메모리 소비율 모니터링을 통한 SYN Flooding 패킷 차단

윤종철*, 곽인섭*, 강홍식**
인제대학교 컴퓨터공학부*
email : {pcman95, kinsub, hskang}@nice.inje.ac.kr

SYN Flooding packet interception through memory specific consumption monitoring .

Jong-Chul Yun*, In-Seub Kwak*, Heung-Seek Kang**,
Dept of Computer Engineering, Inje University

요 약

서비스 거부공격(DoS Attack : Denial-of-Service Attack)이란 공격자가 침입대상 시스템의 시스템 자원과 네트워크 자원을 대량으로 소모시킴으로써 정상 사용자로 하여금 시스템이 제공하는 서비스를 하지 못하도록 하는 공격을 의미한다. TCP SYN Flooding 기법을 이용한 DoS공격은 서비스 자체를 하지 못하도록 하기 보다는 다른 공격을 하기 위한 사전 공격으로써 활용될 소지가 높은 공격법인 것이다. 본 논문에서는 TCP SYN Flooding을 이용한 DoS공격의 근본적인 원인을 분석하고 시스템 보안 관리자의 입장에서 이 공격에 능동적으로 탐지 할 수 있는 해결책을 모색해보고자 한다.

1. 서 론

DoS(Denial-of-Service)공격은 일반인들에게는 다소 생소한 개념이었지만 최근 몇 년 사이 각종 유명 사이트들이 DoS공격을 받아 운영이 불가능해진 사건이 발생하면서 일반인들에게 DoS공격이 점차 인식되기 시작하였다. DoS는 한 사용자가 시스템의 리소스를 독점하거나 모두 사용함으로서 다른 사용자들이 이 시스템의 서비스들을 정상적으로 사용할 수 없도록 만드는 것을 말한다. 이런 의미에서 볼 때 시스템의 정상적인 서비스를 방해 할 수 있는 모든 행위를 DoS이라고 할 수 있다.

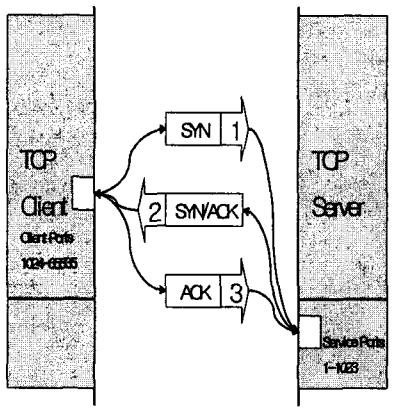
TCP SYN Flooding을 이용한 DoS공격은 한 시스템의 정상적인 서비스를 방해하는데서 그치는 것이 아니라 또 다른 공격을 위한 사전 공격으로 널리 활용되고 있는 크래킹 기법 중에 하나이기도 하다. 이 공격법은 IP Spoofing을 하기 위해 원래의 IP를 가진 시스템을 무력화시키기 위해서 미리 사용된다. 이렇게 함으로써 크래커는 익명성을 가지게 되고 자신의 IP가 아닌 다른 사람의 IP로 안전하게 크래킹을 할 수 있기 때문에 많이 선호하는 기법중의 하나이다. 이런 점들로 미루어 볼 때 DoS공격은 현재를 비롯하여 미래에도 지속적으로 활용될 소지가 높다고 볼 수 있다. 본 논문에서는 TCP SYN Flooding을 이용한 DoS공격에 대해 살펴보고 보다 효율적으로 이에 대처 할 수 있는 방법을 모색해보고자 한다. 논문의 2장에서는 TCP SYN Flooding공격의 원리를 소개하고 3장에서는 DoS공격의 탐지 기법을 소개한다. 그리고 4장에서는 대응방안 및 구현 기법을 간략히 소개한다. 마지막 5장에서는 본 내용의 결론을 정리하고 향후 이 연구의 발전과제에 대해서 언급하고 논문을 맺도록 한다.

2. 관련연구

2.1 TCP SYN Flooding공격 원리

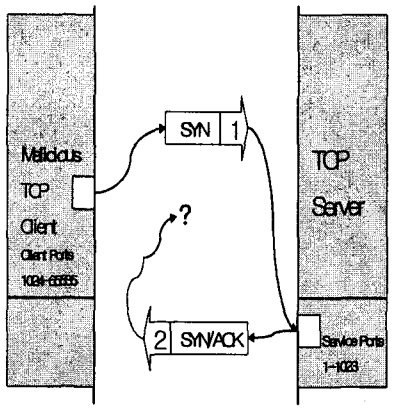
TCP SYN Flooding 공격은 서버와 클라이언트 간에 실질적으로 통신이 이루어지기 위해서는 "3 Way Handshaking"이라는 과정이 있어야 한다는 TCP 프로토콜의 근본적인 취약점을 이용한 방법이다.

[그림1]은 3 Way Handshaking을 잘 보여주고 있다. 먼저 첫 단계에서 클라이언트는 서버에 접속하기 위해서 SYN패킷을 보낸다. 그러면 접속 요청을



[그림1] 3 Way Handshaking

받은 서버 측에서는 SYN/ACK패킷을 클라이언트 측에게 보냄으로써 접속 요청을 승인한다. 그러면 클라이언트는 최종적으로 서버에 ACK패킷을 보내고 연결이 이루어진다. 그 후 본격적인 패킷의 교환이 이루어지게 되는 것이다.



[그림2] 잘못된 3 Way Handshaking

그러나 만일 악의적인 클라이언트가 Source IP 부분을 속여서 보내게 되면 서버는 SYN/ACK패킷을 보낼 수가 없게 된다. 서버 측에서는 회선장애로 판단하고 일정 시간 뒤에 다시 SYN/ACK패킷을 보내게 된다. 악의적인 클라이언트에서 잘못된 패킷을 짧은 시간에 대량으로 보내게 되면 서버에서는 'Half Open' 상태의 패킷이 넘치게 되어 결국은 백로그 큐(Backlog Queue)가 꽉 차게 됨으로서 결과적으로 정상적인 서비스를 할 수 없게 된다.

3. DoS공격의 탐지 기법

리눅스에서 TCP SYN Flooding공격을 확인하는 가장 간단한 방법은 "netstat"라는 명령을 통해서 확인해 볼 수 있다. "netstat -na|grep SYN_RECV"를 입력하면 SYN_RECV 메시지가 나열되는데 SYN_RECV 메시지는 클라이언트로부터 패킷을 기다리는 것을 의미한다. 따라서 이 메시지가 아주 많이 보인다면 TCP SYN Flooding 공격을 의심해 볼 수 있는 것이다. 하지만 정확하게 공격이 되고 있다고는 판단하기 힘들고 관리자에 의해서 임의적으로 실행되어야 하기 때문에 어디까지나 참고적으로 사용되어야 한다.

TCP SYN Flooding공격을 탐지하는 전통적인 방법은 서버에 연결된 패킷들의 정보를 분석함으로써 공격을 탐지하는 것이다. 패킷을 분석해서 네트워크 세션 정보를 만들고 이 정보를 바탕으로 비정상행위를 탐지하는 방법이다. 이 방법은 소수의 공격자에 의한 공격의 경우 네트워크 세션정보를 가공하고 분석하여 공격여부를 탐지 할 수 있지만 DDoS와 같이 짧은 시간 내에 수많은 좀비 머신으로부터의 공격에는 적합하지 않다.

또 다른 훌륭한 탐지기법은 시스템의 리소스를 모니터링 하는 방법이다. 시스템이 가지고 있는 메모리는 소비되는 목적에 따라서 시스템메모리와 서비스 메모리로 구분이 되는데 시스템 메모리는 운영체제나 커널이 작동하기 위해서 사용되는 메모리 이며 서비스 메모리는 각종 데몬을 돌리거나 사용자의 요구에 부응하기 위해서 준비된 메모리를 말한다. 서비스 메모리 중에서도 커널 메모리는 시스템이 수신한 네트워크 패킷을 처리하는 과정에서 메모리를 소모하게 되는데 수신된 패킷의 수에 비례하여 즉각적으로 메모리를 소비하는 특징을 가지고 있기 때문에 이를 감사 자료로 이용해 DoS공격의 여부를 감지해 낼 수 있다.

4. 대응방안 및 구현 기법

4.1 기존 라우터 장비를 이용한 대응방안

그렇다면 이러한 공격에 대한 명백한 대응방안은 없는가? 불행히도 아직까지 완벽한 대응방안은 나오지 않고 있다. 다만 최대한 빨리 공격징후를 포착해서 그 피해를 최소한으로 줄이는 방법뿐이다. 과거 많은 회사들이 이 문제를 해결하기 위해서 많은 노력을 기울였다. 그 중에서 라우터등 네트워크 장비로 유명한 CISCO사에서는 TCP SYN Flooding 공격을 차단하기 위해 'TCP Intercept"라는 솔루션을 제안했었다. 이 솔루션은 외부에서 들어오는 SYN

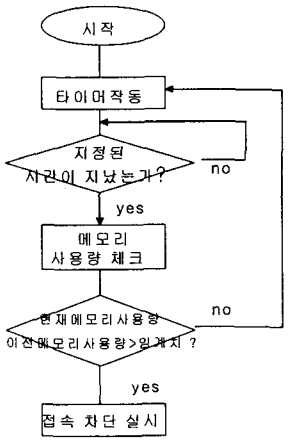
패킷 요청을 일차적으로 수행하여 패킷의 투명성 여부를 판단한 후 다시 서버에게 연결 시켜주는 방식이다. 그 외에도 ingress와 egress 필터링을 적용해 각각의 네트워크에 방화벽을 도입하는 방식을 사용한 솔루션들이 많았다. 이러한 방식들은 실제로 스푸핑된 SYN패킷들은 서버로 보내지 않는 상당한 성과를 나타내었다. 하지만 아쉽게도 네트워크 상의 트래픽이 많이 발생하였을 경우 라우터의 CPU와 Memory부하가 너무 높아지는 경향과 다운 증상을 보였다. 더욱이 이러한 라우터 장비가 고가의 장비라는 점에서 개인 또는 소규모의 서버를 운용하는 회사에서는 큰 경제적 부담으로 작용할 수 있는 단점이 있다.

4.2 대응 방안 및 구현 기법

아무리 고가의 장비로 무장을 하고, 높은 버전의 보안 패치로 보안 안정성을 높였다고 하더라도 앞서 설명한 TCP고유의 허점이 존재하는 한 기존의 TCP를 기반으로 하는 수많은 하드웨어와 소프트웨어 시스템을 일제히 교체하지 않는 한 이 공격을 근본적으로 막을 수 있는 방안은 없다. 다만, DoS공격은 그 공격 징후를 얼마나 빨리 포착하고 대응하느냐에 따라서 피해규모가 달라지는데 본 논문에서는 이러한 공격의 징후를 가능한 빨리 찾아 낼 수 있도록 대응 방안을 마련하고자 한다.

구현프로그램을 만들기 위한 2가지 알고리즘을 간략히 소개하고자 한다. 리눅스 시스템에서는 시스템 상태정보를 제공하는 "/proc" 디렉토리 안에는 커널의 메모리 사용정보를 포함하는 slab라는 파일이 존재한다. slab파일 내에는 ip_conntrack항목이 있는데 수신한 패킷에 의해 세션이 성립되었을 경우 성립된 세션 현황을 보여준다. ip_conntrack항목은 1개당 약 350바이트의 스왑되지 않은 커널 메모리가 사용되고 있다는 것을 의미한다. 즉, 연결된 사용자가 많으면 많을수록 커널은 많은 메모리를 소비하고 있음을 즉각적으로 이 파일에 기록하게 된다. 공격이 되지 않는 평소의 메모리 소비량과 임계치값을 두고, 일정시간이내에 평소 사용량 보다 일정한 임계치값을 초과하는 메모리 사용이 일어난다면 공격받은 것으로 간주 하게 된다.

[그림4]는 쉘 프로그래밍으로도 구현 할 수 있는 간단한 알고리즘이다. 프로그램 내부에서 가동된 netstat 명령이 메시지의 개수를 파악해서 일정 한 개수(프로그램 상에서는 기본적으로 20개로 정해져 있

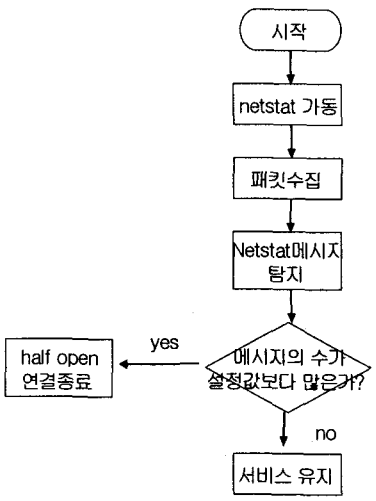


[그림3] 탐지 알고리즘1

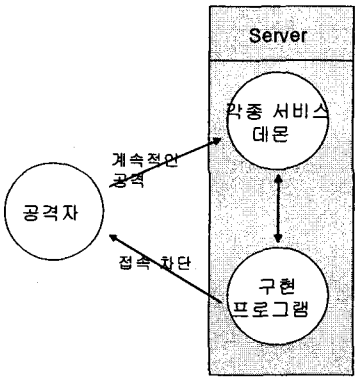
다.) 이상일 경우 HTTP서비스를 멈추었다가 다시 시작한다든가 Half Open된 연결만 종료한다든가 하는 구체적인 대응을 할 수 있다. 다만 HTTP 서비스를 다시 시작한다면 정상적인 사용자 까지도 불편을 겪어야 한다.

아래는 [그림4]의 슈도코드(Sudo Code)이다.

```
...(중략)...
if($TASK){
$TASK_CONFIRM='netstat -na|grep SYN_RECV';
if ($TASK_CONFIRM>20){
'/etc/rc.d/init.d/httpd restart'; ...(후략)...
```



[그림4] 탐지 알고리즘2



[그림5] 대응 프로그램의 동작 형태

5. 결론 및 향후 방향

본 논문에서는 2가지 알고리즘을 이용하여 효율적으로 크래커의 공격을 탐지하고 대처 할 수 있는 방법을 제안하였다. 하지만 이 방법들 역시 단점이 없는 것은 아니다. 다만 기존의 탐지 방법과 연계함으로써 좀더 빨리 DoS공격을 탐지하여 피해를 최소화 하였으면 할 뿐이다. 본 연구에서는 대응 프로그램의 효율성이라든지 성능평가는 이루어지지 않았으나, 차후에 일련의 과정을 통해서 좀더 빠르고 능동적인 알고리즘을 이용한 탐지 및 대응 프로그램을 방안을 제시하도록 하겠다.

참고문헌

[1] 강진석, 강홍식 "개인용 리눅스 서버 기반에서 TCP SYN Flooding탐지를 위한 프로그램의 설계와 구현" 2001.10

[2] 고광선,강용혁,엄영익 "리눅스시스템에서 서비스 자원소비율을 이용한 분산서비스거부공격 탐지기법" 2003.5

[3] 인포북, 리눅스 보안의 모든 것, 2000

[4] 정보문화사, 해커프루프, 1998

[5] 교학사, 네트워킹 프로토콜, 1999

[6] 오정욱, "Hacking How to", 크라운 출판사, 2001

[7] 파워북, 해커를 위한 파워 핸드북, 2000

[8] Rocky K. C. Chang, "Defending against Flooding-Based Distributed Denial of Service Attacks" Proceedings of the tenth international conference on World Wide Web April 2001

[9] Joseph S. Sherif, and Tommy G. Dearmond, "Intrusion Detection : System and Models," Proceedings of Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2002