Port Scanning 기법 기반의 공격을 탐지하기 위한 실시간 스캔 탐지 시스템 구현

송중석° 권용진
한국항공대학교 정보통신공학과
{oaktree°, yjkwon}@tikwon.hangkong.ac.kr

Implementation of Real Time Scan Detector System against Attacks of Applying on Port Scanning Techniques

Joongsuck Song° Yongjin Kwon
Dept. of Information. and Telecomm. Eng., Hankuk Aviation University

요 약

현재 네트워크 보안 취약점을 자동으로 검색해주는 다양한 도구들이 인터넷에 공개되어 있어 이러한 도구들을 이용한 취약점 정보수집 및 네트워크 검색공격으로 비롯된 해킹사고가 크게 증가하고 있다. 이와 같은 검색 공격에 대한 탐지 시스템은 "False-Positive(실제 공격이 아닌데 공격이라고 탐지)"와 "False-Negative(실제 공격인데 공격이 아니라고 탐지)"를 줄이는 것이 중요하다. 그러나 현재 공개되어 있는 실시간 스캔 탐지 시스템은 오탐율이 높을 뿐만 아니라 다양한 스캔 기법에 대해서 탐지를 할 수 없는 것이 사실이다. 본 논문에서는 다양한 포트스캐닝 기법기반의 공격에 대해서 탐지 가능하고 오탐율을 최소화한 실시간 스캔 탐지 시스템을 구현한다.

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, Nmap[1], Nessus 등과 같은 네트워크 보안 취약점을 검색해주는 보안 관리 도구들이 공개되었다. 그러나 시스템 관리자가 아닌 해커들은 이러한 보안 관리 도구들을 이용하여 침입하고자 하는 시스템의 보안 취약점 정보 및 공격대상을 찾는데 활용하고 있다.

포트 스캐닝은 서비스거부공격(DOS), 버퍼오버플로우공격, 포맷스트링공격 등과 같이 시스템에 직접적인 피해를 주지는 않지만 침입하고자 하는 시스템의 취약점 정보를 수집하는 첫 단계이다. 누군가 자신의 시스템에 침입 하고자 한다는 것을 미리 안다면 해킹을 막는데 유용할 것이다. 따라서 이러한 포트 스캐닝에 대한 정확한 탐지는 중요한 문제이다.

포트 스캐닝에 대한 탐지는 "False-Positive(실제 공격이 아닌데 공격이라고 탐지)"와 "False-Negative(실제 공격인데 공격이 아니라고 탐지)"를 낮추는 것이 중요하다. 그러나 현재 공개되어 있는 실시간 스캔 탐지 시스템은 Open Scan, Half-Open Scan, Stealth Scan 등 다양한 스캔 기법에 대해서 탐지를 할 수 없는 것이 사실이고 오탐율 또한 높다.

현재 공개되어 있는 실시간 스캔 탐지 시스템의 이러한 문제점을 보완하기 위해서는 다양한 스캔 기법에 대한 분석을 하고 그에 맞는 탐지를 해야하며 또 해커가 포트 스캐닝을 할 때의 패턴을 분석하여 더욱 정확한 탐지가 가능하도록 해야 한다. 본 논문에서는 다양한 포트스캐닝 기법기반의 공격에 대해서 탐지 가능하고 오탐율을 최소화한 실시간 스캔 탐지 시스템을 구현한다.

* 본 논문은 과학기술부·한국과학재단지정 「한국항공대학교 인터넷정보검색연구센터」의 연구비 지원으로 수행되었음.

본 논문의 구성은 다음과 같다. 2장에서는 실시간 스캔 탐지 시스템에서 패킷을 캡쳐할 때 사용할 Libpcap와 탐지하기 위한 포트 스캐닝 기법에서 대해서 설명하고 3장에서 본 논문에서 구현한 실시간 스캔 탐지 시스템의 전체 구성도, 다양한 스캔 기법기반의 공격들을 탐지하기 위한 방법, 오탐율을 최소화하기 위한 알고리즘을 소개한다. 마지막으로 4장에서 결론을 제시한다.

2. 포트 스캐닝 기법과 Libpcap

2.1 포트 스캐닝 기법

현재 알려진 포트 스캐닝 기법의 종류[2][3][4]는 다음 <표 1>과 같다.

<표 2> 포트 스캐닝 기법의 종류

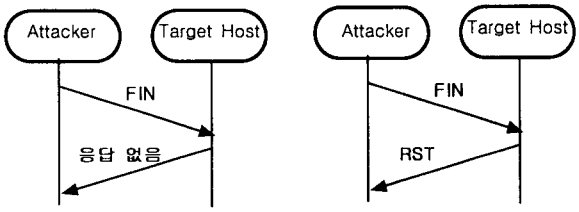
기법	종류
Open Scan	TCP Connect, Ident Scan
Half-Scan	SYN flag
Stealth Scan	FIN flag, NULL flag, XMAS flag, ACK flag, Window Scan, TCP Fragment

TCP Connect Scan은 TCP 스캐닝 중에서 가장 기본적이고 간단한 Port Scan 방법으로 TCP/IP 3-way handshake 연결을 확립한다. O/S가 제공하는 connect() 시스템 콜을 Target Host의 각기 잘 알려진 포트로 접속하기 위해서 사용한다. 공격자는 Target Host의 해당 포트가 열려 있으면 SYN/ACK 패킷을 받고 닫혀 있는 경우에는 RST/ACK 패킷을 받게 된다.

Ident Scan은 서버와 클라이언트 사이에 연결이 되어 있는 상태일 경우 ident 프로토콜이 TCP 방식으로 접속된 그 프로세스의 실행권자 username를 공개시켜 버리는 것을 이용한 Scan 기법이다.

SYN flag Scan은 TCP/IP 연결을 완전히 성립하지 않아 로깅을 교묘하게 피할 수 있는 Scan 기법이다. TCP Connect 기법과는 달리 이 방법에서는 Target Host로부터 SYN/ACK 패킷을 받은 경우 해당 포트가 열려 있는 것만 확인하고 연결을 확립하기 위한 마지막 ACK 패킷 대신에 RST 패킷을 보냄으로써 완전한 연결을 성립하지 않아 로그가 남지 않게 된다. 공격자는 Target Host의 해당 포트가 열려 있으면 SNY/ACK 패킷을 받고 닫혀 있는 경우에는 RST/ACK 패킷을 받게 된다.

Stealth Scan은 Firewall이나 IDS와 같은 보안 시스템들과 필터링들이 중요한 포트에 대한 SYN 패킷을 가지고 검사를 하기 때문에 보안시스템을 통과하기 위해서 패킷에 다른 flag를 설정한다. 한 예로 FIN flag Scan은 TCP 운용에서 하나의 bug를 이용한 것으로 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. FIN flag Scan의 패킷 순서는 (그림 1)과 같다.



a. 포트가 열려 있을 경우 b. 포트가 닫혀 있을 경우

(그림 1) FIN flag Scan의 패킷 순서

2.2 Libpcap

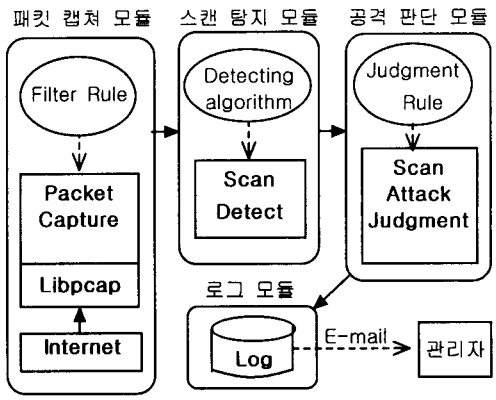
패킷을 캡쳐하기 위한 도구로는 BPF(Berkeley Packet Filter), DLPI, NIT, SNOOP, SNIT, SOCK_PACKET, LSF(Linux Socket Filter), drain등 각 운영체제별로 다양한 도구가 있다. 그러나 이 모든 도구들을 수용하는 Portable한 API가 있는데 이것이 바로 Libpcap[5][6]이다. 라이브러리 사용자는 운영체제의 각기 다른 datalink로의 접근 방법에 상관없이 libpcap을 이용하여 소기의 목적을 달성할 수 있다. libpcap을 이용한 대표적인 tool이 바로 tcpdump이다. 본 논문에서 구현한 실시간 스캔 탐지 시스템도 Libpcap를 이용하여 패킷을 캡쳐하고 Filter Rule에 의해서 캡쳐하기 원하는 패킷만을 캡쳐한다. 패킷 캡쳐를 위해 사용한 함수들은 다음과 같다.

- ○ pcap_lookupdev() 함수는 리눅스 머신의 네트웍 디바이스를 가져온다.
- ○ pcap_open_live() 함수는 실제 기기를 열어주는 기능을 한다.
- ○ pcap_lookupnet() 함수는 열려진 패킷 캡쳐 디바이스에 네트웍 주소와 서브넷 마스크를 넘겨준다.
- ○ pcap_compile() 함수는 정해진 필터 룰에 의해 필터 프로그램을 컴파일 한다.
- ○ pcap_setfilter() 함수는 컴파일 한 필터 프로그램을 패킷 캡쳐 디바이스로 읽어들인다.
- ○ pcap_loop() 함수는 실제 패킷을 잡아서 실행할 함수를 지정해 준다.

3. 실시간 스캔 탐지 시스템 구현

3.1 실시간 스캔 탐지 시스템의 전체 구성도

(그림 2)는 실시간 스캔 탐지 시스템의 전체 구성도를 보여준다.



(그림 2) 실시간 스캔 탐지 시스템의 전체 구성도

네트워크에 지나다니는 모든 패킷들은 Libpcap을 이용하여 캡쳐되고 캡쳐된 패킷들은 실시간 스캔 탐지 시스템을 통과하게 된다. 스캔 탐지 모듈에서는 일반적인 실시간 스캔 탐지 시스템들의 알고리즘을 사용하여 탐지한다. 스캔 탐지 모듈에서 탐지된 정보는 본 논문에서 추가한 공격 판단 모듈을 거치게 된다. 공격 판단 모듈에서는 스캔 탐지 모듈에서 넘어온 정보에 Rule을 적용하여 스캔 공격 여부를 다시 판단하게 되고 스캔 공격이라고 판단되면 이를 로그로 저장하게 된다. 저장된 로그는 E-mail을 통해 관리자에게 보내진다.

3.2 다양한 스캔 기법에 대한 탐지

패킷 캡쳐 모듈에서 다양한 스캔 기법에 대한 탐지를 위해 Filter Rule을 적용하게 된다. 일반적인 스캔 탐지 시스템들은 SYN Scan 공격에 대한 탐지를 하기 위해서는 다음과 같은 Filter Rule만을 적용한다.

"TCP[13] & 2 == 2"

위 Filter Rule은 네트워크에 지나다니는 패킷 중에서 TCP Header에 SYN flag만 설정된 패킷을 캡쳐한다. 또한 이 Filter Rule은 Connect Scan과 RPC Scan도 탐지해낸다.

SYN Scan, Connect Scan, RPC Scan 공격 외의 다른 스캔 공격에 대한 탐지를 위해 본 논문에서 사용한 Filter Rule들은 다음과 같다.

- ○ FIN Scan : "TCP[13] & 1 == 1"
- ○ NULL Scan : "TCP[13] == 0"
- ○ ACK Scan, Window Scan : "TCP[13] & 16 == 16"
- ○ IP Protocol Scan : "ip[1] == 0 and ip[0] == 69 and ip[2:2] == 20 and ip[6:2] == 0"
- ○ UDP Scan : "ip[1] == 0 and ip[0] == 69 and ip[2:2] == 28 and ip[6:2] == 0 and udp"

이러한 다양한 Filter Rule의 적용에 의해 현재 사용되고 있는 다양한 스캔 공격에 대한 탐지가 가능하게 된다. 다음 <표 2>는 정보보호진흥원에서 개발한 RTSD(Real Time Scan Detector)[7][8]와 스캔 공격에 대한 탐지 범위를 비교한 것이다. 실험을 위한 스캔 공격 Tool로 Nmap을 사용하였다.

<표 2> 탐지범위의 비교

	RTSD	본 논문의 시스템
Connect Scan	O	O
SYN Stealth Scan	O	O
FIN Stealth Scan	X	O
NULL Scan	X	O
X-mas Scan	X	O
UDP Scan	X	O
IP Protocol Scan	X	O
ACK Scan	X	O
Window Scan	X	O
RPC Scan	O	O

3.3 정확한 스캔 공격 탐지

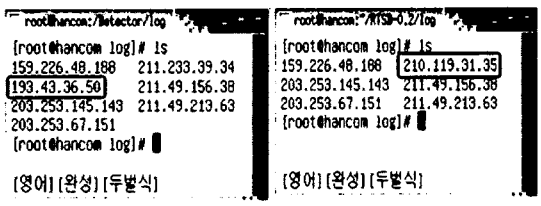
일반적인 실시간 스캔 탐지 시스템들의 알고리즘을 사용한 스캔 탐지 모듈에서의 탐지 정보는 정상적인 ftp 접속이나 web surfing을 스캔 공격으로 탐지할 수 있기 때문에 정확성이 떨어진다. 이러한 오탐지를 최소화하기 위해 본 논문에서는 공격 판단 모듈이 추가되었다. 공격판단 모듈에서는 스캔 공격을 하는 공격자가 스캔 공격 시에 나타나는 다음과 같은 행동 패턴의 특성을 이용하여 공격 여부를 판단한다.

① 하나의 Host에 대해서 Scan을 하면 모두 다른 Port에 대해서 Scan을 한다.
② 네트워크 전체에 대해서 Scan을 하면 하나의 Port에 대해서 Scan을 한다.

위 두 개의 행동 패턴은 공격 판단 모듈에서 각각 다음과 같은 Rule로써 적용되어 스캔 공격 여부를 판단한다.

①′ Destination IP가 모두 같을 경우 Destination Port 번호가 모두 달라야 스캔 공격이다.
②′ Destination Port번호가 모두 같을 경우 Destination IP가 모두 달라야 스캔 공격이다.

(그림 3)은 본 논문의 시스템과 정보보호진흥원에서 개발한 RTSD 시스템과의 스캔 공격에 대한 탐지의 정확도를 비교해 준다. RTSD에서는 210.119.31.35 호스트로부터 스캔 공격이 들어 왔다고 탐지했으나 이것은 실제로 210.119.31.35 호스트에서 정상적인 Web Crawling 작업중인 것을 스캔 공격이라고 잘못 탐지 한 것이다. 반면 본 논문의 시스템에서는 Web Crawling을 스캔 공격으로 탐지하지 않고 있으며 또한 RTSD에서 탐지하지 못한 193.43.36.50 호스트로부터의 스캔 공격을 탐지하였다.

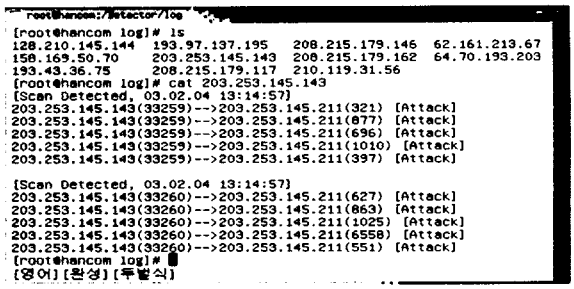


a. 본 논문의 시스템 b. RTSD

(그림 3) 본 논문의 시스템과 RTSD와의 정확성 비교

3.4 공격 로그의 기록

공격 판단 모듈이 공격이라고 판단한 네트워크 공격 정보들은 로그 모듈에서 파일 형태로 기록되며 관리자가 원할 경우 공격정보들은 관리자에게 메일로 보내지게 된다. 로그 모듈에서는 관리자가 공격정보들을 쉽게 관리하게 하기 위하여 공격을 시도한 Host IP를 파일 이름으로 생성하게 된다. (그림 4)는 본 논문에서 구현한 실시간 스캔 탐지 시스템에서 스캔 공격을 탐지한 것이다.



(그림 4) 실시간 스캔 탐지 시스템의 스캔 공격 탐지

(그림 4)에서 공격을 감행한 Host의 IP주소로 로그가 남고 203.253.145.143 Host의 33259번 Port로부터 203.253.145.211 Host로 스캔 공격이 이루어 졌다.

4. 결론 및 향후 연구과제

스캔 공격 기법들이 침입 차단 시스템이나 침입 탐지 시스템을 우회하기 위하여 더욱 다양해지고 있다. 이러한 스캔 공격에 대해서 빠르고 정확하게 대응할 수 있는 탐지 시스템이 요구되고 있다. 기존의 실시간 스캔 탐지 시스템들은 다양한 스캔 기법들에 대해 탐지를 할 수 없고 오탐율 또한 높게 나타난다.

본 논문에서는 기존의 실시간 스캔 탐지 시스템들이 갖고 있는 한계점을 보완하기 위해 다양한 스캔 기법들에 대해 분석하고 패킷 캡쳐 모듈에서 그러한 스캔 기법들에 대한 탐지가 가능하도록 패킷들을 캡쳐하였다. 또한 공격자들이 스캔 공격 시에 나타내는 행동 패턴을 분석하여 Rule화 하였고 그 Rule을 공격 판단 모듈에 적용함으로써 스캔 공격에 대한 정확한 탐지가 가능하도록 하였다.

차후의 실시간 스캔 탐지 시스템에서는 본 논문에서 탐지하지 못하는 스캔 공격기법들의 분석을 통해 해당 공격기법에 대한 탐지가 가능하도록 해야하고 스캔 공격자의 행동패턴에 대한 많은 연구를 통해 탐지의 정확성을 더욱 높이는 것이다. 또한 관리자가 쉽게 사용할 수 있는 GUI 환경 구축도 필요하다.

참고문헌

[1] 유성철, "Analyzing NMAP", 2002.
[2] 정계욱, 홍상국, 유성철, "The Art of Port Scanning", 1997.
[3] Fyodor, "The Art of Port Scanning" Phrack Magazine Volume 7, Issue 51, article 11 of 17, 1997.
[4] 박현미, 오은숙, 이동련, "IP 네트워크 Scanning 기법", 2002.
[5] 노광민, "리눅스에서 pcap library를 사용하여 패킷을 잡아보기 v0.3", 2002.
[6] 유정각, "Packet Capture using libpcap", 2001.
[7] 이현우, "RTSD(Real Time Scan Detector) V.0.1", 1999.
[8] 이현우, 이상엽, 정현철, 정윤종, 임채호, "대규모 네트워크 취약점 검색공격 패턴분석 및 탐지도구", WISC, 1999