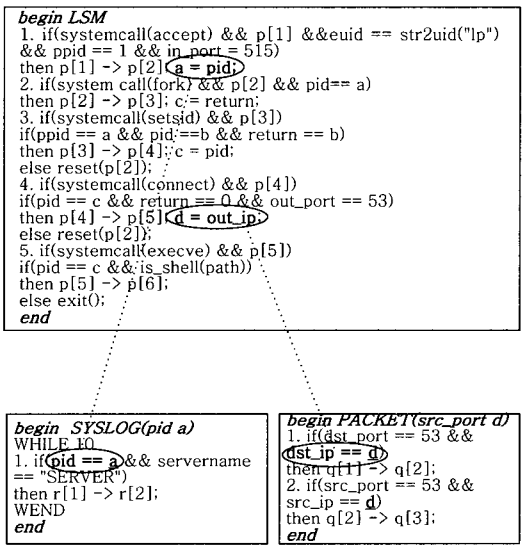
| | |
|---|---|
| 일반 패턴 구조 | PATTERN pattern_name BEGIN<br>event_list<br>PEND |
| 상관관계 패턴 구조 | CORRELATION pattern_name BEGIN<br>event_list<br>CEND |

lpd-formatstring 공격(이하 lpd 공격)을 통해 상관관계를 이용한 탐지패턴을 기술해 보았다. lpd 공격은 프로그램내의 syslog()함수의 포맷스트링 버그에 의한 "%s%s%s%s..."의 입력 값을 받았을 때 원격에서 루트 권한을 침해당하게 되는 공격으로 원격에서 공격을 통하여 root 권한을 획득하는 원격 포맷스트링(remote-formatstring) 공격의 하나이다. lpd 공격을 실행한 후 LSM의 시스템 호출 정보의 상태전이 방법을 표현하면 [그림 4]와 같다. [표 6]에서 보는 것처럼 시간대를 중심으로 세 가지 로깅시스템의 정보를 연결지을 수 있으며, 패킷 로거와 LSM은 도착지 IP와 도착지 port를 통해 연관된 정보를 얻을 수 있다. 또한 LSM과 syslog parser는 time과 pid를 통해 정보를 연결할 수 있다.

〔표 6〕 lpd 공격의 상관관계

| 연결정보 | dst IP | dst port | time | pid |
|---|---|---|---|---|
| packet loggger | ○ | ○ | ○ | |
| LSM | ○ | ○ | ○ | ○ |
| syslog parser | | | ○ | ○ |

[그림 4]는 lpd 공격을 기술한 공격 패턴이다. 이를 if-then-else를 통한 간단한 규칙 기반 접근방법으로 표시하고 그에 대한 상관성을 보여주고 있다.




〔그림 4〕 lpd 공격 상관성

3가지 로깅 모듈은 각각의 패턴으로 표현될 수 있으며, LSM의 시간 축을 중심으로 LSM의 pid와 syslog parser의 pid, LSM의 destination ip와 패킷 로거의 destination ip가 일치하여 로깅 모듈에 기록됨을 볼 수 있다. 로깅 모듈 각각의 패턴구조로 표현하고 상관관계 패턴구조로 표현하면 다음과 같다.

〔표 7〕 lpd 공격의 LSM 패턴

```
PATTERN ptn_lpd_BO_LSM BEGIN
    LSM 1029 : isAccept() AND localport(515);
    LSM 2 : equalPID() AND isUID("root") AND
saveReturnPID();
    EXITON LSM 1 : equalPID();
    LSM 66 : isChild() AND cmpReturnPID();
    LSM 1027 : equalPID() AND isConnect() AND
success();
    LSM 11 : equalPID() AND isEUID("root") AND
is_shell();
PEND
```

[표 7]은 LSM 로깅 모듈을 표현한 모습으로 원격에서 들어온 공격의 상태를 상태전이 형태로 표현한 모습이다.

① 처음의 상태는 accept 시스템호출(1029) 발생을 시작으로 하며, 이는 lp라는 euid를 가지고 부모프로세스가 1이다. 원격에서 프린터포트인 515번 포트로 들어온다.

② 다음 fork 시스템호출(2)이 발생한다.

③ setsid 시스템호출(66)이 발생하고 ppid와 pid 그리고 return value를 체크한다.

④ connect 시스템호출(1027)이 발생하고 외부의 53번 포트를 통해 DNS 질의를 요청한다.

⑤ execve 시스템호출(11)이 발생하고 쉘이 뜨는 것을 확인할 수 있다.

〔표 8〕 lpd 공격의 패킷 로거 패턴

```
PATTERN ptn_lpd_BO_PL BEGIN
   PACKET 53 : equalHostIP();
PEND
```

[표 8]은 패킷 로거 로깅 모듈의 상태로서 도메인 질의를 하는 lpd 공격의 패킷을 탐지한다.

〔표 9〕 lpd 공격의 syslog parser 패턴

```
PATTERN ptn_lpd_BO_SL BEGIN
   SYSLOG 0x0F01: SearchString("SERVER");
PEND
```

[표 9]는 syslog parser 로깅 모듈의 상태로 /var/log/messages 파일에 특정 형태의 로그를 남긴 형태를 syslog parser 로깅 모듈에서 "SERVER"라는 특정 문자열을 탐지하여 공격 정보를 탐지하게 된다.

세가지 모듈의 lpd 공격의 특징을 잡아 각각의 모듈에서 패턴으로 유지하고, 3가지 모듈에서 탐지한 패턴을 [표 10]과 같이 상관관계 패턴으로 기술하게 된다.

[표 10]은 상관관계를 표현한 상관성 패턴으로, ①에서 LSM과 패킷 로거의 30초 시간(TIME)과, 침입자 IP(HOST)의 일치로서의 상관성과, ②에서 LSM과 syslog parser의 30초 시간과, PID의 일치로서의 상관성을 표현 하였다. 각 모듈 간 AND 연산되어 참이 되면 상관성 패턴의 임계 값(threshold value)이 설정되어 공격을 탐지하게 된다.

〔표 10〕 lpd 공격의 상관성 패턴

```
CORRELATION cor_lpd_BO BEGIN
   SAME(HOST, TIME(30)) 50: ptn_lpd_BO_LSM(80) AND
ptn_lpd_BO_PL(50); .....①
   SAME(PID, TIME(30)) 50: ptn_lpd_BO_LSM(80) AND
ptn_lpd_BO_SL(60); .....②
CEND
```

## Ⅳ. 상관관계를 이용한 침입탐지

### 4.1 상관관계의 적용의미

침입탐지 시스템에서 복합적이고 지능적인 공격을 탐지하는데 있어 단일 로그 시스템은 한계성을 가지고 있다. 침입탐지시스템의 제한적인 환경 내에서 최적의 결과를 끌어내기 위해서는 침입탐지시스템의 여러 가지 통합적인 분석을 통해 침입탐지 능력을 향상시키는 방법이 요구되어진다. 본 논문에서는 침입탐지 시스템의 탐지율을 높이는 방안으로 감사로그 시스템의 다양한 정보를 제공하고, 이 3가지 모듈인 네트워크 관련 모듈인 패킷 로거, 시스템 호출 관련 모듈인 LSM, 시스템 자체에서 제공하는 모듈인 syslog parser의 3가지 모듈의 상관성을 통해 침입탐지 능력을 향상시키고자 하였다.

공격의 탐지는 각 감사모듈의 패턴을 중심으로 판단된다. 침입탐지는 하나의 감사모듈에서 탐지할 수도 있고, 3가지 감사모듈이 상관성을 통해 나타난 결과로 탐지할 수도 있다. 상관성을 통한 탐지는 한가지 감사 모듈에서 판단할 수 있는 과탐지와 미탐지의 잘못된 결과를 줄여 정확성을 높일 수 있으며, 일반적인 정상행위와 구별하기 힘든 패턴도 다른 감사모듈과의 상관성을 통해 공격을 판단할 수 있게 된다. 본 논문에서 제안한 상관성 패턴 모듈은 하나의 감사로그 패턴만으로도 침입을 판단할 수 있으며, 3가지 감사로그의 모든 상관성을 연결지을 수도 있고, 4.2절에서 기술한 Loki 탐지 상관패턴을 만들어 내는 것처럼 2가지만으로도 상관 패턴을 만들 수 있다.

3.4절에서 소개한 lpd 공격은 3가지 로깅 모듈의 전체 상관관계를 표현하기 위해 기술했다. lpd 공격은 단일 로그 침입탐지 시스템에서도 탐지가 가능한 것으로, 상관성의 효율성을 좀 더 살펴보기 위해 4.2절에서 covert channel을 생성하여 이루어지는 트로이 목마 종류의 Loki 공격에 대해 상관성을 이용한 방법으로 탐지하는 패턴을 기술하고 시험해 보겠다.

### 4.2 covert channel 에 대한 패턴 - Loki

Loki 공격의 기본적인 개념은 임의의 데이터를 ICMP_ECHO 요청 패킷의 데이터 영역에 숨겨서 전송하게 되며 Loki 데몬에서는 이를 해석하여 통신 채널을 유지하게 된다. 따라서 대부분의 방화벽이나 침입탐지시스템에서 ICMP_ECHO 패킷을 거르지 않기 때문에 해커는 원하는 정보를 ICMP 패킷을 통해 숨길 수 있게 된다. 또한 데이터 영역의 암호화를 지원하여 snort와 같이 패킷을 스니핑하여 탐지하는 경우 어떤 데이터인지 판단하지 못하게 되어 탐지하기가 매우 힘들다.

Loki를 탐지하기 위해 3가지 감사모듈에 나타난 특징을 추출해보면 LSM과 패킷 로거에서 탐지 패턴을 추출할 수 있다. Loki 공격은 원격에서 명령을 실행하는 순간 흐르는 ICMP 패킷과, 그때 호스트 상에서 발생되는 시스템호출의 흔적을 추적하여 나타낼 수 있다.

Loki 공격에 대한 각 감사로그의 패턴과 상관관계 패턴을 기술 하면 다음과 같다.

〔표 11〕 Loki 공격의 LSM 패턴

```
PATTERN ptn_Loki_LSM BEGIN
    LSM 2 : saveReturnPID();  #fork()
    EXITON LSM 1 : equalPID() ;
    LSM 42 : cmpReturnPID() AND success();  #pipe()
    LSM 11 : is_shell() AND success();  #execve()
    LSM 11 : equalPID() AND success();  #execve()
    WHILE 3
      LSM 162 : success() ; #nanosleep()
      LSM 1035 : equalPID() AND success(); #sendto()
    WEND
PEND
```

Loki에 대한 LSM의 탐지 패턴은 [표 11]과 같이 구성된다. 외부에서 icmp를 통해 명령어가 전달되면 Lokid가 존재하는 서버에서 fork()를 통해 프로세스를 생성하고 pipe()를 호출하고 쉘을 실행시켜 원격에서 하달된 명령어를 Lokid가 존재하는 서버에서 실행하고, 명령어가 올 때 nanosleep()과 sendto()가 시스템 호출이 병렬적으로 수행되어지는 패턴이다.

〔표 12〕 Loki 공격의 패킷 로거 패턴

```
PATTERN ptn_Loki_PL BEGIN
   PACKET : isTYPE(8) AND isCODE(0) AND saveID() ;
WHILE 4
   PACKET : isTYPE(0) AND isCODE(0) AND equalID() ;
WEND
PEND
```

패킷 로거의 탐지패턴은 [표 12]와 같이 구성된다. 위 패턴은 하나의 echo request 패킷이 호출되고 echo reply 패킷이 명령어의 크기만큼 보내지는 모습의 패턴이다.

〔표 13〕 Loki 공격의 상관성 패턴

```
CORRELATION cor_Loki BEGIN
  SAME(TIME(10)) 100: ptn_Loki_LSM(80) AND
                      ptn_Loki_PL(80);
CEND
```

Loki의 상관성 패턴은 [표 13]과 같이 구성된다. Loki 공격은 LSM 모듈과 패킷 로거의 패턴으로 공격이 판단될 수 있으며 이에 대한 상관성 패턴을 기술하면, 시간 축을 중심으로 10초의 상관성을 유지하면서, LSM의 임계 값 80(100을 기준으로 하였을 때)과 패킷 로거의 임계 값 80이 만족되면(AND) 침입임을 판단하게 된다. 임계 값은 전체를 100으로 하였을 때 각 패턴의 개수로 나눈 값이며, 여기서는 5개의 패턴이므로 전이가 이루어질 때마다 20씩 증가하며, 침입의 기준이 되는 80은 경험치에 대한 값이며, 80보다 작은 값에서 판단하였을 때는 침입 확률도 그만큼 낮아지게 된다.

```
[root@asadal mids.1.0.5]# ./AID
Making Event Table...
This system name is asadal
Check 1Detecting Intruder...
logfile[0] name is "/var/log/audit/lsm.dat"
logfile[1] name is "/var/log/audit/syslog.dat"
logfile[2] name is "/var/log/audit/plog.dat"
3 Audit log files are opened.
Time is Sun Jan 12 20:20:07 2003
[0.00][0.20][0.00][0.00][0.00][0.00][0.00][0.20][0.00][0.00] ---①
Time is Sun Jan 12 20:24:28 2003
[0.00][0.40][0.00][0.00][0.00][0.00][0.00][0.40][0.00][0.00] ---②
Time is Sun Jan 12 20:25:31 2003
[0.00][0.60][0.00][0.00][0.00][0.00][0.00][0.60][0.00][0.00] ---③
Time is Sun Jan 12 20:29:37 2003
[0.00][0.80][0.00][0.00][0.00][0.00][0.00][0.80][0.00][0.00] ---④
Time is Sun Jan 12 20:39:25 2003
[0.00][1.00][0.00][0.00][0.00][0.00][0.00][1.00][0.00][0.00]----⑤
```

(그림 5) Loki 공격에 대한 탐지

[그림 5]는 침입탐지시스템이 실시간으로 Loki 공격을 탐지하는 모습을 보여주고 있다. 침입은 각 패턴에서 추출한 임계 값을 기준으로 패턴간의 AND 연산과, 각 상관성 패턴에서 정한 임계 값을 통해 결정되게 된다. 이 값이 임계 값 이상이 되면 침입이라고 판단하게 된다. 침입 판정의 임계 값은 1.0이 아

넌 더 작은 값에서 판단하게 되며, 이는 완전한 침입이 이루어진 후 판단해 시스템에 악의적인 행위가 이미 발생됨을 예방하는 차원과 실시간 탐지에서 공격이 완전히 이루어지기 전에 탐지함으로써 예방할 수 있는 여유를 가지게 된다. [그림 5]에서 ④의 임계 값 0.80을 넘어서는 순간 침입을 탐지하게 된다.

### 4.3 탐지 범위

상관관계를 통한 탐지범위는 기존의 단일 로깅시스템의 한계를 뛰어넘는데 있다. 일반적인 네트워크 공격이나 호스트 관점에서의 공격은 단일 로그모듈에서 판단 할 수 있으며, 본 실험결과에서는 정상행위와 유사하거나, 혼합 공격으로 인하여 하나의 로깅시스템으로 판단하기 힘든 부분을 다른 로깅시스템에서 나타나는 특징과 상관관계를 통해 추출하여 탐지가 가능한 부분에 대하여 기술하였다.

[표 14]는 본 논문에서 개발한 침입탐지 시스템에서 상관성을 통해 탐지 가능성이 높아진 공격 유형을 도식화 하였다.

〔표 14〕 상관관계를 통해 탐지한 공격 유형
P: packet logger, L: LSM, S: syslog parser

| 공격 유형 | P | L | S |
|---|---|---|---|
| UDP client | O | O | |
| three failed login | O | O | O |
| 버퍼오버플로우 공격 | O | O | △ |
| 포맷스트링 공격 | O | O | △ |
| process table 공격 | O | O | |
| 패킷 스니핑 | | O | O |
| 포트 스캔 | O | | △ |

UDP client는 백도어 형태로 DDos agent들이 특정한 시간에만 신호를 날려 통신을 유지하는 형태로 패킷 로거와 LSM을 통해 탐지 가능하며, three failed login은 3번 이상 시스템에 login을 실패하는 행위로서 패킷 로거와 LSM, syslog parser를 통해 가능하며, 일반적인 local bof(buffer overflow)나 remote bof는 패킷 로거, LSM, syslog parser의 상관관계를 통해 탐지 가능하다. 포맷 스트링 공격 역시 local, remote 공격에 있어 3가지 로깅 모듈의 상관성으로 탐지가능하다. process table 공격은 외부에서 프로세스를 점유하여 자원을 소모시키는 공격형태로 패킷 로거와 LSM을 통해 탐지 가능하며, 패킷 스니핑은 LSM과 syslog parser를 통해 탐지 가능하였다. 포트 스캔은 패킷 로그와 부분적인 기법에 따라 syslog parser에 나타났다. O는 해당 로그가 발생한 경우이며, △는 공격 방식에 따라 발생되는 경우도 있고, 그렇지 않은 경우를 나타낸다.

## V. 결 론

기존의 단일로그 침입탐지 시스템은 한 가지 감사로그의 정보만으로 침입을 판단하므로 삽입, 회피 등의 우회 공격과 혼합 공격에 있어 한계성이 존재하였다. 따라서 본 논문에서는 호스트에서 얻을 수 있는 3가지 감사로그시스템을 통해 호스트 기반의 침입탐지 시스템을 설계하여 침입탐지에 있어 정확도를 높이는 방안에 대해 제안하였다.

3가지 로깅시스템에서 얻어진 정보를 토대로 정형화된 방안을 기술하고 침입탐지 시스템에 있어서의 효율적인 감사로그 알고리즘을 개발하기 위해, 각각의 공격에 대한 침입패턴을 기반으로 3가지 로깅시스템에서 상관성을 추출하여, 기존의 감사 시스템에서보다 정확성과 효율성을 높이기 위해 상태기반의 패턴과 그 패턴을 규칙기반의 상관관계 패턴으로 표현하는 알고리즘을 통해 구현하였다. 이를 증명하기 위해 lpd 포맷스트링 공격과 covert channel을 생성하는 Loki 공격을 탐지하는 실험을 통해 상관관계 패턴의 탐지를 실험으로 보였으며, 다양한 공격에 대해 탐지 범위를 기술하였다.

향후 보다 다양한 침입패턴에 대한 상관성을 기술하고 이를 토대로 비정상행위에 대한 패턴을 기술할 수 있도록 하며, 패턴언어를 정형화하여 보다 쉽고 간편하게 기술할 수 있도록 하고자 한다. 또한 로깅시스템의 필터링 규칙을 정형화 하여 로그처리에 걸리는 시간을 단축하고, 룰의 간소화로 시스템 부하에 대한 부담을 줄이는 연구를 진행하고자 한다.

## 참 고 문 헌

[1] D. Denning "An Intrusion Detection Model", IEEE Transactions on Software Engineering. No.2, Feb. 1987.

[2] M. Esmaili, R. Safavi-Naini, and J. Pieprzyk, "Intrusion Detection : a survey", International Conference in Computer Communication, pp.409~414, 1995.

[3] daemon9, "L O K I 2(the implementation)", Phrack Magazine Volume 7, Issue 51 September 01, 1997.
[4] R.Sekar, Y. Guang, T. Shanbhag and S. Verma, "A High Performance Network Intrusion Detection System", *ACM Computer and Communication Security Conference,* 1999.
[5] Peter G. Neumann and Phillip A. Porras, "Experience with EMERALD to Date", Proceeding of the Workshop on Intrusion Detection and Network Monitoring, April, 1999.
[6] P. A. Porras and R. A. Kemmerer, "Penetration state transition analysis : A rule-based intrusion detection approach", Proc. 8th Annual Computer Security Application Conference, Dec. 1992.
[7] F. Cuppens, A. Miege, "Alert Correlation in a Cooperative Intrusion Detection Framework", IEEE Symposium on Security and Privacy, May 2002.
[8] Christopher Krugel, Thomas Toth, and Clemens Kerer, "Decentralized Event Correlation for Intrusion Detection", Pre-Proceedings of ICISC 2001.
[9] Kristopher Kendall, "A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems", May 1999.
[10] anonymous, "Once upon a free()", Phrack Magazine Volume 9, Issue 57 August 11, 2001.
[11] 손태식, 김진원, 박일곤, 문종섭, 박현미, 김상철, "보안 솔루션에 대한 우회 공격 기법 분석 연구", 한국정보보호학회 학술대회, pp.324~327, November 2002.
[12] Thomas H. Ptacek, "Insertion, Evasion, and Denial of Service:Eluding Network Intrusion Detection", Secure Networks Inc, 1998.
[13] 박남열, 송춘환, 김정일, 노봉남, "리눅스 보안 모듈설계 및 구현", 제1회 정보보호 연구회 논문발표집, pp.51~54, February 2001.
[14] 김민수, 은유진, 노봉남, "UNIX 환경에서 퍼지 Petri net을 이용한 호스트 기반 침입탐지 시스템 설계", 정보처리논문지, 제 6권, 제 7호, 1999.

## 〈著者紹介〉

**황 현 욱 (Hwang Hyun-Uk)**
2000년 : 조선대학교 정보통신공학과 졸업(학사)
2002년 : 조선대학교 대학원 전자공학과 졸업(공학석사)
2002년~현재 : 전남대학교 대학원 정보보호협동 박사과정 재학
<관심분야> 시스템 보안, 네트워크 보안, 정보보안, 포렌식스, 해킹 등

**김 민 수 (Kim Min-Soo)**
1993년 : 전남대학교 전산통계학과 졸업(학사)
1995년 : 전남대학교 대학원 전산통계학과(이학석사)
2000년 : 전남대학교 대학원 전산통계학과(이학박사)
2000년~2001년 : 한국정보보호진흥원 선임연구원
2001년~현재 : 전남대학교 리눅스시스템보안연구센터 객원교수
<관심분야> 시스템 보안, 네트워크 보안, 정보보안, 신경망 등

**노 봉 남 (Noh Bong-Nam)**
1978년 : 전남대학교 수학교육과 졸업
1982년 : KAIST 대학원 전산학과(이학석사)
1994년 : 전북대학교 대학원 전산통계학과(이학박사)
1983년~현재 : 전남대학교 컴퓨터정보학부 교수
<관심분야> 객체지향시스템, 통신망관리, 정보보안, 시스템 및 네트워크 보안 등