한국시뮬레이션학회 논문지
Vol. 15, No. 1, pp. 87-95 (2006. 3)

SSFNet 환경에서 보안시스템 시뮬레이션을 위한 IDS 모델링 및 구현

김용탁[1†] · 김태석[1] · 권오준[1] · 서동일[2]

Modeling and Implementation of IDS for Security System simulation using SSFNet

Yong-Tak Kim · Tai-Suk Kim · Oh-Jun Kwon · Dong-il Seo

ABSTRACT

We need to check into when a security system is newly developed, we against cyber attack which is expected in real network. However it is impossible to check it under the environment of a large-scale distributive network. So it is need to simulate it under the virtual network environment. SSFNet is a event-driven simulator which can be represent a large-scale network. Unfortunately, it doesn't have the module to simulate security functions. In this paper, we added the IDS module to SSFNet. We implement the IDS module by modeling a key functions of Snort. In addition, we developed some useful functions using Java language which can manipulate easily a packet for network simulation. Finally, we performed the simulation to verify the function if our developed IDS and Packets Manipulation. The simulation shows that our expanded SSFNet can be used to further large-scale security system simulator.

Key words : Simulation, SSF(Scalable Simulation Framework)Net, IDS(Intrusion Detection System), Cyber Attacks, DML(Domain Modeling Language), Security System

요 약

사이버 공격에 대한 새로운 보안시스템의 성능을 검증하기 위해서는 실 네트워크상에서의 검증이 필요하다. 하지만 현실적으로 대규모 분산 네트워크 환경에서의 검증은 어렵다. 이에 성능 검증을 위해 대규모 가상 네트워크 시뮬레이션 시스템이 필요하다. 본 논문에서는 대규모 네트워크를 표현할 수 있고, 프로세스 기반 사건 중심 시뮬레이션 시스템인 SSFNet을 사용하였다. 하지만 대규모 분산 네트워크상에서 보안시스템의 시뮬레이션을 SSFNet에서 수행할 수가 없다. 확장된 SSFNet내에는 IDS(Intrusion Detection System)의 보안모듈이 구성요소로 추가되었다. 추가적으로 IDS는 Snort의 기능을 모델링하여 구현하였다. 네트워크 시뮬레이션의 패킷을 쉽게 조작할 수 있도록 자바언어를 사용하여 패킷조작기를 개발하였다. 최종적으로 개발된 IDS와 패킷 조작기를 기능 검증을 위해 시뮬레이션을 수행하였다. 확장된 SSFNet은 향후 대규모 네트워크의 보안 시스템 시뮬레이터로 사용할 수 있다.

주요어 : 시뮬레이션, SSF(Scalable Simulation Framework)Net, 침입탐지시스템, 사이버공격, DML, 보안시스템

1. 서 론

현대인의 삶 속에서 인터넷이 차지하는 비중은 날이 갈수록 증대되고 있다. 통신 기술의 급속한 진전은 고도의 정보 통신망 구축을 가능하게 하였으며, 정보 통신 서비스의 다양화를 가져왔다. 네트워크를 통한 서비스를 이용하는 사용자들은 편리한 서비스 제공과 더불어 정보에 대한 보호의 필요성을 인식하고 있다.

2003년 1.25 인터넷 대란 이후 기업 및 공공기관은 소규모 네트워크 보안에서 탈피하여 대규모 네트워크를 포함할 수 있는 광범위한 네트워크 보안을 요구하고 있다. 개별시스템 중심의 소규모 보안 형태는 네트워크의 접속점에 위치하여 자기 도메인만을 담당하기 때문에 기능 집중 및 중복에 의한 전체 네트워크의 성능을 저하시키고,

2006년 3월 13일 접수, 2006년 3월 16일 채택
[1] 동의대학교 컴퓨터소프트웨어공학과
[2] 한국전자통신연구소
주 저 자 : 김용탁
교신저자 : 김태석
E-mail; tskim@deu.ac.kr

사이버 테러에 취약점을 가지고 있다. 따라서 네트워크를 대상으로 한 사이버 테러 급증에 따라 네트워크 노드에서 실시간으로 대응할 수 있는 보안 기술과 정보 인프라 구축이 필요하다.[1]

이러한 여러 가지 위험들로부터 시스템을 보호하기 위하여 침입차단시스템(Firewall), 침입탐지시스템(IDS : Intrusion Detection System), 가상 사설망(Virtual Private Network)등 여러 보안 제품들이 사용되고 있으며, 침입탐지 기술은 침입차단 기술과 함께 안정한 정보화 환경을 구축하는데 주목받는 기술로 부각되었다.

현 보안 시스템 중 IDS를 구현하고 설치하였을 때 방어 수준을 연구하기 위해 실제 상황을 시뮬레이션할 수 있는 기반이 필요하다. 이러한 성격을 만족하는 시뮬레이션을 확보하기 위해 대규모 네트워크를 표현하고, 사이버 공격 및 탐지 기능을 확장할 수 있는 SSF(Scalable Simulation Framework)를 사용하였다.[2-4]

SSF의 구성요소인 SSFNet을 확장하여 침입탐지 시스템을 추가하였고, 사이버 공격 프로그램들을 지원하기 위해 패킷 조작기(PM:Packet Manipulator)를 추가하였다. IDS 클래스는 SSFNet의 IP클래스를 확장하여 구현하였고, IP 계층을 지나가는 패킷들의 송신 및 목적지 IP주소, 포트번호, 프로토콜(TCP/UDP, IP, ICMP)종류 및 플래그, 패킷 헤더내의 IP 플래그 등을 바탕으로 패킷의 헤더 내용을 탐지할 수 있도록 구현하였다. IDS의 규칙은 네트워크 모델을 기술할 때 사용된 DML(Domain Modeling Language)로 설계 구현하였다.

Snort[5]의 동작 메커니즘을 가진 IDS에 패킷 조작기를 사용하여 인터넷 패킷을 자유로이 조작하여 공격을 할 경우, 스니퍼를 구동시키고, 스니퍼를 통과한 패킷은 각각 프로토콜 형식에 맞추어 여과 방식으로 구분한다. 구분된 패킷에 대한 규칙 집합이 정의된 데이터베이스에서 해당 패킷과 비교 작업을 수행한다. 최종적으로 해당 패킷의 행동 여부를 판단하도록 구현하였다.

본 논문에서는 확장 구현된 IDS[6-7]의 검증을 위해 실제 상황과 비슷한 네트워크를 모델링[8]하고, IDS를 설치하여 시뮬레이션하였다. 이를 통해 앞으로 다양한 가상 공격과 다양한 지점에서 공격의 대한 대비책을 마련할 수 있다.

2. 사이버 공격을 위한 패킷 조작기구현

2.1 SSFNet의 기본 구조

SSF는 프로세스 기반 이산 사건 중심 시뮬레이션 커널

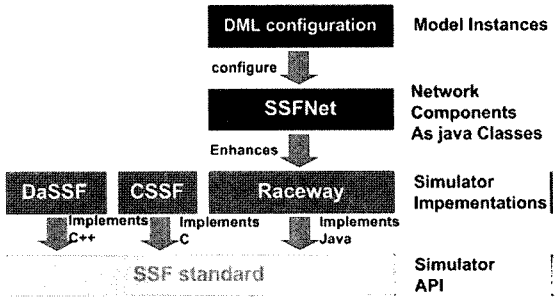


그림 1. SSFNet 계층 구조도

이며, 이를 기반으로 구현된 상위단계의 SSF 구성요소와 더불어 100,000개 이상의 노드로 구성된 대규모 네트워크까지 표현할 수 있는 기능을 제공한다. 이를 표현하기 위해 DML이라는 문서를 통해 네트워크의 구성 요소와 개별 속성을 객체지향 방식으로 정의를 한다. DML과 SSFNet을 이용하여 기존의 프로토콜로 설계된 망이나 확장 구현한 프로토콜을 이용하여 다양한 네트워크 환경을 구현할 수 있다. 그림 1은 SSFNet이 SSF를 기반으로 하여 Raceway에서 오픈 소스로 개방된 라이브러리로 제공되고 있는 것을 보여준다.

본 논문에서는 네트워크 공격에 따른 보안 시스템의 행동을 관찰하기 위해 추가적으로 클래스를 구현하였다. 보안 시스템으로 공격을 탐지 할 수 있는 IDS를 추가하였고, 사이버 공격 프로그램을 지원하기 위해 패킷 조작기를 추가하였다.

2.2 패킷 조작기의 구현

패킷 조작이란 패킷의 생성, 원하는 패킷 필드의 설정, 수정, 정보추출 등을 의미한다. SSFNet은 기존 소켓 API를 이용하여 TCP, UDP 패킷을 전송하고, 수신하는 기본 API만 제공할 뿐, NS(Network Simulation)와 같은 패킷의 헤드 및 내용을 직접 조작할 수 있는 API와 기능을 제공하고 있지 않다. 네트워크 해킹 방법 중 널리 사용하는 패킷을 조작하고, 이를 통한 공격을 시뮬레이션 할 수 있도록 SSFNet 내에 별도의 패킷 조작기 클래스를 추가 개발하였다.

패킷 조작기(PM)의 기본적인 기능으로 필요한 패킷을 조작하고, 공격 호스트로부터 도착한 패킷을 가로채기 할 수 있도록 설계하였다. 그리고 기본적인 소켓 API를 사용하여 TCP 또는 UDP 기반으로 다른 호스들과 통신할 수 있도록 구현하였다.

패킷 조작기(PM)는 공격 시뮬레이션을 위해 IP 패

킷을 직접 조작하고, 이를 공격 프로그램에서 어떠한 형태로든 사용할 수 있도록 IP의 각 필드(Head length, Type of Service, Total length, Identification, Flags, Fragment offset, TTL, Protocol, Checksum, Source IP, Destination IP)의 값을 설정 및 수정할 수 있는 기능을 제공한다. 또한 전송계층의 프로토콜인 TCP의 각 필드(Source port, Destination port, Response Number, Head Length, Flags, Window size, Checksum)의 값을 설정하고 수정할 수 있다. 이외도 UDP, ICMP 패킷 조작도 지원하고 있다.

또 다른 기능인 패킷 가로채기는 공격 호스트로부터 조작한 패킷을 보냈을 때 대상 호스트로부터 응답이 올 수 있으며, 응답한 패킷을 받아 다음 행위를 결정할 수 있어야 한다. 이와 같은 기능을 SSFNet 시뮬레이션에 추가하였다.

그림 2는 PM에서 패킷을 생성하는 클래스들 간의 관계를 보여주고 있다. PM에서는 새로운 패킷을 만들기 위해서 각 프로토콜 생성 클래스에서 패킷을 생성 한 후 IP 헤더를 생성한다. IP에서는 패킷을 보내고 받는 부분만을 책임지고 있다.

SSFNet에서의 IP 클래스는 실제의 IP와 같이 IP 계층에서 새로운 IP패킷을 만들어서 보내는 독립적인 기능을 가지고 있지 않다. 그래서 생성된 패킷은 PM 클래스의 Start- Send() 메소드에서 각각의 프로토콜 메시지에 맞게 원거리 호스트로 전송하게 된다.

그림 3은 SSFNet 시뮬레이션을 실행하면 DML로부터 설정을 불러들이고 이를 토대로 메시지 타입을 결정한다. init() 메소드에서 callback() 메소드를 사용하여 Start-Send() 메소드를 호출한다. StartSend() 메소드 내에서 메시지 타입에 맞는 메시지 전송 메소드를 호출한다. 이와 같은 방법을 통해 메시지 타입을 지원할 수 있도록 패킷 조작기를 구현하였다.

3. IDS 구조 및 모델링

침입탐지시스템은 컴퓨터 시스템 자원의 비밀성과 무결성, 가용성을 저해하는 비정상적인 사용과 오용, 남용 등을 탐지하여 자동으로 대응을 취하거나 관리자에게 경고 메시지를 보내주고 침입에 대응하는 정보보호시스템의 한 종류이다. 단순한 접근 제어 기능을 넘어서 침입 패턴 데이터베이스와 전문가 시스템 등을 사용하여 네트워크나 시스템의 사용을 실시간 모니터링하고 침입을 탐지하는 보안 시스템이다. 즉 침입탐지시스템은 이러한 불법적인 침입행위를 신속하게 감지하고 대응하는 시스템을 말한다.

침입탐지시스템은 침입차단시스템처럼 단순히 네트워크를 통한 외부 침입을 차단하는 단계를 넘어 외부 침입에 의해 침입차단시스템이 해킹되는 순간 혹은 해킹된 후 침입 사실을 탐지해 이에 대응하도록 하는 등 보안 담당자의 보안운용 능력을 증대시켜준다.

침입 차단 시스템은 기술적인 측면에서 접근제어 방법과 암호화 기법이 한계에 도달했고, 내부 사용자의 침입

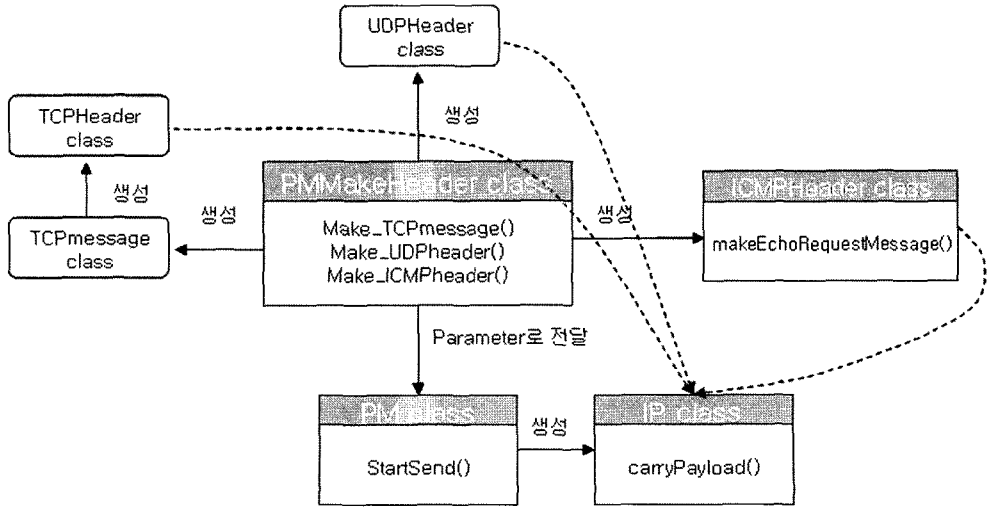


그림 2. PM의 클래스 계층도 및 흐름도

```
[DML     ]
ProtocolSession [name pm use SSF.App.PM.PM]
        PM[
                src_nhi 1:5
                dest_nhi 0:4
                msg_type ICMP       <- 메시지 종류 설정
            ]
-PM class init()
(new SSF.OS.Timer(localHost, start_time){
public void callback() {
                        StartSend(null) <- StartSend 메소드 호출
                        }
}).set(start_time);

-PM class startsend()
protected void StartSend(PMmakeheader msg){
        try {
                switch(msg_type){
                        case TCP: send_TCP(msg);
                        case UDP: send_UDP(msg);
                        case ICMP: send_ICMP(msg);
                  }
              }
}
```

그림 3. 메시지 타입 결정 예제

행위를 방어할 수 없는 문제점을 가지고 있다. 따라서 보다 강력한 보안을 위해서는 사전에 침입 탐지, 침입자 식별 및 침입방지가 필요하며, 그 방법의 하나로 침입탐지 시스템의 비중이 증가되고 있다.

3.1 Snort 기본 구조

본 논문에서는 보안시스템 중 하나인 IDS를 SSFNet시뮬레이션에 모델링하기 위해 Snort엔진을 기반으로 삼았다. Snort의 기본 구조는 그림 4와 같다.

패킷 스니퍼는 네트워크를 도청하는데 쓰이는 장치다. 네트워크 스니퍼를 사용하면 어플리케이션 또는 하드웨어 장치에서 네트워크의 트래픽을 볼 수 있다. 인터넷의 대부분은 IP 트래픽이라 할 수 있다.

전처리기는 원본 패킷을 받아들여서 특정한 플러그-인으로 그 패킷을 보낸다. 이들 플러그-인은 패킷에서 특정한 종류의 행위를 찾는다. 패킷의 특정한 종류의 '행위'를 찾으면 그 패킷은 탐지 엔진으로 전송된다.

탐지 엔진은 전처리기와 플러그-인으로부터 오는 데이터를 받아서 여러 규칙과 비교한다. 만약 패킷과 일치하는 규칙이 있다면 그 패킷은 경고 처리기로 전달한다.

3.2 IDS 규칙 형식

SSFNet에서 IDS 구현은 네트워크 백본을 모니터링하여 공격 서명을 찾는 네트워크 기반으로 작성하였다. 네트워크 트래픽에서의 이상 탐지를 감지한 IDS는 규정된 규칙 집합에 부합된 패킷이 들어올 경우 규칙에 맞는 정의대로 패킷을 처리한다. 패킷에 대한 처리 방법에 대해서는 각 패킷에 대한 행동을 결정하는 데이터가 필요하다.

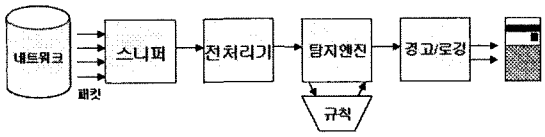


그림 4. Snort 기본 구조

본 논문에서는 규칙데이터베이스를 MySQL에 정의하였다. 규칙자체는 두 부분으로 돼 있다.

규칙헤더는 기본적으로 취할 행동(로그 또는 경고), 네트워크 패킷의 종류(TCP, UDP, ICMP 등), 출발지와 목적지 IP주소, 포트 등으로 이루어진다.

규칙 옵션은 패킷이 규칙과 일치하기 위해 포함해야 하는 컨텐츠이다. IDS는 기본적으로 규칙 문법을 가지고 있으며, 규칙은 종류별로 묶여 정기적으로 갱신이 된다.

3.3 IDS 동작 방식

SSFNet으로 설계한 IDS의 기본 메커니즘은 그림 5와 같다. Snort와 같은 방법으로 패킷에 대한 스니퍼를 구동시킨다. 스니퍼에 통과된 패킷은 각각 프로토콜 형식에 맞추어 여과 방식으로 구분한다. 구분된 패킷에 대해 규칙 집합이 정의된 데이터베이스에서 해당 패킷과 비교작업을 수행한다. 마지막으로 해당 패킷의 행동 여부를 판단하는 것으로 동작이 된다.

3.4 IDS 클래스 구조

SSFNet에서의 IDS 구현을 위해서는 기존의 SSFNet

표 1. 규칙 헤더구조

RuleHeader	
Name	Description
no	규칙 번호
proto_no	프로토콜 번호
name	프로토콜 명시
action	이 패킷에 대한 행동 결정
srcAddr	송신자 IP주소
RuleHeader	
srcPort	송신자 포트번호
destAddr	목적지 IP 주소
destPort	목적지 포트번호
flag	규칙 옵션 수행여부 결정

표 2. 규칙 옵션구조

Rule Option		
Type	name	Description
TCP	tcp_falgs	SYN, FIN, ACK
	tcp_seq	TCP의 시퀀스 번호
	tcp_ack	TCP ACK의 시퀀스 번호
ICMP	icmp_type	ICMP 메시지 형태
	icmp_code	ICMP 메시지 코드
	icmp_id	ICMP 메시지 ID
	icmp_seq	ICMP 메시지 시퀀스 번호

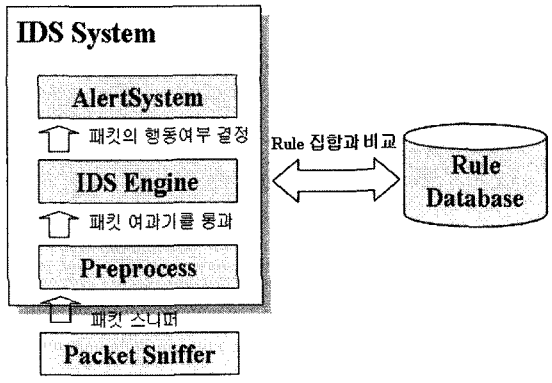


그림 5. IDS 기본 동작

에서 제공되는 IP 계층을 확장하는 방법으로 구현하였다. 그림 6은 SSFNet에 존재하는 IP 클래스를 상속받아 IDS 구성요소로 SIP 클래스를 확장 추가하였다. 그리고 독립된 구성요소로 이루어진 IDS 패키지로 구성하였다. 패킷에 대한 처리 방법을 수행하는 RuleDbc 클래스는 미리 정의된 규칙 집합을 초기화하는 작업을 수행한다. 특정 이상 패킷이 정의된 규칙 집합과 일치하는지 검사하는 SelectObject 클래스와 그것에 대한 행동 여부를 결정하는 SelectOptObject 클래스가 있다. 행동 여부는 마지막으로 이상 패킷에 대한 로그, 경고, 패킷 통과 여부를 결정하는 AlertDbc 클래스로 이루어진다. 그림 7은 IDS의 기본 클래스 구조이다.

4. 시뮬레이션 및 검증

4.1 시뮬레이션을 위한 네트워크 구조

가상 공격을 수행하고 이에 따른 IDS의 변화를 시뮬레

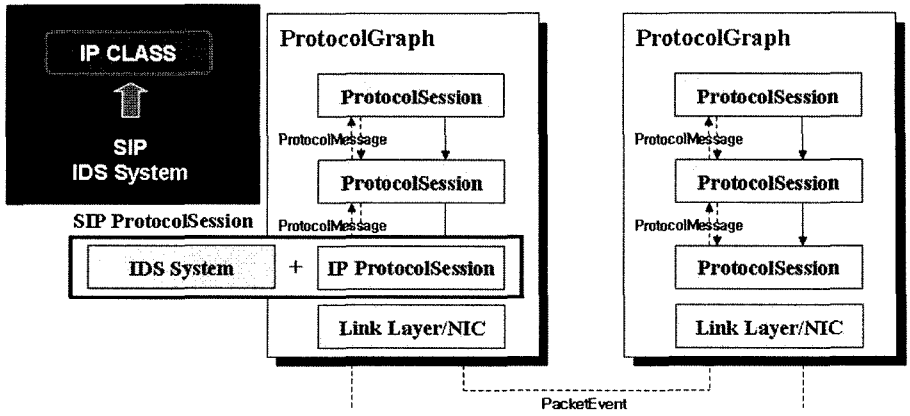


그림 6. IDS 클래스 구조

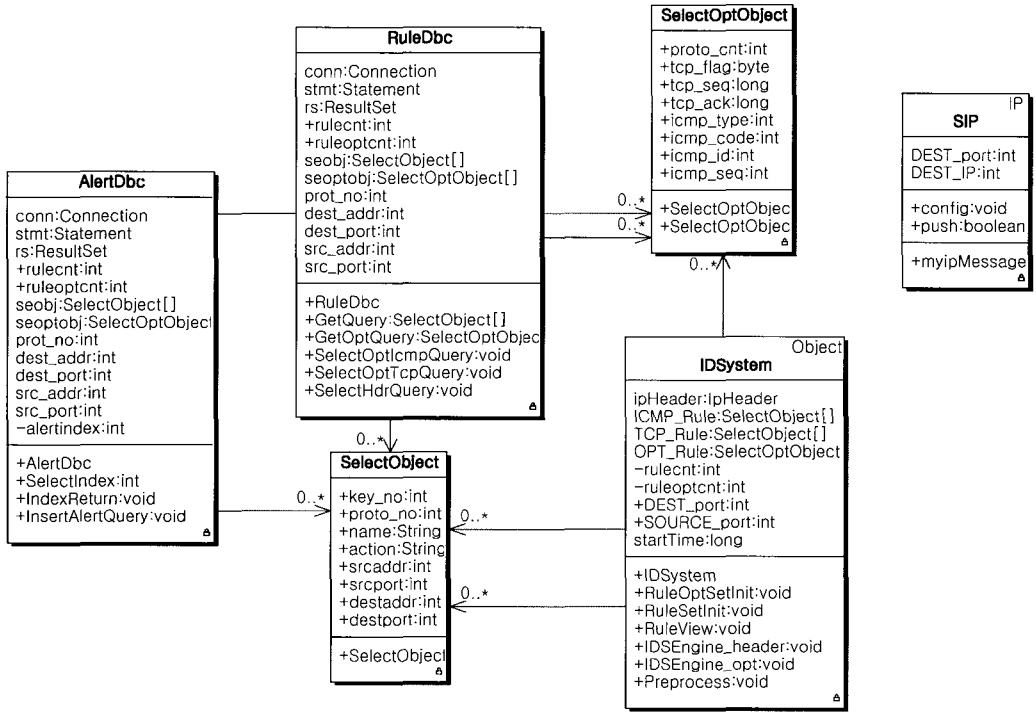


그림 7. IDS 클래스 구조

이션 하기 위해서는 실제 모습을 모델링한 네트워크 구조가 필요하다. 네트워크를 구성하는 서브시스템들의 배치와 그들의 특성을 네트워크에 반영하여 DML로 그림 8과 같이 라우터의 기능, 서버와 클라이언트간의 관계를 파악하여 SSFNet 시뮬레이션 환경을 구성하였다.

그림 8은 NET0 과 NET1의 두 개의 네트워크 그룹으로 구성되어 있다. NET0에 4대의 서버와 라우터로 연결되었다. NET1은 10대의 호스트를 각각 링 구조로, 라우터에 연결하였다. 두 개의 네트워크 그룹은 라우터 2번과 라우터 3번을 통해 패킷을 전달 할 수 있도록 하였다.

4.2 시뮬레이션 검증

IDS 시뮬레이션을 위해 DML 내의 라우터를 기술하는 부분인 ProtocolSession 내의 IP 클래스를 본 논문에서

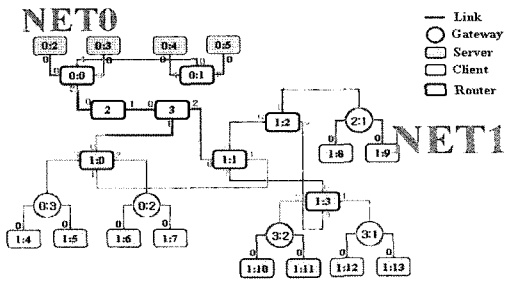


그림 8. 시뮬레이션을 위한 네트워크 구조

제안한 SIP 클래스로 지정해야한다. 이 클래스는 IP 계층으로 전달되는 모든 패킷의 헤더를 분석하여 규칙과 비교, 패킷의 이상 유무를 모니터링 해 준다. 그림 9는 IDS를 정의하는 DML의 예를 보여주고 있다.

그림 8에서 NET0에서 백본으로 이루어지는 라우터는 0:0이다. NET1에서 백본으로 이루어지는 라우터는 1:0, 1:1, 1:2이다. IDS를 NET0의 0:0에 설치 할 경우 NET1 1:0, 1:1, 1:2에 설치했을 경우 IDS가 실제로 제대로 동작하는지 시뮬레이션 하였다.

그림 10은 그림 8의 네트워크 라우터에 IDS 설치했을 경우, 각 패킷의 로그 화면을 보여주고 있다. 이와 같이 SSFNet의 IP 계층을 확장한 SIP 클래스를 지정하여 라우터에 사용할 경우 보안 시스템인 IDS로 제대로 동작하고 있음을 검증하였다.

5. 결 론

본 논문에서는 대규모 네트워크상에서 사이버 공격 시 보안 시스템을 시뮬레이션 할 수 있는 시뮬레이션 환경을 연구하였다. 대규모 네트워크상에서 새로운 보안시스템의 성능 검증은 필요하다. 하지만 현실적으로 이러한 환경 조성이 어려움으로 시뮬레이션 시스템을 사용하여 검증을 한다.

이를 위해 대 규모 네트워크를 표현 할 수 있고, 프로세스 기반 사건 중심 시뮬레이션 시스템인 SSFNet을 사용하였다. 하지만 보안 시스템을 검증할 수 있는 라이브러리를 지원하지 않는다.

본 논문에서는 SSFNet내의 IP 클래스를 상속받아 SIP 클래스를 개발하였다. SIP 클래스에는 IDS로 동작될 수 있는 기능을 구현하였다. 그리고 SSFNet은 패킷의 헤드를 직접 조작할 수 있는 API를 제공하고 있지 않다. 본 논

```
router [
        idrange [from 1 to 2]
  graph [
      ProtocolSession [
            name ip use SIP
      ]
  ]
  interface [
    id 0
    buffer 8000 extends
.dictionary.100BaseT
  ]
  interface [
    idrange [from 1 to 2]
    buffersize 16000
  ]
  route [dest default interface 0]
]
```

그림 9. IDS를 모델링하는 DML 표현

```
1          1861913663 tcp 10:1600 -> 6:10001
2          1867016863 tcp 10:1600 -> 6:10001
.....
.....
.....
32         1867016863 tcp 13:1600 -> 4:10001
33         1861913663 tcp 13:1600 -> 4:10001
34         1867016863 tcp 12:1600 -> 5:10001
35         1861913663 tcp 12:1600 -> 5:10001
36         1867016863 tcp 17:1600 -> 10:10001
37         1861913663 tcp 17:1600 -> 10:10001
38         1867016863 tcp 27:1600 -> 14:10001
39         1861913663 tcp 27:1600 -> 14:10001
40         1867016863 tcp 24:1600 -> 16:10001
41         1861913663 tcp 24:1600 -> 16:10001
42         1867016863 tcp 18:1600 -> 8:10001
43         1861913663 tcp 18:1600 -> 8:10001
44         1867016863 tcp 11:1600 -> 2:10001
45         1861913663 tcp 11:1600 -> 2:10001
.....
.....
```

그림 10. IDS 시뮬레이션 로그화면

문에서는 패킷을 조작 가능하게 하고, 이를 통한 공격을 시뮬레이션 할 수 있도록 SSFNet 내에 별도의 패킷 조작기를 개발하여 추가하였다. 패킷 조작기의 구성 요소들은 Java로 작성되었으며, 시뮬레이션을 위하여 사용되는 가상 공격 프로그램도 Java로 작성하였다. 패킷 조작기를 통해 다양한 사이버 공격을 할 수 있는 환경도 제공된다.

실제로 대규모 네트워크상에서 발생 가능한 사이버 공

격을 확장된 SSFNet 시뮬레이션 시스템을 통해 시뮬레이션함으로서 사전에 문제점을 파악할 수 있는 장점을 가지고 있다.

참 고 문 헌

1. 정보보호시스템 Technology&Market An- alysis, 정보통신연구진흥원.
2. SSFNet HomePage, http://www.ssfnet.org.
3. SSFNet 2.0 API Documents, http://www.ssfnet. org/javadoc/
4. James H. Cowie,Editor "Scalable Simulation Framework API Reference Manual", March 1999.
5. SNORT HomePage, http://www.snort .org.
6. R. Durst, T.Champion, B. Witten, E. Miller, and L. Spagnuolo. "testing and Eva- luating Computer Intrusion Detection System." CACM, 1999.
7. Jae-Hyuk Lee, Eul Gyu Im, Joo Beom Yun, Seung-Kyu Park, "Network intrusionand defense simulation framework based on SSFNet", The 6th International Conference on Volume 1, 2004.
8. 정우식, "IDS/Firewall/Router 통합 로그 분석기 설계", 한국정보과학회 학술발표논문집 2003. pp.37-43, 2003.
9. 김병구, 정태명, "침입탐지 기술의 현황과 전망", 정보과학회지 제 18권 제 1호 , 2000.
10. Information Warfare and Security, Addison-Wesley, 1999.
11. Jea-hyuk Lee, Bul-gyu Im, Joo-beom Yoon, Seung-koo Park, "Network Intrusion and Defense Simulation Framework based on SSFNet", The 6th International conference on advanced communication technology, 2004.

김 용 탁 (63164@deu.ac.kr)

1998 동의대학교 공과대학 컴퓨터공학과 학사
2003 동의대학교 공과대학 컴퓨터공학과 석사
2006 동의대학교 공과대학 컴퓨터·소프트웨어공학과 박사수료

관심분야 : 모바일 프로토콜, 무선 네트워크, 네트워크 보안, 인터넷 QoS

김 태 석 (tskim@deu.ac.kr)

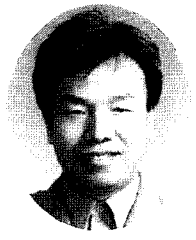
1982 경북대학교 전자공학과 공학사 졸업
1992 일본 KEIO대학 이공학부 계산기과학전공 공학박사 졸업
1992 일본 KEIO대학 이공학부 객원연구원
2000～2003 동의대학교 전산정보원장
2003～2005 동의대학교 교무처장
1993～현재 동의대학교 컴퓨터소프트웨어공학과 교수

관심분야 : 인터넷응용, 원격강의, 자연어처리

권 오 준 (ojkwon@deu.ac.kr)

1986 경북대학교 전자공학과(공학사)
1992 충남대학교 전산학과(이학석사)
1998 포항공대 전자계산학과(공학박사)
1986～2002 한국전자통신연구원 선임연구원
2000～2002 동의대학교 자연과학대학 컴퓨터통계학과
2002～현재 동의대학교 공과대학 컴퓨터소프트웨어공학과 교수

관심분야 : 컴퓨터네트워크, 정보보호, 인공신경망



서 동 일 (bluesea@etri.re.kr)

1994 포항공과대학교 정보통신학과 공학석사
2004 충북대학교 전산학과 이학박사
1989～1992 삼성전자 종합연구소
1994～현재 한국전자통신연구원, 선임연구원(팀장)
2002～현재 ASTAP Forum IS-EG 의장
2003～현재 정통부지정 IT국제표준화전문가
2004～현재 TTA TC1 부의장

관심분야 : 네트워크보안, 해킹, 인터넷정보보호, 보안장비 시험