Proceedings of KFIS Fall Conference, 2003. 5. 3

TCP 프로토콜을 사용하는 서비스 거부 공격 탐지를 위한 침입시도 방지 모델

An Intrusion Prevention Model for Detection of Denial of Service attack on TCP Protocol

이세열, 김용수
대전대학교 컴퓨터공학부

Se-Yul Lee, Yong-Soo Kim
Division of Computer Engineering, Daejeon University
E-mail : ailab@dju.ac.kr, kystj@dju.ac.kr

요 약

해킹을 방지하기 위한 목적으로 개발된 보안 도구들 중 네트워크 취약점을 검색할 수 있도록 만들어진 프로그램들이 있다. 네트워크 취약점을 자동 검색해 주는 보안 관리 도구를 역이용하여 침입하고자 하는 시스템의 보안 취약점 정보를 알아내는데 사용하여, 알아낸 정보들을 가지고 공격 대상을 찾는데 활용하고 있다. 해킹 수법들에는 서비스 거부 공격, 버퍼오버플로우 공격 등이 있다. 따라서, 해커들이 침입하기 위하여 취약점을 알아내려고 의도하는 침입시도들을 탐지하여 침입이 일어나는 것을 사전에 방어할 수 있는 침입시도탐지가 적극적인 예방 차원에서 더욱 필요하다. 본 논문에서는 이러한 취약점을 이용하여 침입시도를 하는 사전 공격형태인 서비스 거부 공격 중 TCP 프로토콜을 사용하는 Syn Flooding 공격에 대하여 패킷분석을 통하여 탐지하고 탐지된 경우 실제 침입의 위험수준을 고려하여 시스템관리자가 대처하는 방어수준을 적절히 조절하여 침입의 위험수준에 따른 방어대책이 가능한 침입시도 방지 모델을 제시한다.

1. 서론

최근 네트워크 기술 발전으로 인하여 사회 전반에 걸쳐 인터넷 활용 의존성이 매우 높아지고 있다. 이러한 네트워크 기술 발전의 반대급부로 악의적 목적을 둔 침입을 위한 서비스 거부 공격이 점차 늘어나는 추세이다. 여기서 서비스 거부 공격이란 일반적으로 시스템의 자원을 고갈 또는 마비시켜 서비스제공을 방해하는 일련의 침입시도라고 볼 수 있다. 서비스 거부 공격 중 가장 대표적인 서비스 거부 공격으로는 최근 몇 년 동안 지속적인 형태로 나타나는 Syn Flooding 공격을 둘 수 있다. Syn Flooding 공격은 TCP/UDP 프로토콜 네트워크 환경에서 모두 가능하다. TCP 프로토콜이 신뢰성 연결 지향적 전송서비스라는 점에서 특히 TCP Syn Flooding 공격이 많이 사용되고 있으며 인터넷환경에서 많이 사용되어지는 TCP기반의 프로토콜 서비스를 지원하는 시스템에 크게 영향을 미치고 있다. 서비스 거부 공격은 크게 주요 파일을 훼손시켜 대상 시스템의 동작을 방해하는 우회적 서비스 거부 공격과 대상 시스템의 자원 및 네트워크 데이터전송을 위한 흐름제어 자원을 고갈시키는 공격으로 나눌 수 있다[1].

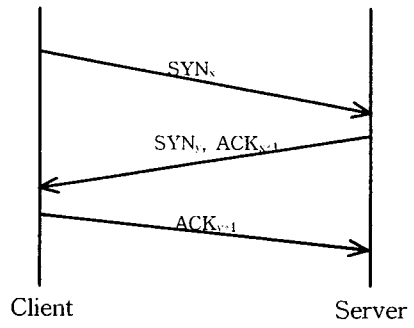
현재 이를 해결하기 위한 여러 대안이 연구되어 지고 있으나 TCP/IP의 설계상의 문제점에 기인한 것이므로 새로운 프로토콜을 사용하지 않는 한 완벽한 보호 대책은 존재할 수 없다. 이에 본 논문에서는 모니터링과 보안관리로 최소한의 피해를 막을 수 있는 방법을 제시한다. 본 논문의 구성은 제2장에서 서비스 거부 공격에 대해서 살펴보고 피해를 줄이기 위한 방안을 알아본다. 제3장에서는 이러한 방안 중 TCP프로토콜의

3-Way Handshake 연결과정에서 발생하는 Half Open State를 실시간으로 탐지하여 퍼지 인식도(FCM : Fuzzy Cognitive Maps)를 적용하여 침입정도의 시스템 위험도를 결정하는 탐지위험도 판단모듈을 제안하며 마지막장에서 향후 연구방향과 결론을 제시한다. 참고로 UDP Syn Flooding 공격은 다루지 않는다.

2. Syn Flooding 공격

2.1 TCP Syn Flooding 공격

TCP Syn Flooding 공격은 앞에서 거론되었듯이 TCP의 취약점을 이용한 공격형태이다. 일반적으로 TCP는 신뢰성 지향적 연결서비스이며 서버와 클라이언트간에 연결 설정에는 그림 1과 같은 '3-Way Handshake'라는 정상적 연결 흐름이 이루어진다.



[그림1] 3-Way Handshake

여기서 클라이언트가 SYN_x를 요청하고 서버로부터 SYN_y와 ACK_{x+1}을 받은 후 ACK_{y+1}을 보내지 않으면 서버에서는 클라이언트로부터 응답이 올 것을 기대하고 반쯤 열린 'Half Open State'가 된다. 물론 얼마간 이런 상태가 유지된 후 다음 요청이 오지 않으면 해당 연결을 reset하게 된다. 이때 reset되기 전까지 메모리에는 backlog queue가 계속 쌓이게 되는데 이러한 reset이 되기 전에 지속적으로 이와 같은 요청이 아주 빠르게 이루어진다면 Syn Packet은 backlog queue에 쌓이게 되어 결국 메모리 용량을 넘어서게 되면 해당 포드에 대한 연결을 받아들일 수 없는 상태인 서비스 거부 상태가 된다.

2.2 해결 방안

2.2.1 Backlog queue 와 Half Open Time

실제 서비스 거부가 발생하는 원인으로 backlog queue에 더 이상 받아들일 수 있는 조건이 되지 않기 때문이다. 이는 원천적으로 해결방안이라고는 할 수 없지만 공격에 대하여 어느 정도 경감시킬 수 있는 해결방안으로써 backlog queue 크기를 증가시켜주는 것과 Half Open State의 대기시간을 줄이는 방법을 적용할 수 있다. 그러나 H/W 및 OS마다 서로 다른 메모리 용량과 backlog queue 크기가 할당되어 있어 정확한 크기증가 선정이 어려워진다. 다음은 리눅스 시스템에서 설정한 예시이다.

```
# cat/proc/sys/net/ipv4/tcp_keepalive_time-->7200
# cat/proc/sys/net/ipv4/tcp_keepalive_probes-->9
# cat/proc/sys/net/ipv4/tcp_max_ka_probes-->5
```

위와 같이 설정을 하고 지속적인 공격 테스트를 해 본 결과 공격이 이루어지는 순간동안 아주 짧은 순간이나마 시스템이 다운되는 현상이 주기적 반복형태를 띄고 있는 결과를 나타내었다. 그리하여 추가적인 해결을 위하여 다음과 같이 추가조치를 하였다. 바로, tcp_max_syn_backlog와 syncookies의 수치를 조절하는 것이다.
/sbin/sysctl -w net.ipv4.tcp_max_syn_backlog의 기본값인 256을 1280으로 설정하는 것이다.

```
/sbin/sysctl -w net.ipv4.tcp_max_syn_backlog=1280
```

위와 같이 socket queue의 크기를 높여주는 방법이다. 그러나, 이러한 대안은 지속적인 공격측면에서 볼 때 효율적이지 못하므로 적절한 대안이라 할 수 없다.

2.2.2 Syncookies

syncookies에는 크게 Berkeley, Linux, Reset cookie가 있으며 '3-Way Handshake'에서 TCP 헤더의 Syn's sequence number, 소스 및 목적주소에 단방향 해쉬함수를 적용한 암호화 알고리즘을 이용한 방식으로 연결 설정이 정상적으로 이루어지지 않으면 더 이상 소스 경로를 따라 가지 않고 정상적 연결 요청에 대해서만 연결 설정을 하여 자원의 낭비를 줄이는 방법이다[2]. 아래는 리눅스 시스템에서 공격 테스트를 위한 syncookies 설정값으로써 공격시 시스템 다운 현상을 어느 정도 차단효과를 볼 수 있다.

```
/sbin/sysctl -w net.ipv4.tcp_syn_cookies=1
```

2.2.3 Packet Monitoring

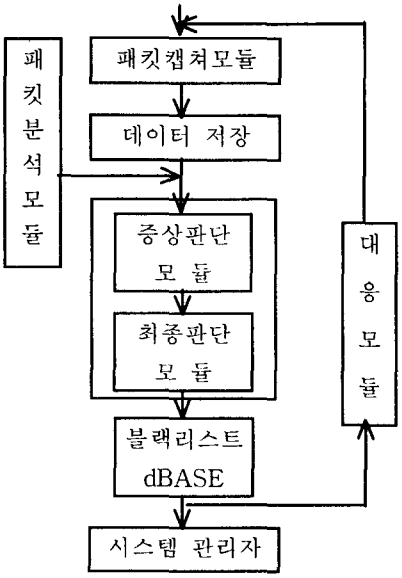
라우터 및 게이트웨이를 통과한 후 시스템 접근에 앞서서 모니터링을 하는 방법으로써 들어오는 패킷을 잡아 분석하여 'Half Open State'를

요청하는 포트 및 IP Address를 탐지하여 Reset 등으로 연결 해제하는 방법이다. 본 논문에서는 제안하는 모니터링을 통한 탐지 또한 이 범주에 속한다[3].

3. 모델 제안

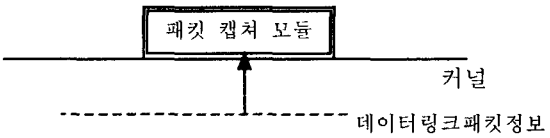
3.1 모델 구조

본 논문에서 제안하는 모델구조는 그림2와 같이 여러 모듈로 이루어져 있다



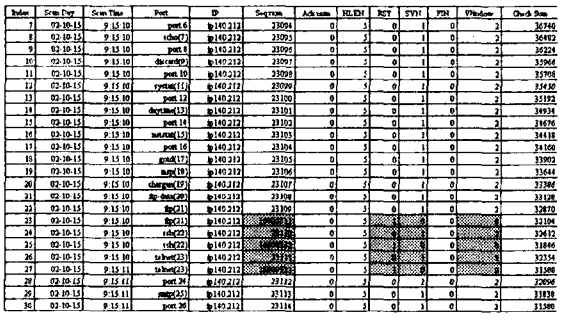
[그림 2] 모델 구조

전체적인 모델구조는 들어오는 패킷을 이용하여 패킷을 분석하고 제어하는 모듈과 데이터저장 후 'Half Open State'를 판단하는 판단모듈로 되어 있으며 추가로 대응모듈이 있다. 여기서 패킷캡쳐모듈은 그림 3과 같이 promiscuous mode에서 데이터링크층의 패킷을 캡쳐한 소프트웨어 방식이며 이를 프로토콜별 패킷을 받아서 세션별로 저장한다. 여기서 세션이란 Source IP와 Destination IP 그리고 프로토콜 종류가 같은 것들끼리 모아서 저장 DB에 저장하는 동시에 패킷들을 판단 모듈로 보낸다. 패킷을 파싱(parsing)하여 로컬포트, IP Address, Sequence Number, 윈도우 크기 및 공격시간 등으로 저장시키고 패킷분석모듈을 통하여 Syn패킷과 정상패킷으로 구분하여 1차 Half Open State를 탐지하게 된다. 여기서 탐지된 IP Address는 퍼지인식도를 이용한 판단모듈을 통하여 블랙리스트 데이터베이스에 저장되고 시스템관리자에게 통보 하게된다. 이런 일련의 과정을 통하여 재차 공격시에는 블랙리스트 dBASE와 비교하여 공격을 탐지하고 대응모듈을 가동하게 된다.



[그림 3] 패킷 캡쳐 모듈

그림 4는 데이터저장모듈에 저장된 탐지로그 항목이다. 여기서 'SYN'과 'RST'는 연결제어와 관련된 TCP 헤더의 Flags부분이다. 각각의 수치가 '0 -> 1' 또는 '1 -> 0' 변경된 영역은 '3-Way Handshake'에서 클라이언트에서 3번째 연결설정부분을 수행하지 않은 경우이며 서버에서는 'backlog queue'상태가 된다. 즉, 'Half Open State'이다. 그림 4에서는 3회 연속으로 'Half Open State'가 된 경우이다. 또한, TCP의 순서적 전달인 'Sequence Number' 가 갑작스럽게 순서적으로 전달되지 못하고 비순서적인 'Sequence Number'로 도착하였으며 이는 외부의 비정상적인 연결설정이 있음을 의미한다. 아울러 흐름제어영역인 'Window'에서는 크기가 순간적으로 변경된 것을 알 수 있는데, 이 역시 'Half Open State' 상태에 나타나는 특징들이다. 바로 이러한 항목들의 반복적 패턴을 감시하면 실시간으로 'Half Open State'를 탐지 할 수 있다.

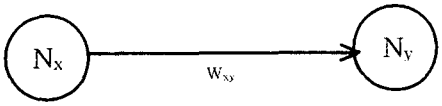


[그림 4] 탐지로그항목

3.2 판단 모듈

판단 모듈은 퍼지 인식도(FCM)의 Causal knowledge reason을 이용하여 지능적 판단모듈 구조를 설계하였다. FCM은 주어진 문제영역내의 각 개념들 사이에 존재하는 인과관계(Cause-effect relationship)를 나타내는 유향성 그래프(Directed graph)이다. 그림 5는 퍼지 인식도를 표현한 것으로써 각 노드와 노드사이의 가중치(링크)가 W_{ij}=0인 경우에는 각 노드사이에는

아무런 관련이 없는 것을 의미하며 $W_{xy}\neq 0$ 경우에는 그림 5와 같은 의미를 부여한다. 단순한 FCM에서는 인과관계값을 {-1, 0, 1}으로 취할 수 있다. 따라서 이경우의 인과관계는 최대 또는 최소의 정도로 발생한 것을 의미한다[4].

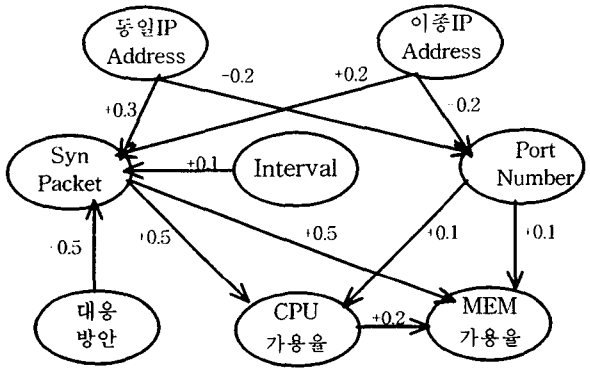


$W_{xy}>0$; N_x 수치 증가로 인한 N_y 수치 증가인 경우

$W_{xy}<0$; N_x 수치 증가로 인한 N_y 수치 감소인 경우

[그림 5] 퍼지인식도

판단모듈에서 여러 가변 요소 중 어떤 요소에 의존성을 부여함으로써 가장 최적의 탐지를 할 수 있는 것이 가장 큰 관건이다. 그뿐만 아니라 탐지한 IP Address를 침입시도로 간주하고 블랙리스트 dBASE에 저장하여야 하는지도 결정하여야 한다. 퍼지인식도는 이러한 여러 가변 요소를 적용하여 최적의 판단을 내리게 한다[5, 6].

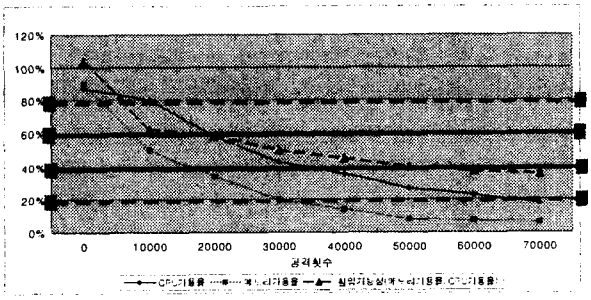


[그림 6] 가변요소 적용한 퍼지인식도

그림 6은 가변요소를 적용한 판단모듈의 퍼지인식도를 나타낸 것으로써, 판단모듈에 의존성을 갖는 가변요소로 IP Address의 동일성 여부, Port Number Count, 'Half Open State'의 시간간격 그리고 각 프로세서의 CPU가용율과 메모리가용율 및 판단모듈 후 재차 공격시 대응모듈의 처리로 인한 공격성 IP Address에 대한 Syn 패킷 조절을 둘 수 있다. 가변요소를 노드(N_x)와 다음 노드(N_y)에 두고 두 노드의 링크인 가중치(W_{xy})를 적용하는 것이다. 예를 들면, Syn Packet과 CPU가용율에서는 Syn Packet의 용량이 증가할수록 CPU가용율이 증가하므로 이때 가중치는 0보다 크게 된다. 이때 임의의 노드에 가해지는 수치는 노드와 가중치를 연결한 네트워크를 통과할수록 그리고 반복횟수에 따라서 달라지게 된다. 이를 수식화 하면 다음과 같다.

$$N_k(t_{n+1}) = \sum_{i=1}^{n} W_{ik}(t_n) N_i(t_n)$$

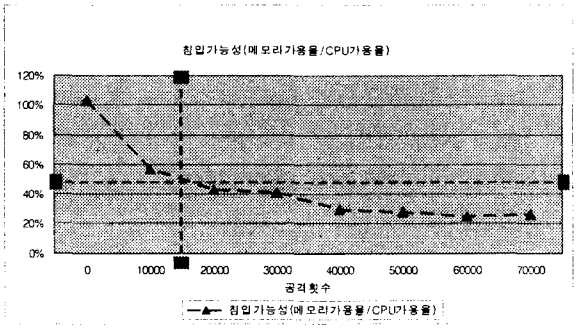
단, 가중치(W_{xy})의 증감부호는 다음 노드에 미치는 영향에 따라서 결정을 내렸으며 수치는 의미 있는 규칙기반에 의한 수치로 결정하였다. 결정하는 과정에서의 방법론으로는 신경망, 통계분석, 행렬 등 여러 방법론 중 이 실험에서는 경로분석에 의한 방법론을 적용하였다. 우선 초기값은 반복적 실험치를 위한 표본데이터로 KDD'99 데이터를 이용하여 결정하였으며 1시간동안의 발생한 이벤트를 다시 경로분석을 통하여 가중치를 업데이트 하였다. 시뮬레이션은 시험망 환경에서 테스트 하였으며 테스트 결과, 공격횟수 0~70,000번으로 연결요청이 증가 할수록 하드웨어 가용율(CPU 와 메모리)이 최종적으로 시스템부하로 적용되며 시스템부하율의 변화추이로 최대 침입시도 공격시 시스템 위험도를 확인할 수 있게 된다. 그림 7에서 침입시도시 하드웨어 가용율의 임계값 및 침입시도 판단 및 결정하는 데드라인으로 40% 가용율영역대를 설정하고 60% 가용율영역대를 시스템관리자에게 통보하는 임계값으로 설정한다. 이는 실시간 대응처리를 고려한 수치와 시험망이 아닌 네트워크에서 분산형 서비스 거부(DDoS)공격을 위한 것을 감안한 수치로써 반복적 테스트에 의한 평균값을 적용하였다[5, 6]. 아울러 점선으로 표시한 20%, 80% 가용율영역대를 다시 세분화하여 시스템에 끼치는 위험도에 의하여 가중치의 업데이트 시간간격을 조절할 수 있도록 연구진행중이다.



[그림 7] 시험망에서의 공격횟수에 대한 하드웨어 가용율

그림 7에서 공격횟수가 50,000번 이상인 경우 시스템의 부하율이 40% 데드라인을 넘게된다. 그러나 실제 네트워크에서는 그림 8에 나타나듯이 테스트결과보다 더 낮은 공격횟수에서 발생하

였는데, 초당 16,000번 이상으로 interval을 불규칙적으로 발생하였을 경우 시스템부하율이 50%보다 낮은 수치를 보였다. 이런 결과는 실제 네트워크에서 발생할 수 있는 여러 가능 요소인 트래픽(UDP Syn Flooding 공격 제외), 전파지연시간 그리고 hop count를 무한루프로 돌리는 비정상 패킷 등에 대한 고려사항을 감안하지 않은 결과로 본다.



[그림 8] 네트워크에서 공격횟수에 대한 하드웨어 가용율

실제 네트워크상의 테스트는 2002년 9월에서 2003년 2월까지 3차례 3주 동안의 결과의 평균값을 나타낸 것이며 100Mbps의 이더넷환경에서 클라이언트/호스트 모두 펜티엄4 1.7GHz, DDR 512M 메모리환경에서 측정하였다. 표 1은 측정일자 및 측정시각과 시간을 나타낸 것이다.

[표 1] 네트워크 테스트 일정

일자(3주동안)	측정 시각				측정 시간
2002년09월3주간	00	06	12	18	1시간
2002년12월3주간	00	06	12	18	1시간
2003년02월3주간	00	06	12	18	1시간

4. 결론

본 논문에서는 TCP Syn Flooding 공격에 대해서 살펴보았으며 해결책으로 여러 대안 중에서 backlog queue와 syncookies 수치를 조절한 1차 해결안을 채택하였으며 아울러 데이터링크계층의 패킷을 캡쳐 및 분석하여 침입시도탐지기능을 수행하는 네트워크 기반 탐지모델을 제안하고 시험망에서 테스트하였다. 여기서, 탐지성능을 좌우하는 요소들간의 상호 관계로부터 퍼지인식도를 이용한 침입시도 위험도를 판단하였는데, 퍼지인식도에서 가장 중요한 가중치를 결정하는 수치에 대해서는 여러 방법론적 연구가 진행중이나 현시점에선 경로분석에 의한 방법론을 취하여 실험하였다. 아울러, 침입시도 여부를 판단하는 하드웨어 가용용량 구역대를 정확히 선정하기 위하여 실시간 처리 가능한 데드라인 시간과 초당 발생하는 Flow 임계값을 설정하여 침입여부를 결정하는 방법에선 5단계의 위험수준을 고려하는 방법을 취한 실험이 진행 중이다. 향후 연구과제로 패킷캡처를 소프트웨어로 처리하기 전에 하드웨어적으로 캡쳐하는 전처리기능을 두고 소프트웨어 처리로 인하여 미쳐 캡쳐하지 못한 패킷에 대한 분석을 하여 완벽한 탐지 및 방지를 하고자 한다. 특히, 분석 및 판단모듈을 각각의 독립시스템에 두고 FCM의 방법론을 다양하게 하여 테스트를 하여 판단모듈에 의해 학습된 지능형 대응모듈기능을 추가한 지능적 탐지 및 방지시스템을 연구목표로 삼는다.

5. 참고문헌

[1] Computer Emergency Response Team, "TCP Syn Flooding and IP Spoofing Attacks," CERT Advisory: CA, 96-21, 1996.

[2] Syncookies mailing list. ftp://koobera.math.uic.edu/pub/docs/syncookies-archive, 1999.

[3] Aman Garg and A.L.Narasimha Reddy, "Policy Based End Server Resource Regulation," IEEE/ACM Transactions on Networking, Vol. 8, No. 2, pp. 146-157, 2000.

[4] C. L. Schuba, I. V. Krsul, M. G. Khun, E. H. Spaford, A. Sundram, and D. Zamboni, "Analysis of a denial of service attack on tcp," IEEE Symposium on Security and Privacy, 2002.

[5] S.Y. Lee and Y.S. Kim, "A RTSD Mechanism for Detection of DoS Attack on TCP Network," Proceedings of KFIS 2002 Spring Conference, pp. 252-255, 2002.

[6] K.B Sim, J.W. Yang, D.W. Lee, S.Y. Lee, Y.S. Kim, et Al. "Intrusion Detection System of Network Based on Biological Immune System," Journal of Fuzzy Logic And Intelligent Systems, Vol. 12, No. 5, pp. 411-416, 2002.