# DDoS 공격 완화를 위한 분산 방화벽 모델

방세중*, 이승하*, 김양우*
*동국대학교 정보통신공학과
e-mail: {neovega, lesh915, ywkim}@dongguk.edu

# A Model for the Distributed Firewall to Mitigate Distributed DoS Attacks

Sechung Pang*, SeungHa Lee*, Yang-woo Kim*
*Dept of Information and Communications Engineering,
Dongguk University

요 약

현재 사이버 공간에서 일어나고 있는 정보보호 위반사건들은 이미 실생활에 구체적인 악영향을 미치고 있고 이런 정보보호 관련 이슈사항에 대한 여러 보완 및 개선 방안이 제시되고 있다. 그런데 분산서비스거부공격(DDoS)에 의한 피해규모는 나날이 커지는 반면에 인터넷의 구조적 특성으로 명확한 대응책보다는 조기탐지를 통해 사전에 대응함으로써 피해규모를 줄이거나 공격을 완화시켜 가용성을 확보하는 방법만 고안되고 있다. 그러나 우리는 공격을 완화시켜 주는 추가적인 고가의 네트워크 장비 구축 없이 기존 시스템을 활용한 분산 방화벽 모델을 제안한다. DDoS 공격이 이루어질 때 방화벽의 세션테이블과 간단한 관리기능을 그리드 컴퓨팅 기법 중 하나인 워크릿(worklet)으로 구성하여 방화벽 하위 웹 서버 군(group)에 배포시켜 각각의 웹 서버가 방화벽의 세션테이블 기능을 부분적으로 수행하는 것이다. 이렇게 함으로써 공격이 진행되는 중이라도 기존 구조에서는 할 수 없었던 정당한 인터넷 서비스 요청에 응답할 수 있어 가용성이 증대되는 효과를 얻을 수 있다.

## 1. 서론

인터넷 공간에서 벌어지는 각종 침해사고는 과거 단순한 해킹에서 기존 worm/virus 등의 네트워크 침해사고로 발전하여 현재 Phishing, 명의도용 등으로 변화하면서 웹 서버, DB의 개인정보 유출과 범죄에 이용하는 형태로 변하였다. 그러나 인터넷의 구조적 취약성인 발신자의 익명성과 요청 정당성을 확인할 수 없기에 2000년 초 대규모 트래픽으로 인한 eBay.com, CNN.com, Yahoo.com 및 Amazno.com 등의 사이트가 서비스거부(Denial of Service: DoS) 공격에 마비된 것을 시작으로 전 세계적으로 인터넷 사용에 장애가 발생했던 2003년 1.25인터넷침해사고를 거쳐 지금까지 지속적으로 크고 작은 분산 서비스 거부(Distributed Denial of Service: DDoS) 공격이 사이버 공간에서 벌어지고 있다[1].

DDoS 공격도 다른 인터넷 침해사고와 마찬가지로 초기에 단순한 호기심으로 트래픽 발생기 등을 통한 공격이 주였는데 현재 악성 봇(bot) 등을 이용하여 좀비(zombie) 컴퓨터 등으로 공격자를 지능적으로 감추고 상용 서비스를 제공하는 서버 또는 네트워크를 대상으로 다량의 트래픽을 보냄으로써 서비스를 정지시키고 서비스 재개의 대가로 금전적 요구를 하는 경우로 바뀌었다. 이에 대한 대응방안으로 인터넷 상용 서비스를 제공하는 사업자들 중 일부는 DDoS 공격을 완화시키는 고가의 네트워크 장비를 도입하여 구축하기도 한다. 그러나 하나의 완화 장비가 모든 DDoS 공격에 효율적으로 대응하는 것이 아니기에 각 특성에 맞는 여러 대의 장비를 도입하여 구축할 수밖에 없다. 그렇기에 금전적인 이유로 대부분의 사업자/기관은 여전히 DDoS 공격의 위험에 그대로 노출되어 있는 상태로 웹 서비스를 포함한 여러 인터넷 서비스를 운영하고 실정이다.

그래서 우리는 현재 운영 중인 방화벽과 그 하단에 운영 중인 서버군(group), 대표적으로 웹 서버 군의 구성에 추가적인 공격완화 네트워크 장비 없이 DDoS 공격을 완화하여 효과적으로 서비스 가용성을 확보할 수 있는 모델을 제안한다. 분산서비스거부공격이 특정 서버 군 또는 네트워크 영역에 가해질 때를 생각해 보자. 이때, 방화벽이 설치되어 운영되고 있다면 서버 등의 컴퓨팅 자원 고갈로 서비스가 중단되기 이전에 방화벽이 기능을 수행하지 못하여 서버의 서비스가 중단되는 것이 대부분이다. 또 방화벽의 주요기능은 정책에 따라 패킷을 내부 네트워크로 받아들이거나 거부하는 것으로 처리되는 형태로 패킷의 순서와 상관없이 매우 계산 지향적 작업을 수행한다.

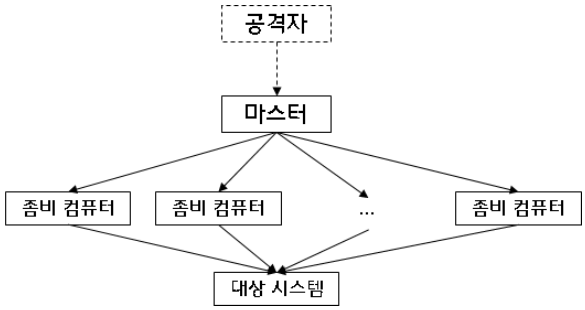
따라서 분산서비스거부공격이 발생할 때 공격대상 서버/네트워크에 속해 있는 방화벽이 처리해야하는 대규모의 트래픽을 방화벽 하단의 서버 군과 함께 그리드(Grid) 구조로 분산 처리함으로써 정당한 서비스 요청을 처리할 수 있는 비율을 높이고자 한다.

## 2. 관련연구 및 요구사항

### 2.1 분산서비스거부 공격과 대응방안

DDoS 공격은 공격자가 감염시킨 여러 피시 또는 서버 등이 공격을 하게 하여 특정 시스템의 자원을 고갈시킴으로써 시스템이 더 이상 정상적인 서비스를 할 수 없도록 만드는 공격 방법이다. 또는 특정 웹 사이트를 공격하기 위해 공격자가 DoS 공격을 위한 도구들을 여러 컴퓨터에 설치해 놓고 공격 대상 컴퓨터 시스템이 처리할 수 없는 많은 양의 패킷을 동시에 전송시켜 네트워크의 성능 저하나 시스템 마비를 유발하는 공격이라 정의할 수 있다.

과거 공격도구를 이용하여 DDoS 공격방법으로 사용되었던 SYN Flooding, ICMP Flooding 등의 방법은 운영체계에 대응방안이 탑재되고 하드웨어의 발전으로 더 이상 의미 있는 피해를 주지 못하는 상태이다. 그러나 정상적인 세션을 다수 생성하여 대상 시스템의 자원을 고갈시키는 공격은 여전히 유효하다[2]. 따라서 DDoS 공격이 이루어질 때 대상 시스템의 자원 고갈을 최대한 지연 또는 완화시키거나 차단하는 방안들이 고안되었다[3,4]. 또, DDoS 공격을 조기 탐지하여 사전에 대응함으로써 피해를 최소화하는 방법들도 제안되었다[5,6].




(그림 1) DDoS 공격 개념도

그렇지만 이러한 방안과 제안들은 대부분 현재 운영되고 있는 시스템의 구조를 변경해야 하거나 추가적인 시스템을 도입하여 구축, 운영해야 되는 구조이다.

### 2.2 방화벽(Firewall)

방화벽은 네트워크의 요구사항에 따라 초기 패킷 필터 기능을 제공하면서 출발하여 세션을 관리하고 검사하는 기능을 갖는 상태점검(Stateful Inspection) 방화벽으로 발전하며 근래 패킷의 내용, 즉 애플리케이션에 영향을 미칠지를 분석하는 방화벽까지 나타났다. 여기서 논의될 방화벽은 세션을 관리하는 보편적인 기능을 보유한 것으로 한정하고자 한다[7].

표.1은 방화벽의 패킷 필터를 하기 위한 접근제어 목록을 보여주는 것이고 표.2는 상태 유지 패킷 필터 기능을 위해 필요한 정보를 기술한 표이다. 즉, 패킷이 방화벽에 도착하면 규칙 A부터 E까지 순차적으로 점검하며 해당하는 규칙에 따라 액션을 취하는 것이다. 그리고 상태점검 방화벽은 방화벽을 통해 흐르는 모든 연결의 상태를 기록하고 패킷을 거부하기 위한 기초자료로 연결 상태를 이용하는 것이다. 이것을 위해 먼저 네트워크 및 세션 계층의 상태에 대한 상태 테이블을 메모리에 생성하고 현재 세션 테이블에 있는 허용된 포트(서비스)와 관련된 패킷만 통과시키고 TCP 종료 세션이 발생하거나 몇 분이 지나면 해당 목록을 삭제하여 관계없는 패킷을 거부하는 것이다.

<표 1> Packet Filter 구조 예시

| 규칙 | 방향 | 출발지 주소 | 목적지 주소 | 프로토콜 | 목적지 포트 | 액션 |
|---|---|---|---|---|---|---|
| A | In | External | Internal | TCP | 25 | Permit |
| B | Out | Internal | External | TCP | >1023 | Permit |
| C | Out | Internal | External | TCP | 25 | Permit |
| D | In | External | Internal | TCP | >1023 | Permit |
| E | Either | Any | Any | Any | Any | Deny |

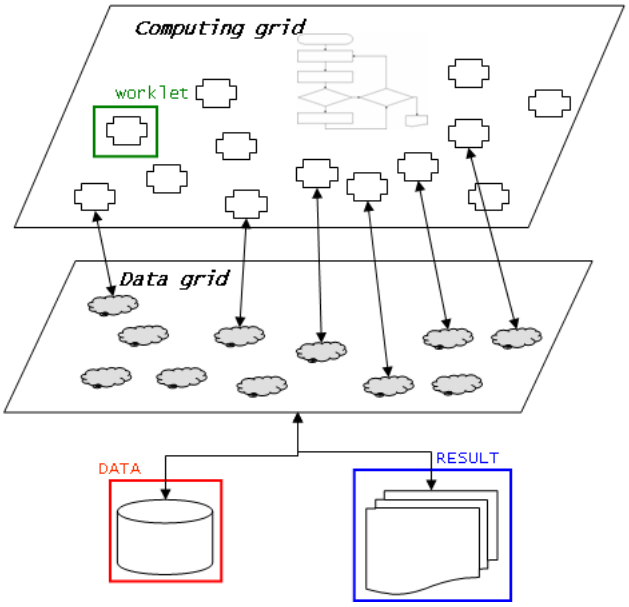
<표 2> Session Table 구조 예시

| 종류 | 내부 IP | 내부 port | 외부 IP | 외부 port | 상태 |
|---|---|---|---|---|---|
| TCP | 60.55.xx.12 | 62600 | 123.80.xx.34 | 80 | OK |
| UDP | 60.55.xx.12 | 63206 | 222.8.xx.4 | 69 | OK |

그래서 상태점검 방화벽은 단순히 규칙만 비교하는 패킷필터 방화벽보다 세션 테이블을 생성하고 상태를 검색하는 부하가 증가하기에 처리속도가 늦다. 또한 이러한 구조로 인하여 DoS 공격의 대상이 되기도 한다.

### 2.3 그리드 컴퓨팅

그리드 컴퓨팅에서 하나의 서비스를 더 세부적인 일로 나눌 수가 있는데 이에 해당하는 일을 워크릿(worklet)이라고 할 수 있다. 그리드 형태의 서비스에서 어떠한 프로그램의 수정도 없이 처리용량을 추가적으로 이용할 수 있다. 워크릿이 동시에 컴퓨터 자원에 접근을 함으로써 가능하다[8]. 특히 계산 지향적인 애플리케이션의 경우, 서로 독립적이며 동시에 다른 시스템에서 수행될 수 있는 단위 작업을 워크릿이라 한다. 그리드 환경에서 평행하게 실행되는 워크릿은 수많은 시스템에서 계산작업을 수행하는데 이용된다. 그림 2는 데이터가 존재하는 그리드에서 해당 데이터를 갖고 와 계산 그리드에서 독립적이며 평행하게 계산을 수행하고 그 결과를 다시 데이터 그리드로 이관시키는 그리드 모델을 형상화한 것이다[8]. 이 모델은 에너지 탐사, 제약 및 금융서비스 개발 등의 분야, 즉 많은 데이터로부터 데이터를 분석 처리하여 의미 있는 결과(데이터)를 얻고자하는 여러 분야에서 활용되고 있다. 그리드 기술은 시간 절약뿐만 아니라 강력하며 유연하고 비용절감을 할 수 있는 컴퓨팅 환경을 제공하고 있다.



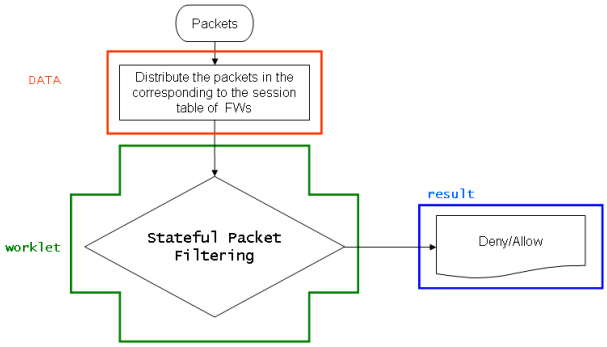

(그림 2) 워크릿 개념

## 3. 제안된 분산 방화벽 모델

앞서 살펴 본 바와 같이 구조변경을 최소화하면서 추가적인 시스템 없이 DDoS 공격에 대응하는 모델이 없는 실정이다. 그리고 방화벽은 Stateful Packet Filtering를 하기 위해 방화벽이 세션테이블을 구성해 관리하고 있는데 이 방화벽에 의해 보호되고 있는 인터넷 서비스 제공 서버 군(group)들이 DDoS 공격 대상이 되었을 때 보호대상 서버군의 자원이 고갈되어 서비스 불능상태가 되기 이전에 방화벽의 세션테이블 관리기능이 마비가 되어 버려 서비스가 제공되지 않는다. 따라서 그리드 컴퓨팅 기법 중 하나인 워크릿(worklet) 모델을 적용하여 부하분산을 하고자 한다. 우선, 방화벽 안쪽으로 들어오는 패킷과 방화벽의 세션 테이블 일부를 데이터로 구성하고 세션테이블 관리 기능과 패킷 필터링 기능을 워크릿으로 구성하여 방화벽 하위 웹 서버 군에 배포시켜 각각의 웹 서버가 방화벽의 세션테이블 기능을 부분적으로 수행하는 것이다. 구성별 구조와 기능은 다음과 같다.

### 3.1 제안된 워크릿(worklet) 구조

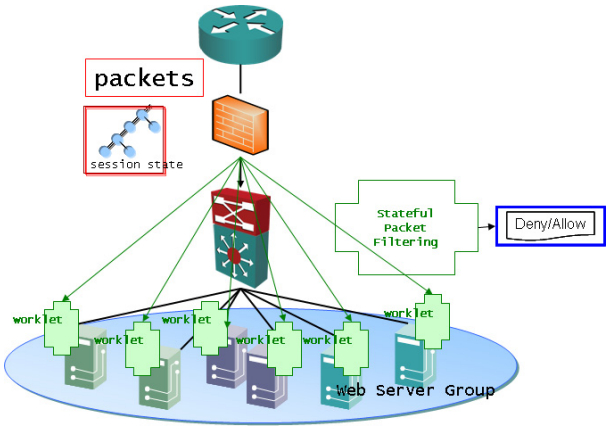
방화벽의 기능에서 독립적으로 동시에 다른 시스템에서 수행될 수 있으며 또한 반복적인 단위작업은 패킷을 세션테이블과 비교하여 규칙에 맞는 것은 통과시키고 그렇지 않은 것을 거부하는 기능으로 단순하지만 많은 컴퓨팅 리소스가 필요한 작업이다. 특히 대규모의 트래픽이 발생할 때, 주소지 등의 패킷 헤더 정보가 유효하다면 세션테이블에 신규 등록이 되어 세션테이블도 급격히 증가할 것이다. 따라서 Stateful Packet Filter 기능을 워크릿으로 구성하고 패킷 또는 일부 세션테이블 내용을 데이터로 구성할 수가 있다. 그림 3은 일반적인 워크릿 구조에 기존 상태점검 방화벽 기능을 적용한 것을 형상화했다.



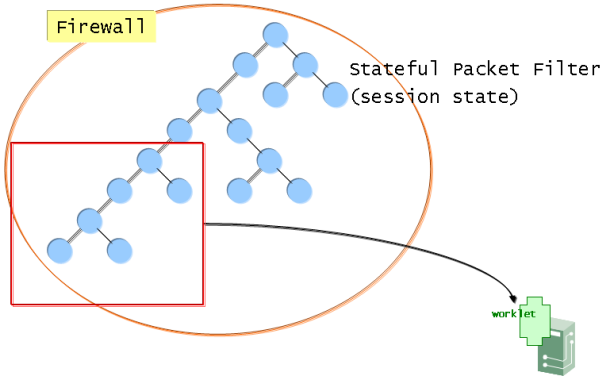

(그림 3) 방화벽 기능을 수행하는 워크릿 구조

### 3.2 분산 방화벽 모델

여기서 구성된 워크릿을 기존 방화벽이 보호하고 있는 인터넷 서비스 제공 서버(웹 서버)에 배포를 시키고 방화벽에 대규모의 트래픽(DDoS 공격)이 도달하여 세션테이블도 급격히 증가할 때, 미리 정해 놓은 일정한도의 크기를 세션테이블의 크기가 넘어서면 세션테이블의 최종 규칙에 패킷을 전달/분산시킬 서버의 주소를 추가하고 더 이상 세션테이블의 크기를 증가시키지 않는다. 그러면 기존 방화벽의 세션테이블의 규칙에 따른 점검을 진행하면서 신규 세션에 대해서는 방화벽 하단의 서버로 분산되어 이 서버에 있는 방화벽 워크릿에 의해 처리된다. 그림 4는 이 전체 모델을 도시한 것이고 그림 5는 세션테이블 내용 일부를 워크릿에서 처리하는 것을 형상화한 것이다.



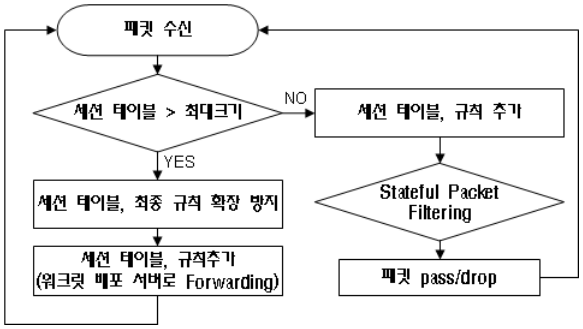

(그림 4) 워크릿을 이용한 분산 방화벽 모델




(그림 5) 세션테이블 구조와 워크릿

### 3.3 분산 방화벽 모델의 적용 및 운용

이와 같은 모델을 적용하기 위해 기존 방화벽에 추가할 것은 그림 6에서 기술한 기능을 수행하는 에이전트를 추가하고 방화벽하단에 같은 기능의 에이전트와 방화벽 워크릿을 추가하기만 하면 된다.




(그림 6) 세션테이블 점검 알고리즘

즉, 세션테이블 점검 기능이 탑재된 방화벽이 운용되다가 세션테이블의 크기가 일정 크기를 넘어서면 세션테이블의 규칙을 더 이상 증가시키지 않고 하단의 서버 중 하나로 패킷을 포워딩시키는 규칙을 추가함으로써 트래픽을 분산시킨다. 그리고 그 트래픽을 받는 서버 내에 있는 방화벽 워크릿이 그림 3과 그림 6과 같이 작동을 하며 패킷을 점검하게 된다. 또한 이 서버의 세션테이블도 일정 크기를 넘어서면 다시 이웃의 다른 서버에게로 신규 세션을 필요로 하는 트래픽을 분산하도록 세션테이블의 마지막 규칙을 추가함하고 그림 3과 그림 6과 같이 반복 수행함으로써 방화벽 하단의 서버 리소스만큼 가용성을 증대시킬 수 있는 것이다.

## 4. 정리

지금까지 제안한 모델은 DDoS 공격이 진행되는 중이라도 추가적인 구성 및 시스템 없이 정당한 인터넷 서비스 요청에 응답할 수 있도록 가용성을 증대시키는 것이다. 이 모델에서 상태점검 방화벽의 신규로 발생하는 상태점검 테이블(세션 테이블)의 일부를 패킷과 함께 워크릿에서 처리하도록 함으로써 DDoS 공격으로 인한 방화벽의 부하를 분산하는 효과를 얻을 수 있는 것이다. 향후 여기서 제안한 모델을 구현함과 동시에 성능측정을 통해 DDoS 공격으로 인한 방화벽의 부하를 분산하는 능력을 측정할 예정이다. 구현과 분석평가의 용이성을 위해 방화벽은 소프트웨어 형태를 선택하여 그림 4와 같이 테스트 베드를 구성하여 단위 시간 당 웹 페이지 요청 건수의 증가에 따른 응답시간 등의 테스트 시나리오로 분석평가를 진행할 것이다. 또한 방화벽이 사용하고 있는 세션테이블 관리 알고리즘, B-tree 또는 RB(red-black) tree 등에 따른 차이도 살펴볼 것이다.

## 참고문헌

[1] 한국정보보호진흥원, http://www.kisa.or.kr

[2] 한국정보보호진흥원, “분산 서비스 거부 공격 차단 및 분석 기술”, 인터넷침해사고대응지원센터(KISC), 2004.09.

[3] 김미희, 채기준, “계층적 오버레이를 이용한 DDoS 공격 감내 네트워크”, 한국정보처리학회 논문지 C, VOL. 14-C NO. 01, p45~54, 2007.02.

[4] 박필용, 홍충선, 최상현, “검증된 IP 테이블을 사용한 통계 기반 DDoS 대응 시스템”, 한국정보처리학회 논문지 C, VOL. 12-C NO. 06, p827~838, 2005.10.

[5] 강길수, 이준희, 최경희, 정기현, 심재홍, “DDoS 공격 탐지를 위한 패킷 샘플링 기법들의 성능 분석”, 한국정보처리학회 논문지 C, VOL. 11-C NO. 06, p0711~718, 2004.12.

[6] 이철호, 최경희, 정기현, 노상욱, “웹 서버에 대한 DDoS공격의 네트워크 트래픽 분석”, 한국정보처리학회 논문지 C, VOL. 10-C NO. 03, p253~264, 2003.06.

[7] Elizabeth D. Zwicky, Simon Cooper, D. Brent Chapman, “Building Internet Firewalls (2nd Edition)”, O'REILLY, 2000.

[8] Michael Di Stefano, "Distributed Data Management for Grid Computing", A John Wiley & Sons, 2005.