2001 년 한국정보처리학회 추계 학술발표논문집 제 8 권 제 2 호

정보보호 서비스를 위한 규칙기반 프로토콜 보안평가 시스템 설계

현정식* 권혁찬** 나재훈** 손승원**
*충북대학교 전자계산학과
**한국전자통신연구원 정보보호기술연구본부 네트워크보안연구부
e-mail : hyun708@chongju.ac.kr

Design of the Rule Based Protocol Security Evaluation System for Internet Secure Service

Jeung-sik Hyun*, Hyeok-chan Kwon**, Jae-hoon Nah**, Sung-won Sohn**
*Dept. of Computer Science, Chungbuk National University
**Information Security Technology Division, ETRI

요 약

인터넷에서 정보보호 서비스를 제공하는 시스템은 일반적인 시스템보다 보안성 유지의 필요성이 더욱 중요하다. 그렇기 때문에 시스템의 안전성, 즉 시스템의 구현상의 적합성과 보안성을 평가하는 기술이 필요하다. 특히 고도로 발전하는 해킹기술에 대해 시스템이 얼마만큼의 정보보호 서비스를 제공하는지에 대해 평가할 수 있어야 그 시스템의 적합성과 보안성을 확인할 수 있다. 이러한 보안성 평가 기술은 정보보호 서비스를 제공하는 시스템에 독립적으로 구동 되어야 하고, 고도로 발전하는 해킹기술에 대해 유연히 대처할 수 있어야 한다. 본 논문에서는 프로토콜 레벨의 정보보호 서비스를 제공하는 시스템에 대해 다양한 규칙을 적용하여 시스템의 적합성 및 보안성을 객관적으로 평가할 수 있는 규칙기반 프로토콜 보안평가 시스템을 설계한다.

1. 서론

최근 인터넷과 WWW(World Wide Web)의 이용이 폭발적으로 증가하면서 인터넷 정보보호 서비스에 대한 많은 연구가 진행되었고, 그 중 상당수의 시스템이 이미 인터넷 정보보호 서비스를 제공하고 있다. 하지만 고도로 발전하는 해킹기술로부터 이들 시스템이 제공하는 정보보호 서비스를 보호하고 유지하기 위해서는 그에 적합한 보안성 취약점 점검을 수행하고 기능을 향상시킬 수 있는 보안도구가 필요하다.

현재까지 알려진 보안도구 중에는 COPS, tripwire, SATAN[1], ISS[2], netlog 등이 대표적이다. 이들 보안도구의 대부분은 시스템의 보안관련 파일을 점검하거나 감시하여 시스템의 보안 취약점을 검사한다. 그리고 그 중 일부만이 네트워크에 대한 보안 취약점을 검사하기 위한 포트스캔과 패킷 모니터링을 수행한다[3]. 결국 네트워크 프로토콜이 가지고 있는 고유의 보안 취약점에 대한 보안성 평가를 수행하는 보안도구는 전무하다.

네트워크 프로토콜의 취약점을 이용한 해킹방법으로는 SYN flooding, TCP Sequence number 공격, ICMP 공격, IP Spoofing, TCP Connection Hijacking 등이 있다[4]. 물론 침입탐지 시스템[5]에 의해 네트워크에 접속하는 사용자들의 패킷을 감시하고, 프로토콜의 취약점을 이용한 공격에 대한 패턴분석으로 해킹에 대해 방어 할 수 있다. 하지만 정보보호를 목적으로 개발되는 프로토콜들에 대해 개발단계에서 그들의 보안 취약점을 평가할 수 있다면 네트워크 프로토콜을 통한 불법적인 행동이 발생하기 전에 해킹에 대한 위험요소를 제거할 수 있다.

2. 참조모델

본 논문에서 설계한 프로토콜 보안평가 시스템은 Cisco 의 NetSonar[6]를 기본 모델로 참조하여 설계하였다. NetSonar 는 네트워크 취약점 목록 데이터베

이스를 이용하여 해커가 네트워크상의 보안 취약점을 이용하기 전에 취약요소를 발견하고, 관리자에게 이를 통보해 준다. NetSonar 는 취약 호스트와 OS 의 결함, 취약 보안 레벨 및 취약점에 대한 설명, 그리고 관련된 데이터를 수집하는 등과 같은 세부기능을 제공하고, 독립된 네트워크 뿐만 아니라 인터넷에 연결된 모든 IP 기반의 네트워크를 검사할 수 있다. 하지만 NetSonar 또한 네트워크와 관련된 파일의 점검 및 감시와 네트워크 포트스캔 및 패킷 모니터링을 통해 보안 취약점을 검사하므로 프로토콜이 가지고 있는 보안 취약점에 대해서는 평가할 수 없다. NetSonar 의 기능과 수행절차는 [그림 1]과 같다.

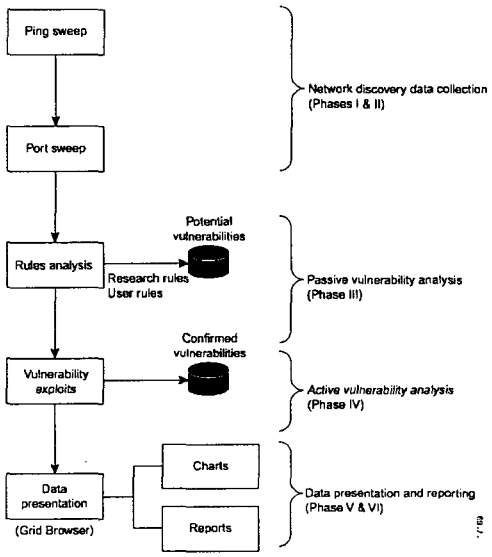

그림 1. NetSonar 단계별 기능 및 수행절차

3. 프로토콜 보안평가 시스템 설계

본 논문에서 설계한 프로토콜 보안평가 시스템은 개발중인 정보보호 프로토콜의 적합성 시험과 기존에 알려져 있는 프로토콜 취약점 공격에 대한 보안성 평가를 수행하고, 향후 발생할 수 있는 프로토콜의 보안 취약점에 대해 대비할 수 있다. 그리고 확장성을 고려한 규칙기반 시스템으로 고도로 발전하는 해킹기술에도 유연하게 대처할 수 있다. [그림 2]는 이러한 프로토콜 보안평가 시스템의 형상을 나타낸다.

프로토콜 보안평가 시스템은 크게 시스템을 총체적으로 제어하는 Java 기반의 GUI, 시스템에 필요한 데이터를 관리하는 DBMS, 평가에 사용할 데이터를 수집하는 에이전트, 그리고 평가 규칙에서 사용하는 모듈로 이루어져 있다. 그리고 이러한 프로토콜 보안평가 시스템에서 가장 중요한 것은 수집된 데이터에 대해 평가규칙을 유연하게 적용하고 실행할 수 있는 인터프리터이다. 인터프리터는 평가규칙을 DBMS 의 규칙 데이터(Rule Data)로부터 순차적으로 읽은 다음, 수행절차에 따라 명령을 해석하고 실행한다. 인터프리터의 구조는 [그림 3]과 같다.

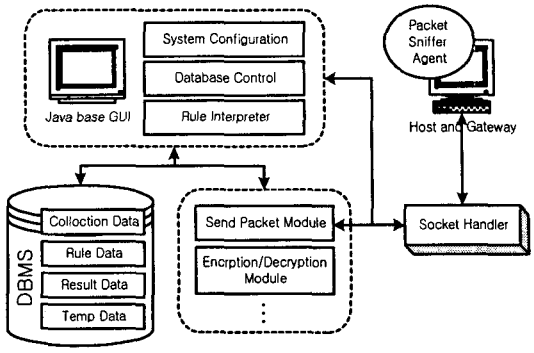

그림 2. 시스템 형상

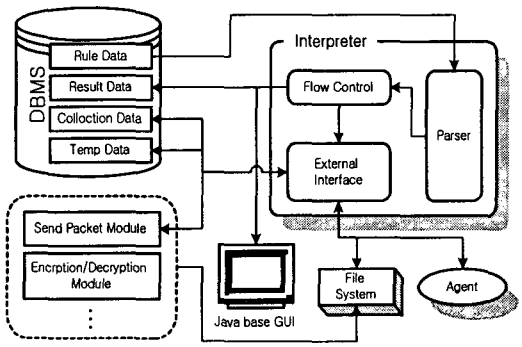

그림 3. 규칙 인터프리터

DBMS 의 규칙 데이터는 GUI 의 데이터베이스 제어부분에 의해 정의되고 편집된 평가규칙이 저장되며, 각 필드의 구조는 [표 1]과 같다. 이 중 프로그램 필드에는 인터프리터에서 해석되고 수행될 평가 프로그램이 작성되어 저장된다.

Name	Type	Description
ID	INTEGER	규칙의 고유 ID
NAME	CHAR	규칙명
DESCRIPTION	CHAR	규칙에 대한 설명
PROGRAM	TEXT	평가 프로그램

표 1. 규칙 데이터 필드구조

인터프리터의 파서(Parser)는 규칙 데이터의 프로그램 필드에 저장되어 있는 평가 프로그램을 읽은 다음, 각 단어를 분리하여 흐름제어(Flow Control)로 보낸다.

흐름제어는 파서로부터 받은 각 단어에 할당되어 있는 제어명령 등을 수행하고, 명령수행 과정에 있는 함수들을 외부 인터페이스(External Interface)로 제공한다. 그리고 명령수행 과정에서 문법적 에러가 발생하면 GUI 의 로그창(log window)와 DBMS 의 결과 데이터(Result Data)에 해당 에러메시지를 출력하고 실행중인 평가 프로그램을 종료한다.

외부 인터페이스는 수집 데이터(Collection Data)나 그 외 평가규칙에 의해 생성되거나 삭제되는 임시 데이터(Temp Data)와 같은 DBMS 를 조작하고, 평가규칙에서 사용하는 에이전트와 모듈을 제어하는 함수를 수행한다. 그리고 파일 시스템을 사용하여 규칙실행에 필요한 모듈과의 인터페이스를 제공한다. 이와 같은 인터프리터의 수행절차는 [그림 4]와 같다.

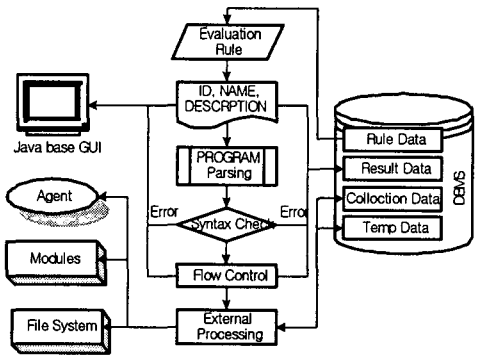

그림 4. 인터프리터의 수행절차

인터프리터는 먼저 DBMS 의 규칙 데이터로부터 평가규칙을 읽은 다음, ID 필드와 NAME 필드, 그리고 DESCRPTION 필드를 GUI 의 로그창과 DBMS 의 결과 데이터(Result Data)에 출력하고, PROGRAM 필드의 내용을 파싱한 다음, 명령어들의 문법적 형식을 검사한다. 그리고 문법적 형식이 틀릴 경우에는 해당 에러메시지를 GUI 의 로그창과 DBMS 의 결과 데이터에 출력하고 현재 적용중인 평가규칙을 종료한다.

흐름제어는 평가규칙의 흐름을 제어하는 제어명령 등을 수행하는 곳으로 베이직과 유사한 문법적 구조를 사용한다. 흐름제어에서 사용하는 제어명령들은 [표 2]와 같다.

Command	Syntax

IF 문	IF 조건식 THEN
	  실행문
	  ...
	[ELSE]
	  실행문
	  ...
	ENDIF

FOR 문	FOR 초기값, 최종값 [STEP 증가치]
	  실행문
	  ...
	NEXT

DO 문	DO [WHILE 조건식]
	  실행문
	  ...
	LOOP [UNTIL 조건식]

BREAK 문	BRAKE
PRINT 문	PRINT 출력문
COMMENT 문	// ~ 또는 /* ~ */

표 2. 제어명령

제어명령내에서 사용하는 조건식은 베이직의 관계연산자와 논리연산자를 사용하여 구성된다. 그리고 PRINT 문은 출력문을 GUI 의 로그창과 DBMS 의 결과 데이터에 출력하는데 사용하고, 마지막으로 BRAKE 문은 C 언어의 BRAKE 문과 동일하게 제어명령의 범위에서 벗어나는데 사용한다.

흐름제어에서 사용하는 제어명령 이외의 평가규칙을 수행하는데 필요한 명령들은 모두 함수형태로 지원되고, 이는 외부 인터페이스에 의해 수행된다. 외부 인터페이스에 의해 수행되는 함수는 [표 3]과 같다.

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

표 3. 외부 인터페이스 지원 함수

SQL 함수는 입력 데이터베이스로부터 SQL 질의를 실행하고 그 결과를 출력 데이터베이스에 저장한다. 그리고 SQL 함수는 SQL 질의에 해당하는 결과 데이터가 존재할 경우 TRUE 을 복귀시키고, 존재하지 않을 경우 FALSE 을 복귀시킨다.

AGENT 함수는 GUI 를 통해 등록되어 있는 에이전트를 실행시키거나 종료시키는 함수로, AGENT 함수를 통해 실행될 에이전트의 이름과 실행옵션은 command 매개변수에 의해 처리된다. 그리고 에이전트가 정상적으로 실행되거나 종료되면 AGENT 함수는 TRUE 를 복귀시키고, 그렇지 않으면 FALSE 를 복귀시킨다.

MODULE 함수는 AGENT 함수와 매우 유사하며, 그 차이점이라고는 에이전트 대신 모듈을 실행한다는 것밖에는 없다.

마지막으로 SAVE 함수는 입력 데이터베이스로부터 SQL 질의를 실행하고, 그 결과를 패킷 전송용 데이터타입으로 파일에 저장한다. 이 함수도 SQL 함수와 같이 질의결과가 있을 경우에는 TRUE 를, 없을 경우에는 FALSE 를 복귀한다.

지금까지 설명한 인터프리터에서 사용하는 평가규칙의 예를 들어보면 다음과 같다.

- ID: 1
- DESCRIPTION : 프로토콜 Reply 공격에 대한 평가
- PROGRAM :

```
PRINT " 패킷 수집 에이전트 구동 ..."
AGENT(sniffer, START)
PRINT " ICMP 패킷 검색 ..."
DO
LOOP UNTIL SQL(TEMP_DB, select * from ICMP
  where TYPE=" ICMP_ECHO" , SNIFF_DB)
PRINT " REPLY 공격용 ICMP 패킷 생성"
SQL(TEMP_DB, update ICMP SET ID=1000 WHERE
  TYPE=" ICMP_ECHO" , TEMP_DB)
SAVE(icmp.echo, select * from IP)
```

```
PRINT " REPLY 공격용 ICMP 패킷 전송..."
MODULE(sndpkt icmp.echo)
PRINT " 프로토콜에 대한 REPLY 공격에 대한
  결과 도출 중 ..."
FOR 1, 100
  IF SQL(SNIFF_DB, select * from ICMP
      where TYPE=" ICMP_REPLY"  AND
      ID=1000, TEMP_DB) THEN
    PRINT " REPLY 공격 위험이 존재합니다."
    BREAK
NEXT
PRINT " 평가 종료"
```

4. 결론

본 논문에서 설계한 규칙기반 프로토콜 보안평가 시스템은 일차적으로 차세대 인터넷 정보보호 프로토콜인 IPSec을 목표로 하고 있다. 그러므로 다음과 같은 프로토콜의 적합성 시험과 보안성 평가를 수행할 수 있다.

- 재현공격에 대한 평가
- 비연결형 무결성에 대한 평가
- 원격지 인증에 대한 평가
- 접근제어에 대한 평가
- 기밀성에 대한 평가
- 서비스 거부 공격에 대한 평가
- IP Spoofing 공격에 대한 평가
- TCP Connection Hijacking 공격에 대한 평가
- SYN flooding 공격에 대한 평가

그리고 다음과 같은 특징을 가진다.

- 다양한 프로토콜들의 패킷을 모니터링할 수 있다.
- 에이전트를 사용하여 원거리 호스트에 대해 평가할 수 있다.
- 평가규칙에 필요한 기능을 모듈로 관리하므로 확장이 용이하다.
- 가상적으로 프로토콜을 구현하여 시험할 수 있다.

하지만 앞으로 개발될 보다 다양한 프로토콜에 대해 보다 복잡한 평가규칙을 적용하기 위해서는 평가규칙의 평가 프로그램에 변수의 개념이 추가되어야 할 것이다. 왜냐하면 변수의 개념이 없는 지금의 평가 프로그램으로는 패킷 데이터를 섬세하게 조작할 수 없기 때문이다. 그리고 Java 기반의 GUI 와 DBMS 의 사용으로 패킷에 대한 생성 및 재가공 속도가 너무 늦어 평가 대상 호스트와의 네트워크가 끊어질 우려가 있으므로 패킷에 대한 응답속도를 향상시킬 필요가 있다.

참고문헌

[1] Larry J. Hughes, Jr., "Actually Useful Internet Security Techniques," New Riders Publishing, 1995.
[2] "Network and Host-based Vulnerability Assessment," http://documents.iss.net/white papers/nva.pdf
[3] "인터넷 보안 검사 도구," http://wwwcs.dongguk.ac.kr/~dh999/Security/chap3.html
[4] "프로토콜 관련 보안," 월간 LAN Times, June, 1996.
[5] V. Paxson, "Bro: A System for Detecting Network Intruders in Real-Time," Computer Networks, vol. 31, no. 23-24, pp. 2435-2463, Dec. 1999.
[6] Cisco, "NetSonar Security Scanner," Release Notes, May, 1999.