IP Security 엔진을 위한 규칙기반 보안평가 시스템의 설계 및 구현

권 혁 찬[†] · 현 정 식[††] · 김 상 춘[†††] · 나 재 훈[††††] · 손 승 원[†††††]

요 약

IPsec은 인터넷에서 필수적인 암호와 인증 서비스를 구조적으로 제공하며, 동시에 안전한 키교환이나 재현공격 등을 방어할 수 있는 메커니즘을 제공하는 프로토콜로서, 현재 다양한 플랫폼에서 구현 중에 있다. 그러나 현재까지 IPsec엔진이 탑재된 서버들이 제대로 동작하는지 그리고 보안서비스를 적절히 제공하는지를 평가하기 위한 도구는 존재하지 않고 있다. 따라서 본 논문에서는 IPsec엔진의 보안성을 평가하기 위한 자동화된 규칙기반 보안평가 시스템을 설계 및 구현하였다. 본 시스템은 Windows와 UNIX 환경에서 수행이 가능하며 Java와 C언어로 구현되었다.

A Design and Implementation of A Rule-based Security Evaluation System for IP Security Engine

Hyeok Chan Kwon[†] · Jung Sik Hyun[††] · Sang Choon Kim[†††]
Jae Hoon Nah[††††] · Sung Won Son[†††††]

ABSTRACT

IPsec offers not only Internet security service such as Internet secure communication and authentication but also the safe key exchange and anti-replay attack mechanism. Recently IPsec is implemented on the various operating systems. But there is no existing tool that checks the servers, which provide IPsec services, work properly and provide their network security services well. In this paper, we design and implement the rule based security evaluation system for IPsec engine. This system operated on Windows and UNX platform. We developed the system using Java and C language.

키워드 : 보안성 평가(Security Evaluation), IPsec, ISAKMP, 에이전트(Agent), 패킷 스니퍼(Packet Sniffer), 규칙 해석기(Rule Interpreter)

1. 서 론

최근 인터넷의 이용이 폭발적으로 증가하면서 인터넷 정보보호 서비스에 대한 필요성이 증대되었고 관련된 많은 연구들이 진행되어 왔다. 이러한 연구의 일환으로 IETF Security Area의 IPsec WG은 1993년 6월부터 작업을 시작하여 현재 IPsec 아키텍처를 기술한 RFC2401을 비롯한 18개의 RFC를 작성하였다[1]. IPsec WG은 인터넷 정보보호에 관한 기본구조를 연구하고 있는 그룹으로서, 인증을 위한 헤더인 AH(Authentication Header)와 기밀성을 위한 헤더인 ESP(Encapsulation Security Payload)의 두 가지 확장 헤더와 키교환을 위한 IKE(Internet Key Exchange)를 정의하였다[2-5]. IPsec이 제공할 수 있는 정보보호 서비스는 접근제어, 데이터 원적지 인증, 비연결형 무결성, 데이터 기밀성, 재현공격방지, 제한된 트래픽 흐름 기밀성이다. 현재 IPsec은 리눅스, FreeBSD, Window2000 등 여러 가지 플랫폼에서 구현되고 있으며, 리눅스 기반의 FreeS/WAN, FreeBSD 기반의 KAME 등 공개된 프로젝트도 존재한다[6-8].

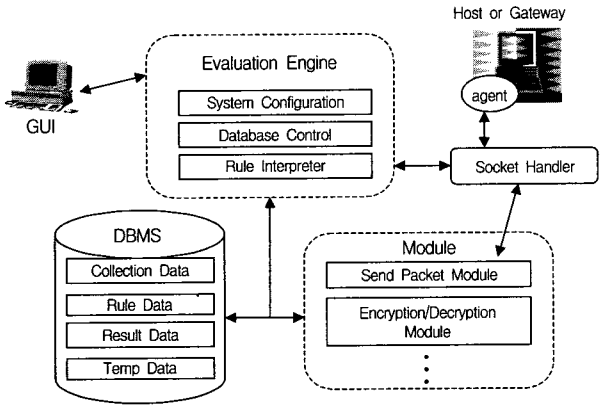
현재 많은 플랫폼에서 IPsec을 구현 중에 있으나 아직까지 IPsec 엔진에 대한 보안성과 적합성을 평가하기 위한 자동화된 도구는 나와있지 않다. 보안 호스트에 대한 보안 취약성 분석 툴로 AXENT의 NetRecon[9], Cisco의 Cisco Scanner[10] 그리고 LANguard network&port scanner[11] 등이 있으나, 이러한 툴들은 단지 호스트나 네트워크에 대한 스캐닝 기능만을 제공하며 실제 IPsec 보안 서비스를

† 준 회 원 : 한국전자통신연구원 정보보호연구본부 선임연구원
†† 준 회 원 : 한국전자통신연구원 정보보호연구본부 위촉연구원
††† 정 회 원 : 삼척대학교 정보통신공학과 교수
†††† 정 회 원 : 한국전자통신연구원 정보보호연구본부 팀장 / 책임연구원
††††† 정 회 원 : 한국전자통신연구원 정보보호연구본부 네트워크보안연구부장 / 책임연구원
논문접수 : 2002년 1월 21일, 심사완료 : 2002년 3월 25일

제공하는 특정 호스트에 대한 보안성을 평가하는 기능은 제공해 주지 못한다.

본 논문에서는 IPsec엔진의 보안성을 평가하기 위한 자동화된 규칙기반 보안평가시스템을 설계 및 구현하였다. 제안하는 보안평가시스템은 ETRI에서 개발중인 C-ISCAP(Controlled Internet Security Connectivity Assurance Platform) [12]이라고 명명한 통합 IPsec엔진에 대한 보안성을 평가하고 디버깅하기 위한 목적으로 개발되었다. 그러나 본 보안평가시스템은 독립적으로 존재하는 평가시스템이며 IPsec 프로토콜에 대한 국제 표준인 IETF의 RFC문서(2401, 2402, 2406, 2407등)를 기반으로 개발된 도구이므로 C-ISCAP 뿐 아니라 기타의 개발 중이거나 개발이 완료된 IPsec엔진에 대해서도 아무런 변경 없이 평가를 수행할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 설계한 보안평가시스템을 모듈별로 기술한다. 3장에서는 구현과 수행과정에 대한 내용을 기술하며 4장에서 결론을 맺는다.



(그림 1) 보안평가 시스템의 구조

2. 보안평가시스템의 설계

(그림 1)은 보안평가 시스템의 기본구조를 보여준다. 보안평가 시스템은 크게 다음과 같은 요소로 구성된다.

- 시스템을 총체적으로 제어하는 평가엔진(Evaluation Engine)
- 시스템에 필요한 데이터를 관리하는 DBMS
- 평가에 사용할 데이터를 수집하는 에이전트(agent)
- 평가 규칙에서 사용하는 모듈 (module)
- 평가규칙을 실행하기 위한 규칙 해석기(rule interpreter)

2.1 평가엔진(Evaluation Engine)

평가엔진은 (그림 1)에서 볼 수 있듯이, System Configuration, Database Control, Rule Interpreter로 구성된다. Rule Interpreter는 2.2절에서 기술되며 본 절에서는 System Configuration과 Database Control 모듈에 대해 설명한다.

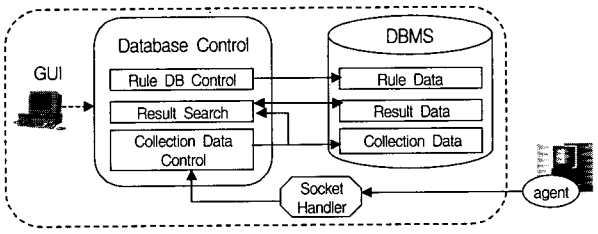
2.1.1 System Configuration

System Configuration은 세부적으로 Access Control, Agent Registration, Module Registration, Directory Setup 모듈로 나눌 수 있다.

Access Control은 보안평가 시스템 사용을 위한 사용자 인증을 수행하는 부분이다. Agent Registration에서는 에이전트 이름, 에이전트타입, 데이터베이스 이름, 현재 에이전트가 설치되어 있는 호스트 정보등에 대한 등록 작업을 수행한다. Module Registration에서는 평가규칙에서 사용하게 될 각 모듈에 대한 정보를 등록하는 곳으로 모듈 이름, 모듈 설치 경로, 모듈에 대한 설명 등을 등록한다. Evaluation Control에서는 실제 평가할 호스트들을 선택하기 위해 현재 에이전트가 설치된 호스트들과 평가 시스템간의 Network Map을 제시한다. 그리고 현재 Rule Data에 등록되어 있는 규칙 중 어떤 평가규칙을 수행할 것인지에 대한 평가범위를 결정하기 위한 인터페이스를 제공한다. Directory Setup에서는 평가규칙 수행 시 사용되는 임시 데이터와 관리자가 임의적으로 만든 패킷 데이터가 저장될 Directory Path를 정의한다.

2.1.2 Database Control

(그림 1)의 Database Control은 DB내의 테이블 생성, 데이터 추가, 수정, 삭제 등의 작업을 수행한다. (그림 2)에서 Database Control의 구조를 볼 수 있다. (그림 2)에서 Rule DB Control에서는 관리자가 평가규칙을 정의, 수정, 삭제하고 평가규칙을 수행할 때 사용하기 위한 패킷 데이터 생성을 위한 인터페이스를 제공한다. 생성한 패킷 데이터는 System Configuration의 Directory Setup에 정의된 위치에 저장된다. Result Search에서는 현재까지 수행한 평가결과를 검색하여 보여주는 기능을 제공한다. Collection Data Control은 평가 수행 중에 대상 호스트의 에이전트로부터 전송 받은 데이터를 Socket Handler에 의해 수신하고, 수신한 데이터에 대한 정보를 Collection Data DB에 저장하는 기능을 갖는다. (그림 2)에 보이는 DBMS의 각 테이블의 구조는 2.3절에서 기술된다.

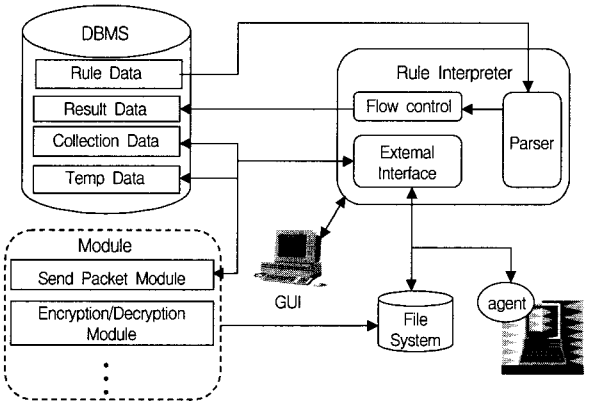


(그림 2) Database Control 구조

2.2 규칙 해석기(Rule Interpreter)

규칙 해석기는 평가규칙을 DBMS의 규칙 데이터(Rule Data)로부터 순차적으로 읽은 다음, 수행절차에 따라 명령

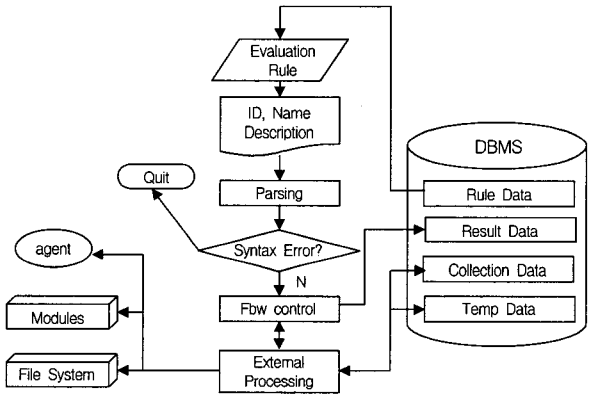
을 해석하고 실행하는 기능을 갖는다. 규칙 해석기의 구조는 (그림 3)과 같다.



(그림 3) 규칙 해석기(rule interpreter)

규칙 해석기의 파서(parser)는 규칙 데이터의 프로그램 필드에 저장되어 있는 평가 프로그램을 읽으며 문법검사를 수행한 후 문법오류가 없으면 각 단어를 분리하여 흐름제어(Flow Control)로 보낸다. 흐름제어는 파서로부터 받은 각 단어에 할당되어 있는 제어명령을 수행하고, 명령수행 과정에 있는 함수들을 외부 인터페이스(External Interface)로 제공한다.

외부 인터페이스는 수집 데이터(Collection Data)나 그 외 평가규칙에 의해 임시 저장된 임시 데이터(Temp Data)와 같은 DB를 조작하고, 평가규칙에서 사용하는 에이전트와 모듈을 제어하는 함수를 수행한다. 그리고 규칙실행에 필요한 모듈과의 인터페이스를 제공한다. 규칙해석기의 수행절차는 (그림 4)와 같다.



(그림 4) 규칙 해석기의 수행절차

<표 1>은 규칙 해석기에서 처리 가능한 제어 명령이다.

제어명령에서 사용하는 *조건식*은 베이직(BASIC)의 관계연산자와 논리연산자를 사용하여 구성된다. PRINT문은 출력문을 GUI의 로그 창과 DBMS의 결과 데이터에 출력하는데 사용하고, BRAKE문은 제어명령의 범위에서 벗어나는 명령어이다. 제어명령은 흐름제어에서 처리한다.

〈표 1〉 흐름제어명령

Command	Syntax

IF문	IF *조건식* THEN
	실행문 …
	[ELSE]
	실행문 …
	ENDIF

FOR문	FOR *초기값, 최종값* [STEP *증가치*] *실행문* … NEXT
DO문	DO [WHILE *조건식*] *실행문* … LOOP [UNTIL *조건식*]
BREAK문	BRAKE
PRINT문	PRINT *출력문*
COMMENT문	// … 또는 /* … */

<표 1>의 제어명령 이외의 평가규칙을 수행하는데 필요한 명령들은 모두 함수형태로 지원되며, 외부 인터페이스에 의해 수행된다. 외부 인터페이스에 의해 수행되는 함수는 <표 2>와 같다.

〈표 2〉 외부 인터페이스 지원 함수

Name	Syntax
SQL	SQL (output DB, Query, input DB)
AGENT	AGENT (command, START [or STOP])
MODULE	MODULE (command)
SAVE	SAVE (filename, query, input DB)

<표 2>의 SQL함수는 input DB로부터 SQL 질의를 실행하고 그 결과를 output DB에 저장한다. AGENT함수는 등록된 에이전트를 실행시키거나 종료시키는 함수이며, command 매개변수를 이용하여 실행옵션을 지정한다. MODULE함수는 모듈을 실행시키기 위한 함수로 모듈이름과 실행옵션은 command 매개변수에 의해 지정된다. SAVE함수는 입력 데이터베이스로부터 SQL 질의를 실행하고, 그 결과를 패킷 전송용 데이터 타입으로 파일에 저장하는 함수이다.

MODULE 함수와 AGENT 함수의 파라미터로 사용되는 'command'에 포함되는 명령어로는 현재 sniffer와 sndpkt가 구현되어 있다. sniffer는 주어진 규칙에 맞는 packet을 네트워크상에서 또는 타겟 호스트의 에이전트로부터 혹은 시스템 내부의 DB로부터 스니핑하는 명령어이다. 현재 Sniffer로 모니터링 할 수 있는 프로토콜로는 ARP, IP, AH, ESP, TCP, UDP, ICMP, ISAKMP가 있다. Sndpkt는 시스템 내에 저장된 패킷이나 현재 편집한 패킷을 Raw Socket을 이용하여 목적 호스트로 전송하기 위한 명령어이다. 현재 Sndpkt로 전송할 수 있는 프로토콜로는 ARP, IP, AH, ESP, TCP, UDP, ICMP, ISAKMP가 있다. 규칙 해석기에서 사용하는 평가규칙의 예는 3장에서 볼 수 있다.

2.3 DBMS

보안평가시스템에서 사용되는 DB로는 Collection Data, Rule Data, Result Data, Temp Data가 있다.

Collection Data는 에이전트로부터 전송 받은 각종 프로토콜 데이터를 저장하는 곳으로, Database Control의 Collection Data Control에 의해 자동으로 각 프로토콜 별로 테이블을 생성하고 저장한다. 자동으로 생성되는 테이블은 Ethernet, ARP, IP, AH, ESP, TCP, UDP, ICMP, ISAKMP이며 이 중 AH, ESP, ICMP, ISAKMP에 대한 테이블의 구조를 <표 3>에서 볼 수 있다.

<표 3> Collection Data DB내의 각 테이블의 구조

• AH

Field Name	Type	설 명
ID	INTEGER	패킷에 대한 고유번호
LINKID	INTEGER	한 패킷내에서의 AH 순서번호
SERIAL	INTEGER	한 패킷내에서의 프로토콜 순서번호
NEXTHDR	CHAR	상위 계층 프로토콜의 유형
PAY_LEN	INTEGER	AH헤더의 전체길이
SPI	INTEGER	SA 식별자
SEQUENCE	INTEGER	단순 증가 카운터
AUTHENTICATION	MEDIUMTEXT	Authentication Data
TIME	VARCHAR	수행된 시간
RUNCOUNT	CHAR	수행 횟수

• ESP

Field Name	Type	설 명
ID	INTEGER	패킷에 대한 고유번호
SPI	INTEGER	SA 식별자
SEQUENCE	INTEGER	단순 증가 카운터
ENCRYPTION	MEDIUMTEXT	Encryption Data
TIME	VARCHAR	수행된 시간
RUNCOUNT	CHAR	수행 횟수

• ISAKMP

Field Name	Type	설 명
ID	INTEGER	패킷에 대한 고유번호
INIT_COOKIE	CHAR	SA을 시작한 Entity의 COOKIE
RESPOND_COOKIE	CHAR	SA에 응답 하는 Entity의 COOKIE
NEXT_PAY	CHAR	메시지내의 첫 번째 Payload의 타입
MAJ_VER	INTEGER	ISAKMP의 Major버전
MIN_VER	INTEGER	ISAKMP의 Minor버전
EXCHANGE_TYPE	CHAR	현재 사용되고 있는 Exchange타입
FLAGS	CHAR	Exchange에 대한 특정한 선택사항
MESSAGE_ID	INTEGER	메시지 식별자
TOTAL_LEN	INTEGER	헤더를 포함한 전체 Payload길이
ENCRYPTION	MEDIUMTEXT	Encryption된 데이터
TIME	VARCHAR	수행된 시간
RUNCOUNT	CHAR	수행 횟수

Rule Data는 평가에 대한 규칙을 저장하는 곳으로, Database Control의 Rule DB Control에 의해 평가규칙이 정의되거나 수정 및 삭제되며, 규칙 해석기에 의해 평가규칙이 해석되고 수행되어 진다. <표 4>는 Rule Data에 저장되는 평가규칙의 구조를 설명한 것이다.

<표 4> Rule Data DB

Field Name	Type	설 명
ID	INTEGER	평가 규칙의 고유 ID
NAME	CHAR	평가 규칙 이름
DESCRIPTION	CHAR	평가 규칙에 대한 설명

Result Data는 평가규칙이 수행된 결과를 저장하는 곳으로, Rule Interpreter의 Flow Control에 의해 저장되어지며, 구조는 <표 4>의 Rule Data DB에 TIME(규칙수행시간), PKTPTR(저장된 패킷 데이터에 대한 포인터) 필드가 추가된 형태를 갖는다. Result Data의 각 필드에 저장되는 값은 평가규칙이 수행되는 과정에서 화면에 출력되어지는 내용으로, 이들은 나중에 평가수행 기록과 결과를 보고자 할 때 사용된다. Temp Data는 평가규칙 수행 중에 발생하는 프로토콜 데이터의 임시 저장 공간이다.

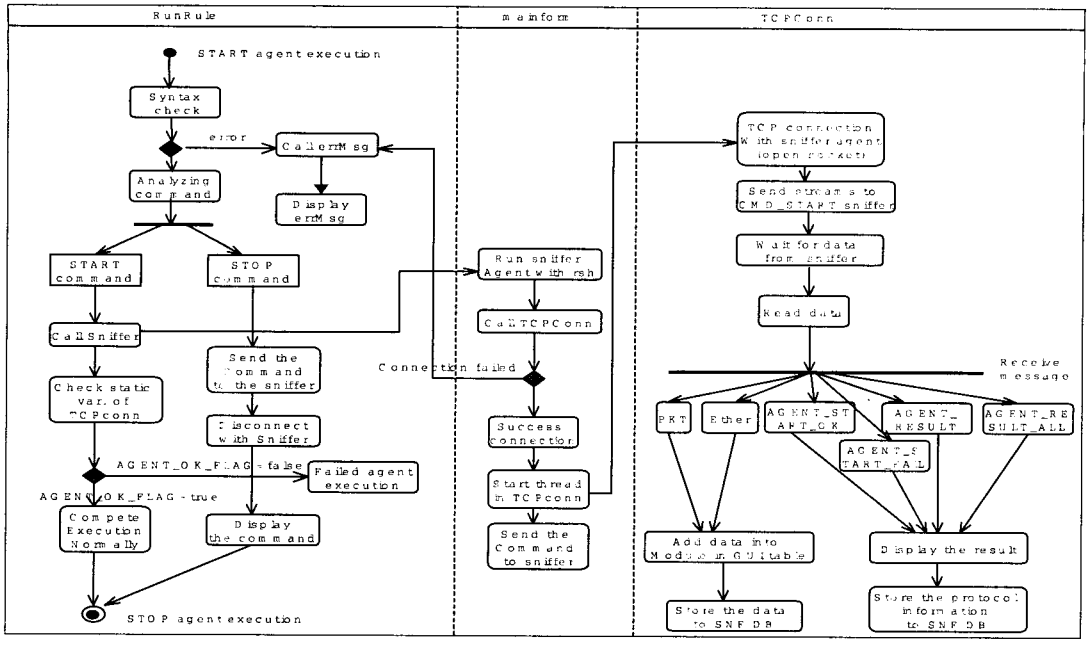
2.4 에이전트

에이전트는 평가 대상 시스템에서 수행되며 네트웍 단자를 통해 송수신되는 패킷들을 평가시스템으로 전달하는 기능을 수행한다.

평가시스템과의 접속이 성공되면 에이전트는 평가시스템으로부터 전송되는 명령을 우선 기다린다. 평가시스템은 START, STOP, HALT, RESUME의 4가지 명령을 송신할 수 있는데 START 명령을 송신하는 경우에는 스니퍼링할 패킷을 선별하기 위한 option 값을 함께 전송할 수 있다. 각 명령 및 OPTION 값은 다음과 같이 정의된다.

- START prot host1 ip1 host2 ip2 : 스니퍼링을 시작하라는 명령이며 스니퍼링을 위한 옵션으로 프로토콜(prot)과 패킷의 발신지(src), 패킷의 목적지(dest) 호스트를 지정할 수 있다. 옵션이 있는 경우 옵션에서 지정한 조건에 맞는 패킷만 스니퍼링을 수행한다.
- STOP : 스니퍼링을 종료하고 평가시스템과의 접속을 해지하라는 명령이다. 이 명령을 수신하게 되면 에이전트의 동작이 종료되게 된다.
- HALT : 스니퍼링을 일시 중지하라는 명령이다.
- RESUME : 스니퍼링을 계속하라는 명령이다.

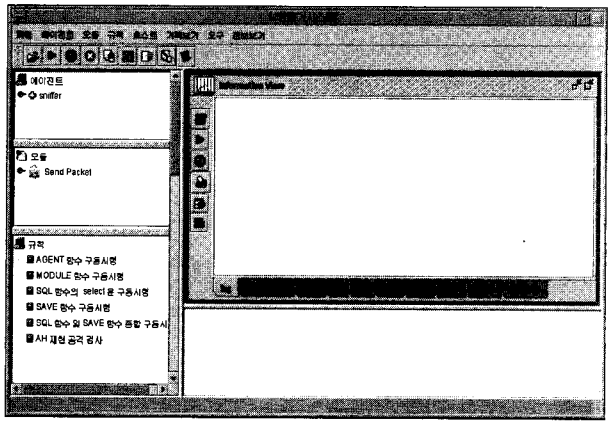
평가시스템과의 접속이 이루어지면 에이전트는 WAIT 상태에서 평가시스템으로부터의 명령을 기다린다. START 명령이 수신되면 상태를 ACTIVE로 전이시키고 스니퍼링을 위한 준비 작업을 수행한다. 이는 PCAP 라이브러리를 초기화하는 작업을 포함한다. 에이전트가 ACTIVE 상태에서는 10ms마다 PCAP 함수를 호출하여 패킷을 스니퍼링한다. (그림 5)에서는 에이전트의 수행과정을 도식화하였다.



(그림 5) 에이전트 함수의 수행 과정

3. 구 현

본 논문에서 제안한 보안평가 시스템은 Windows와 UNIX 환경에서 수행이 가능하며 Java와 C언어로 구현되었다. DB는 my-sql로 구현하였다. 시스템의 메인 화면은 (그림 6)과 같다.



(그림 6) 메인 화면

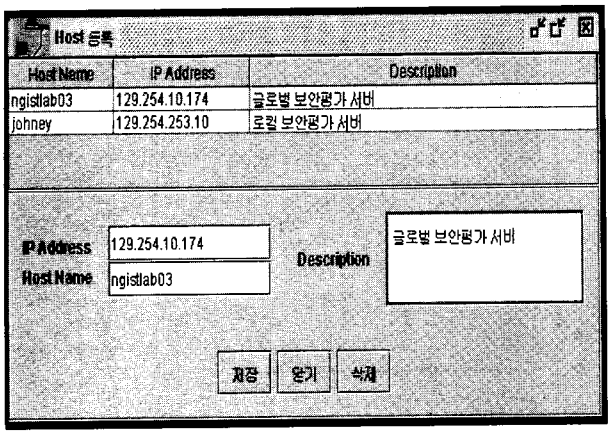
(그림 6)에서 좌측의 세 개의 창은 등록된 에이전트, 모듈, 규칙의 리스트를 tree형식으로 보거나 선택할 수 있는 창이다. 우측 상단의 창은 실행 로그 파일이나 에이전트에서 오는 각 프로토콜의 수집데이터를 보여주는 창이며 창 하단의 log는 log 기록보기를 선택하기 위한 버튼이며, Packet data는 수집된 packet 전체를, ARP는 수집된 packet중 ARP 패킷만, AH는 수집된 packet중 AH packet만 보기 위한 버튼이다. 나머지 IP, TCP, UDP, ICMP, ISAKMP모두 동일한 방식으로 원하는 프로토콜 packet만을 볼 수 있다. 이처럼 패킷을 각 필드별로 구분하여 보여줌으로 본 시스템은 IPsec 프로토콜 개발시 디버깅 툴로도 사용이 가능하다. 우측 하단의 창은 에이전트에서 오는 패킷 데이터가 수집되는 과정을 보여주는 텍스트 영역이다.

3.1 에이전트, 모듈, 호스트, 패킷 관리

(그림 6)상단의 메뉴 중 '에이전트'는 에이전트를 추가, 변경, 선택, 삭제할 때 사용하며, '모듈'은 모듈을 추가, 변경, 선택, 삭제할 때 사용된다. 마찬가지로 '규칙'은 규칙을 추가, 수정, 삭제할 때 사용되는 메뉴이다. '호스트' 메뉴는 타겟 호스트를 등록하고, 설치할 에이전트등의 정보를 입력하고 수정할 때 사용된다. '도구'메뉴는 사용자 설정, DB configuration등의 환경설정과 패킷 관리등에 사용된다.

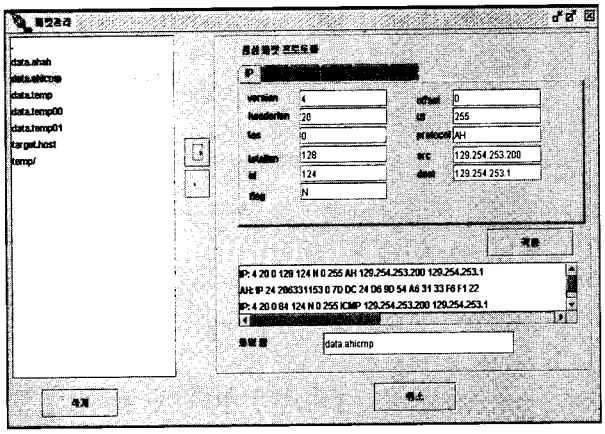
(그림 7)은 모듈 변경을 (그림 8)은 호스트 등록을 위한

(그림 7) 모듈 변경 창

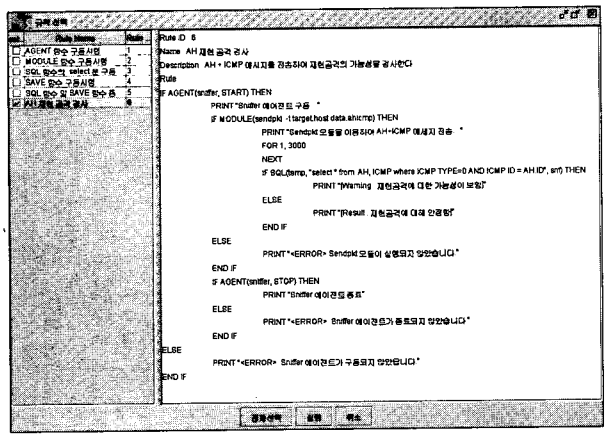


(그림 8) 호스트 등록 창

윈도우이다. (그림 9)는 패킷을 가공하기 위한 윈도우이다. (그림 9)의 패킷관리 창에서는 기존에 수집된 패킷을 보여주고 특정 필드의 값을 변경시켜 저장하거나 새로운 패킷을 만드는 기능을 제공한다. (그림 9)의 좌측 창에는 패킷 디렉토리내의 패킷 파일 리스트를 보여주며 우측 상단은 패킷을 편집하는 영역이며 우측하단은 선택된 파일에 저장된 패킷 리스트를 보여준다.



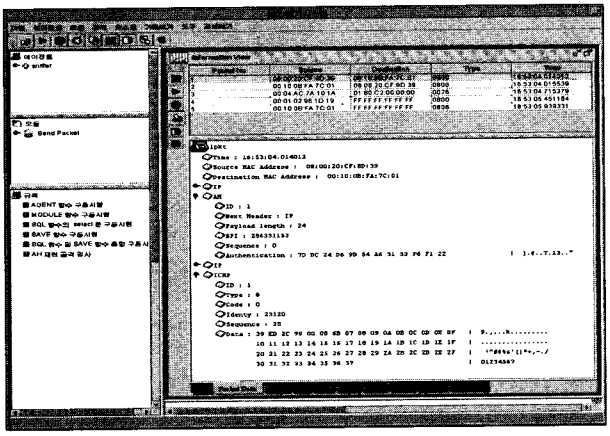
(그림 9) 패킷관리 창



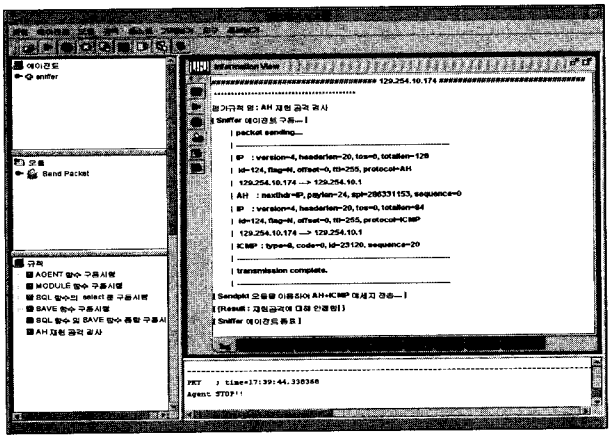
(그림 10) 규칙 선택

3.2 규칙 편집 및 실행

(그림 6)의 '규칙' 메뉴는 규칙을 입력, 수정, 선택, 삭제할 때 사용된다. (그림 10)은 이미 입력된 규칙 중 실행하고자 하는 하나의 규칙을 선택하기 위한 윈도우이다. (그림 10)에 기술된 규칙의 의미는 다음과 같다. "목적 호스트로 전달되는 AH가 적용된 ICMP패킷을 스니퍼링하여 저장한 후 동일한 패킷을 재전송 하여보고 결과를 분석한다. 만약 재전송한 패킷에 대한 응답이 목적 호스트로부터 오게 된다면 '재현공격에 대한 가능성이 보임' 이라는 경고메시지를 출력하고, 응답이 오지 않는다면 '재현 공격에 대해 안전함'이라는 메시지를 출력한다." (그림 11)과 (그림 12)는 (그림 10)의 규칙을 수행하였을 때 보여지는 화면이다. (그림 11)은 AH가 적용된 ICMP패킷을 목적호스트로부터 스니퍼링하는 단계에서 AH패킷을 보여주는 윈도우이며, (그림 12)는 log 탭을 선택한 경우로서 규칙에 대한 실행 기록을 보여준다. C-ISCAP에 대해 실험을 한 결과 재현공격에 대한 위협은 존재하지 않았다.



(그림 11) 스니퍼링 된 AH 패킷



(그림 12) 규칙 실행

3.3 보안성 평가

IPsec에서 제공하는 4가지 보안성을 보안평가시스템을

이용하여 평가하는 방법을 요약하면 다음과 같다.

- 기밀성(confidentiality)
  전송중인 암호화된 ESP 패킷을 수집하여 메시지의 내용을 알아 볼 수 있는지 확인하고 임의의 키로 복호화 해본다.
- 원적지 인증(Data Origin Authentication)
  전송중인 AH 혹은 ESP 패킷을 실시간으로 수집한 후 수집한 패킷 헤더 내의 Source IP 주소를 변경하여 목적지로 전송하고, 그 결과를 분석한다.
- 접근 제어(Access Control)
  임의의 키(key)를 이용하여 AH 혹은 ESP 패킷을 구성하여 목적지로 전송하고, 그 결과를 분석한다.
- 비연결형 무결성(Connnectionless Integrity)
  수집한 패킷의 특정 필드를 변경한 후 ICV값을 재 계산하여 변조한 후 전송하고, 그 결과를 분석한다.
- 재현공격 방어(Anti-replay)
  수집한 패킷을 복사하여 동일한 목적지로 전송하여 본다. 또는 수집된 패킷의 IPsec AH/ESP 헤더내의 SN (Sequence Number)값을 감시하여, 새로운 SN을 생성하거나 수집한 SN을 변경하여 전송하고, 그 결과를 분석한다.

실제 제안한 보안평가시스템을 이용하여 ETRI에서 개발중인 C-ISCAP (Controlled Internet Security Connectivity Assurance Platform)이라고 명명한 통합 IPsec엔진의 보안성을 평가하였다. C-ISCAP의 보안성을 평가하기 위해 75가지의 시험항목을 작성하였으며 각각의 시험항목을 보안평가시스템을 이용하여 평가하였다. 평가결과 C-ISCAP은 총 72가지의 보안성 평가 항목을 통과하였다. <표 5>는 75개의 평가항목 중 5개만을 평가결과와 함께 샘플로 보여준다.

〈표 5〉 보안성 평가항목과 평가결과의 일부

1. Detect modified AH packet (Modification of IP dst.)	RESULT [PASS]
H1 =======> H2	ICMP Echo request (with AH, modification of IP dst.)
H1 <===X=== H2	No ICMP Echo reply (Drop packet)
2. Inbound ESP packet with Fragmentation(Authentication : NONE, Encryption : DES-CBC)	RESULT [PASS]
H1 ==========> H2	Send Fragmented TCP message (with ESP) (1^{st} /2^{nd} / 3^{rd} ··· fragment)
3. Inbound Tunnel AH packet (Authentication : HMAC-MD5)	RESULT [PASS]
H1 ===> GW1 ===> GW2 ===> H2	ICMP Echo request (with Tunnel AH)
|----------Tunnel------|	
H1<=== GW1 <=== GW2 <=== H2	ICMP Echo reply (with Tunnel AH)
4. Outbound AH + ESP packet with SA bundles	RESULT [FAILED]
H1 ==> GW1 ==> GW2 ==> H2	ICMP Echo request (with SA bundles)
|-------ESP SA---|	
|------------AH SA-------|	
H2 <== GW1 <== GW2 <== H1	ICMP Echo reply (with SA bundles)
|-----ESP SA-------|	
|----------AH SA---------|	
5. Inbound AH+ESP (policy = drop)	RESULT [PASS]
H1 ======> H2	ICMP Echo request (with AH + ESP)
H1 <==X == H2	No ICMP Echo reply (drop packet)

4. 결 론

본 논문에서는 IPsec엔진의 보안성을 평가하기 위한 자동화된 규칙기반 보안평가시스템을 설계 및 구현하였다. 제안하는 보안평가시스템은 ETRI에서 개발중인 C-ISCAP(Controlled Internet Security Connectivity Assurance Platform)이라고 명명한 통합 IPsec 엔진에 대한 보안성을 평가하고 디버깅하기 위한 목적으로 개발되었으나, 독립적으로 존재하는 평가시스템으로 C-ISCAP 뿐 아니라 기타 현재 개발중인 IPsec 엔진에 대해서도 아무런 변경 없이 평가를 수행할 수 있다.

본 평가시스템은 다음과 같은 특징을 갖는다.

- 네트워크 상의 다양한 프로토콜들의 패킷을 수집하고 분석하는 기능을 갖는다.
- 에이전트를 사용하여 원거리 호스트에 대한 평가가 가능하다.
- 규칙기반으로 동작하므로 자동화된 보안성 평가가 가능하다.
- 규칙에 대한 문법을 단순화하여 손쉽게 규칙을 정의할 수 있다.
- 평가규칙에 필요한 기능을 모듈로 관리하므로 확장이 용이하다.

현재 보안평가시스템은 100% 자동화되지는 않았으며 일부 수동 작업이 필요하며, 계속 보완해 나갈 예정이다. 향후 과제로서 보안평가시스템의 규칙을 추가하여 다양한 방식의 보안성 평가가 가능하도록 확장하는 작업이 필요하다. 현재 평가시스템은 Java와 DBMS를 이용하여 구현되어서 규칙을 수행하는 시간이 오래 걸리므로 평가 대상 호스트와의 연결이 끊어지는 현상이 종종 발생한다. 이러한 속도문제를 해결하기 위한 연구도 현재 진행중이다. 또한 기타의 개발된 다른 IPsec엔진에 대한 테스트를 수행하는 작업도 현재 진행중이다.

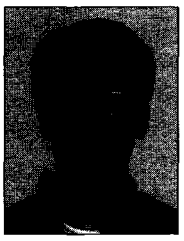
참 고 문 헌

[1] IETF, http : //www.ietf.org.
[2] S. Kent and R. Atkinson, Security Architecture for the Internet Protocol, RFC2401, Nov., 1998.
[3] S. Kent and R. Atkinson, IP Authentication Header, RFC 2402, Nov., 1998.
[4] S. Kent and R. Atkinson, IP Encapsulating Security Payload, RFC2406, Nov., 1998.
[5] D. Harkins, D. Correl, Internet Key Exchange, RFC2409, Nov., 1998.
[6] USAGI Project, http : //www.linux-ipv6.org/.
[7] FreeS/WAN, http : //www.ipv6.iabg.de/.

[8] KAME, http : //www.kame.net.
[9] ISS, Network and Host-based Vulnerability Assesment, AXENT, http : //www.axcent.com.
[10] Cisco Scanner, http : //www.cisco.com/univercd/cc/td/doc/pcat/nssq.htm.
[11] LANguard Network&Port scanner, http : //www.gfi.com/languard/lanscan.htm.
[12] J. H. Jeong, J. H. Nah, S. W. Sohn and J. T. Lee, "C-ISCAP : Controlled-Internet Secure Connectivity Assurance Platform," Proc. of the IEEE International Conference on Enterprise Information Systems(ICEIS2001), Vol.2, pp.920-925, Setubal, Protugal.
[13] ISS, "Securing Operating Platforms : A solution for tightening system security," Jan., 1997.

권 혁 찬

e-mail : hckwon@etri.re.kr
1994년 서원대학교 전자계산학과 공학사
1996년 충남대학교 전산학과 이학석사
2001년 충남대학교 컴퓨터과학과 이학박사
2001년~현재 한국전자통신연구원 정보보호연구본부 선임연구원
관심분야 : 네트워크 보안, IPv6, IPsec, 에이전트 시스템



현 정 식

e-mail : hjs62845@etri.re.kr
1999년 청주대학교 컴퓨터정보공학과 공학사
2001년 청주대학교 전자계산학과 공학석사
2001년~현재 충북대학교 전자계산학과 박사과정
2000년~현재 한국전자통신연구원 정보보호연구본부 위촉연구원
관심분야 : 네트워크 보안, IPv6, Ad hoc Network, P2P Network

김 상 춘

e-mail : kimsc@samchok.ac.kr
1986년 한밭대학교 전자계산학과 졸업
1989년 청주대학교 전산학과 공학석사
1999년 충북대학교 컴퓨터과학과 이학박사
1983년~2001년 한국전자통신연구원 정보보호연구본부 선임기술원
2001년~현재 삼척대학교 정보통신공학과 교수
관심분야 : 네트워크 보안, IPsec, 정보보호

나 재 훈

e-mail : jhnah@etri.re.kr
1985년 중앙대학교 컴퓨터공학과 공학사
1987년 중앙대학교 대학원 컴퓨터공학과 공학석사
1987년~현재 한국전자통신연구원 정보보호연구본부 팀장 / 책임연구원
관심분야 : 네트워크 보안, IPsec, Active Network, Secure OS, 무선인터넷 보안

손 승 원

e-mail : swsohn@etri.re.kr
1984년 경북대학교 전자공학과 공학사
1994년 연세대학교 대학원 전자공학과 공학석사
1999년 충북대학교 대학원 컴퓨터공학과 공학박사
1996년 정보통신 기술사 취득
1983년~1986년 삼성전자(주) 연구원
1986년~1991년 LG전자(주) 중앙연구소 HI8mm 캠코더 팀장
1991년~현재 한국전자통신연구원 정보보호연구본부 네트워크 보안연구부장 / 책임연구원
관심분야 : IC card, Biometry, Active Network, 생체인식 분야