넷필터 프레임워크를 이용한 침입 탐지 및 차단 시스템 개발*

백승엽** · 이건호** · 이 극**

요 약

네트워크 및 인터넷시장의 발전과 더불어 침해사고도 급증하고 있으며 그 방법도 다양해지고 있다. 이를 방어하기 위해 여러 가지 보안시스템이 개발되어 왔으나 관리자가 수동적으로 침입을 차단하는 형식을 띄고있다. 본 논문에서는 침입탐지 시스템의 패킷 분석능력과 공격에 대한 실시간 대응성을 높이기 위하여 리눅스 OS에서 방화벽기능을 담당하는 넷필터를 이용해 침입탐지 및 차단시스템을 설계 하였다.

A Development of Intrusion Detection and Protection System using Netfilter Framework*

Seoung Yub Baek** · Geun Ho Lee** · Geuk Lee**

ABSTRACT

Information can be leaked, changed, damaged and illegally used regardless of the intension of the information owner. Intrusion Detection Systems and Firewalls are used to protect the illegal accesses in the network. But these are the passive protection method, not the active protection method. They only react based on the predefined protection rules or only report to the administrator. In this paper, we develop the intrusion detection and protection system using Netfilter framework. The system makes the administrator's management easy and simple. Furthermore, it offers active protection mechanism against the intrusions.

Key words : Netfilter Framework, Intrusion Detection System

* 본 연구는 2005 한남대학교 교비 연구비 지원으로 이루어졌습니다.
** 한남대학교 컴퓨터공학과

1. 서 론

컴퓨터의 급속한 발전과 초고속 인터넷의 보급은 사람들에게 편리성을 제공하는 반면, 보안이라는 중대한 문제점을 안겨주었다. 초고속 통신 보급의 확산으로 인하여 각 기관이나 업체뿐만 아니라 각 가정의 컴퓨터도 바이러스나 인터넷 웜, 해킹 등의 위험에 노출되고 그에 따른 악영향이 증가하고 있다. 이를 방어하기 위해 여러 보안시스템도 개발되어 왔으나 이를 대처하기 위한 문제는 쉽게 해결할 수 없는 실정이다.

침입탐지 시스템은 정보시스템을 실시간으로 모니터링하고 분석하여 침입을 탐지하는 역할을 수행하며 하나의 패킷에 대한 분석과 탐지를 할 수 있다. 하지만 다수의 패킷을 이용하는 세션 기반 탐지는 수행하기 어렵다는 문제점이 있고, 네트워크 IDS는 실시간 패킷감지를 하지만 공격을 차단하지 못하는 문제점을 가지고 있다. 침입차단시스템 역시 네트워크상에 있는 패킷들을 감지하지만 차단하지 못하기 때문에 대부분의 패킷은 관리자가 침입에 대응하기 전에 공격하고자 하는 목적지에 도달하여 침입에 성공하게 된다.

이러한 침입탐지 시스템과 방화벽의 단점을 보완하고 관리자의 관리 편의성 및 신속하고 빠른 조취를 수행하고자 본 논문에서는 리눅스 시스템에서 패킷을 필터링하는 기능을 담당하는 넷필터 프레임워크를 이용하여 침입이 탐지되면 해당하는 패킷을 실시간으로 차단할 수 있는 시스템을 설계하였다. 2장에서는 기존의 침입탐지시스템과 침입차단시스템에 대해서 분석하였고, 3장에서는 시스템의 설계 및 동작원리에 대하여 설명하였으며, 4장에서는 결론에 대해 기술하였다.

2. 관련 연구

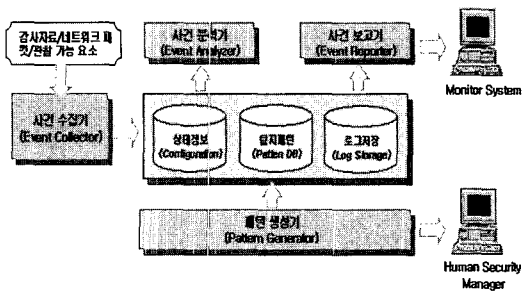
본 절에서는 네트워크상에서 침입을 탐지하고 차단하는 시스템을 개발하는데 필요한 기존의 기법들을 살펴보도록 한다.

2.1 침입탐지 시스템

침입탐지 시스템(IDS : Intrusion Detection System)은 허가되지 않은 사용자로부터 접속, 정보의 조작, 오용, 그리고 남용 등 컴퓨터 시스템 또는 네트워크 상에서 시도됐거나 진행중인 불법적인 예방에 실패한 경우 취할 수 있는 방법으로서 의심스러운 행위를 감시하여 가능한 침입자를 조기에 발견하고 실시간 처리를 목적으로 하는 시스템이다[1].

2.1.1 침입탐지 시스템의 구조

침입탐지 시스템의 전반적인 구조는 정보수집 단계, 정보가공 및 축약단계, 분석 및 침입탐지, 그리고 보고 및 조치단계로 구성되어 있다. 정보수집단계 에서는 호스트나 네트워크 패킷과 같은 정보를 이용하여 데이터를 수집하며, 수집된 데이터는 특정 침입 탐지 모델을 적용하여 분석된다. 분석 결과는 보고 및 조치단계를 통하여 대응 및 후속조치를 수행하게 된다. (그림 1)은 침입탐지 시스템의 기본 구성 요소들을 보여 주고 있다.



(그림 1) 침입탐지 시스템의 기본 구성

「감사자료/네트워크 패킷/관찰가능 요소」에 대

한 사건의 발생은 사건 수집기에 의해서 수집되어, 사건 분석기에서 침입여부에 대한 분석을 수행하게 된다. 이러한 과정에서 탐지규칙에 대한 패턴 데이터베이스와 로그가 이용되며, 침입이 발생한 경우 이는 사건 보고기에 의한 정보전달을 통하여 시스템 관리자에게 침입사실에 대한 정보들을 전달하게 된다. 또한 침입탐지 시스템의 신뢰성을 높이기 위해서는 일반적인 서비스와 침입에 대한 패턴을 주기적으로 갱신할 필요성이 있으며, 보안 관리자는 패턴발생기(Pattern Generator)를 통해서 새로운 공격방식에 대한 탐지규칙의 생성 및 관리역활을 수행한다.

2.1.2 침입탐지 시스템의 분류

침입탐지 시스템은 각각의 기능요소별로 어떠한 방식을 채택했는지에 따라서 다양하게 분류가 가능하다. (그림 2-9)와 같이 감사자료에 의한 분류, 탐지 방법에 의한 분류, 자료 공급자에 의한 분류, 탐지 시간에 의한 분류, 감사 자료 분석에 의한 분류, 대응 방법에 의해 IDS가 분류되고 있다.

(1) 감사자료에 따른 분류(Audit Source)

침입탐지 시스템에서 감사자료의 생성에 기반이 되는 데이터를 기준으로 구분한 것으로 시스템 기반 IDS와 네트워크 패킷 기반 IDS로 나눈다.

- 시스템 자료기반(System Data Based) : 시스템에서 발생하는 시스템 호출에 관한 이벤트 또는 로그파일등 시스템에서 생성되는 자료를 이용하여 침입을 탐지한다.
- 네트워크 패킷 기반(Network Packet Based) : 보안영역의 호스트에 전송되어지는 네트워크 패킷에 대한 정보를 분석하여 침입을 탐지한다.

(2) 자료 공급자에 따른 분류

자료 공급자에 의한 분류는 탐지 시스템의 탐지 영역을 중심으로, 단일 호스트 기반, 다중 호스트 기반, 네트워크 기반, 혼합형 기반 IDS로 분류된다.

- 단일 호스트 기반의 IDS(Single Host Based)는 감사 자료를 단일 호스트에서 수집하고 시스템 취약점을 검사한다. 단일 호스트로부터 생성되고 수집되어진 감사(audit) 데이터를 침입 탐지에 사용하게 된다.
- 다중 호스트 기반의 IDS는 감사 자료를 다중 호스트로부터 수집하여 단일 또는 다중의 시스템에서 침입을 탐지한다. 이것은 여러 호스트를 통하여 이루어지는 것을 감시하기 위한 방법이다. 네트워크 기반 탐지법으로 탐지가 어려운 경우 여러 호스트 정보를 수집하여 비정상행위를 수집한다. IP spoofing이나 packet sniffing, 그리고 협동 공격(coordinated attack) 등의 경우 네트워크나 호스트 침입탐지 시스템에서 탐지하기가 어렵지만, 각 호스트의 에이전트에서 발생한 정보와 네트워크에서 얻어지는 정보를 결합하면 정확한 판단을 내릴수 있다.
- 네트워크 기반 IDS는 시스템간의 네트워크 망을 통하여 교환되어지는 네트워크 패킷을 분석하여 수집되어진 감사(audit) 데이터를 침입 탐지에 사용하게 된다. 호스트 기반 IDS는 모니터링하려는 시스템마다 하나씩 설치가 되어야 하지만, 네트워크 기반 IDS는 네트워크 단위에 하나만 설치하면 된다.
- 혼합형 IDS는 호스트와 네트워크 기반 IDS를 결합한 형태로써 근래에는 두 개의 다른 타입의 IDS를 혼합시킨 혼합형 IDS가 가장 많이 개발되고 있다. IP spoofing이나 packet sniffing, 그리고 협동 공격(coordinated attack) 등의 경우 네트워크나 호스트 침입탐지 시스템에서 탐지하기가 어렵지만, 각 호스트의 에이전트에서 발생한 정보와 네트워크에서 얻어지는

정보를 결합하면 정확한 판단을 내릴 수 있다.

2.1.3 침입탐지 시스템의 한계

침입탐지시스템은 탐지 위주의 매커니즘 설계로 인해 몇 가지 한계점을 가지고 있다. 첫 번째 오탐지와 미탐지의 문제이다. 침입행위가 늘어나면서 네트워크 IDS나 호스트 IDS의 제한된 탐지 능력으로 공격시도들에 대해 적절하게 침입을 구분해 내기 어려워졌다[2]. 두 번째로 네트워크 침입탐지 시스템은 실시간으로 공격을 막을 수 없다는 것이다. 이는 네트워크 상에 있는 패킷들을 감지하지만 차단하지 못하기 때문이며 대부분이 패킷은 네트워크 침입탐지 시스템이 판별하기 전에 침입에 성공하게 된다[3]. 현재의 침입탐지 시스템은 이 같은 결점들을 보강하기 위하여 여러 가지 방법을 제공하고 있으나 오탐지 문제와 다량의 로그, 그리고 실시간 방어문제는 쉽게 해결할 수 없는 실정이다.

2.2 침입차단 시스템

침입차단 시스템(방화벽 혹은 firewall)은 외부로 부터의 불법적인 접근이나 해커의 공격으로부터 내부 네트워크를 방어하기 위해 내부 인트라넷과 외부 인트라넷 사이에 유일한 통로에 설치하여 두 네트워크간에 이루어지는 접근을 제어하는 장치이다[4]. 침입차단 시스템을 양방향 트래픽의 병목점에 설치함으로써 내부 네트워크의 취약한 부분이 외부에 노출되어 위험을 감소시킬 수 있다. 외부 인터넷과 조직 내부의 전용통신망 경계에 건물의 방화벽과 같은 기능을 가진 시스템, 즉 라우터나 응용 게이트웨이 등을 설치하여 모든 정보의 흐름이 이들을 통해서만 이루어진다.

2.2.1 방화벽의 한계

방화벽의 기본원리는 관리자가 정해놓은 룰에 따라서 외부 네트워크에서 오는 패킷을 차단하는 것이다. 이렇게 수동적이고 정해진 패킷에 대해서만 차단하는 측면에서 방화벽은 다음과 같은 문제점을 가지게 된다[5, 6].

필수적인 서비스를 위해 오픈해야 하는 서비스
- 침입을 능동적으로 차단 불가능
- 알려진 공격에 대한 제한적인 차단
- 내부 이용자에 의한 공격 탐지 불가능
- 침입에 대한 보고기능이 없음

2.3 넷필터(Netfilter)

넷필터는 리눅스 커널내부에 위치하여, 패킷을 필터링 하는 기능을 담당한다.

2.3.1 패킷필터링 과정

넷필터는 지나가는 패킷의 헤더를 살펴보고 그 패킷의 운명을 결정하며, 이것은 패킷을 DROP(패킷거부)하던가, ACCEPT(패킷허용)하던가 또는 다른 무엇(target)을 할 것인가를 결정할 것이다[7]. 모듈은 넷필터에게 다음의 동작을 하도록 요청한다.

- ACCEPT : 패킷을 계속 진행
- DROP : 패킷을 거부
- STOLEN : 패킷을 접수. 즉 계속 진행시키지 않음
- QUEUE : 패킷을 큐로 보낸다

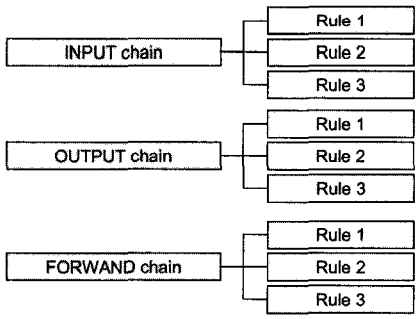
커널은 '필터' 테이블에 세 가지 기본체인 INPUT · OUTPUT · FORWARD 체인을 갖고 시작한다.

패킷이 한 체인에 이르면 그 패킷의 운명이 결정되는데 체인이 그 패킷을 'DROP'이라고 명령하면 패킷은 그곳에서 삭제되고 그 체인이 'ACCEPT'라고 명령하면 다음 부분으로 패킷은 계속 전달된다.

체인에는 한 개 이상의 규칙(rule)을 지정할 수 있다. 각 규칙은 패킷이 일치되어야할 상태를 설정하고 일치되었을 때 무엇을 할 것인가(target)를 나타낸다. 패킷은 그 규칙과 일치하지 않으면 다음규칙을 참고한다. 마지막으로 더 이상 고려할 규칙이 없으면 그 체인의 기본 정책을 확인하여 패킷의 운명이 결정된다. 기본적으로 만들어진 체인은 제거할 수 없다.

컴퓨터로 들어오는 패킷은 INPUT chain을 통과하고 컴퓨터에서 나가는 모든 패킷은 OUTPUT chain 을 통과한다. 그리고 하나의 네트워크에서 다른 곳으로 보내는 패킷은 FORWAND chain을 통과한다.

다음 (그림 2)는 세 가지의 chain의 형태이다.



(그림 2) 기본 세 가지 chain

각각의 chain에 rule을 추가하면 아래에서 부터 하나씩 추가 된다.

2.3.2 iptables

'iptables'는 커널의 패킷 필터링 테이블에 필터링 규칙을 삽입하거나 삭제하는 사용자 도구이다.

iptables 자체가 패킷을 필터링 하는 것이 아니고 패킷필터링은 커널에 탑재된 'netfilter'의 기능이며 iptables는 단지 netfilter의 규칙을 수립해 주는 사용자 도구일 뿐이다[8].

iptables에서 사용하는 명령어는 다음과 같다.

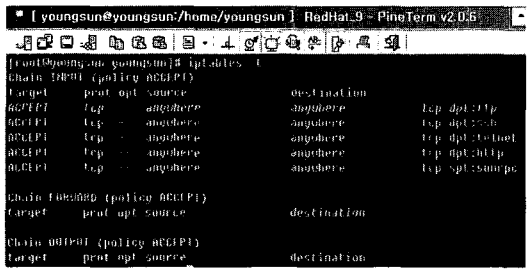
〈표 1〉 iptlables 명령어

명령어	내 용
N	새로운 체인 생성
X	비어있는 체인 제거
P	미리 만들어진 체인의 기본 정책 바꾸기
L	rule 목록을 보고 싶을때 사용
F	chain의 모든 rule을 삭제 할때 사용
A	INPUT, OUTPUT, FORWAND 중 하나를 선택해야하는데 chain의 rule을 적용시킬지를 결정
I	새로 추가하는 rule을 먼저 작동 시킬때 사용
R	이미 위치된 rule을 다른위치로 바꾸고 싶을때 사용
D	rule을 삭제하고 싶을때 사용
s	source ipsj DNS name을 지칭할 때 사용
d	목적지 ip 주소 지정할 때 사용
p	TCP, UDP .ICMP의 프로토콜을 결정할때 사용
i	입력 인터페이스 지정
o	출력 인터페이스 지정
j	패킷을 어떻게 처리할지를 결정한다. 일반적으로 ACCEPT(패킷을 허용하는 옵션), DENY(컴퓨터가 연결을 허용하지 않는다고 메시지를 돌려보내는 옵션), DROP(패킷을 완전히 무시하는 옵션)이다.

위의 iptables의 명령어를 가지고 패킷을 차단하거나 허용할 수 있는데 웹서버용 IPTABE 설정할때는(포트 범위 : 1~65535) 열어둘 포트는 21 : ftp, 22 : ssh, 23 : telnet, 80 : http, 111: portmap(NFS)을 다음과 같이 설정한다.

```
iptables -A INPUT -p tcp --dport 21 -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p tcp --dport 23 -j ACCEPT
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --sport 111 -j ACCEPT
```

결과는 (그림 3) 다음과 같다.



(그림 3) 웹서버 iptables설정

특정 ip 203.247.42.156 에 대해서 모두 차단하고 싶을 때는 다음과 같이 설정한다.

iptables -A INPUT -s 203.247.156 -j DROP

결과는 (그림 4) 다음과 같다.

```
[root@youngsun youngsun]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  203.247.42.156       anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

(그림 4) 특정ip만 차단하였을 경우

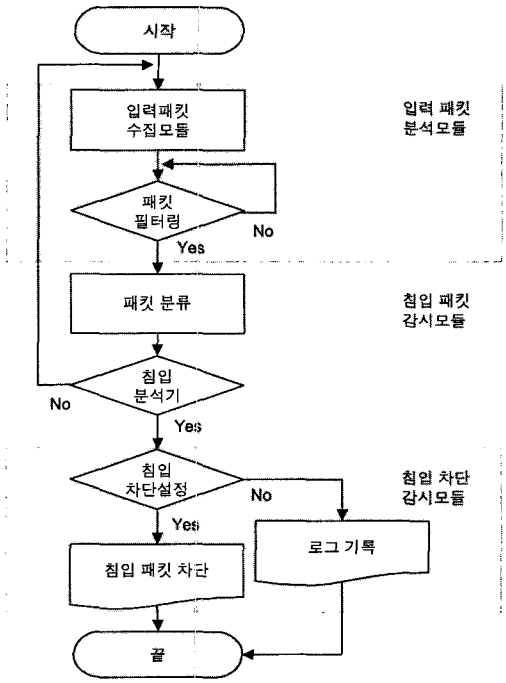
3. 침입탐지 및 차단 시스템 설계

본 논문에서 제시하는 침입탐지 및 차단시스템은 외부 네트워크와 내부 네트워크 사이에 위치하게 되며, 외부 네트워크로부터 들어온 패킷은 입력패킷 분석모듈을 거쳐서 프로토콜별로 분석이 된다. 침입패킷 분석모듈은 룰 DB와 분석된 패킷을 비교하여 침입을 분석하게 된다. 침입차단정보 관리모듈은 침입차단정보를 넷필터 프레임워크에 전달하여 해당 패킷을 차단하고 차단정보에 대하여 로그를 기록하게 된다.

3.1 침입탐지 및 차단 알고리즘 설계

침입탐지 및 차단 알고리즘은 크게 3개의 모듈로 구성이 되어있다. 입력패킷 분석모듈은 입력된 패킷을 수집하고 수집된 패킷을 필터링하며, 침입패킷 감시모듈은 입력된 패킷을 분류하고 공격패킷에 대한 분석을 한다. 침입차단 관리모듈은 패킷의 차단을 설정하고 로그를 남김으로써 침입패킷에 대한 관리를 수행한다.

(그림 5)는 침입탐지 및 차단 시스템의 알고리즘 설계를 나타낸다.



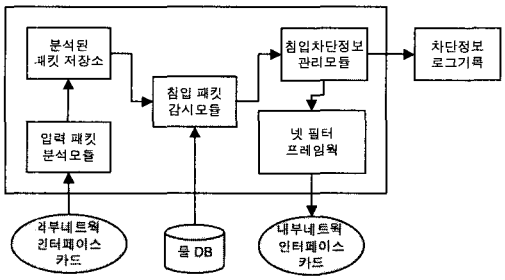
(그림 5) 침입탐지 및 차단 알고리즘 설계

3.2 침입탐지 및 차단 시스템 구성

본 논문에서 제시하는 침입탐지 및 차단시스템은 (그림 6)과 같이 구성되어 있다.

시스템은 외부 네트워크와 내부 네트워크 사이에 위치하게 되며, 외부 네트워크로부터 들어

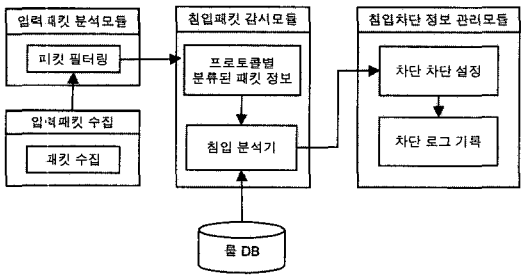
온 패킷은 입력패킷 분석모듈을 거쳐서 프로토콜별로 분석이 된다. 침입패킷 분석모듈은 룰 DB와 분석된 패킷을 비교하여 침입을 분석하게 된다. 침입차단정보 관리모듈은 침입차단정보를 넷필터 프레임워크에 전달하여 해당 패킷을 차단하고 차단정보에 대하여 로그를 기록하게 된다.



(그림 6) 침입탐지 및 차단 시스템 모델

3.3 침입차단 시스템 모듈 구성도

(그림 7)는 침입탐지 및 차단 시스템의 세부 모듈들의 구성도를 나타낸다.



(그림 7) 시스템 모듈 구성도

3.3.1 입력패킷 분석모듈

입력패킷 수집모듈로부터 패킷을 받아서 TCP/IP 프로토콜스택에 따라서 네트워크 인터페이스 계층, 인터넷 계층, 전송 계층, 응용계층으로 분리하여 각 계층의 헤더정보를 분석하게 된다.

- Analyze_Ethernet_Packet() : 패킷을 이더넷 구조체로 형변환 후 하드웨어주소와 상위 프로토콜을 저장
- Analyze_IP_Packet() : 패킷을 IP 구조체로 형변환 후 IP헤더 내용을 저장하고 상위 프로토콜에 따라 TCP, UDP, ICMP 패킷을 분석하게 된다.
- Analyze_TCP_Packet() : TCP패킷 헤더 분석
- Analyze_UDP_Packet() : UDP패킷 헤더 분석
- Analyze_ICMP_Packet() : ICMP패킷 헤더 분석

3.3.2 침입패킷 감시모듈

입력패킷 분석모듈로부터 TCP/IP 프로토콜스택에 따라 각 계층별로 분석된 헤더 정보와 룰 DB의 침입 유형 정보를 비교 분석하여 해당 패킷의 침입 여부를 판단하게 된다. 침입 분석기는 해당 패킷이 침입이라고 판정되면 침입차단 정보 관리모듈로 침입차단 설정정보를 보내게 된다.

- Parse_Rule_DB() : 룰 DB로부터 1라인씩 룰을 읽어 들여서 구조체에 침입 유형 정보를 저장
- Check_Rule() : 침입 유형 정보와 계층별로 분석된 헤더 정보를 비교하여 침입을 탐지하고 침입차단 설정정보 메시지를 작성한다.

3.3.3 침입차단 정보 관리모듈

침입차단 정보 관리모듈에서는 침입패킷 감시모듈에서 받은 침입차단 설정정보를 통하여 침입차단을 설정하게 되고 침입에 해당하는 패킷에 대하여 로그를 기록한다. 침입차단 설정정보에는 탐지된 침입을 즉시 차단해야 하는지, 경고메시지 형식으로 시스템에 로그를 기록할 것인지와 같이 침입에 대한 대응정도를 결정하기 위한 레벨정보가 포함된다.

침입차단 정보 관리모듈에서는 패킷의 흐름을 차단하는 역할을 하는 넷필터 프레임워크과 상

호 연동하기 위하여 libiptc 라이브러리를 이용한다. libiptc는 넷필터 프레임워크의 패킷 필터링 기능을 설정하는 라이브러리이다.

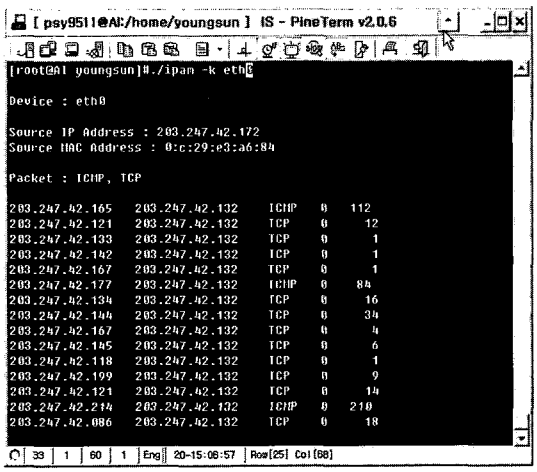
사용자의 요청 또는 시스템의 초기화가 발생될 때 규칙기반 접근제어 모듈과 침입방지 모듈에게 침입 설정 정보를 전송하고 로그 모듈은 침입탐지 정보를 받아서 로그정보로 유지하도록 한다.

- Set_Intrusion_Detection() : libiptc를 이용하여 침입차단 설정정보에 따라서 침입을 차단하게 된다. 침입차단 설정정보에는 해당 패킷의 소스주소와 포트번호 등이 포함되게 된다.
- log() : 프로토콜별로 분석된 패킷의 로그를 남기고 침입차단 설정정보를 로그로 기록한다.

4. 침입탐지 및 차단 시스템 구현

4.1 입력패킷 분석 모듈 실행 화면

입력패킷 수집모듈로부터 받은 패킷들은 입력패킷 분석모듈에서 프로토콜별로 분석되어진다.



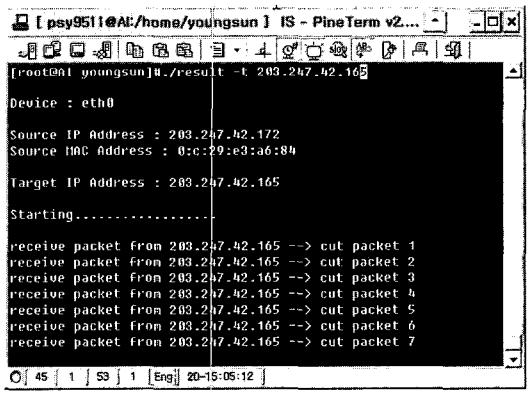
(그림 8) 입력 패킷분석모듈 실행 결과

(그림 8)은 입력 패킷에 대한 분석 결과 화면으로 해당 패킷에 대한 프로토콜의 종류와 패킷의 크기 등을 출력한 결과이다.

입력패킷 분석모듈로부터 프로토콜 스택에 따라 각 계층별로 분석된 헤더정보와 룰DB의 침입 유형정보를 비교 분석하여 해당 패킷의 침입 여부를 판단하게 된다. 침입분석기는 해당패킷이 침입이라고 판정되면 침입차단 정보관리모듈로부터 침입차단설정 정보를 보내게 되어 송신된 ip주소를 cut 하게 된다.

4.2 침입 차단 관리 모듈 실행 화면

(그림 9)는 침입 분석기를 통해 침입 패킷으로 설정된 IP 203.247.42.165의 패킷들에 대하여 침입 차단관리 모듈이 패킷을 차단하는 화면이다.



(그림 9) 침입 차단 로그 결과

5. 결 론

사회가 발전하고 정보화로의 발전이 급속도로 진행되는 시점에서 내부 정보망의 취약점은 증가할 수밖에 없는 상황이며, 이런 시점에서 내부 정보망의 정보유출을 탐지하고 차단할 수 있는 기법이 필요하다. 여러 개의 네트워크가 모여 인터

넷을 이루므로 작게는 개인에서부터 크게는 회사, 국가에 이루기까지 시스템 정보가 유출될 수 있으므로 심각한 문제가 야기될 수 있다. 시스템 보안과 정보의 유출에 대한 지식이 부족한 현실에서 손쉽게 다가갈 수 있는 도구가 절실할 때이다.

본 논문에서는 네트워크보안강화 및 침입탐지와 침입차단의 사전예방으로 네트워크를 사용하는 가개인의 보안 및 업무의 안정성향상을 높이고자 넷필터를 이용하여 침입이 탐지되면 해당하는 패킷을 실시간으로 넷필터로 차단할 수 있는 시스템을 설계하였다.

넷필터는 침입기술에 대해 다양한 방법의 보안기술을 이용해 침입이 일어나기 전에 실시간으로 침입을 막고 알려지지 않은 방식의 침입으로부터 네트워크와 호스트를 보호할 수 있는 시스템으로 보안기능을 강화하고 패킷처리 능력을 향상 시키는데 보다 나은 서비스를 제공할 수 있는 기반이 될 것이다.

참 고 문 헌

[1] R. G. Bace, 'Intrusion Detection', Macmillan Technical Pub., 2000.

[2] Steve Schupp, "Limitation of Network Intrusion Detection", http://www.sans.org/rr/whitepapers/detection/.

[3] M. Esmaili, R. SafaviNaini, and J. Pieprzyk, "Intrusion Detection: a Survey", Proceedings of ICCC, pp. 409-414, 1995.

[4] 조대일, 송규철, 노병구, "네트워크 침입탐지와 해킹 분석 핸드북", 인포북, 2001.

[5] B. H. Jeong, J. N. Kim, and S. W Sohn, "Current Status and Expectation of Techniques for Intrusion Protection System", http://kidbs.itnd.or.kr/WZIN/jugidong/1098/109801.htm.

[6] D. C. Sim, "A trend of Intrusion Detection System", KISDI IT FOCUS 4, Korea Information Strategy Development Institute, pp. 61-65, 2001.

[7] A. Jones, "Netlter and IPTables : A Structural Examination", http://www.sans.org/rr/whitepapers/rewalls/1392.php.

[8] R. Russell and H. Welte, "Linux netlter Hacking HOWTO", http://www.netlter.org//documentation/HOWTO/netlter-hacking-HOWTO.html.

백 승 엽

2004년 한남대학교 수학과(이학사)
2005년~현재 한남대학교 일반대학원 컴퓨터공학과 재학

이 건 호

1985년 홍익대학교 도시계획과 (공학사)
1996년 국방대학교 전산학과 (공학석사)
2004년~현재 한남대학교 일반대학원 컴퓨터공학과 재학
2004년~현재 육군본부 지휘통제부 정보보호 담당

이 극

1983년 경북대학교 전자과 (공학사)
1986년 서울대학교 전산학과 (공학석사)
1993년 서울대학교 전산학과 (공학박사)
1988년~현재 한남대학교 컴퓨터공학과 교수
2003년~현재 한남대학교 민군겸용보안공학 연구센터 소장