연구자료 등록서

작성자	작성부서	승인자	작성일자	페이지
고 규 만	기반보호사업단 기반보호팀	이 강 신(인)	2002-08-19	20

제출번호	구분 연도 일련번호	출연(위탁)기관		변경코드	
	기반보호 - 2002 - 001				

기술보호등급	1. A급 2. B급 ③ C급	기술보호기간 (C급에 한함)	

승인문서관리자	등록일자
고 승 철 (인)	2002-08-31

제목	한 글	분산반사 서비스 거부에 대한 고찰
	영 문	A Study of Distributed Reflection Denial of Service(DRDoS)
색인어	한 글	서비스 거부 공격, 분산 반산 서비스 거부 공격
	영 문	TCP, DoS, DDoS, DRDoS
요 약 (300자 이내)		지난 1월 미국에서 DRDoS가 처음 발견되었다. 이 공격의 대상인 된 깁슨리서치는 DDoS 공격과는 달리 정상적으로 동작하고 있는 베리오(Verio.net), 퀘스트(Qwest.net), 어보브(Above.net)의 라우터와 야후의 웹서버들로부터 공격을 당했다. DDoS 공격이 여러 서버에 설치된 특정 에이전트에 의해 공격이 진행되는 반면 DRDoS 공격은 정상적인 서비스를 운영하고 있는 서버를 에이전트로 활용하기 때문에 해커들이 손쉽게 이용할 수 있다. 또한 공격의 근원지를 추적하기 어렵고 공격을 막을 수 있는 방법이 그리 용이하지 않다. 이 TM에서는 DRDoS를 설명하기 앞서 TCP 프로토콜의 기본적인 동작, SYN Flood, 대역폭 고갈 공격, DoS(Denial of Service), DDoS(Distributed denial of service)공격의 특징을 간단하게 설명하고 DRDoS(Distributed Reflection Denial of Service)공격의 원리와 특징 및 방어 방법을 기술한다.
공동 작성자		
관 련 문 서		
배 포 처		

종합관리번호	보존등록일	확인

DRDoS

(Distributed Reflection Denial of Service)

2002.8.19
기반보호팀/ 고규만

1. 개 요

지난 1월 미국에서 DRDoS가 처음 발견되었다. 이 공격의 대상인 된 깁슨리서치는 DDoS 공격과는 달리 정상적으로 동작하고 있는 베리오(Verio.net), 퀘스트(Qwest.net), 어보브(Above.net)의 라우터와 야후의 웹서버들로부터 공격을 당했다. DDoS 공격이 여러 서버에 설치된 특정 에이전트에 의해 공격이 진행되는 반면 DRDoS 공격은 정상적인 서비스를 운영하고 있는 서버를 에이전트로 활용하기 때문에 해커들이 손쉽게 이용할 수 있다. 또한 공격의 근원지를 추적하기 어렵고 공격을 막을 수 있는 방법이 그리 용이하지 않다.

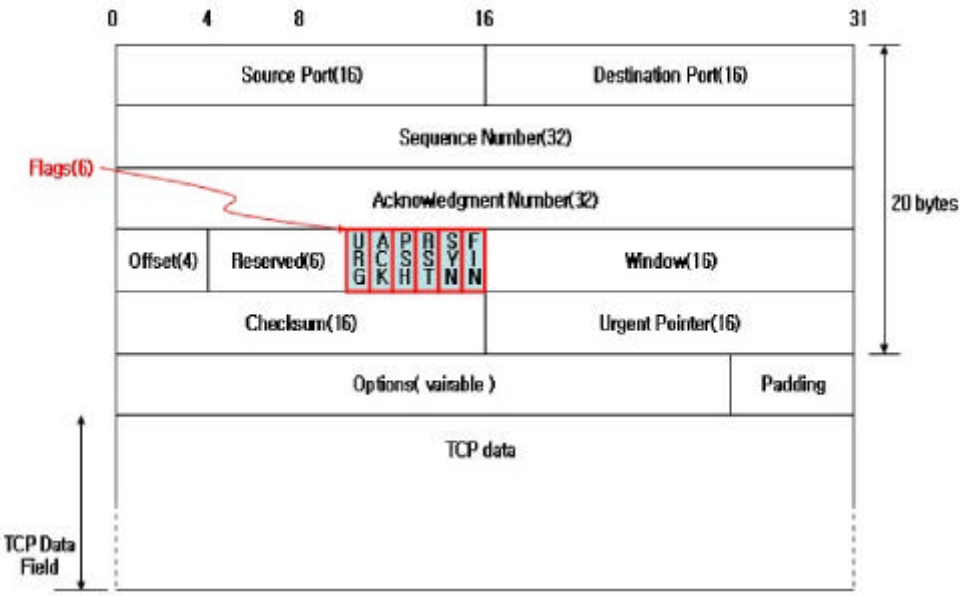
이 TM에서는 DRDoS를 설명하기 앞서 TCP 프로토콜의 기본적인 동작, SYN Flood, 대역폭 고갈 공격, DoS(Denial of Service), DDoS (Distributed denial of service)공격의 특징을 간단하게 설명하고 DRDoS(Distributed Reflection Denial of Service)공격의 원리와 특징 및 방어 방법을 기술한다.

2. TCP 프로토콜의 기본적인 동작

DRDoS 공격의 구조를 이해하기 전에 TCP 프로토콜의 동작을 알아야 한다.

글로벌네트워크에서 서로 다른 서버들이 어떻게 연결될 것인가? 두 서버가 "접속협약(connection agreement)"을 맺는 방법으로 각자의 데이터 패킷에 주소를 붙여서 전송할 수 있다. 이러한 연결을 가상 TCP 연결(Virtual TCP Connection)이라고 한다.

개개의 TCP 패킷은 각 패킷의 목적과 내용을 명시하는 플래그 비트(FLAG BITS)를 포함한다. 예를 들어, SYN(synchronize)플래그 비트가 세팅(1)된 패킷은 송신자와 수신자 사이에 접속을 초기화한다. ACK(acknowledge) 플래그 비트가 세팅(1)된 패킷은 송신자로부터 정보를 잘 받았음을 확인하는 것이다. FIN(finish) 플래그 비트가 세팅(1)된 패킷은 송신자와 수신자 사이의 접속을 종료함을 의미한다.



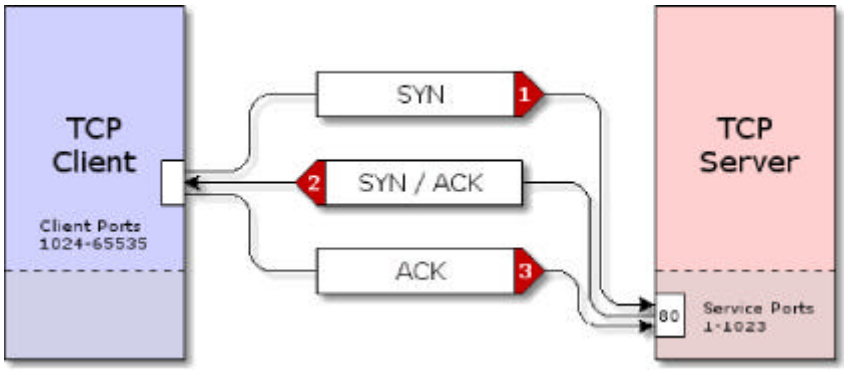
[그림 1] TCP Header

TCP는 접속 기반(connection-oriented) 프로토콜이다. TCP 세그먼트 헤더의 플래그 필드에 적당한 값(0,1)을 설정함으로서 연결을 제어할 수 있다. TCP 헤드에서 사용되는 Flag는 다음과 같다.

- URG : Urgent Pointer field의 값이 유효함을 나타낸다.
- ACK : ACK field가 유효함을 나타낸다.
- PSH : 해당 세그먼트의 데이터가 즉각 상위 응용 프로그램에 전달되어야 함을 나타낸다. 따라서, 이 세그먼트에 대한 확인은 해당 확인 번호 이전의 모든 데이터들이 이미 통신 상대자에게 무사히 전달되었음을 의미한다.
- RST : 접속의 재설정이나 무효한 세그먼트에 대한 응답을 나타낸다.
- SYN : 접속 설정 요청으로서 세그먼트에 대한 확인이 반드시 있어야 한다.
- FIN : 통신 상대자중 한 쪽에서 접속을 종료함을 나타내며, 이 방향으로 더 이상의 데이터 흐름이 끝났음을 의미한다. 이 세그먼트에 대한 확인이 반드시 이루어져야 한다.

● Three-way handshake

TCP 연결 확립 과정은 세 개의 세그먼트가 교환되기 때문에 "three-way handshake"라고 하며 다음 세 단계의 과정을 통해 이루어진다.



[그림 2] Three-way handshake

1. SYN

TCP 클라이언트(웹브라우저, FTP, Telnet 클라이언트)는 TCP 서버의 포트와 TCP 클라이언트의 $ISN_{클라이언트}$(Initial Sequence Number)가 명시된 SYN패킷을 TCP 서버에게 보냄으로서 TCP 서버와 접속을 초기화한다.

[그림 2]에서, SYN 패킷은 일반적으로 1024에 65535사이의 클라이언트 포트에서 1에서 1023사이의 서비스 포트로 보내진다. TCP 클라이언트에서 동작하고 있는 클라이언트 프로그램은 원격서버에 연결하기 위해 OS에게 포트 할당을 요구한다. TCP 서버에서 동작하고 있는 서버프로그램은 OS에게 특정 포트에서 들어오는 패킷을 리스닝(Listening)할 수 있도록 요구한다. 예를 들어, 웹서버 프로그램은 전형적으로 포트 80번을 통해 패킷을 받아들이고 클라이언트의 웹브라우저는 일반적으로 서버의 포트 80번으로 서비스를 요청한다.

2. SYN/ACK

접속을 요구하는 SYN 패킷을 "열려진(Open)" TCP 서비스 포트가 받았을 때, TCP 서버는 자신의 $ISN_{서버}$를 포함하고 $ISN_{클라이언트}+1$의 값을 Acknowledgement Number로 설정한 SYN/ACK 패킷을 SYN 패킷에 대한 응답으로 보내게 된다.

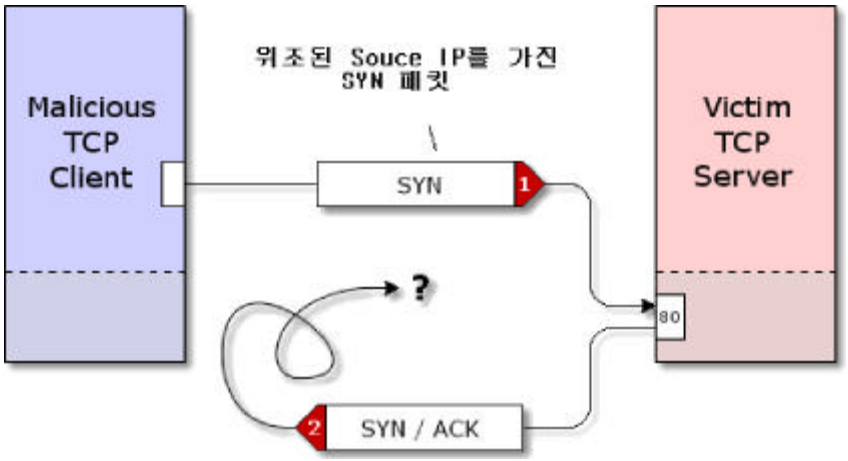
3. ACK

TCP 클라이언트는 TCP 서버의 SYN/ACK에 대한 응답으로 $ISN_{서버}+1$의 값을 Acknowledgement Number로 설정하여 ACK 패킷으로 응답한다.

3. TCP 오용(Abusing TCP) : 전통적인 TCP SYN Flood

TCP SYN Flood 공격은 TCP 프로토콜 자체의 문제점을 이용한 것으로, 서버가 클라이언트에게 SYN/ACK를 보낸 후, 클라이언트로부터 다시 ACK를 받기 이전의 시점에서 발생한다. 이 상태를 "불완전한(half open)" 연결이라고 부른다. 서버는 진행중인 연결에 대한 정보를 저장하기 위해 시스템 메모리에 자료구조를 구축하며, 이 자료구조는 그 크기가 제한되어 있다. 따라서, 계속적인 "불완전한" 연결을 생성함으로써 서버의 자원이 고갈된다.

TCP 동작을 보여주는 위의 [그림 2]처럼, 클라이언트의 SYN 패킷을 받은 서버는 클라이언트와 연결(connection)을 준비하게 된다. 일반적으로 TCP 연결을 하기 위해 필요한 데이터를 보내고 받기 위해 메모리가 할당되고, 클라이언트 IP, 연결 포트번호 등의 정보를 기록할 것이다. 이러한 방법으로 서버는 클라이언트의 SYN 패킷에 대한 응답을 준비하고 그 응답으로 SYN/ACK 패킷을 클라이언트에게 보내게 된다. 서버가 ACK 패킷을 받지 못하면 서버는 SYN/ACK 패킷이 도중에 잃어버렸거나 중간의 인터넷 라우터가 버렸을 거라고 가정하고 클라이언트에게 SYN/ACK 패킷을 재전송 할 것이다. 공격이 이루어지는 과정은 다음과 같다.



[그림 3] TCP SYN Flood

1. 공격자는 타겟서버에 TCP 연결을 원하는 SYN를 보낸다. 이 때 자신이 아닌 위조된 다른 서버의 IP 주소를 사용한다.

2. 타겟서버는 위조된 IP의 주소를 갖는 제 3의 서버로 SYN/ACK를 보내어 연결준비를 알리고, 응답을 기다린다.

4. 1과 2의 과정을 무수히 반복하면서 타겟서버의 메모리 버퍼가 고갈됨으로 오버플로우가 발생한다.

서버가 SYN 패킷을 받게 되면 메모리와 연결을 위해 서버의 자원(connection resources)이 할당된다. 해커들은 TCP 서버가 "불완전한 연결(half open connection)을 통제하는 데 한계가 있다는 것을 알고 이 한계를 초과하여 연결을 시도하는 간단한 방법을 발견했다.

"Raw Sockets"을 사용하면 패킷의 리턴주소(source IP)를 위조할 수 있다. 위조된 근원지 IP를 가진 SYN 패킷이 서버에 도착하게 되면 서버는 유효한 연결 요청으로 인식한다. 그 즉시, 서버는 메모리 버퍼를 할당하고 새로운 연결에 대한 정보를 기록한 후 클라이언트에게 SYN/ACK 패킷으로 응답을 하게 된다.

그러나, SYN 패킷에 포함된 근원지 IP는 랜덤숫자(Random number)로 위조되었기 때문에, 서버는 SYN/ACK 패킷을 클라이언트의 위조된 근원지 IP로 보내게 될 것이다. 위조된 IP가 실제 존재하는 유효한 IP 라면 그 주소의 머신은 자신이 서버에게 연결요청을 하지 않았다는 것을 알리기 위해 RST 패킷으로 응답을 하게 될 것이다. 하지만, 그 주소에 머신이 존재하지 않는다면 SYN/ACK 패킷은 버려지게 될 것이다.

악의적인 클라이언트의 연결 요청이 위조된 것이라는 것을 알 방법이 없기 때문에 서버는 정상적인 연결요청으로 간주하게 되고 Three-way handshake를 완료하기 위해 기다리게 된다. 서버가 얼마의 시간동안 ACK를 받지 못하게 되면, 서버는 클라이언트에게 SYN/ACK를 재전송 하게 되는 문제가 발생한다.

이러한 TCP 연결 관리방식은 제한되어 있는 서버의 자원을 낭비하게 된다. 그 사이에도, TCP 클라이언트는 계속해서 위조된 SYN 패킷을 서버에게 보내게 되고, 서버는 "불완전한" 연결(half open connection)을 위해 계속해서 메모리와 자원을 할당한다. 일정시간이 되면, 서버는 더 이상 불완전한 연결을 수용할 수 없게 되어 결과적으로 서버의 자원이 악의적으로 낭비(고갈)되기 때문에 유효한 연결요청도 수용할 수 없게 된다.

※ 대역폭 고갈(Bandwidth Consumption)

대부분의 분산(Distributed)공격은 여러 서버가 연결되어 있는 인터넷의 대역폭을 고갈시키는 공격 방법이다. 개개의 패킷은 별로 해가 되지 않을지도 모르지만 이러한 패킷이 여럿 모이게 되면 타겟이 되는 서버의 인터넷 연결이 안되거나 패킷 처리 리소스가 고갈되게 된다. 결과적으로 유효한 트래픽이 적절한 서비스를 받지 못하게 되는 것이다.

● SYN Flood는 대역폭을 고갈시키는 것이 아니다.

이러한 SYN Flood의 효과를 반감시키기 위해서 OS의 기능이 개선되기 전에는 심지어 다이얼업 연결을 사용하는 하나의 악의적인 머신으로도 성능이 좋은 서버의 연결큐를 고갈시킬 수 있었다. SYN spoofing 취약점에 대한 방어기술이 많이 있음에도 불구하고 효과적인 해결책은 거의 없는 실정이다.

초기의 위조된 근원지 IP(spoofed-source-IP) SYN 공격이 대역폭을 고갈 공격이 아니라

는 것은 중요한 내용이다. 즉, 네트워크의 대역폭 자원(bandwidth resource)을 소모하는 것보다, 서버의 연결자원(connection resource)을 낭비하는 것이다.

또한, DoS(Denial of Service)공격은 분산공격이 아니라는 것을 주목하자. 즉, DOS 공격은 DDoS 공격의 형태가 아니었다. 하나의 악의적인 SYN 패킷 생성 머신이 자신의 IP 주소를 숨기고 위장된 IP SYN 패킷을 웹사이트에 보냄으로써 서비스를 중단시킬 수 있었다.

- **SYN spoofing 문제를 해결하는 방법**

OS 벤더들은 위조 SYN 패킷 DoS 공격에 대응하기 위해 TCP 프로토콜 스택을 강화했다. 이런 방법들의 대부분은 양적인 개선이었다. 그러나, 이 방법들이 문제를 완전하게 해결하지는 못했다.

하지만, 다음 두 가지 방법은 만족할 만한 해결책을 제시하고 있다. 자세한 내용은 링크된 문서를 참조하기 바란다.

  - SYN-Cookies(http://cr.yp.to/syncookies.html)
  - GENESIS(http://grc.com/r&d/nomoredos.htm)

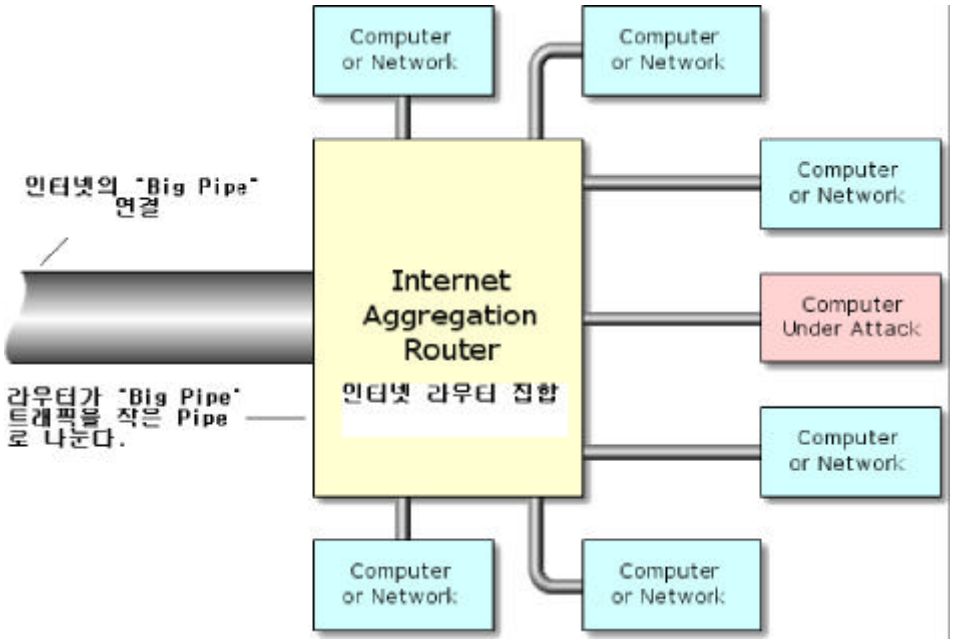
위의 두 가지 방법은 TCP 프로토콜과 호환을 유지하고 있고 동작의 원리는 SYN 패킷을 받은 후에 서버의 할당된 모든 자원을 제거하고 SYN/ACK 패킷을 보내게 되어 있다.

4. 대역폭 고갈 공격

해커의 기술이 전문화 되어감에 따라, 이미 손상된(compromised) 인터넷 호스트의 목록이 빠르게 추가되고 있고 대역폭 고갈 공격인 DDoS(Distributed denial of service)공격이 일반화되었다. 즉, 강력한 공격 도구를 이미 어린아이조차 양심의 가책 없이 사용할 수 있게 되었다.

- **대역폭 고갈 공격동안 발생하는 현상**

DoS 공격과는 달리 대역폭 공격은 타겟 서버의 대역폭이나 서버의 네트워크 연결 대역폭(server's network connection bandwidth)을 고갈시키기 위해 악의의 의미 없는 인터넷 트래픽을 생성한다. 악의의 패킷 흐름(malicious packet flood)이 네트워크의 유효한 트래픽(valid traffic)과 경쟁하게 되고 결과적으로 유효한 패킷들이 흐름에서 살아남을 가능성이 낮아지는 것이다. 네트워크의 서버들은 인터넷 서비스를 중단하게 된다.



[그림 4] 대역폭 고갈

- **패킷 버리기(Discarding packets)**

[그림 4]는 대역폭 공격의 결과를 잘 보여준다. 컴퓨터나 네트워크는 라우터에 의해서 서비스가 된다. 이 라우터는 ISP(Internet Service Provider)네트워크의 "고객 가장자리(customer edge)"에 위치하고 작은 규모인 고객 네트워크의 트래픽을 모으고 분산한다. 그러므로, 범용 인터넷으로 라우팅하기 위해 대역폭이 작은 수많은 네트워크들이 하나의 높은 대역폭을 가지는 네트워크로 집결된다.

"Big Pipe"로 인터넷 트랙픽이 들어와서 저장되고 대역폭이 낮은 클라이언트 네트워크로 포워딩(Store and forward 방식)될 것이다. 그러나, "Big Pipe"가 단 하나의 고객 네트워크으로 향하는 패킷 흐름으로 채워진다면 어떤 일이 발생하게 될까? "Big pipe"에서 작은 pipe로 너무 많은 패킷을 보내게 되면 라우터는 작은 파이프로 향하는 많은 패킷을 고의로 버리게 된다

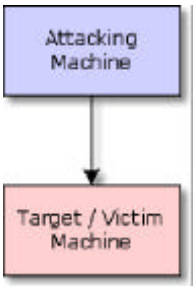
작은 pipe의 네트워크 자원에 접근을 시도하는 클라이언트가 정상적인 패킷을 보내게 되더라도 패킷을 잃게 되고 그 버려진(dropped) 패킷을 다시 전송하게 될 것이다. 그러나, 클라이언트는 일반적으로 몇 번의 시도 후에는 포기 할 것이다. 타겟 네트워크는 악의적인 트래픽 흐름에 의해 서비스가 중단될 것이다.

5. DoS versus DDoS

● **DoS(Denial of Service)**

일반적으로 DoS 공격은 공격자의 컴퓨터로부터 타겟서버와 그 서버가 속한 네트워크에 과다한 데이터를 보냄으로써 시스템과 네트웍의 성능을 급격히 저하시켜 타겟서버에서 제공하는 서비스들을 인터넷 사용자들이 이용하지 못하도록 하는 것이다.

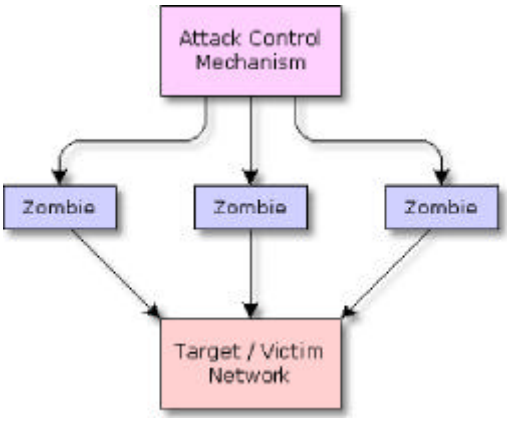
DoS 공격은 하나의 머신이 다른 머신을 공격한다.



[그림 5] DoS 공격

● **DDoS(Distributed Denial of Service)**

DDoS 공격은 많은 수의 호스트들에 패킷을 범람시킬 수 있는 DoS공격용 프로그램들이 분산 설치되어 이들이 서로 통합된 형태로 어느 타겟서버(네트웍)에 대하여 일제히 데이터 패킷을 범람시켜서 그 타겟서버(네트웍)의 성능저하 및 시스템 마비를 일으키는 기법을 말한다.

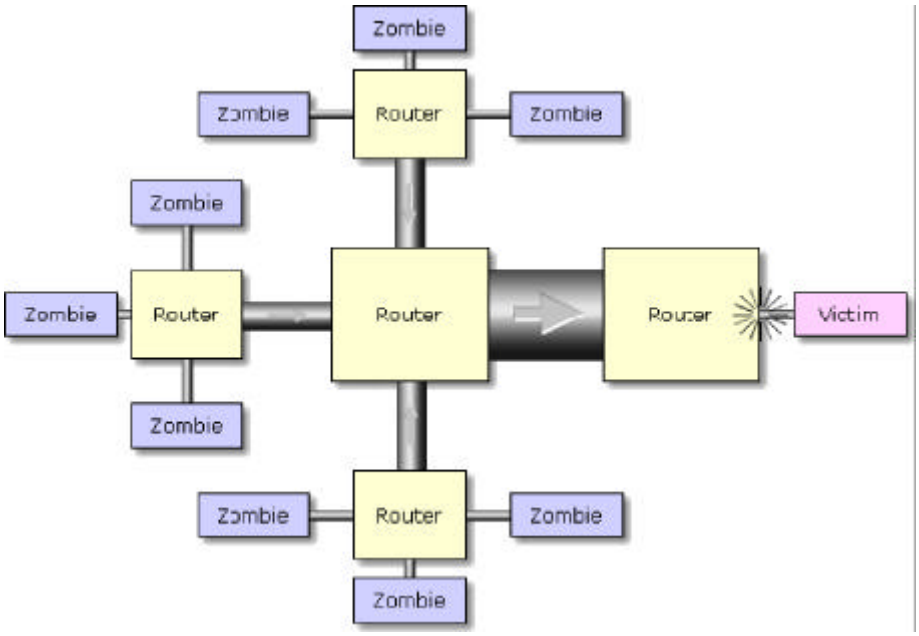


[그림 6] DDoS 공격

[그림 6]에서 보는 것처럼 이 구조는 일반적으로 DDoS 공격에서 사용된다. 원격으로 조정되는 "Zombie" 공격프로그램이 설치되어 있는 머신은 중앙 조정 센터(central control agency)인 "Zombie Master"에 의해 공격을 지시 받게된다. Zombies 네트워크가 공격지시를 받으면 하나의 타겟 머신이나 네트워크를 목표로 하는 악의적인 트래픽 흐름(a flood of malicious traffic)을 생성하기 시작한다.

o 분산된 zombie 트래픽 결합(Distributed zombie traffic aggregation)

개개의 트래픽 흐름은 많은 분리된 소스로부터 인터넷을 통하여 움직이게 되기 때문에, 그것들은 하나의 거대한 흐름(a single massive flood)을 형성하기 위해 인터넷 라우터에서 결합된다.



[그림 7] 분산된 zombie 트래픽

흐름이 마지막으로 타겟의 ISP 라우터(target network's service provider aggregation router)를 만나게 되면 네트워크 트래픽의 대부분이 버려지게 된다. 라우터는 유효한 트래픽과 유효하지 않은 트래픽을 구분할 수 없기 때문에 (라우터는 모든 패킷을 동일하게 본다) 네트워크의 유효한 트래픽 또한 버려지게 될 것이다.

6. DRDoS(Distributed Reflection Denial of Service) : (차세대 DDoS 공격: A Next Generation DDoS Attack)

2002년 1월 11일 AM 2:00에 grc.com은 좀더 진일보한 악의적인 패킷 흐름을 발견했다. 이 새로운 스타일의 DDoS 공격을 DRDoS(Distributed Reflection Denial of Service)라고 부른다.

가. 사건배경(A packet flood mystery)

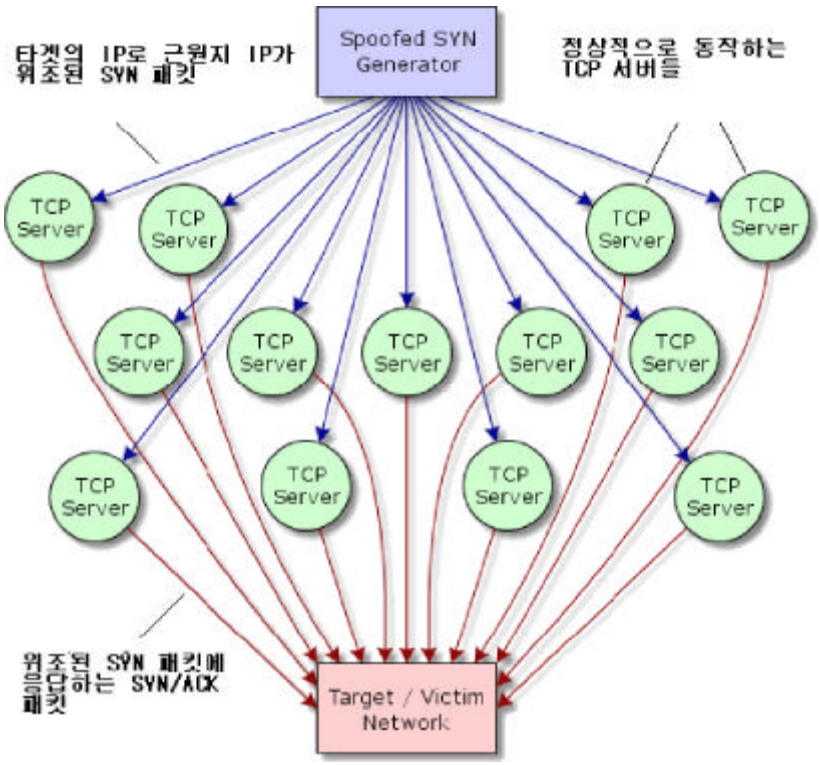
미국의 네트워크 회사인 깁슨리서치(http://grc.com)는 베리오(Verio : Internet Service Provider)에서 T1을 통해 인터넷 서비스를 받고 있다. 2002.1.11 AM 2:00에 인터넷 공격이 시작되어 깁슨리서치의 웹서버는 서비스를 할 수 없는 상황에 처하게 되었다.

깁슨리서치 웹서버로 흘러드는 패킷을 분석하면 Source IP가 Verio.net Qwest.net, Above.net 네트워크에 속하는 주소로 나타났다. 이 패킷은 근원지 포트가 179이고 정상적인 SYN/ACK 패킷이다. 웹서버가 웹서비스 요청에 대해 포트 80으로 응답하는 것처럼 이 SYN/ACK 패킷도 BGP포트 179으로부터 나오고 있었다.

Network	Source IP	Machine Name
verio.net	129.250.28.1	ge-6-2-0.r03.sttlwa01.us.bb.verio.net
	129.250.28.3	ge-1-0-0.a07.sttlwa01.us.ra.verio.net
	129.250.28.20	ge-0-1-0.a12.sttlwa01.us.ra.verio.net
	129.250.28.33	ge-0-0-0.r00.bcrtfl01.us.bb.verio.net
	----------	----------
	----------	----------
	129.250.28.254	ge-2-1-0.r01.snjsca0-3.us.bb.verio.net
qwest.net	205.171.31.1	iah-core-01.inet.qwest.net
	205.171.31.2	iah-core-01.inet.qwest.net
	205.171.31.5	iah-core-01.inet.qwest.net
	205.171.31.6	iah-core-01.inet.qwest.net
	----------	----------
	----------	----------
	205.171.31.81	iah-core-01.inet.qwest.net
above.net	208.184.232.13	core1-atl4-oc48-2.atl2.above.net
	208.184.232.13	core2-atl4-oc48.atl2.above.net
	208.184.232.13	core1-atl4-oc48.atl2.above.net
	208.184.232.13	core2-core1-oc48.atl2.above.net
	----------	----------
	----------	----------
	216.200.127.226	dfw2-atl2-oc48.dfw2.above.net

[표 1] 200개 이상의 라우터로부터 공격을 당하고 있다.

라우터는 이웃라우터와 라우팅테이블을 교환하기 위해 BGP(Border Gateway Protocol)를 사용한다. 라우팅테이블은 각 라우터가 포워드할 수 있는 IP 범위에 대한 정보를 가지고 있다. 라우터(Intermediate router)는 포트 179에 대한 TCP 접속 요청을 받아들일 것이다. 다시 말하면, 라우터는 포트 179로 들어오는 SYN 패킷에 대해 SYN/ACK 패킷으로 응답할 것이다.



[그림 8] DRDoS 공격

이런 수백 개의 라우터에는 어떤 Zombie 프로그램도 설치되어 있지 않고 정상적인 서비스를 수행하고 있었다. 즉, 이 라우터(BGP 서버)들은 grc.com이 서비스 포트 179로 TCP 연결을 요청한다고 판단하여 grc.com으로 SYN/ACK 패킷을 보내고 있었던 것이다.

즉, 악의의 해커는 SYN 패킷을 보냄으로 TCP 연결을 요청하려고 한다. 이 SYN 패킷의 source IP는 grc.com의 IP로 위조되어 있다. 그러므로 라우터는 SYN 패킷이 grc.com으로 온 것이라고 믿고 SYN/ACK 패킷으로 응답을 하게 되는 것이다.

※ 악의의 SYN 패킷들은 정상의 TCP 서버에 의해 SYN/ACK 패킷으로 반사되고 grc.com의 대역폭을 공격하는 데 사용되고 있었다.

o **BGP 서비스 포트 179 차단**

grc.com은 ISP가 아니기 때문에 원격 BGP를 쓰는 라우터에 접속할 필요가 없다. 그러므로, grc.com의 관리자는 BGP 서비스 포트 179로부터 생성되는 인바운드 트래픽을 막아 줄 것을 요청했다. 해커의 SYN 패킷 목적지 포트가 라우터의 포트 179번으로 되어 있기 때문에, 반사된 패킷도 이 포트로부터 생성이 될 것이다.

grc.com에 서비스를 제공하고 있는 라우터에 필터링을 추가했다. 라우터의 서비스 포트 179에서 생성된 패킷을 차단하는 것이다.

o **서비스 포트(22,23,53,80)와 클라이언트 포트(4001,6668)의 SYN/ACK 패킷**

포트 179번으로부터 유입되는 패킷을 차단하고 나서도 새로운 SYN/ACK 패킷이 흘러 들어오고 있음이 발견되었다. 이 패킷들은 BGP 트래픽이 아니라, 포트 22(Secure Shell), 23(Telnet), 53(DNS), 80(HTTP)에서 유입되는 것들이었다. 또한 포트 4001(aprosy server port), 6668(IRC chat)로부터 흘러드는 패킷들도 있었다.

HTTP 포트 80으로부터 유입된 SYN/ACK 트래픽의 다음과 같다.

Source IP	Machine Name
64.152. 4. 80	www.wwfsuperstars.com
128.121.223.161	veriowebsites.com
131.103.248.119	www.cc.rapidsite.net
164.109. 18.251	whalenstoddard.com
171. 64. 14.238	www4.Stanford.EDU
205.205.134. 1	shell1.novalinktech.net
206.222.179.216	forsale.txic.net
208. 47.125. 33	gary7.nsa.gov
216. 34. 13.245	channelserver.namezero.com
216.111.239.132	www.jeah.net
216.115.102. 75	w3.snv.yahoo.com
216.115.102. 76	w4.snv.yahoo.com
216.115.102. 77	w5.snv.yahoo.com
216.115.102. 78	w6.snv.yahoo.com
216.115.102. 79	w7.snv.yahoo.com
216.115.102. 80	w8.snv.yahoo.com
216.115.102. 82	w10.snv.yahoo.com

[표 2] 반사공격에 이용된 웹서버

포트 80 으로부터 유입된 SYN/ACK 패킷의 리스트가 부분적이지만 대체적으로 잘 알려진 서버들이다. 이 서버들은 대개 흩어져 있고, 패킷반사공격(packet reflection attack)에 사용하기 위해 알려지거나 있음직한 웹서버의 IP 주소를 모아 둔 것이다. 인터넷 라우터와는 별

개인 TCP 서버를 이용한 것을 보면 해커가 어떤 TCP 서버가 패킷반사공격에 사용될 수 있는 지를 잘 알고 있다는 것이다.

※ 공격이 끝날 때까지 베리오 라우터는 10억개(1,072,519,399)의 악의적인 SYN/ACK 패킷을 버려야 했다.

나. 반사공격(reflection attack)

여러 가지 수집된 증거를 바탕으로 판단을 해보면 인터넷 서버와 관련된 TCP 서비스 포트 번호가 잘 정리된 목록이 축적되어 왔다. 이 목록에는 BGP(포트 179) 프로토콜을 사용하는 수백 대의 라우터가 있고 일반적인 포트에 서비스(SSH, TELNET, DNS, HTTP, IRC)를 하고 있는 수많은 서버들이 포함되어 있다.

● **반사서버 목록(Reflection server list)**

인터넷에서 접속이 가능한 TCP 서버들의 목록을 "**반사서버 목록**"이라고 한다. 이 목록은 해커에 의해 쉽게 생성, 추가, 유지될 수 있다.

예를 들어 이 목록을 유지하고 추가하는 방법은 아래와 같은 것들이 있다.

○ "Trace Router" command
이 명령은 원격의 서버와 tracer사이의 경유하는 인터넷 라우터의 IP 주소를 모두 보여준다.

○ 간단한 스크립트가 임의의 인터넷 라우터 IP를 수집하는데 사용될 수 있다.

○ 포트 스캔
'Yahoo.com" 같은 사이트에는 이용 가능한 웹서버가 많이 있다. 간단한 포트 스캔을 통하여 공개적으로 이용 가능한 TCP 서버를 찾을 수 있다.

○ 인터넷 검색 엔진
인터넷 검색 엔진을 사용하면 잠재적으로 사용이 가능한 수많은 웹사이트의 도메인 주소를 찾을 수 있고 분류할 수 있다.

이러한 반사서버 목록은 끊임없는 확인 과정을 거쳐 유지된다. 유효한 SYN 패킷을 보내 그 서버가 SYN/ACK로 응답을 하게 되면 그 서버가 네트워크상에 존재하고 있다는 것을 확인하게 되고 무의식적으로 반사공격에 사용될 수 있다는 것을 의미한다.

● **반사서버목록 사용(Using the reflection server list)**

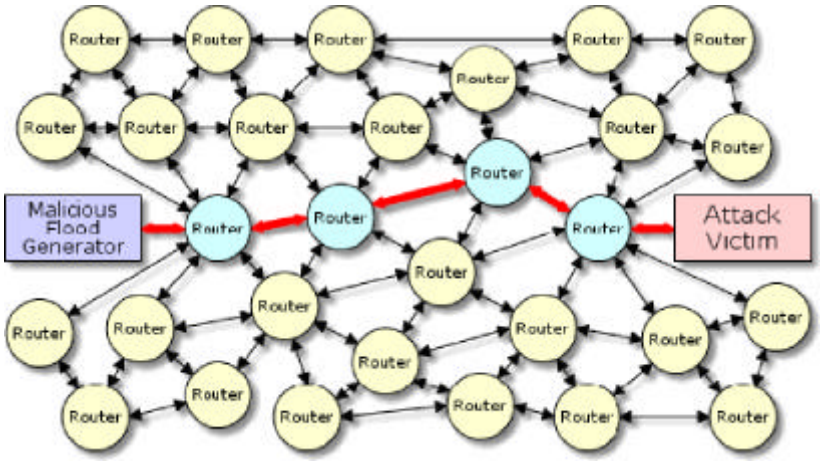
Raw Socket이 가능한 호스트(Unix, Linux, Windows 2000/XP)는 반사공격에 사용되는 SYN 패킷을 쉽게 생성할 수 있다. 공격을 위해 사용되는 서버의 선택은 타겟서버나 네트워크를 충분히 다운시킬 수 있는 대역폭(total flooding bandwidth)에 따라 결정 될 것이다.

각 SYN 패킷을 위조하는 공격 서버는 반사서버목록에 있는 서버에게 SYN 패킷을 보낸다. 위조된 SYN 패킷은 각 서버의 TCP 포트에서 타겟 네트워크나 호스트로 향하는 반사패킷(SYN/ACK)을 생성하게 할 것이다. 또한, 반사공격에 사용된 서버들은 자신이 불법적으로 공격에 이용되고 있다는 사실을 깨닫지 못하고 단지 미세한 "SYN 유동(SYN Flux)"만을 감지하게 될 것이다.

다. 위협이 되는 이유

타겟서버를 직접 공격하기 위해 악의의 호스트를 확보해야 하는 DDoS 공격보다 반사공격이 나은 이유는 아래와 같다.

① 패킷경로확산(packet path diffusion)



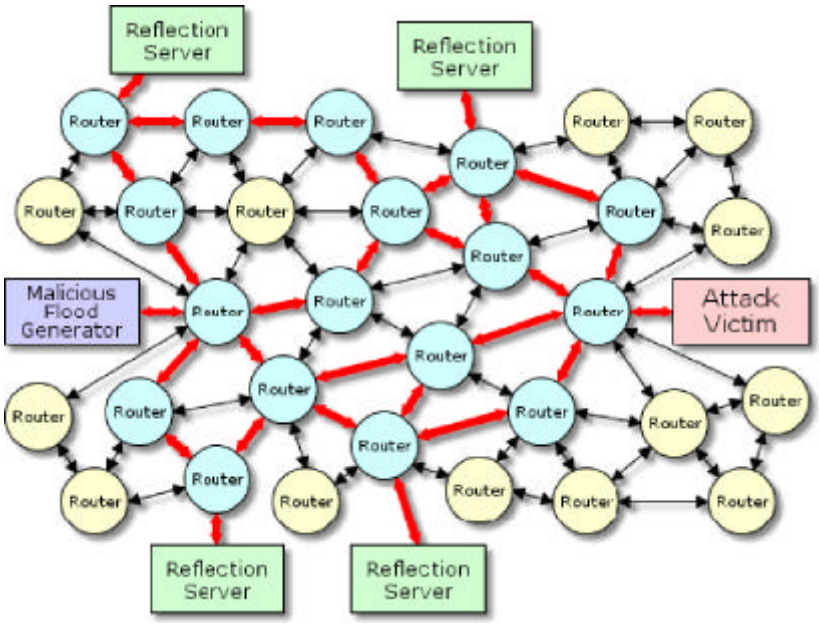
[그림 9] 단일경로

공격트래픽이 수많은 TCP 서버를 경유할 때 "패킷경로확산"의 정도가 높아야 공격자에게 유리하다. [그림 9]는 단일 해커와 타겟 사이의 트랙픽 경로를 나타낸다.

인터넷은 본질적으로 개개의 호스트와 라우터가 연결된 큰 네트워크이다. 인터넷 종단간의 특정 패킷경로는 혼잡한 라우터나 여러 가지 요인 때문에 변할 수도 있다. 그러나, 대부분의 패킷들은 거의 변하지 않는 "최상의 경로(best path)"를 통해 움직이게 될 것이다.

인터넷 라우터가 이전에 라우팅 된 패킷의 정보를 보관하고 있지 않기 때문에, 타겟으로부터 공격자까지 공격에 대한 "역추적(backtracking)"을 하는 것은 하나의 라우터로부터 그 전 라우터에 이르는 업스트림 패킷 흐름(flood)을 수동적으로 따라 갈 수 있는 지에 달려 있다.

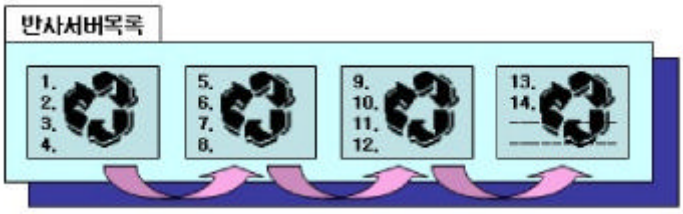
패킷의 흐름을 명백하게 알 수 있음에도 불구하고 패킷 경로를 역추적 하는 것은 시간이 많이 걸리고 수동적인 처리과정이 필요하고 매우 어렵다. 인터넷에 널리 퍼진 많은 수의 패킷반사 서버가 공격시스템에 추가되어질 때 어떤 일이 생기게 될까?



[그림 10] 경로확산

[그림 10]에서 정상적인 반사서버의 추가는 공격의 형태를 변화시키는 것을 보여준다. 악의적인 SYN 패킷은 공격하는 서버에서 떠나자마자 즉시 흩어지게 된다. 직접 타겟으로 향하는 것이 아니라 대신 넓게 퍼진 TCP 반사서버로 보내지게 되고 이 서버들은 인터넷 곳곳에 흩어져 있다. 공격지로부터 단지 몇 "경로 홉수(router hops)"를 제외하고는 하나의 경로가 아니라 여러 이웃 라우터로 패킷이 산개되기 때문에 대량의 패킷 흐름을 인식할 수 없을 것이다. 타겟서버는 수많은 정상적인 TCP 서버의 SYN/ACK 패킷으로부터 공격을 당하게 된다. 각각의 SYN/ACK 패킷 흐름 자체만으로는 타겟에 피해를 주지 않지만 인터넷 도처에서 도착하는 SYN/ACK 패킷의 집중인 공격으로 타겟 시스템이 충분히 다운될 수 있다. 즉, 잠재하는 수많은 서버로부터 원치 않는 공격에 직면한 타겟서버나 네트워크는 심각한 문제에 빠지게 된다.

② 반사서버의 단계적인 사용(Reflector usage phasing)



[그림 11] 반사서버목록

앞 절에서 거대한 인터넷 반사서버 목록을 얼마나 쉽게 얻고 관리할 수 있는지를 알아보았다. 충분히 거대한 반사서버 목록을 가지고 있다면 전체 반사서버 목록의 어떤 일부분을 순환하여 사용함으로서 타겟 네트워크에 집중적으로 지속적인 트래픽을 보낼 수 있다. 즉, 공격 중 어느 시점에도 전체 반사서버 목록 중 단지 작은 일부만 사용되고 있는 것이다.

이 반사서버 목록의 일부분을 단계적으로 사용하는 것은 많은 공격의 경로를 끊임없이 변경할 수 있게 하는 효과가 있다. 공격을 역추적 하는 방법은 수동적으로 처리해야 하는 성격(manual nature)이 강하기 때문에 각 반사서버의 미미한 패킷 흐름을 추적하는 것은 어렵다. 사실 현재 라우터 기술로는 거의 불가능하다.

③ 반사서버 확산(Reflector diffusion)

큰 반사서버 목록을 사용해서 얻게 되는 또 다른 결과는 반사서버 확산(reflector diffusion)이다. 공격자가 SYN 패킷을 수많은 반사서버에 흩어 전송하게 되면 SYN 패킷을 받아서 문제가 발생하는 반사서버(single reflection server)는 거의 없을 것이다. 불완전 연결(half-open connection)은 몇 분 후에 종료되기 때문에 불완전 연결이 결코 한계점을 넘을 만큼 생기지는 않는다. 서버는 이러한 연결을 클라이언트가 중단한 연결(서버와 클라이언트 사이에 종종 생긴다)로 간주하기 때문에 대부분의 반사서버에서는 아마 어떤 경고도 하지 않을 것이다. 주의 깊은 관리자는 불완전 연결에 대해 관심을 가질 수도 있지만, 서버에 어떤 피해도 주지 않기 때문에 아마 개의치 않을 것이다.

④ 대역폭 증대(bandwidth multiplication)

TCP 반사 공격의 중요한 특징은 반사 서버는 받아들이는 SYN 패킷 보다 몇 배 더 많은 SYN/ACK 공격 패킷을 타겟서버로 보내게 된다는 것이다. 반사서버는 SYN/ACK 패킷에 대한 응답이 없을 경우 재전송을 하기 때문에 자연스럽게 SYN 패킷 보다 많게 된다.

TCP 반사서버는 하나의 SYN 패킷에 대해 보통 4개의 SYN/ACK 패킷을 보내게 될 것이다. 이렇게 SYN/ACK 패킷을 재 전송하게 되는 이유는 라우터가 많은 트래픽을 처리할 수 없게 되어 패킷을 버리기 때문이다. 반사서버는 SYN/ACK 패킷에 대한 응답을 받지 못하면 전송도중에 패킷이 사라졌다고 인식할 것이다. 그리고 사라진 SYN/ACK 패킷을 몇 번 더 재 전송하게 된다.(보통 3번 시도)

⑤ 개선된 관리(Improved manageability)

전통적인 DDoS 공격들은 아래 두 가지 목적 때문에 공격서버들로 구성된 큰 네트워크(Large networks of attacking machines)를 이용했다.

- 큰 공격 패킷 흐름
- 공격의 근원지를 분산(diffuse the source of their attack)

주어진 타겟에 대해 "대역폭 증대"와 "경로 확산"을 조합함으로서 다음 두 가지의 이점이 생긴다.

- 대역폭 증대로 인해 공격에 필요한 호스트의 수를 줄일 수 있다.
- 인터넷으로 경로가 확산되기 때문에 경로 역추적이 어려워진다.

이런 특징을 가진 분산반사공격의 등장으로 인해 공격에 사용되는 TCP 서버를 찾고 관리하는 일이 좀 더 쉬워졌고 공격의 근원지를 밝혀내는 것도 어려웠다.

⑥ 스텔스(Stealth)

반사공격에 사용되는 전통적인 위조 SYN 패킷 흐름(traditional spoofed source IP floods)은 모니터링 스크린에 잡히지 않을 것이다. 즉, 전통적인 위장된 SYN 패킷의 근원지 IP는 랜덤으로 생성된다. 이런 전통적인 위조된 근원지 IP 공격과는 달리 반사 공격에서는 타겟 서버의 IP로 근원지 IP를 위조하게 된다. 결과적으로, 이 공격을 밝혀내기란 쉽지 않다. (하지만, "reflection honeypots"을 구성해서 반사공격을 모니터 할 수 있다.)

위의 여러 가지 이유로 인해, DRDoS 공격은 쉽게 시도할 수 있고 유지 또한 쉽다. 반면, 공격을 역추적 하는 것은 거의 불가능하다.

라. 반사공격 방어 및 예방

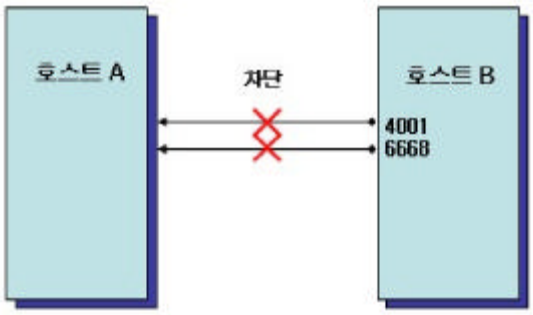
인터넷상의 서버는 클라이언트와 서버의 역할로 나누어져 있다. 이 역할은 호스트의 사용 목적에 따라 바뀔 수도 있다. 그러나, 대부분의 TCP 연결은 일반적으로 클라이언트 서버관계를 포함하고 있다. 클라이언트는 주로 클라이언트 포트(1024-65535)에서 서버로 연결을 시도한다. 서버는 서비스 포트(1-1023)에서 클라이언트의 서비스 요청을 기다리고 있다.

SYN/ACK 패킷들은 TCP 서버를 통해 반사되고 거의 모든 서비스포트는 1-1023의 범위에 다 있다. 서비스 포트에서 생성되는 인바운드(Inbound) 패킷들을 모두 차단하게 되면 반사서버에서 생성된 정상적인 트래픽까지 대부분 차단하게 될 것이다.

① 서버 보호(Protecting a server)

보호해야 하는 대상 시스템이 서비스를 제공하고 있는 서버일 경우 반사공격에 대한 방어 방법과 그에 대한 문제점들은 다음과 같다.

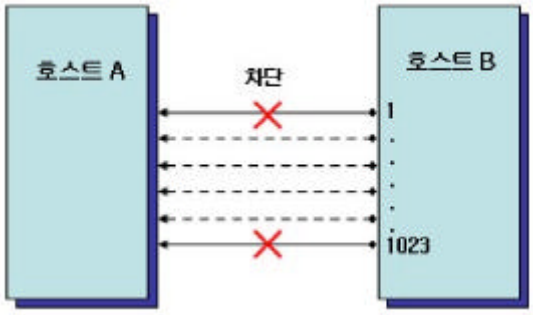
○ 클라이언트 포트 차단



[그림 12] 클라이언트 포트 차단

grc.com은 반사서버의 4001과 6668 포트에서 생성된 SYN/ACK 패킷 흐름에 의해서도 공격을 당하였다. 이처럼, 서비스 포트(1- 1023) 이외의 포트에서 생성된 SYN/ACK 트래픽이 많이 들어온다면 또한 차단해야 할 것이다. 서비스포트로부터 생성된 인바운드(Inbound) 트래픽을 차단할 때 문제는 호스트 A의 정상적인 클라이언트 호스트 B가 이 차단된 포트(4001, 6668)를 통해 연결을 시도할 수도 있다는 것이다. 이런 반사공격필터링은 정상적인 클라이언트 연결 요청까지도 차단하게 될 것이다.

○ 서비스 포트 차단



[그림 13] 서비스 포트 차단

호스트 A가 호스트 B의 서비스포트(1- 1023)로부터 생성된 인바운드 트래픽을 전체적으로 차단하게 될 경우 호스트 A는 호스트 B의 클라이언로 동작할 수 없게 된다.

예를 들면, SMTP(eMail) 서버의 클라이언트로 동작하는 웹서버는 SMTP 서버의 포트 25번으로부터 응답을 기다리게 될 것이다. 하지만, 반사공격필터에 의해 SMTP 서버의 포트 25로부터 생성된 인바운드 트래픽은 모두 차단 된다. 이 문제를 해결하기 위해서 신뢰(friendly)하는 서버의 트래픽은 반사공격필터를 통과할 수 있도록 해야 한다.

② 클라이언트 보호(Protecting a client)

어떤 TCP 서비스도 제공하지 않는 종단사용자(end user) 같은 경우, 막을 수 없다. 대부분의 클라이언트는 인터넷의 원격서버(무의식적으로 클라이언트를 공격하고 있을지도 모르는 서버)에 연결하기 위해 시간을 다 소비하게 되고 대부분 서버의 서비스포트(low-numbered service ports)에 접근을 요청하게 된다.

결과적으로 프락시나 ISP NAT을 이용하여 종단사용자를 보호할 수 있다.

③ 반사서버 악용 막기(Preventing reflection server exploitation)

반사공격의 실정을 이해하고 있는 관리자는 자신의 서버가 반사공격에 악용되고 있다는 것을 알 수도 있다. 또한, 자신의 서버가 반사공격에 악용되는 것을 막을 방법에 대해 궁금해 할 것이다.
이 방법은 이론적으로는 간단하다. 이것을 하기 위해서는 절대 연결을 완료할 수 없는 근원지 IP를 가지고 있는 SYN 패킷을 인식해야 한다. 짧은 시간에 일어나는 연결 실패는 일상적인 현상이 아니기 때문에, 반사공격의 타겟이 정해졌을 수도 있다. 반사공격의 동적인 방어(Dynamic reflection attack prevention)는 연결을 완료할 수 없는 IP로부터 도착하는 SYN 패킷에 대한 블랙리스트(black list)를 작성하는 것이다.

반사서버 악용 방어 시스템은 서버에 있는 방화벽(Firewall)에 쉽게 구축할 수 있다. 그러나, 인터넷상의 모든 서버에 이런 방화벽이 설치되어 있을 수는 없다. 위장패킷을 네트워크 Egress filtering을 할 수 있도록 ISP 업체에 요청을 하는 것이 좀 더 현실적인 방법이다.

④ ISP 책임 (The ISP's responsibility)

반사공격에 이용되는 트래픽 생성은 근원지 IP 위조에 달려 있다. ISP 업체가 그들의 통제하에 있는 네트워크로 유입되는 위조된 패킷을 차단할 수 있다면 이 공격은 막을 수 있을 것이다. 악의의 해커들이 Egress filtering 된 ISP 네트워크에서는 그들의 공격방법이 제 역할을 하지 못할 것이기 때문에 ISP는 그들의 고객을 위한 보안을 향상시킬 수 있다.

⑤ 공격에 사용되는 플랫폼의 책임(The attacking platform's responsibility)

데스크탑 PC에 raw socket API를 포함시키는 것은 신중해야 한다. MS가 윈도우 2000/XP에 raw socket을 포함시킨 것은 잘못된 일이라고 사람들은 말하고 있다.

MS에서는 악의적인 인터넷 트래픽을 생성하는 다른 방법들이 있다고 말하고 있지만 사실 Raw socket을 사용하는 것보다 더 쉬운 방법은 없다. 모든 윈도우 2000/XP에 Raw socket 기능을 포함시키는 것은 결과적으로 악의적인 반사공격 생성기를 쉽게 만들 수 있도록 한다. 이미 늦은 감이 있지만 MS가 패치와 업그레이드를 통하여 윈도우 2000/XP에서 raw socket을 제거할 수 있도록 해야 한다.

7. 결론

대부분의 인터넷 공격은 인터넷 프로토콜이나 많은 기반 구성요소의 동작상의 취약점을 이용한다. 결과적으로, 반사공격은 마음만 먹으면 언제나 누구나 쉽게 가능하다. DRDoS 공격은 공격방법을 파악할 방법이 없고 그 피해도 크기 때문에 앞으로 이를 방어하고 해결하기 위한 연구개발이 시급하다.

8. 참고문헌

[1] Steve Gibson, "DRDoS(Distributed Reflection Denial of Service)" February 2002

[2] 한국정보보호진흥원, "DDoS 공격자 근원지 분석기술 연구 및 프로토타입 구현", 2001,11

[3] Tom Vogt, "Application-Level Reflection Attacks", May 2002