네트워크 시뮬레이터를 이용한 DDoS공격의 효율적인 완화시스템 설계

최향창*, 채명훈*, 이형효**, 노봉남*

전남대학교 정보보호협동과정*
원광대학교 정보·전자상거래학부**

Design of an Efficient Mitigation System for DDoS Attacks using a Network Simulator

Hyang-chang Choi*, Myung-hun Chae*, Hyung-hyo Lee**, Bong-nam Noh*

*Interdisciplinary Program of Information Security, Chonnam National Univ.
**Division of Information and Electronic Commerce, Wonkwang Univ.

요 약

현재의 네트워크 공격 중 대부분은 특정시스템의 서비스를 거부하도록 하여 서비스에 대한 가용성을 제공하지 못하도록 하는 공격이다. 이러한 공격의 유형은 DoS로부터 시작해서 현재에는 DDoS, DRDoS등의 보다 지능화된 새로운 공격으로 발전하고 있다. 이러한 공격은 탐지 자체도 어려울 뿐만 아니라 탐지하더라도 이에 대응하기 위한 방법 또한 어려운 것이 현실이다.

본 논문에서는 시뮬레이션 도구를 이용해서 보호하고자 하는 네트워크를 가상으로 구성하고 서비스공격 행위의 패턴들을 분석, 적용함으로써 통합보안관제 시스템에서 최적의 서비스 거부 공격 완화 방법을 설계하는 시스템을 제안한다.

I. 서론

네트워크상에서 행해지는 서비스 거부 공격에는 일반적으로 DoS, DDoS, DRDoS등의 공격이 있다.

DDoS 공격은 Trinoo, TFN, TFN2K등의 공격 방법이 있다[7, 8, 9]. 이들의 목표는 네트워크상에 이루어지는 특정서비스의 정상동작을 방해함으로서 합법적인 사용자가 제대로 된 서비스를 받지 못하도록 하는데 있다. 또한 취약점을 가지고 있는 시스템들을 공격에 동원하기 때문에 탐지를 더욱 어렵게 한다. 예를 들면, 공격에 사용되는 Master가 공격을 수행하는 Agent를 간단히 제어하여, IDS 탐지가 어렵도록 불규칙적인 DoS공격을 수행한다. 또한 하나의 공격에 여러 호스트가 사용되기 때문에 공격에 대한 방어 또한 매우 어렵다. 이외에 서비스 거부 공격의 일종인 웜은 네트워크 트래픽을 증가시켜 네트워크의 정상적인 서비스에 대한 장애를 유발한다. 이러한 웜은 인터넷상에서 취약한 시스템에 쉽게 복제되기 때문에 서비스거부 공격의 좋은 도구가 된다.

이러한 공격들은 각각에 대한 원인과 결과의 분석이 힘들다는 어려움이 있다. 따라서 공격 방어를 위한 방법들을, 보호하고자 하는 네트워크에 최적화되도록 설계하지 않는다면 오히려 네트워크에 악영향을 미칠 수 있다[1].

본 논문은 이러한 문제점을 해결하기 위해, 보호하고자 하는 네트워크에 적합한 DDoS 방어를 적용하기위해 가능한 방법에 대해 제안한다.

II장에서는 DDoS공격의 완화방법과 NS에 대해서 알아본다. III장에서는 DDoS공격 패턴 분석

을 통하여 보호 네트워크에 최적화된 공격 완화 기법을 도출하는 시스템에 대한 설계, Ⅳ장은 문제점 및 해결전략을 살펴보고, 끝으로 결론 및 향후연구를 제시한다.

II. 관련연구

서비스 거부 공격은 그 형태를 다양하게 변형시키면서 현재에도 수없이 많이 발생한다. 이장에서는 이러한 공격을 완화하기 위한 전략들에 대해 살펴보고, 이 완화 전략을 테스트하기 위한 네트워크 시뮬레이터[5, 6]에 대해서 알아본다.

1. DDos 공격 완화방법

공격 방어를 위한 완화 전략은 크게 그림 1과 같은 종류가 있을 수 있다[1].

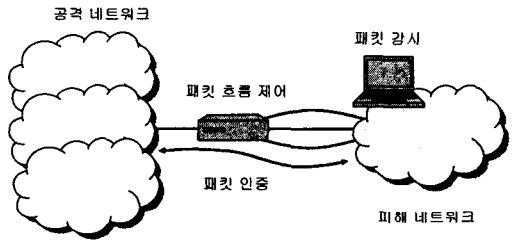


그림 1: DDoS 공격 완화방법

먼저 패킷 감시는 실시간으로 오는 패킷을 감시하고 공격에 관련된 패킷을 필터링하여 서비스 거부공격을 막아낸다. 하지만 이것은 보호하고자 하는 네트워크의 패킷을 실시간으로 계속 감시해야 하고, 또한 정상적인 패킷을 공격 행위로 간주하여 정상적인 서비스에 장애의 요인이 되기도 한다.

두 번째로 패킷흐름 분산은 피해시스템으로 집중되는 패킷을 다른 곳으로 분산함으로써 트래픽이 가중되는 것을 완화시키는 기술이다. 예를 들어 침입 감내 시스템에서 사용자가 특정 서비스를 부여받을 때 해당 서비스에 요구가 많으면 이와 같은 서비스를 수행할 수 있는 다른 시스템에 패킷을 중계하여 트래픽을 분산하도록 하는 것이다[4]. 또 다른 예는 어느 특정한 라우터가 많은 패킷의 집중으로 인해 라우팅 기능이 저하될 때 주변의 라우터로 패킷을 분산시켜 공격 받는 라우터의 수행능력이 떨어지는 것을 방지하는 방법이다.

세 번째로 서버가 클라이언트를 인증하는 메커니즘이다. 이 방법은 특정 서버가 DDoS공격 표적이 될 수 있는 주요한 시스템일때 쓰이는 방법으로 클라이언트는 서버에 서비스를 요청하기 이전에 패킷을 보내도 좋다는 인증을 받은 후 패킷을 전송할 수 있도록 하여 불법적인 사용자에 의한 패킷의 요구를 차단한다.

2. 네트워크 시뮬레이터(NS)

NS는 UC Berkely에서 C++과 OTcl을 기반으로 개발한 객체지향(Object-Orient) 이산사건 구동 시뮬레이터로서 LAN과 WAN, 그리고 다양한 TCP/IP 네트워크를 시뮬레이션 하는데 매우 유용하다.

NS의 특징은 표 1과 같다

표 1: NS의 특징

기반 언어	tcl, otcl, c++
지원 플랫폼	FreeBSD, Linux, Windows, SunOS/Solaris, Mac, HP, SGI
그래픽 환경	NAM(Network Animator)
trace 분석	Awk, Perl, Xgraph, Tcl
개발	UC Berkeley
특징	Event 구동 네트워크 시뮬레이터
구현	TCP/IP 네트워크 프로토콜 - FTP, Telnet, Web, CBR & VBR Router queue 관리 메카니즘 - Drop Tail, RED, CBQ Routing 알고리즘 - Dijkstra
구성	- 시뮬레이션 event scheduler - network 구성 object libraries - network setup - module libraries

본 논문에서 사용하는 네트워크 시뮬레이터는 NS-2로서 다양한 인터넷 프로토콜에 대한 시뮬레이션을 수행하기에 적절한 여러 환경을 제공한다.

III. 시스템 설계

서비스 거부공격에 노출되어있는 내부 네트워크에서 공격 완화 방법을 적용할 네트워크인 보호네트워크와 이러한 공격을 방어하기 위한 방법을 제시한다.

1. 동작 구조 및 방법

서로 같은 형태의 공격이라도 그 결과는 각각의 시스템의 성능이나 외부적인 요인에 따라 많은 차이가 발생한다. 따라서 관리대상이 되는 네트워크 상황을 분석 유지하여 각 공격의 상황에 효율적으로 대처하기 위한 새로운 방법이 필요하다. 여기서는 관리 하에 있는 네트워크를 보호대상 네트워크로 하고 표2와 같은 정보를 유지한다.

표 2: 보호대상 네트워크 유지정보

Topology	네트워크 구성요소들의 위상
Bandwith	각 Router의 처리 율
Vulnerability	각 시스템의 취약점
Services	각 시스템 별 제공 서비스
허용 IP	인가된 호스트 IP

서비스 거부 공격에 대한 최적의 공격방어를 위한 단계는 그림 2와 같이 크게 3단계로 나뉜다.

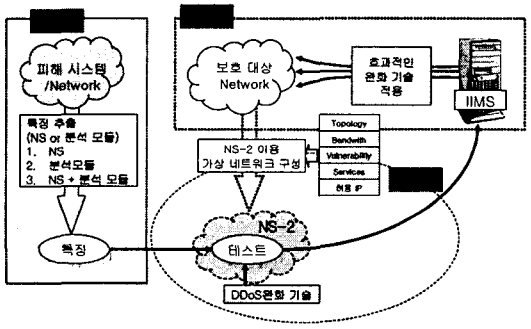


그림 2: 최적의 공격방어를 위한 세부전략

단계1은 피해를 입은 시스템을 기반으로 서비스 거부 공격의 특징을 추출한다. 실제 패킷을 NS에서 재현할 수 있는 모듈과, 통계적 기반으로 피해시스템의 패킷유형과, 흐름을 분석하는 모듈을 수행하고 이들의 결과를 조합해 봄으로서 공격특징을 추출한다.

단계 2는 테스트 단계로서, 표 2의 각각의 요소들을 정량화 한 후 그림 2의 단계 2에서 보호대상 네트워크 상황을 시뮬레이션 한다. 네트워크에 피해시스템의 특징을 적용해 시뮬레이션 하는 동안 DDoS완화 방법을 적용하여 그 결과를 분석한다. 이후 얻을 결과를 정보 통합관리시스템(Integration IDS Management System)[10]에 보고한다.

단계3은 IIMS에 의해서 공격에 따른 최적의 완화 정책을 보호대상네트워크에 적용한다.

이러한 3단계의 보호방법설계에 의해 최적의 상황을 보호대상 네트워크에 적용함으로서 공격에 따른 최적의 방어시스템을 제공한다.

2. 보호네트워크 시뮬레이터

시뮬레이터로 DDoS 공격의 특징을 추출하고, 완화정책을 적용, 테스트해 보기위해 피해시스템으로부터 얻어낸 Tcpdump와, 표2 와 같은 네트워크 정보를 이용한다.

NS-2의 시뮬레이션 재현 도구인 NAM은 trace file인 .nam을 통해 패킷흐름을 시각적으로 표현한다. .nam 파일의 각 요소는 그림 3과 같다.

```
h -t 0 -s 2 -d 10 -p tcp -e 40 -c 1 -i 0 -a 1 -x {0.0.3.0 0.0.2.0 0 -------- null}
-t <time> time
-s <int> source id
-d <int> destination id
-e <int> extent
-a <int> attribute
-i <int> id
-l <int> energy
-c <string> conversation
-x <comment> comment
-p <string> packet type
-k <string> packet type
-R <double> wireless broadcast radius
-D <double> wireless broadcast duration
```

그림 3: .nam파일 분석

이들의 각 요소는 Tcpdump의 패킷헤더를 다음과 같이 대체함으로서 생성될 수 있다.

T : Tcpdump, **N** : NS-2의 .nam파일

- **T**_time -> **N**_Time
- **T**_Source_Ip -> **N**_SourceId
- **T**_ Destination_IP -> **N**_DestinationId
- **T**_Identification -> **N**_Id
- **T**_Port(type) -> **N**_PacketType
- **T**_DataSize -> **N**_Extent

또한 Tcpdump파일을 .nam파일로 변환하는 과정에서 라우팅 경로는 목적지주소에 해당하는 최종의 라우터를 명시한다.

표 2와 같은 보호대상 네트워크의 정보를 통해 시뮬레이션상의 네트워크를 구성할 수 있다. 예를 들어 DARPA 99'의 네트워크를 NS-2에서 구성한 것은 그림 4에서 보여준다.

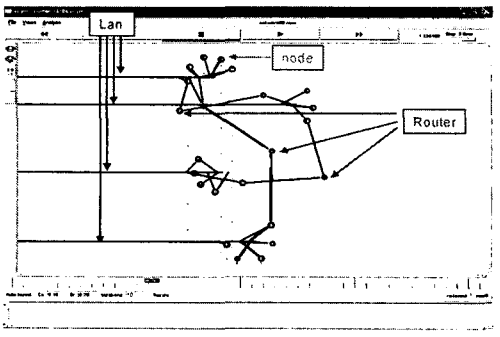
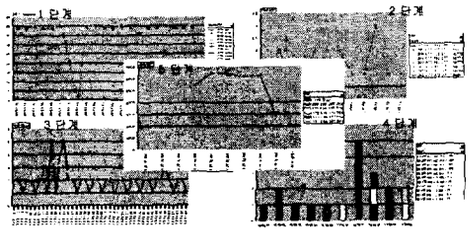


그림 4: DARPA 99' 네트워크 구성 예

위와 같이 구성된 네트워크 상황에 공격재현을 위해 생성된 .nam파일을 적용한다.

3. DDoS공격 분석설계

그림 2의 단계1에서 추출한 피해시스템의 시스템 로그를 기반으로, 공격의 특징을 추출한다. 예를 들어 그림 5는 DARPA 2000년 데이터셋의 Tcpdump 데이터의 단계별 특징을 추출한 것이다.



1단계 : IPSweep - Target IP수의 통계 값 산출
2단계 : Probe - SunRPC/udp : 111
3단계 : Breaking - SunRPC/udp : 111
4단계 : Installing - Telnet : 23, Shell : 514
5단계 : Starting the DDoS - Victim 호스트 IP 통계

그림 5: DARPA 2000년 데이터 셋 분석

그림 5에서 제공되는 기본적인 DDoS특성을 추출하기 위해서는, 표2의 각 내용을 시간에 따라 유지한 후 데이터로 정량화한 후 분석한다. 이렇게 하기 위해서는 표3과 같은 기능이 필요하다.

표 3: DDoS특성 추출을 위한 기본기능

Tcpdump Data	Target Ip별 통계	시간 간격 별 패킷 통계산출
	Source Ip별 통계	
	포트별 통계 (DDoS관련 Port)	
BSM Data	Master, Agent탐색	관련파일 탐색

각각의 기능은 그림 6과같이 동작한다.

그림 6에서 Configuration Files은 각각의 Cluster 모듈에서 수행하는 군집화 알고리즘[2]을 위한 기본 설정과, 군집화 된 결과 중에서 유용한 특성만을 추출하기 위한 판정 값을 설정한다. 또한 BSM 로그에서 DDoS와 관련된 파일을 추출하기 위한 패턴을 기술한다.

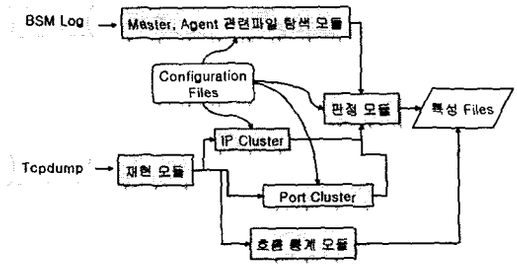


그림 6: 분석모듈 동작

그림 7은 패킷의 흐름을 분석하기 위한 흐름통계모듈의 간단한 예를 보여준다.

Packets	packets	bytes
TCP:	37555	11423536
Pop3 (110):	10	
www:	9063	
nntp (119):	0	
Other:	2653	169837
Tcp streams:		
in progress:	842	
total:	1303	
damaged:	0	
Speed		
current (Mbps):	25.373	
average (Mbps):	29.395	

그림 7: 흐름 통계모듈 실행 예

위와 같은 방법에 의해 분석된 결과는 DDoS공격에 대한특징으로 IIMS에 DDoS공격 패턴의 일부로 저장된다.

4. 최적의 완화기술 적용설계

관련연구에서 제시한 완화기술을 시뮬레이터에 추가하고 이를 통해 보호네트워크를 구성한 후 적용 테스트한다.

앞의 II장 관련연구에서 제시한 DDoS 공격의 완화기술은 네트워크 시뮬레이터에서 재현되는데 방법은 다음과 같다. 먼저 패킷감시는 위조된 패킷을 필터링하는 규칙을 적용한다. 위조된 패킷의 탐지는 다음과 같은 항목과 일치하는 패킷을 탐지하여 차단하는 모듈을 시뮬레이터 환경에 추가함으로써 패킷 감시 기술을 재현할 수 있다[3].

- TTL에 따른 위조된 패킷 탐지
  - 일반적으로 패킷은 두 호스트 사이에 전송될 때, 보통 같은 경로에 한해서 Hop의 수도 같고, 같은 양으로 TTL이 감소된다. 따라서 소스와 목적지 주소가 같으면서 현격하게 TTL값의 차이를 보인다면 위조된 패킷이다.

- IP Identification Number
  - 탐침(Probe) 패킷의 유형으로 판단되어지는 패킷유형이 있을 때, 이 패킷의 Identification Number와 가까운 값을 유지하고 있으면서 소스가 같은 패킷은 공격에 사용된 탐침 패킷이다.

- TCP Specific Methods
  - TCP패킷은 전송간 신뢰를 보증한다. 따라서 송신자의 위조된 패킷은 연결요청에 대해 응답하지 않게 된다. 즉 이러한 패킷은 위조된 패킷이다.

패킷 감시를 통한 완화기술은, 위와 같은 패킷 유형으로 분류되어지는 패킷을 학습시키고, 이와 일치되는 유형의 패킷을 네트워크 시뮬레이션동안 차단하면서 보호된 네트워크의 상황을 확인한다. 하지만 이것으로 인해 정상적인 패킷이 거부되어질 수 있다. 따라서 이를 해결할 수 있는 추가적인 방법은 보호대상 네트워크에서의 정상적인 패킷을 통계적으로 분석하고, 이때 사용되는 IP와 Port를 신뢰호스트로 분류하여 패킷 필터링에 규칙에서 제외한 후, 이외의 패킷에 대해서만 위조된 패킷으로 간주하여 필터링한다.

패킷흐름 분산은 공격의 대상이 되는 시스템으로 흘러가는 패킷을, 같은 서비스를 수행하는 호스트로 분기시킴으로서 패킷 과중을 막는다. 하지만 이 방법만으로는 현실세계의 보호네트워크 시스템자원을 추가로 발생시킬 수 있는 문제가 발생할 수 있다. 따라서 여기서는 실생활에서 응용이 쉽도록 DDoS공격동안 필수 서비스만은 꼭 동작하도록 하는 침입 감내 시스템의 기본기술에 우선을 둔 패킷흐름 분산을 적용한다[4]. 먼저 필수서비스에 대한 우선순위를 유지하고 패킷흐름이 시스템자원에 영향을 미치는 수위까지 올라가면 우선순위가 가장 낮은 패킷들을 같은 서비스를 제공하는 추가된 시스템에 분산시켜 서비스를 수행한다. 즉 이 방법은 보호네트워크의 시스템 자원의 여유와 필수서비스의 범위에 따라서 능동적으로 현실세계의 시스템에 적용 가능하다.

끝으로 서버의 클라이언트 인증방법은 다음과 같다. 서버가 신뢰하는 접속호스트의 IP와 수행가능한 서비스를 유지하고 이외의 패킷은 시뮬레이션 상에서 차단한다. 그런데 만약 서버가 신뢰하는 목록이 없는 네트워크라면 공격이 없는 평시에 서버에 접속하는 호스트와 서비스를 감시하여 이때 출현하는 IP와 서비스를 테이블로 추가해 놓고 이외의 패킷을 시뮬레이터 상에서 차단하는 기능을 수행한다.

5. 효과적인 완화전략 적용

시스템의 전체적인 동작의 순서는 Ⅲ장에서 언급한 그림2와 같이 공격특징 추출(단계1), 보호대상 네트워크 환경구성 및 공격에 따른 최적의 완화를 위한 정책조율(단계2), 공격개시에 대한 방어(단계3)의 순으로 동작한다.

단계3에 해당하는 공격개시에 대한 방어는, 보호대상 네트워크에서 공격이 발생할 때 공격을 감지하고, IIMS는 침입으로 감지한 공격에 대해 Ⅲ장의 4에서 제시한 최적의 완화기술 적용 설계 방법에 의해 최적화로 조율되어지는 효과적인 완화전략을 보호네트워크에 적용한다. 이러한 방법에 의해 각 공격에 대한 효과적인 완화전략은 IIMS를 통해 정책적으로 적용되어진다.

Ⅳ. 문제점 및 해결전략

지금까지 DDoS공격을 효과적으로 방어하기 위한 방법에 대해 제안했다. 하지만 최적의 완화전략을 제공하기 위해서는 다음과 같은 문제점에 대한 고려가 추가적으로 필요하다.

- DDoS공격이 보호대상 네트워크에 발생하였을 때 이를 어떻게 감지할 것인가?
- DDoS의 완화정책을 적용할 시점은 어느 때가 가장 적당한가?
- 현재 표2에 관한 정보만으로 보호대상 네트워크에 대한 DDoS공격에 대한 방어를 제공하는데 추가적으로 필요한 정보데이터는 없는가?
- DDoS공격을 완화시키는 추가적인 효율적인 완화 기술은 없는가?
- IIMS의 영향범위는 공격유형에 따라서 어디까지가 적당하며, 이것으로 인해 발생할 수 있는 추가적인 취약점은 어떤 것들이 있는가?

이러한 문제점을 해결하기위한 접근방법으로는, 각각의 단계를 정형화된 수단으로 네트워크 상황에 대해 모델링하고, 또한 광범위하고 이질적인 데이터 셋을 테스트해봄으로써 그에 따른 결과를 분석하여 시스템의 성능을 개선할 수 있다.

Ⅴ. 결론 및 향후 연구

DDoS공격은 일반적으로 시스템이나 네트워크들을 공격의 목표로 하며, 다른 어떠한 공격의 유형보다도 강력하다. 또한 이러한 공격들은 TCP/IP의 근본적인 프로토콜의 변화 없이는 근본적인 문제의 해결이 어렵다. 더욱 이 공격은 새로운 공격방법으로 전이되어 앞으로도 계속 빈번하게 발생할 수 있다.

이러한 문제점을 해결할 대안으로 새로운 완화 방법들이 개발되고 있다. 하지만 이러한 방법들은, 공격이 발생한 시점에, 각각의 네트워크 상황에 맞게 최적의 대응 전략으로 조율되어야만 최대의 효과를 제공할 수 있다. 또한 이렇게 함으로서 완화정책으로부터 오는 역기능을 해소할 수 있다.

본 논문에서는 시뮬레이터를 이용해 DDoS공격을 분석 및 테스트를 하고, 이에 대한 최적의 DDoS 공격 완화 전략을 보호대상 네트워크에 적용하기위한 방법을 제안했다.

향후 연구방향은 3장에서 논의했던 문제점을 해결할 수 있는 방안을 수행하여, 보호대상네트워크에 보다 최적화된, DDoS공격에 대한 대응방법을 구축하고자 한다.

참고문헌

[1] W. J. Blackert, D. M. Gregg, A. K. Castner, , "Analyzing Interaction Between Distributed Denial of Service Attacks And Mitigation Technologies," *IEEE DARPA Information Survivability Conference and Exposition-Volume1*, April. 2003

[2] Ian H. Witten, Eibe Frank, "Data Mining," Morgan Kaufmann Publishers, 2000

[3] Steven J. Templeton, Karl E. Levitt," Detecting Spoofed Packets," *IEEE DARPA Information Survivability Conference and Exposition-Volume1*, April. 2003

[4] Laura Feinstein, Dan Schnack enberg, Ravindra Balupari, Darrell Kindred, "DDoS Tolerant Networks," *IEEE DARPA Information Survivability Conference and Exposition-Volume1*, April. 2003

[5] Jae Chung, Mark Claypool, "NS by Example," http://nile.wpi.edu/NS/

[6] Kevin Fall, Kannan Varadhan, "The ns Manual," http://www.isi.edu/nsnam/ns/doc/index.html

[7] David Dittrich, "TFN2K - An Analysis," http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt, 1999

[8] David Dittrich, "Tribe Flood Network," http://staff.washington.edu/dittrich/misc/tfn.analysis.txt, 1999

[9] David Dittrich, "The DoS Project's "trinoo - distributed denial of service attack tool," http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt, 1999

[10] 이성호, 박용철, 이형효, 노봉남, "침입정보 통합관리시스템을 위한 테스트베드 구축", 정보처리학회, 5, 2003