포트 스캐닝 기법 기반의 공격을 탐지하기 위한 실시간 스캔 탐지 시스템 구현

(A Real Time Scan Detection System against Attacks based on Port Scanning Techniques)

송 중 석 † 권 용 진 ††
(Joong-Suck Song) (Yong-Jin Kwon)

요 약 포트 스캐닝 탐지 시스템은 "False Positive"(실제 공격이 아닌데 공격이라고 탐지, 오탐지)와 "False Negative"(실제 공격인데 공격이 아니라고 탐지, 미탐지)가 낮아야 하는 등의 시스템 성능에 관한 요구사항과, 해당 탐지 시스템을 활용한 보안관리가 용이해야 하는 등의 사용자 친화적인 요구사항을 만족할 필요가 있다. 그러나 공개되어 있는 실시간 스캔 탐지 시스템은 False Positive가 높고 다양한 스캔 기법에 대한 탐지가 잘 이루어지지 않고 있다. 또한 실시간 스캔 탐지 시스템의 대부분이 명령어 기반으로 이루어져 있기 때문에 이를 활용하여 시스템 보안 관리를 수행하는데 많은 어려움이 있다. 따라서 본 논문에서는 새로운 필터 룰 집합의 적용에 의해 포트 스캐닝 기법 기반의 다양한 공격을 탐지 할 수 있고, 공격자의 행동 패턴으로부터 유도된 ABP-Rule의 적용에 의해 False Positive를 최소화할 수 있는 실시간 스캔 탐지 시스템(TkRTSD)을 제안한다. 또한 Tcl/Tk를 이용하여 GUI환경을 구축함으로써 사용자가 쉽게 보안관리를 할 수 있는 사용자 친화적인 탐지 시스템을 제안한다.

키워드 : 스캔 탐지 시스템, 스캔 기법, 오탐지, 미탐지 포트 스캐닝, Tcl/Tk

Abstract *Port scanning detection systems should rather satisfy a certain level of the requirement* for system performance like a low rate of "False Positive" and "False Negative", and requirement for convenience for users to be easy to manage the system security with detection systems. However, public domain Real Time Scan Detection Systems have high rate of false detection and have difficulty in detecting various scanning techniques. In addition, as current real time scan detection systems are based on command interface, the systems are poor at user interface and thus it is difficult to apply them to the system security management. Hence, we propose TkRTSD(Tcl/Tk Real Time Scan Detection System) that is able to detect various scan attacks based on port scanning techniques by applying a set of new filter rules, and minimize the rate of False Positive by applying proposed ABP-Rules derived from attacker's behavioral patterns. Also a GUI environment for TkRTSD is *implemented by using Tcl/Tk for user's convenience of managing network security.*

Key words : Scan detection system, Scanning techniques, False Positive, False Negative, Port Scanning, Tcl/Tk

1. 서 론

전 세계를 연결해 주는 인터넷의 발전으로 어느 곳에서나 컴퓨터를 이용해서 쉽고 편리하게 원하는 정보를 얻을 수 있게 되었다. 이러한 인터넷의 발전과 더불어 해킹기법 또한 함께 발전하고 있다. 현재까지 SATAN, Mscan, Sscan, Nessus, Nmap[1-5] 등과 같은 네트워크 보안 취약점을 검색해주는 보안 관리 도구들이 공개되어 있다. 그러나 시스템 관리자가 아닌 해커들은 이러한 보안 관리 도구들을 이용하여 침입하고자 하는 시스템의 보안 취약점 정보 및 공격대상을 찾는데 활용하고 있다.

포트 스캐닝은 침입하고자 하는 시스템의 열려있는 포트를 알아보기 위한 기법을 말하며 시스템에 직접적인 피해를 주지는 않지만 침입하고자 하는 시스템의 취약점 정보를 수집하는 첫 단계이다. 따라서 포트 스캐닝은 해킹의 징조라고 할 수 있다. 누군가 자신의 시스템

† 학생회원 : 한국항공대학교 정보통신공학과
oaktree@tikwon.hangkong.ac.kr
†† 비 회 원 : 한국항공대학교 전자.정보통신.컴퓨터공학부 교수
yjkwon@tikwon.hangkong.ac.kr
논문접수 : 2003년 4월 17일
심사완료 : 2003년 12월 22일

에 침입 하고자 한다는 것을 미리 안다면 해킹을 막는데 유용할 것이다. 그러므로 포트 스캐닝에 대한 정확한 탐지는 네트워크 시스템 보안에 있어서 중요한 문제이다. 본 논문에서의 "스캔 탐지"는 포트 스캐닝에 대한 탐지를 말한다.

포트 스캐닝 공격을 탐지하기 위한 탐지 시스템은 "False Positive"(실제 공격이 아닌데 공격이라고 탐지, 오탐지)와 "False Negative"(실제 공격인데 공격이 아니라고 탐지, 미탐지)이 낮아야 하는 등의 시스템 성능에 관한 요구사항과, 해당 탐지 시스템을 활용한 보안 관리가 용이해야 하는 등의 사용자 친화적인 요구사항을 만족할 필요가 있다. 그러나 공개되어 있는 실시간 스캔 탐지 시스템은 Open Scan, Half-Open Scan, Stealth Scan, UDP Scan 등 다양한 스캔 기법에 대해서 탐지를 할 수 없으며 False Positive 또한 높다. 더욱이 실시간 스캔 탐지 시스템의 대부분이 명령어 기반으로 이루어져 있기 때문에 이를 활용하여 시스템 보안 관리를 수행하는데 많은 어려움이 있다.

본 논문에서는 포트 스캐닝 기법 기반의 다양한 공격을 탐지 할 수 있고 False Positive를 최소화할 수 있는 실시간 스캔 탐지 시스템(TkRTSD)을 제안한다. 이를 위하여 효율적인 필터 룰 집합과 공격자의 행동 패턴으로부터 유도된 ABP-Rule을 제안한다. 또한 Tcl/Tk를 이용하여 GUI환경을 구축함으로써 사용자가 쉽게 보안 관리를 할 수 있는 사용자 친화적인 탐지 시스템을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 포트 스캐닝 탐지 시스템에 대해 설명한다. 3장에서는 제안된 TkRTSD의 전체 구성도, 다양한 스캔 기법 기반의 공격들을 탐지하기 위한 방법, False Positive를 최소화하기 위한 알고리즘을 소개한다. 4장에서는 사용자가 쉽게 사용할 수 있도록 Tcl/Tk를 이용하여 GUI환경으로 구축한 TkRTSD를 소개한다. 마지막으로 5장에서 결론을 제시한다.

2. 포트 스캐닝 탐지 시스템

2.1 포트 스캐닝 기법

가장 일반적인 포트 스캐닝 기법[7-9]의 종류는 표 1과 같다.

Open Scan은 열려진 포트에 대한 신뢰할 수 있는 정확한 결과를 얻을 수 있고 Unix-based 시스템 하에서 root 권한 없이도 가능하지만 쉽게 로깅 되고 탐지될 수 있으며 Firewall과 같은 침입 차단 시스템에 의해 쉽게 필터링 된다. Half-Open Scan은 Three-way handshake를 피할 수 있어 Open Scan 보다는 탐지가 어렵다. 하지만 일반적인 SYN 패킷을 만들기 위해서는 root 권한을 가져야 하고 많은 SYN 시도는 방화벽에서 필터링 된다. Stealth Scan은 Firewall이나 IDS(Intrusion Detection System)와 같은 보안 시스템들을 우회하기 위한 Scanning 기법으로 패킷에 SYN 외에 다른 flag를 설정하는 방법이다. 하나의 예로, FIN Stealth Scan의 경우 닫혀진 포트들은 FIN 패킷에 RST로 응답하는 경향이 있고 열려진 포트들은 FIN 패킷을 무시해 버린다. 즉, 포트가 열려 있으면 패킷이 drop 되어 응답이 없고 닫혀 있으면 RST 응답이 전송된다.

표 1 포트 스캐닝 기법의 종류

기법	종류
Open Scan	TCP Connect, Reverse Ident Scan
Half-Open Scan	SYN flag
Stealth Scan	FIN flag, NULL flag, XMAS flag, ACK flag, Window Scan, TCP Fragmentation
기타	UDP Scan, IP Protocol Scan

2.2 스캔 탐지 시스템

2.1에서 설명한 것처럼 보안 시스템들을 우회하기 위하여 포트 스캐닝 기법들이 다양해지고 있지만, 기존의 탐지 시스템은 다양한 스캔 기법을 탐지할 수 없으며 False Positive가 매우 높다. 예를 들어, 한국 정보보호진흥원에서 개발한 RTSD(Real Time Scan Detector)[10,11]는 표 1의 스캐닝 기법 중 TCP Connect Scan과 SYN flag Scan만이 탐지 가능하고 명령어 기반으로 이루어져 있기 때문에 이를 이용한 보안 관리가 쉽지 않다. NIDS(Network-based Intrusion Detection System)중에서 가장 잘 알려진 오픈 소스인 Snort[12,13]는 포트 스캐닝에 대한 탐지 결과를 Portscan.log에 기록 하게 되는데, 표 1의 스캐닝 기법 중 UDP Scan, IP Protocol Scan, Window Scan은 탐지를 하지 못한다. 한편, Snort는 SnortSnarf[14]라는 GUI 환경을 지원한다. 실시간 스캔 탐지 시스템으로 잘 알려진 PortSentry[15,16]는 Open Scan, Half-Open Scan, Stealth Scan 및 UDP Scan의 탐지가 가능하지만 TCP 및 UDP에 대해서만 지원을 하기 때문에 표 1의 스캐닝 기법 중에서 IP Protocol Scan은 탐지를 하지 못한다. 더욱이 PortSentry는 False Negative가 높은 경향이 있고 GUI 환경을 지원하지 않는다.

2.3 스캔 탐지 시스템의 스캔 탐지 방법

기존의 스캔 탐지 시스템은 하나의 IP로부터 일정한 시간 간격으로 동일한 시그너쳐를 갖는 패킷이 임계 값 이상 캡쳐 되었을 때 스캔 공격으로 탐지한다. 예를 들어 임계 값이 5이고 시간 간격이 1초라고 하자. 만약 한 호스트로부터 1초 간격으로 동일한 시그너쳐를 갖는 패

킷이 5개 이상 캡쳐되면 스캔 공격으로 탐지 한다. 이러한 상황에서, 정상적인 ftp connect, web surfing, web crawling 등도 하나의 IP에서 일정한 시간 간격으로 시그너쳐가 동일한 여러 개의 패킷을 보낼 수 있기 때문에 기존의 스캔 탐지 시스템에 의해서는 공격으로 탐지된다. 따라서, 위의 방법으로는 False Positive를 낮추는데 한계가 있다. 그러므로 기존의 스캔 탐지 시스템의 False Positive는 높아진다.

스캔 탐지 시스템은 패킷을 감사하기 위해 패킷을 캡쳐해야 한다. 대부분의 스캔 탐지 시스템은 Libpcap 라이브러리를 사용하여 패킷을 캡쳐하기 위한 루틴을 개발하고, 필터 룰에 의해서 캡쳐하는 패킷 종류를 제어할 수 있다. 필터 룰 형식의 정의는 다음과 같다. 예를 들어, "TCP[13] == 2"라는 필터 룰은 TCP Header의 13번째 바이트(Control)의 값이 2(SYN)이면 해당 패킷을 캡쳐 한다는 의미이다. 또한 "IP[2:2] == 20"이라는 필터 룰은 IP Header의 2번째 바이트부터 2바이트(Total length)의 값이 20(Header length + Data length = Total length = 20bytes)이면 해당 패킷을 캡쳐 한다는 의미이다. Destination Port Number가 80인 UDP 패킷을 캡쳐하기 위해서는 "UDP[0:2] == 80"이라는 필터 룰을 사용한다. 필터 룰은 스캔 탐지 시스템의 성능을 결정하는데 있어서 주요한 역할을 한다. 따라서 본 논문에서는 효율적인 필터 룰 집합을 제안한다.

3. TkRTSD

3.1 TkRTSD의 전체 구성도

그림 1은 본 논문에서 제안한 TkRTSD의 전체 구성도이다.

본 논문에서 제안한 TkRTSD의 동작은 다음과 같다. TkRTSD가 설치된 호스트의 네트워크에 지나다니는 모든 패킷들을 패킷 캡쳐 모듈에서 Libpcap을 이용하여 캡쳐 하고, 캡쳐된 패킷들은 스캔 탐지 모듈로 보내진다. 스캔 탐지 모듈에서는 기존의 실시간 스캔 탐지 시스템의 알고리즘을 사용하여, 패킷 캡쳐 모듈에서 보낸

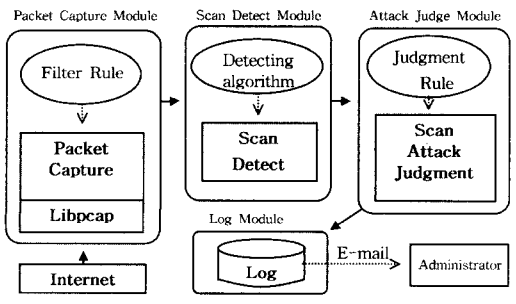


그림 1 TkRTSD의 전체 구성도

패킷들을 감사하여 실제로 스캔 공격일 가능성이 있는 패킷(이하 혐의 패킷(Suspicious Packet), 기존의 스캔 탐지 시스템에 의해 공격으로 판단된 패킷)을 탐지한다. 스캔 탐지 모듈에서 탐지된 혐의 패킷들은 본 논문에서 제안한 공격 판단 모듈로 보내진다. 공격 판단 모듈에서는 혐의 패킷이 공격인지 아닌지를 최종적으로 검사하게 된다. 공격 판단 모듈에서 공격으로 판단된 혐의 패킷은 로그 파일에 저장된다. 저장된 로그는 이메일을 통해 관리자에게 보내진다.

각 모듈에 대한 설명은 다음과 같다.

패킷 캡쳐 모듈에서는 본 논문에서 제안한 필터 룰을 사용하여 네트워크를 지나다니는 패킷들을 캡쳐 한다. 제안된 필터 룰은 시스템의 성능에 영향을 주지 않으면서, 기존의 스캔 탐지 시스템에 의해 탐지 될 수 없었던 스캔 공격을 탐지할 수 있다. 자세한 내용은 3.2에서 설명한다.

스캔 탐지 모듈은 기존의 실시간 스캔 탐지 시스템이 사용하는 알고리즘을 사용하여, 패킷 캡쳐 모듈에서 캡쳐한 패킷 중에서 혐의 패킷을 탐지해낸다. 즉, 하나의 IP로부터 일정한 시간 간격으로 동일한 시그너쳐를 갖는 패킷이 임계 값 이상 캡쳐 되었을 때, 캡쳐된 패킷은 혐의 패킷이 된다.

공격 판단 모듈에서는 본 논문에서 제안한 ABP-Rule(Attacker's Behavioral Pattern Rule)이 적용된다. ABP-Rule은 혐의 패킷이 실제 공격인지 아닌지를 판단하기 위하여 공격자의 행동 패턴을 고려한다. ABP-Rule의 적용에 의해 TkRTSD는 기존의 스캔 탐지 시스템의 높은 False Positive를 낮출 수 있다. 자세한 내용은 3.3에서 설명한다.

3.2 패킷 캡쳐 필터 룰

현재 다양한 스캔 공격 기법이 알려져 있고, 공격자는 알려져 있는 스캔 공격 기법을 이용하여 공격하고자 하는 목적 시스템의 취약점을 찾기 위해 포트 스캐닝을 한다. 따라서 스캔 탐지 시스템에서 다양한 스캔 공격 기법의 탐지 여부는 중요한 문제이다. 하지만 기존의 스캔 탐지 시스템에서는 해당 탐지 시스템이 공개 된 이후에 알려진 스캔 공격 기법에 대해서는 탐지가 불가능할 뿐만 아니라, 공개되기 이전에 알려진 스캔 공격 기법의 경우라도 해당 스캔 공격 기법에 대한 분석이 제대로 이루어지지 않아, 탐지를 위해 필요한 패킷을 캡쳐하지 못하기 때문에 탐지가 불가능하다. 하나의 예로, 정보보호진흥원에서 개발한 RTSD(Real Time Scan Detector)[10][11] 시스템은 스캔 공격을 탐지하기 위해서 다음과 같은 필터 룰을 사용한다.

• TCP[13] == 2

이 필터 룰은 네트워크에 지나다니는 패킷 중에서

TCP Header에 SYN flag가 설정된 패킷만 캡처한다. 하지만 이 필터 룰로 탐지해낼 수 있는 스캔 공격 기법은 SYN Scan, Connect Scan, RCP Scan 뿐이다.

기존의 스캔 탐지 시스템이 탐지하지 못하는 스캔 공격 기법을 탐지하기 위해 TkRTSD에서는 패킷 캡처 모듈을 위한 여러 가지 필터 룰을 제안한다. SYN Scan, Connect Scan, RCP Scan 등의 공격과 그 외의 다양한 스캔 공격을 탐지하기 위해 본 논문에서 제안한 필터 룰을 정리하면 다음과 같다.

- FIN Scan : TCP[13] & 1 == 1

이 필터 룰은 TCP Header에 FIN flag가 설정된 패킷을 캡처한다. FIN Scan이 TCP Header에 FIN flag를 설정하여 보내는 공격이기 때문에, 이 스캔 공격을 탐지하기 위해서는 TCP Header에 FIN flag가 설정된 패킷을 캡처해야 한다.

- NULL Scan : TCP[13] == 0

이 필터 룰은 TCP Header에 어떤 flag도 설정되지 않은 패킷을 캡처한다. NULL Scan이 TCP Header에 flag설정을 하지 않는 공격이기 때문에, 이 스캔 공격을 탐지하기 위해서는 TCP Header에 flag가 설정되지 않은 패킷을 캡처해야 한다.

- ACK Scan, Window Scan : TCP[13] & 16 == 16

이 필터 룰은 TCP Header에 ACK flag가 설정된 패킷을 캡처한다. ACK Scan과 Window Scan이 TCP Header에 ACK flag를 설정하여 보내는 공격이기 때문에, 이 스캔 공격을 탐지하기 위해서는 TCP Header에 ACK flag가 설정된 패킷을 캡처해야 한다.

- IP Protocol Scan : ip[1] == 0 and ip[0] == 69 and ip[2:2] == 20 and ip[6:2] == 0

이 필터 룰은 IP Header에 Service type의 8비트가 0, 버전과 IP Header 길이의 값이 69, IP 데이타 그램의 전체길이가 20, Flags와 Fragmentation offset 값이 0으로 설정된 패킷을 캡처한다. IP Protocol Scan 공격 시에 IP Header에서 위와 같은 패턴이 나타나기 때문에, 이 필터 룰로써 IP Protocol Scan 공격을 탐지할 수 있다.

- UDP Scan : ip[1] == 0 and ip[0] == 69 and ip[2:2] == 28 and ip[6:2] == 0 and udp

이 필터 룰은 IP Header에 Service type의 8비트가 0, 버전과 IP Header 길이의 값이 69, IP 데이타 그램의 전체길이가 28, Flags와 Fragmentation offset 값이 0인 UDP 패킷을 캡처한다. UDP Scan 공격 시에 IP Header에서 위와 같은 패턴이 나타나기 때문에, 이 필터 룰로써 UDP Scan 공격을 탐지할 수 있다.

위와 같은 필터 룰을 적용함으로써 현재 알려져 있는 다양한 스캔 공격을 탐지할 수 있다. 다음 표 2는 RTSD, Snort, PortSentry[10,11,12,13,15,16], TkRTSD의 스캔 공격에 대한 탐지 범위를 비교한 것이다. 실험을 위한 스캔 공격 툴로는 가장 잘 알려진 포트 스캐닝 도구인 Nmap을 사용하였다. Nmap은 표 2의 스캔 기법들을 제공한다. 실험 결과, TkRTSD가 RTSD, Snort, PortSentry보다 더 많은 스캔 기법을 탐지할 수 있었다. 또한 위의 효율적인 필터 룰로부터 알 수 있듯이, TkRTSD는 실험이 진행된 호스트의 성능에 영향을 주지 않았다.

표 2 탐지 범위의 비교

Technique	RTSD	Snort	PortSentry	TkRTSD
Connect Scan	O	O	O	O
SYN Stealth Scan	O	O	O	O
FIN Stealth Scan	X	O	O	O
NULL Scan	X	O	O	O
X-mas Scan	X	O	O	O
UDP Scan	X	X	O	O
IP Protocol Scan	X	X	X	O
ACK Scan	X	O	O	O
Window Scan	X	X	O	O
RCP Scan	O	O	O	O

3.3 공격 판단 ABP-Rule

스캔 탐지 시스템의 탐지 정확도를 떨어뜨리는 두 가지 요인은 False Positive와 False Negative이다. False Positive는 실제로 스캔 공격이 아닌데 스캔 탐지 시스템이 스캔 공격이라고 탐지하는 것을 말하고, Flase Negative는 실제로 스캔 공격이지만 스캔 탐지 시스템이 스캔 공격이 아니라고 탐지하는 것을 말한다. 따라서 스캔 탐지 시스템의 탐지 정확도를 높이기 위해서는 False Positive와 False Negative를 낮추어야 한다. 본 논문에서는 False Positive를 최소화하기 위하여 기존의 스캔 탐지 시스템에 공격 판단 모듈을 새롭게 제안하여 추가하였다. 또한 공격 판단 모듈에서 스캔 탐지 모듈로부터 얻어진 혐의 패킷이 실제 공격인지 아닌지를 판단하기 위하여 공격자의 행동 패턴으로부터 유도된 새로운 룰을 제안한다.

일반적으로 공격자는 스캔 공격 시에 다음과 같은 행동 패턴을 보인다.

(패턴 I) 공격자가 하나의 Host에 대해서 모두 다른 Port로 Scan을 한다.

(패턴 II) 공격자가 네트워크 전체에 대해서 하나의 Port로 Scan을 한다.

위의 패턴 I, II로부터 아래와 같은 ABP-Rule I과 ABP-Rule II를 유도하여, 혐의 패킷이 실제 공격인지 아닌지를 판단하기 위해서 공격 판단 모듈에 적용한다:

(ABP-Rule I) 혐의 패킷의 destination IP가 모두 같

을 경우 destination Port 번호가 모두 달라야 스캔 공격이다.

(ABP-Rule II) 혐의 패킷의 destination Port번호가 모두 같을 경우 destination IP가 모두 달라야 스캔 공격이다.

그림 2는 ABP-Rule I과 ABP-Rule II에 해당하는 공격자의 행동 패턴을 보여준다.

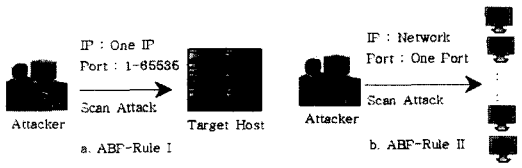


그림 2 공격자의 행동 패턴

기존의 스캔 탐지 시스템의 스캔 탐지 알고리즘에 의해 탐지된 혐의 패킷에는 ftp connect, web surfing, web crawling로부터 발생하는 많은 False Positive가 존재 한다. 이러한 False Positive의 대부분은 패턴 I, II 중 어느 하나에 속하게 된다. 따라서 ABP-Rule I과 ABP-Rule II을 적용함으로써 패턴 I, II에 속하는 False Positive는 필터링 된다. 즉, 기존의 스캔 탐지 시스템에서 나타나는 높은 False Positive를 낮출 수 있다. 본 논문에서 제안된 공격 판단 모듈은 위의 ABP-Rule I과 ABP-Rule II를 기반으로 구현되었다. 그러므로 TkRTSD는 다른 시스템보다 False Positive가 매우 낮아진다.

3.4 탐지 정확도 비교 실험

ABP-Rule I, II가 False Positive에 미치는 영향을 알아보기 위하여 실제 네트워크 상에서 TkRTSD와 기존 시스템과의 False Positive를 비교하는 실험을 하였다. 각 스캔 탐지 시스템은 내부에 20대의 컴퓨터가 있는 게이트웨이에 설치하였으며 20대의 컴퓨터에는 파일 서버(1대), 웹 서버(1대), 웹 크롤링 서버(2대)가 있다. 게이트웨이에 설치된 각 스캔 탐지 시스템들로부터 탐지된 로그들을 일별로 저장 하였다. 20일 동안 저장된 로그파일은 약 23MB(TkRTSD : 1.6MB, RTSD : 14.1 MB, Snort : 7.4MB, PortSentry : 20KB)였으며 탐지 시스템들이 하루 동안 기록한 로그의 크기는 RTSD가 약 500-900KB, Snort가 약 500KB, TkRTSD가 약 100KB, PortSentry가 약 1KB 였다. 각 스캔 탐지 시스템들의 로그를 분석하여 False Positive를 다음과 같이 계산하였다.

$$\text{False Positive} = \frac{\text{실제로 공격이 아닌 로그의 수}}{\text{스캔 탐지 시스템이 공격이라고 탐지한 로그의 수}}$$

여기서 실제로 공격이 아닌 로그는 내부호스트, DNS 서버, 신뢰호스트 등과 같이 공격이 아니라고 확신되는 것들이다. 다음은 실제로 각 스캔 탐지 시스템의 False Positive를 계산한 하나의 예이다.

$$\text{RTSD} = \frac{9400}{11024} = 85.26\%$$

$$\text{Snort} = \frac{587}{6931} = 8.46\%$$

$$\text{TkRTSD} = \frac{56}{1680} = 3.33\%$$

그러나 PortSentry같은 경우 로그의 크기에서도 알 수 있듯이 스캔 공격으로 탐지하는 로그가 극히 적었다. 즉, 하루 동안 탐지한 공격 IP가 대부분 2-3개 뿐 이었다. 이는 PortSentry에서 False Negative가 높게 나타났음을 말해준다. 따라서 각 스캔 탐지 시스템의 False Positive 비교에서 PortSentry는 제외하였다. 그림 3은 20일 동안 일별로 각 스캔 탐지 시스템의 False Positive를 비교한 것을 보여준다. 그림 3에서 TkRTSD가 RTSD와 Snort보다 False Positive가 낮음을 알 수 있다. RTSD의 평균 False Positive는 약 85%, Snort는 10%, TkRTSD는 3%를 기록하였다. RTSD같은 경우 웹 크롤링 서버의 IP를 대부분 스캔 공격으로 탐지함으로써 False Positive가 상대적으로 높게 나타났다.

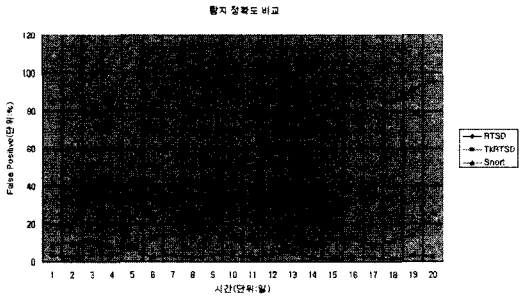


그림 3 탐지 정확도의 비교

4. TkRTSD의 GUI환경 구축

4.1 GUI 환경의 전체 구성도

네트워크 보안 관리에 있어서, 실시간 스캔 탐지 시스템의 사용자 친화적인 인터페이스는 중요한 문제이다. 왜냐하면 그러한 인터페이스는 효율적이고 편리한 보안 관리를 위해 중요한 역할을 하기 때문이다. 그러나 기존의 실시간 스캔 탐지 시스템은 명령어 기반으로 이루어져 있기 때문에 이를 활용하여 시스템 보안 관리를 수행하는데 많은 어려움이 있다. 그래서 본 논문에서는 Tcl/Tk를 이용하여 GUI환경의 TkRTSD를 구현한다. 그림 4는 TkRTSD의 전체 구성도를 보여준다. 시스템

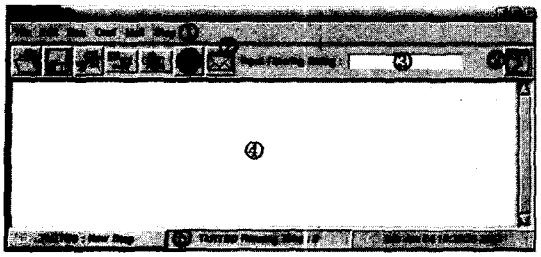


그림 4 TkRTSD의 실행 화면

보안 관리를 위한 모든 작업은 이 윈도우에서 수행된다.

그림 4에서와 같이 TkRTSD는 5부분으로 나누어지고 각 부분에 대한 설명은 다음과 같다.

① 메뉴바

② 툴바

③ 문자열을 입력하기 위한 TextField

④ 탐지된 스캔 공격의 정보를 출력하기위한 메인 윈도우

④ TkRTSD의 현재 실행 정보를 보여주는 상태 바

메뉴 바(①)는 TkRTSD를 제어하기 위한 모든 기능을 포함하고 그 중에서 자주 사용되는 기능을 툴 바(②)로 만들어 두었다. TextField(③)는 메인 윈도우(④)에 출력된 정보를 임의의 문자열로 필터링하기 위해 관리자가 문자열을 입력하는 입력창이다. 메인 윈도우(④)에서 관리자는 실시간으로 공격정보들을 볼 수 있을 뿐만 아니라 탐지된 공격정보들에 대한 편집 및 저장을 할 수 있다. 또한 이전에 탐지되었던 공격정보의 로그 파일, 설정파일 등의 불러오기가 가능하다. 상태 바(⑤)에는 TkRTSD의 실행 정보가 나타난다. 즉, TkRTSD의 실행시간, 캡쳐모드, 현재시간 등이 표시된다.

4.2 TkRTSD의 사용 방법

그림 5는 TkRTSD를 FIN Stealth Scan 공격(위 부분)을 탐지하기 위해, FIN 패킷 캡쳐 모드로 실행한 것이다. 또한 실행 정보들(아래 부분)이 표시되고 있음을 알 수 있다. 메뉴 바의 Run에는 FIN Stealth Scan 공격 외에도 SYN Scan, ACK Scan, NULL Scan, IP Scan, UDP Scan 등 다양한 스캔 공격을 탐지하기 위한 항목이 있다.

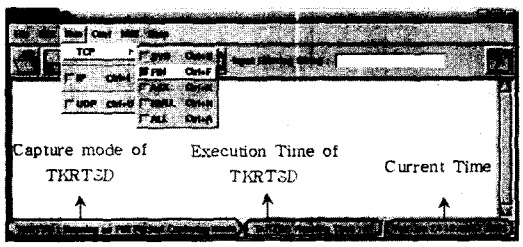


그림 5 FIN 패킷 캡쳐를 위한 TkRTSD의 실행

그림 6은 TkRTSD에 의해 203.253.145.143 호스트로부터 203.253.145.211 호스트로의 FIN 스캔 공격이 탐지되었음을 보여준다.

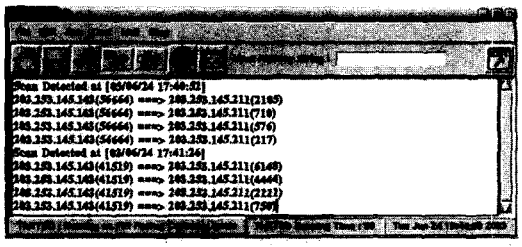


그림 6 TkRTSD의 FIN 스캔 공격 탐지

그림 7은 TkRTSD의 필터링 기능을 보여준다: TextField에 입력된 문자열 "56664"에 의해, "56664"라는 문자열을 가진 모든 정보가 메인 윈도우에 출력되었다.

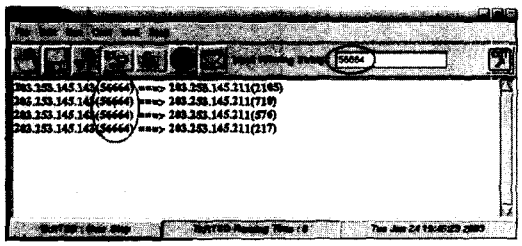


그림 7 TkRTSD의 필터링 기능

그림 8은 TkRTSD가 탐지한 공격 정보들을 메일로 관리자에게 송신할 것인지의 여부를 설정하는 화면이다. Mail -> Send Mail 체크박스가 ON이면 TkRTSD가 탐지한 공격 정보는 관리자에게 메일로 전송된다.

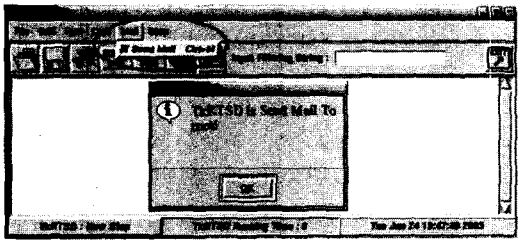


그림 8 TkRTSD의 메일 설정 화면

그림 9는 TkRTSD 프로그램의 설정화면이다. Conf -> rule.h를 선택하면 설정화면이 나타난다. 관리자는 스캔 탐지 모듈에서 사용되는 시간 간격과 임계 값을 이 rule.h 파일에서 지정할 수 있다. 또한 탐지의 정확도를 높이기 위해, 신뢰하는 네트워크, 호스트, 포트 등

에 대해서는 TkRTSD가 스캔 탐지를 하지 않도록 설정할 수 있다.

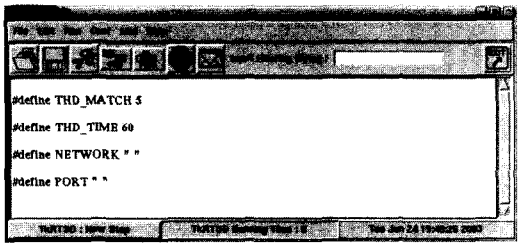


그림 9 TkRTSD의 설정 화면

그림 10과 11은 각각 불러오기와 저장하기 화면이다. TkRTSD가 탐지한 공격정보를 저장할 수 있으며, 저장되어 있는 이전의 공격정보에 대해서도 관리자가 필요할 때 불러올 수 있다.

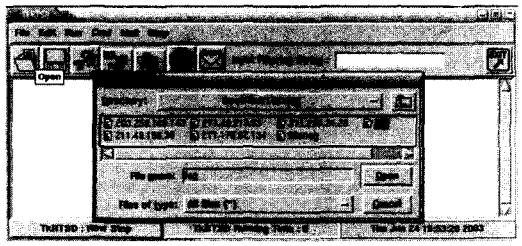


그림 10 TkRTSD의 불러오기

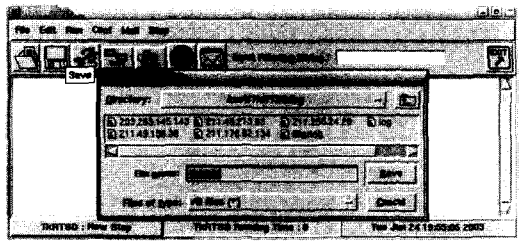


그림 11 TkRTSD의 저장하기

5. 결 론

본 논문에서는 기존의 실시간 스캔 탐지 시스템의 성능을 향상시키기 위하여 다음과 같은 두가지 수법을 제안하고 이를 기반으로 실시간 스캔 탐지 시스템 TkRTSD를 구현했다. 첫 번째 수법은 효율적인 필터 룰 집합을 패킷 캡쳐 모듈에 적용함으로써 기존의 스캔 탐지 시스템이 탐지할 수 없었던 다양한 스캔 공격의 탐지가 가능하게 하였고, 두 번째 수법은 공격자의 행동 패턴으로부터 유도된 ABP-Rule을 공격 판단 모듈에 적용함으로써 기존의 실시간 스캔 탐지 시스템의 높은 False Positive를 낮추었다. 또한 TkRTSD는 사용자가 네트워크 보안 관리를 쉽게 할 수 있도록 GUI 환경을 지원한다.

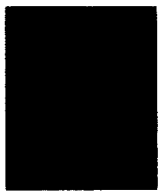
향후의 실시간 스캔 탐지 시스템에서는 높은 False Negative를 낮추기 위한 연구와 탐지율의 정확성을 더욱 높이기 위한 공격자의 행동 패턴에 대한 연구가 필요하다.

참 고 문 헌

[1] http://security.xmecca.com/AnalyzingNmap.pdf
[2] Martin freiss, "Protecting Newtorks with SATAN," *O'REILLY*, 1998.
[3] http://www.nessus.org/
[4] Kim Sang-Jung, "An analysis report of the Mscan," Technical Report of the Korea Information Security Agency(KISA), 1998.
[5] Chung Hyun-Chul, "An analysis report of the Sscan," Technical Report of KISA, 1999.
[6] J. K. Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley Professional Computing Series (1994).
[7] Guo Xiaobing, Qian Depei, Liu Min, Zhang Ran, Xu Bin, "Detection and Protection against Network Scanning: IEDP," Proc. of the 2001 IEEE International Conference on Computer Networks and Mobile Computing, pp. 487-493, 16-19 Oct. 2001.
[8] Fyodor, "The Art of Port Scanning" Phrack Magazine Volume 7, Issue 51, article 11 of 17, 1997.
[9] Park Hyun-Mi, Oh Eun-Suk, Lee Dong-Ryun, "Technique of IP Network Scanning," Technical Report of KISA, 2002.
[10] Lee Hyun-Woo, Lee Sang-Yeop, Chung Hyun-Chul, Chung Yoon-Jong, Lim Chae-Ho, "Pattern analysis and detection tools against scan attack to network vulnerability," WISC, 1999.
[11] http://www.ccrtcc.or.kr/
[12] Martin Roesch, "Snort-Lightweight Intrusion Detection for Networks," Proc. of LISA '99: 13th Systems Administration Conference, Seattle, Washington, USA, November 7-12, 1999.
[13] http://www.snort.org/
[14] Hoagland, J.A, Staniford, S, "Viewing IDS alerts: lessons from SnortSnarf," Proc. of the 2001 IEEE DISCEX '01 on DARPA Information Survivability Conference & Exposition II, pp. 374-386, 12-14 June. 2001.
[15] David Sarmanian, "Deploying PortSentry-A Simple and Free Barrier From Inside Hackers," http://www.giac.org/practical/gsec/David_Sarmanian_GSEC.pdf.

[16] http://www.psionic.com/
[17] S. McCanne, C. Leres and V. Jacobson, libpcap, available via anonymous ftp to ftp.ee.lbl.gov, 1994.

송 중 석
2003년 2월 한국항공대학교 항공통신정보공학과 졸업. 2003년 3월~현재 한국항공대학교 대학원 석사과정 재학중. 관심분야는 정보보호, 암호이론



권 용 진
1986년 2월 한국항공대학교 항공전자공학과 졸업. 1990년 3월 일본교토대학 정보공학과 대학원 졸업(공학석사). 1994년 3월 일본교토대학 정보공학과 대학원 졸업(공학박사). 1994년 3월~현재 한국항공대학교 전자·정보통신·컴퓨터공학부 부교수. 관심분야는 정보보호, 놀리설계 및 합성, 정보검색