韓國컴퓨터情報學會 論文誌
第14卷 第4號, 2009. 4.

2009-14-4-3-2

싱크홀 라우팅을 이용한 유해 트래픽 제어

장 문 수*, 이 정 일*, 오 창 석**

Harmful Traffic Control Using Sink Hole Routing

Moon-Soo Chang*, Jeong-Il Lee*, Chang-Suk Oh**

요 약

인터넷을 구성하는 IP 기반의 네트워크 구성은 다양한 회사의 라우터와 스위치 장비들로 구성되어 있다. 다양한 장비의 구성은 웜, 바이러스, DDoS 등과 같은 유해트래픽을 필터링하기 위하여 각 회사마다 서로 다른 형태의 문자 명령어 기반인 CLI가 주로 사용되고 있어 관리 및 제어의 복잡성이 높다. 이의 대안으로 IETF에서는 XML 기반으로 구성관리 표준을 NETCONF 작업 그룹에서 제정하고 있지만, NETCONF의 명령어 처리 단계에서 처리되는 명령어 몇 가지만 표준으로 정의되어 있고, 유해트래픽을 차단하기 위한 XML 명령어는 각 회사 장비마다 서로 다르게 되어 있으므로 이 기종 장치간의 일관된 제어 명령어 처리가 어렵다. 본 논문에서는 이 기종 장치의 일관된 제어 명령어를 통하여 네트워크로 유입되는 유해트래픽을 싱크홀 라우터로 우회시키고, 유입된 트래픽을 대상으로 유해성 여부를 판단하여 다양한 공격을 효과적으로 차단할 수 있는 제어시스템을 설계하여, 유해트래픽으로 부터 보호되고, 보다 안정된 네트워크 효율을 높일 수 있는 방법을 제안하였다.

Abstract

The construction of Internet IP-based Network is composed of router and switch models in a variety of companies. The construction by various models causes the complexity of the management and control as different types of CLI is used by different company to filter out abnormal traffics like worm, virus, and DDoS. To improve this situation, IETF is working on enacting XML based configuration standards from NETCONF working group, but currently few commands processing at the level of operation layer on NETCONF are only standardized and it's hard for unified control operation process between different make of system as different company has different XML command to filter out abnormal traffics. This thesis proposes ways to prevent abnormal attacks and increase efficiency of network by re-routing the abnormal traffics coming thru unified control for different make of systems into Sinkhole router and designing a control system to efficiently prevent various attacks after checking the possibility of including abnormal traffics from unified control operation.

▸ Keyword : 싱크홀라우팅(Sink Hole Routing), 트래픽 제어(Traffic Control), NETCONF(Network Configuration), CLI(Command Line Interface)

• 제1저자 : 장문수 교신저자 : 오창석(csoh@chungbuk.ac.kr)
• 투고일 : 2009. 03. 05, 심사일 : 2009. 03. 26, 게재확정일 : 2009. 04. 28.
* 충북대학교 컴퓨터공학과 ** 충북대학교 전기전자컴퓨터공학부
※ 본 논문은 2008년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었습니다.

Ⅰ. 서 론

인터넷이라는 도구가 정보화 사회에서 유비쿼터스 사회로 진화하고 있으며, 융합 인터넷 서비스의 확대에 따라 실생활에 밀접한 생활 도구로 자리 잡고 있다. 최근 인터넷을 이용하여 악의적인 목적으로 유해트래픽을 발생하여 개인 컴퓨터와 같은 단일 시스템을 공격대상으로 하던 패턴에서 벗어나 학내망, ISP와 같은 네트워크 인프라 자체를 위협하고, 보다 큰 규모의 시스템 및 네트워크를 공격하는 형태로 발전하고 있다[1]. 이미 여러 매체를 통해 알려진 바와 같이 2000년 야후, 아마존, CNN 등 굴지의 인터넷 관련 기업들이 수많은 좀비 시스템들로부터 대규모 DDoS 공격을 받아 천문학적인 피해를 입었으며, 국내에서도 2003년 1.25 인터넷 침해사고 시 수 많은 ISP를 대상으로 Slammer SQL monitor port인 1434 포트를 통해 대량의 트래픽이 발생하여 생기는 네트워크 과부하를 막지 못하여 발생된 인터넷 장애 때문에 수많은 피해를 입기도 하였다[2]. 또한 최근에는 인터넷을 통해 웜이나 Bot과 같이 내재된 좀비 시스템들에 의해 발생되는 유해트래픽으로 인하여 네트워크 트래픽 폭주나 불안정으로 인하여 잠재적인 위협 및 추가적인 비용도 증가하고 있다.

따라서 본 논문에서는 추가적인 설비투자 없이 기존의 IP 기반 라우터와 스위치 장비로 구성된 네트워크 인프라를 대상으로 회선 차단, ACL, Rate-Limit, 트리거 라우터 설정, 싱크홀 라우터 설정, BGP(Border Gateway Protocol)등의 라우터 기반 제어기술에 대해서 알아보고, CLI와 XML기반의 NETCONF[3]를 이용한 이기종 장비 접근 방법 설계와 코어 네트워크로 실시간 유입되는 트래픽을 분석하여 정상적인 트래픽은 네트워크 인프라를 통해 서비스 되고, 유해트래픽을 이용한 공격트래픽은 공격탐지 및 차단할 수 있는 시스템을 설계하여, 기존의 코어 네트워크 시스템에 부하를 주지 않으며, 보다 안정되고 효율적인 네트워크 트래픽 관리 및 유해트래픽을 차단할 수 있는 제어시스템을 설계하여 제안 시스템의 안정성 및 효율성을 실험 결과를 토대로 고찰하였다.

Ⅱ. 트래픽 제어 기술

2.1 ACL

ACL(Access Control List)은 가장 일반적인 유해트래픽 차단 기술이다. 라우터를 경유하는 모든 트래픽에 대해서 제어를 할 수 있으며, 방화벽과 같은 보안기능 및 예기치 않은 트래픽이 네트워크를 경유 또는 접근하는 것을 방지하거나, 허가되지 않은 이용자가 네트워크의 특정 자원을 접근하는 것을 차단하는 기능을 한다. 〈그림 1〉은 라우터 내에서 ACL 프로세스를 나타낸다.

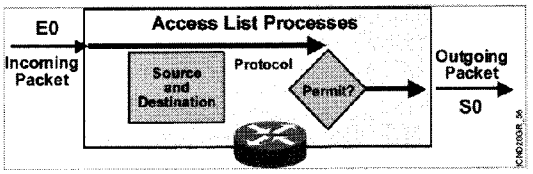


그림 1. ACL 프로세스
Fig 1. ACL Processes

접근 통제를 위한 ASIC화된 모듈이 라우터 내에 내장되지 않을 경우 성능저하에 원인이 되기도 하지만, 대부분의 라우터들이 ASIC화된 모듈을 사용하여 성능저하의 원인을 해소 하고 있다. ACL은 3계층, 4계층, 7계층 레벨의 제어 수준으로서 IP 주소, 서비스 포트 등 컨텐츠 기반의 차단이 가능한 기술이다. 제어를 목적으로 라우터에 ACL을 설정하게 되면 라우터로 들어오는 인그레스 트래픽, 라우터에서 나가는 이그레스 트래픽 모두 제어할 수 있다. ACL의 종류는 표준 ACL, 확장 ACL, 명명된 ACL로 구성되어 있다. 표준 ACL은 트래픽 정의를 근원 네트워크, 와일드카드 마스크로 정의하고, ACL 번호는 1~99, 1300~1999 이다. 확장 ACL은 트래픽 정의를 근원 네트워크, 목적지 네트워크, 와일드카드 마스크, 프로토콜 타입, 옵션으로 정의하고, ACL 번호는 100~199, 2000~2699 이다. 명명된 ACL은 트래픽의 주소를 이용하기 보다는 워드 텍스트를 이용하여 트래픽을 정의하며 이름을 부여하여 표준 또는 확장 형식을 구분한다.

2.2 Rate Limit

Rate Limit은 특정 서비스 또는 패킷이 단위 시간동안 인터페이스가 갖고 있는 대역폭을 대상으로 패킷이 일정량 이상 초과할 경우 그 이상의 패킷을 통과시키지 않도록 하는 것이다. 일정한 시간을 정해서 해당 비율에 맞게 버스트 크기를 정하여 버스트 크기 이상의 패킷이 지나가지 않도록 하여 비율을 제한하는 방식이다. 〈그림 2〉는 Rate Limit 적용전과 후의 트래픽 흐름 상태를 나타낸다.

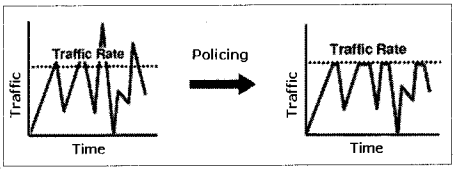


그림 2. Rate Limit
Fig 2. Rate Limit

트래픽 비율에 부합하지 않는 트래픽을 버리거나 양을 조절하며, MAC, 목적지 주소, 소스 주소, L4 프로토콜, 목적지 포트, 소스 포트별로 제어가 가능하다.

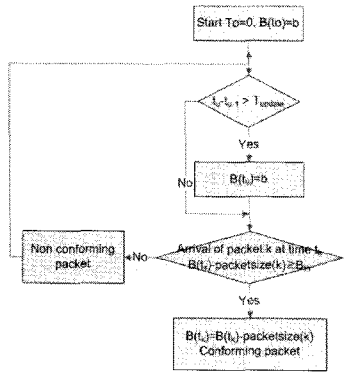


그림 3. Rate Limit 알고리즘
Fig 3. Rate Limit Algorithm

〈그림 3〉은 Rate Limit 알고리즘을 나타내며 Rate Limit 설정 시 옵션은 다음과 같다.

- burst-normal : 초과 허용 대역폭 (bytes)
- burst-max : 초과 정책을 적용할 대역폭 한계 (bytes)
- conform-action : 한계를 넘지 않을 때 취할 행동
- exceed-action : 초과 시 취할 행동

2.3 uRPF

uRPF(unicast Reverse Path Forwarding)는 출발지 IP 주소를 위장하여 공격하는 IP 스푸핑 공격을 차단해 줄 수 있는 기술이다. 지속적으로 업데이트되는 라우팅 테이블을 참조하기 때문에 동적 필터라고 한다. 패킷 반환 경로 상의 인터페이스와 패킷이 도착한 인터페이스가 동일한지를 라우팅 테이블을 참조하여 검사하며, 위조된 원천지 주소를 갖는 패킷을 필터링한다. 만약 비대칭 경로가 존재하는 경우에는 부적절하기 때문에 백본 라우터보다는 에지 라우터에 적용하는 것이 바람직하다. 싱글-홈 SP(Service Provider) 고객 에지 라우터를 대상으로 하는 스트릭트 모드 uRPF와 멀티-홈 SP 고객 에지 라우터를 대상으로 하는 루즈 모드 uRPF로 나눌 수 있다[4].

2.4 Triggered Blackhole Routing

에지 라우터의 null0 인터페이스 설정을 통하여 악의적인 목적의 트래픽을 에지 라우터들에 의해서 설정된 null0 인터페이스로 트래픽을 포워딩하여 차단할 수 있다. 목적지 주소 위주의 패킷 필터링이 적용되며, ASIC기반으로 동작하므로 CPU와 ACL 처리 시간을 단축시키며 성능 저하가 없어 트래픽 폭주로 인한 네트워크 다운이나 서비스 중지가 발생할 경우 유용하게 적용할 수 있다. tag 기반으로 필터링을 하며, 라우터로 유입되는 트래픽을 대상으로 제어하고자 하는 트래픽을 대상으로 지정된 tag를 붙이고, 라우터에서 특정 tag는 null0 인터페이스로 포워딩하는 환경설정을 통하여 유해한 트래픽을 폐기하는 형태로 트래픽을 차단할 수 있다.

트리거 블랙홀 라우팅을 활용하려면 iBGP 광고를 위하여 기본적으로 BGP 설정이 되어 있어야 한다[5]. iBGP를 통해서 AS(Autonomous System) 내의 에지 라우터들에게 정적 라우트를 재분배할 수 있도록 설정해야 한다[6]. DDoS와 같은 공격 발생 시 새로운 라우팅 정보를 에지 라우터들에게 알려주는 역할을 수행한다.

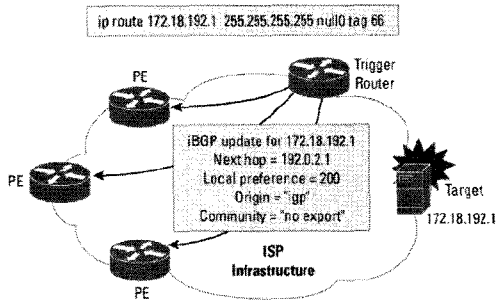


그림 4. 목적지 주소기반 원격 트리거 블랙홀 필터링
Fig 4. Triggering Destination-Based Remotely Triggered Black Hole Filtering

〈그림 4〉는 목적지 주소 기반의 트리거 블랙홀 라우팅 구조도를 나타낸다.

2.5 Sinkhole Routing

유해트래픽과 관련된 패킷들을 특정 네트워크로 우회하여 트래픽을 분석하는 방법이다. 트리거 블랙홀 라우팅과 같이 iBGP 광고를 위하여 기본적으로 BGP 설정이 되어 있어야 한다. 태그 기반으로 필터링을 하며, 에지 라우터의 null0 인터페이스의 다음 홉을 싱크홀 주소로 명시하여 구성할 수 있다. 싱크홀 주소는 라우터나 트래픽 분석을 위한 시스템이 될 수 있다. 싱크홀로 처리하기 위하여 태그된 트래픽은 싱크홀 주소로 포워딩 된다.

싱크홀은 사용되지 않는 어드레스 영역을 '_BOGON'이나 '_DarkIP'로 표시해, 지역 레지스트리에 의해 할당되지 않았다는 IP 주소를 나타내주기 때문에 유효하지 않은 소스나 목적지가 된다. 트래픽이 네트워크의 에지에서 삭제되거나 이러

한 삭제된 트래픽의 소스가 싱크홀에 표시된 것과 동일하다면 ICMP는 싱크홀로 트래픽을 라우팅한다. 싱크홀은 이러한 ICMP 메시지의 기록을 모니터링하고 진입 라우터를 규명해 시스템 에지에 대한 트래킹을 실행하게 된다[7].

III. 유해트래픽 제어시스템 설계

본 논문에서 제안한 유해 트래픽 제어 시스템은 DDoS나 웜과 같은 불특정한 유해 트래픽을 대상으로 회선차단, 서비스 포트 기반 차단, IP주소 기반 차단하는 방법으로 크게 3가지로 구분하여 제어시스템을 구성하였다. 서비스 포트 기반 차단의 경우에는 ACL, Rate Limit을 주로 활용하였으며, IP주소 기반 차단의 경우에는 RTBH(Remote Triggered Black Hole routing), 싱크홀 라우팅 방법을 활용하였다. 특히 트래픽을 분석하기 위하여 라우터로 유입되는 유해 트래픽을 싱크홀로 우회하여 트래픽 분석 시스템을 통해 유해성 여부를 판단하여 유해한 트래픽은 백본 네트워크로 유입되기 전에 폐기 처리하여 백본 네트워크에 영향을 주지 않게 하였으며, 정상적인 트래픽은 제어시스템에서 판단하여 정상적으로 서비스가 이루어질 수 있도록 설계하였다.

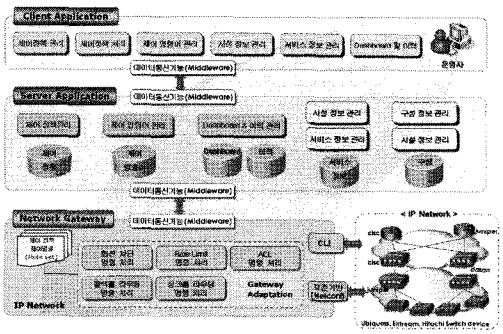


그림 5. 유해트래픽 제어시스템 구성도
Fig 5. Block Diagram of Harmful Traffic Control System

〈그림 5〉에서 보는 것과 같이 본 논문에서 제안한 유해트래픽 제어시스템은 이 기종간의 라우터를 기반으로 하는 네트워크 인프라를 대상으로 하며, 3 티어 아키텍처로 구성되어 있다. 사용자 인터페이스를 제공하는 클라이언트 계층과 비즈니스 로직 및 데이터 처리를 관장하는 비즈니스 계층, 이기종간의 제어처리 및 데이터를 수집하는 게이트웨이 계층으로 구성된다.

CLI를 이용한 제어와 NETCONF를 이용한 제어기능을 제공하는 IP 기반 장치에 적용 가능하다. CLI는 거의 모든 벤더에서 환경 설정을 위하여 텔넷을 통하여 접속할 있도록 제공하고 있으며, 표준기반인 NETCONF의 경우에는 향후에 더 많은 벤더가 참여하여 구성이 이루어지겠지만, 현재로서는 주니퍼, 시스코 장치만 제공하고 있으므로 제어 대상 장치가 제한적이다.

```
configure terminal
access-list 2500 permit UDP any any eq 3333
access-list 2500 permit TCP any any eq 5555
access-list 2500 permit UDP any any
access-list 2500 permit TCP any any
interface FastEthernet5/0/0
rate-limit out access-group 2500 50000000 9375000 18750000 conform-
action transmit exceed-action drop
exit
exit
write
```

그림 6. 시스코 CLI 제어 명령어
Fig 6. CLI Control Operation in Cisco

〈그림 6〉에서와 같이 CLI 명령어는 텍스트 형태의 명령어와 파라미터 조합으로 구성되어 있다. CLI 제어명령어는 벤더나 각 장치의 OS 버전에 따라 차이가 있어, 서로 다른 벤더의 장치를 일관되게 제어할 경우 용이하지 않으며, CLI 명령어 처리 후 write, commit 명령어와 같이 EEPROM에 환경설정을 저장하려고 시도한 후에 결과 메시지를 리턴 받은 후에야 정상, 구문 오류, 의미 오류를 알아볼 수 있는 단점이 있다. NETCONF의 경우에는 CLI보다는 구조화된 형태의 XML 태그로 구성되어 가독성이 높으며 명령어 수행 시 각 라인단위 또는 명령어 구조 단위로 오류체크가 이루어져 디버깅이 수월하며, 직관적인 응답처리가 가능한 장점이 있다.

```
<rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <candidate/>
    </target>
    <config>
      <configuration>
        <interfaces>
          <interface>
            <name>fe-0/2/0</name>
            <fastether-options>
            <ingress-rate-limit>80</ingress-rate-limit>
            </fastether-options>
          </interface>
        </interfaces>
      </configuration>
    </config>
  </edit-config>
</rpc>
]]>]]>
```

그림 7. 주니퍼 NETCONF 제어 명령어
Fig 7. NETCONF Control Operation in Juniper

NETCONF 제어 명령어는 XML 태그로 구성되어 있다. 그 구성은 컨텐츠, 명령어, RPC, 전송 프로토콜의 네 단계로 구성되며, 〈rpc〉, 〈rpc-reply〉 태그로 구성된 RPC 부분과, 〈get-config〉, 〈edit-config〉 등과 같이 NETCONF 명령어로 구성된 명령어 부분과, 실제 제어 명령어의 파라미터와 데이

터로 구성된 환경설정 데이터 부분으로 구성된다. 〈그림 7〉은 주니퍼사의 IOS에서 제공하는 NETCONF 표준 제어명령어를 나타낸다.

CLI 기반으로 제어하기 위해서는 제어 대상 IP기반 장치에 텔넷으로 연결할 수 있으면 CLI 기반 제어가 가능하며, NETCONF로 연결하여 제어하기 위해서는 SSH[8], BEEP[9], SOAP[10]와 같은 프로토콜을 통해서 연결하여 제어할 수 있다. 본 논문에서는 CLI기반 제어는 텔넷을 이용하였으며, NETCONF를 이용하여 IP기반 장치를 제어하기 위하여 NETCONF 표준에서 권고하는 SSHv2를 연결 인터페이스 대상으로 설정하였다.

이기종간의 제어 명령어 처리를 위해서 명령어 구문을 CLI기반과 NETCONF 기반으로 구분하여, 각 장비별, 제어 명령어 별로 구분하여 제어명령어 구문 테이블을 작성하였다. 제어 명령어 구문 테이블은 일관된 제어 명령어를 처리할 수 있도록 사용자 인터페이스에서 입력되는 파라미터와 맵핑할 수 있도록 하였다. CLI 기반 장비제어 명령어는 텍스트 형태이며, NETCONF 기반 장비제어 명령어는 XML 태그 기반으로 구성되어 있다. 파라미터에 해당하는 부분은 서비스 포트 번호, IP 주소, 태그 번호 등으로 구성되어 서비스 포트 번호에 해당하는 데이터로 구성되어 있으며, 장비 종류에 맞게 해당 제어 명령어가 맵핑될 수 있게 처리하여 이기종간의 장비 제어를 원활하게 처리할 수 있도록 하였다. 〈그림 8〉은 본 논문에서 제안한 유해 트래픽 제어처리 흐름을 나타낸다.

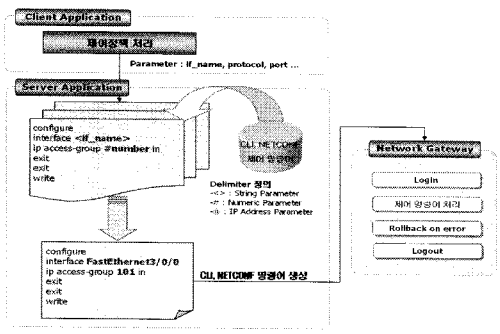


그림 8. 제어시스템 제어처리 흐름
Fig 8. Process of Harmful Traffic Control System

제어 처리는 제어 정책을 기반으로 하는 일괄제어 처리와 각각의 장치와 인터페이스에 개별적으로 제어를 처리할 수 있는 개별 제어 처리로 나눌 수 있다. 일괄제어의 경우 규칙 관리를 통해서 생성된 각각의 제어종류 즉, 회선차단, ACL, 링크 Rate-Limit, 서비스 포트 Rate-Limit, 블랙홀, 싱크홀에 따라 생성된 규칙을 제어대상에 적용할 수 있도록 구성하여 특정 목적의 제어대상을 규칙 기반으로 구성하여 일괄적으로 제어할 수 있도록 구성하였다. 개별제어의 경우에는 특정한 장치와 인터페이스를 대상으로 위에 나열된 제어종류를 개별적으로 파라미터 설정을 통하여 직접적으로 제어기능을 설정할 수 있도록 구성하였다.

구성된 서비스망을 대상으로 현재 제어가 적용된 장치와 인터페이스 정보를 표시하기 위한 상태확인 기능을 통하여 실시간으로 발생된 일괄제어와 개별제어를 표시할 수 있다.

제어 이력 조회 기능을 통하여 일괄제어와 개별제어로 발생된 이력을 관리하여 챠트 형태로 볼 수 있다.

〈그림 9〉은 유해트래픽 제어시스템의 개발환경을 나타낸다

항목 / Tier	개발언어	개발환경	개발도구
Client	Visual C++	MS Windows	Visual Studio T-Chart
AP	C	SUN	VI, Editor
GW	C	IBM AIX	VI, Editor
DBMS	Oracle PL/SQL	IBM AIX	TOAD
Middleware	C	TP monitor	

그림 9. 개발환경
Fig 9. Development Environment

IV. 실험

4.1 실험환경

본 논문에서 제안한 유해트래픽 제어시스템의 제어 대상 장치의 실험환경은 〈그림 10〉과 같다.

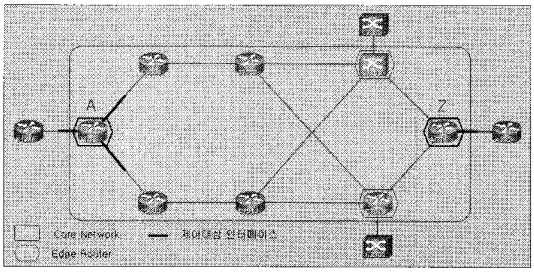


그림 10. 실험 네트워크 토폴로지
Fig 10. Network Topology of Testbed

실험을 위하여 가운데 코어 네트워크를 구성하였으며, 에지 라우터로 4군데 포인트를 설정하여 각 에지 라우터에 null0 인터페이스를 설정하고, 코어 네트워크는 각 라우터간 BGP 광고 설정을 위하여 BGP 라우팅 프로토콜에 대한 환경설정을 하였다. 주요 장비는 시스코와 주니퍼 장비를 사용하여 코어 네트워크를 구성하였다.

실험에 사용된 도구로는 트래픽제너레이터를 사용하였으며, 실험 네트워크 내에 구성된 A라우터와 목적지 에지 라우터인 Z 라우터에 공격포트로 가장한 포트번호를 설정하여 공격 트래픽을 전달하였다.

〈그림 11〉은 공격트래픽으로 가정하여 10MB의 트래픽을 공격 대상 IP인 192.168.100.204 목적지의 UDP 3333 서비스 포트로 공격 트래픽을 보내고 있다. 실험을 위하여 10MB의 공격 트래픽을 5 대의 노트북을 이용하여 약 50MB의 공격 트래픽을 발행하였다.

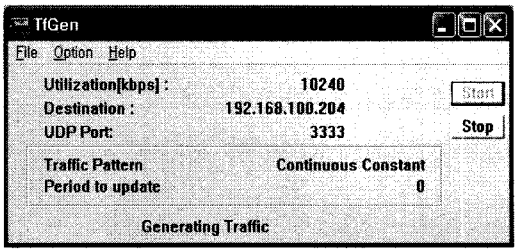


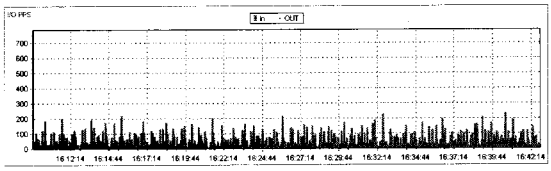
그림 11. 트래픽 제너레이터
Fig 11. Traffic Generator

4.2 실험결과

본 연구의 유해트래픽 제어시스템은 에지 라우터 인터페이스의 pps(packet per second) 데이터를 이용하였다. 실제 운용자 관점에서는 bps(bits per second)가 관심 사항이지만, 네트워크 관리자 측면에서는 이상 트래픽이 발생하거나 발생한 것을 알기 위해서는 pps 데이터가 좀더 유익한 정보를 제공하여 준다.

실험은 유해트래픽을 가장하여 트래픽제너레이터를 이용 트래픽을 발생하여, 실시간 트래픽을 수집하였으며, 본 논문에서 제안한 유해트래픽 제어시스템을 통해서 회선차단, ACL, Rate Limit, Null0 라우팅 제어 기능이 발생했을 때 트래픽 흐름을 파악하여 제어가 원할게 이루어졌는지 확인할 수 있었다.

〈그림 12〉은 정상상태를 나타내는 평상시의 에지 라우터의 트래픽 흐름을 나타낸다.



A 라우터

그림 12. 평상시 edge router의 트래픽
Fig 12. Traffic of edge router on general state

트래픽제너레이터를 이용하여 50MB의 유해트래픽을 발생하여 실시간으로 수집한 A와 Z의 에지 라우터의 트래픽 흐름은 〈그림 13〉과 같다.

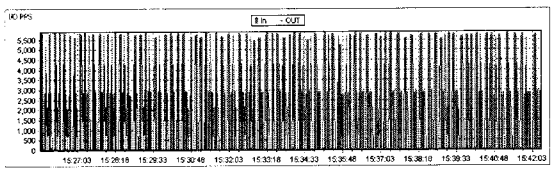


그림 13. 유해 트래픽 Z 라우터
Fig 13. Harmful Traffic

첫 번째 트래픽 테스트는 A와 Z 라우터의 인터페이스에 공격으로 가장한 실험 트래픽을 보내고 백본 네트워크에 진입하기 전인 A 라우터의 인터페이스에 회선차단, ACL, 블랙홀 라우팅인 Null0 라우팅으로 제어를 하였다. 〈그림 14〉의 위에 그림은 A 라우터의 트래픽 흐름을 나타내고, 아래 그림은 Z 라우터의 트래픽 흐름을 나타낸다. 〈그림 14〉에서의 결과처럼 A라우터의 실시간 트래픽 수집을 보면 일정한 트래픽이 지속적으로 A 라우터에 입력되는 것을 확인할 수 있으며, Z 라우터는 회선차단의 경우 백본네트워크로 진입되는 트래픽이 차단되어 Z 라우터에는 수집된 트래픽이 없는 것을 확인할 수 있으며, A 라우터에서 회선차단 시 A 라우터의 롤백으로 인하여 입력 트래픽이 증가한 것을 확인할 수 있다. ACL의 경우 공격트래픽으로 가장한 목적지 3333 포트를 기반으로 제어를 한 후의 Z 라우터의 트래픽 흐름을 볼 수 있다. 상대적으로 50MB의 트래픽 보다는 약 15%정도 트래픽이 필터링 되어 수집되는 것을 확인할 수 있었다. Null0 라우팅은 목적지 주소기반으로 제어를 한 경우로 192.168.100.204 가 목적지로 설정된 트래픽을 제어하였으며, 〈그림 14〉에서 보는 것과 같이 A 라우터에서 Null0 인터페이스로 폐기되어 목적지인 Z 라우터까지 트래픽 흐름이 없음을 알 수 있다.

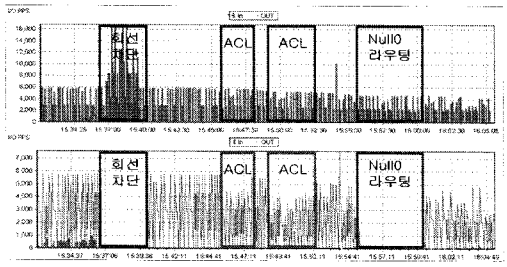


그림 14. 회선차단, ACL Null0 실험결과 A 라우터
Fig 14. The Result of Test 1

또한 물리적인 인터페이스를 차단하는 회선차단과 공격형태의 목적지 주소를 Null0 인터페이스로 drop 하는 형태와 비교를 해 보면 백본 네트워크의 트래픽 부하는 없지만 인그레스 라우터인 A 라우터 자체의 트래픽은 롤백되어 3배 이상 폭주하여 트래픽이 발생하는 것을 확인 할 수 있었다. 상대적으로 Null0 인터페이스로 폐기 하는 형태가 라우터 자체에 부하를 주지 않는 이점이 있다는 것을 실험 결과를 토대로 알 수 있었다.

두 번째 트래픽 테스트는 Rate-Limit을 이용하여 유해트래픽 과부하시 적정 기준의 임계치 정보를 이용하여 유해하다고 판단되는 포트번호의 트래픽을 제어하였다. 〈그림 15〉는 Rate-Limit 실험 결과를 나타낸다. 실험으로 나타난 결과와 같이 Rate-Limit 1은 약 60%의 트래픽 흐름을 유지하는 결과이며, Rate-Limit 2는 약 20%의 트래픽 흐름을 유지하고 있는 결과이다. 유해트래픽 발생 시 폭주하는 트래픽을 적정수준으로 차단하여 원활한 트래픽 흐름을 유도할 수 있다.

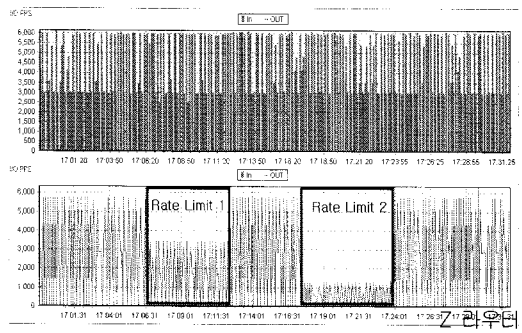


그림 15. Rate-Limit 실험결과
Fig 15. The Result of Test 2

50MB의 트래픽이 유입되기 시작하여 CPU의 사용률이 증가하고 있지만, 〈그림 16〉에서와 같이 5%~10%의 범위 내의 CPU 사용률을 나타내고 있으며, 안정된 상태를 나타내며 CPU의 과부하는 받지 않는다는 것을 알 수 있다.

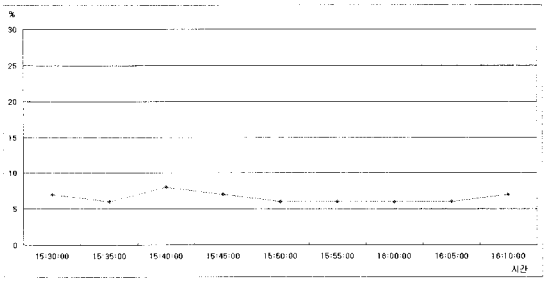


그림 16. CPU 사용률
Fig 16 Usage rate of CPU

〈그림 17〉은 실험 시 발생한 메모리 사용률을 나타낸다.

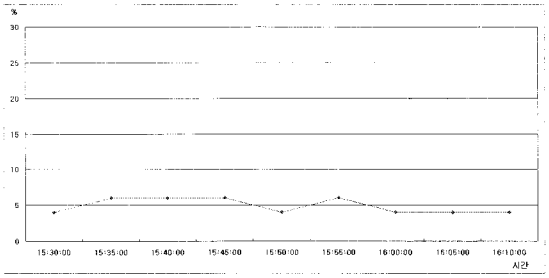


그림 17. 메모리 사용률
Fig 17. Usage rate of Memory

실험의 경우 트래픽에 대하여 제어가 발생할 경우 메모리 사용률의 변화도 거의 없고 안정된 상태를 유지하고 있다.

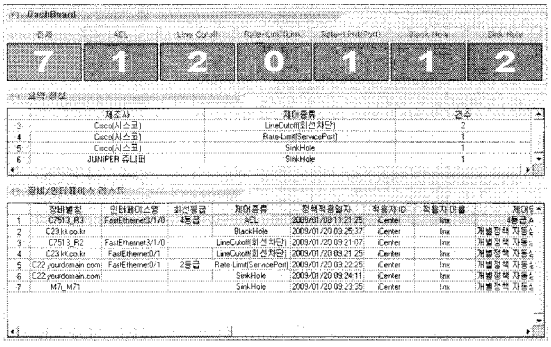


그림 18. 대쉬보드
Fig 18. Dashboard

〈그림 18〉은 실험환경을 기반으로 제어시스템을 통해 제어가 발생한 인터페이스나 대상 장비에 대한 카운트 정보를 표시하고 있다. 이 정보는 현재 제어상태를 나타내며, 유해트래픽을 검증하여 새로운 제어가 발생하게 되면 실시간으로 반영되어 직관적으로 현재 상태를 표시한다.

V. 결론

본 논문에서는 싱크홀 라우팅을 이용하여 트래픽을 수집하여 분석하였으며, 유해트래픽 제어시스템을 제안하였다. 실험결과를 토대로 본 논문에서 제안한 유해트래픽 제어시스템은 실험으로 발생시킨 유해트래픽을 회선차단, ACL, Rate Limit, 블랙홀, 싱크홀 라우팅을 이용하여 서비스 포트기반, IP 주소 기반의 유해한 트래픽을 차단하는 결과를 볼 수 있었다. 또한 코어 네트워크로 유입되기 전에 에지 라우터의 인그레스나 이그레스 인터페이스에 제어시스템을 통해 제어가 발생하여 코어 네트워크로 유입되는 유해트래픽을 차단하여 보다 안정된 코어 네트워크의 트래픽 흐름을 보장할 수 있었다.

본 논문의 향후 연구과제는 향후 All-IP기반으로 진화하는 백본네트워크와 전송네트워크의 경계가 없어질 것으로 전망된다. CLI기반 제어와, 표준기반인 NETCONF 기반 제어의 확장을 고려하여, 제안한 유해트래픽 제어 시스템을 적용하여 에지 구간의 안정된 제어를 위하여 에러 발생 시 롤백하는 기능과, 제어명령어 적합성 확인 기능을 강화하여 보다 안정된 유해트래픽 제어시스템을 구축하는 것이다.

참고문헌

[1] 한국정보보호진흥원, "*분산 서비스 거부 공격 차단 및 분석 기술*," 한국정보보호진흥원, 2004년

[2] 장문수, 구향옥, 오창석 "유해 트래픽 분석을 이용한 침입 방지," 한국컴퓨터정보학회논문지, 제10권 4호(pp.173-179), 2005년 9월

[3] Enns, R., "NETCONF Configuration Protocol", RFC4741, IETF, Oct. 2006.

[4] cisco, "Unicast Reverse Path Forwarding Enhancements for the Internet Service Provider - Internet Service Provider Network Edge", *http://www.cisco.com/warp/public/732/Tech/security/docs/urpf.pdf*, 2005.

[5] cisco, "Remotely Triggered Black Hole Filtering Destination based and Source based", *http://www.cisco.com/warp/public/732/Tech/security/docs/blackhole.pdf*, 2005.

[6] Traina, P., "*Autonomous System Confederations for BGP*," RFC1965, Cisco Systems, June 1996.

[7] cisco, "Cisco Service Provider 18," *http://www.cisco.com/web/KR/about/packet/sps/18_2.html*

[8] Wasserman, M., "Using the NETCONF Configuration Protocol over Secur SHell (SHH)," RFC4742, IETF, 2006. 10.

[9] Lear, E., Crozier, K., "Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)," RFC4744, IETF, Oct. 2006.

[10] Goddard, T., Crozier, K., "Using NETCONF over the Simple Object Access Protocol (SOAP)," RFC4743, IETF, Oct. 2006.

저 자 소 개

장 문 수

1997년 2월 충주대학교 전자계산학과 (공학사)

2005년 8월 충북대학교 컴퓨터공학과 (공학석사)

2007년 3월~현재 충북대학교 컴퓨터공학과 박사과정

〈관심분야〉 정보보안, 네트워크 보안, 시스템 보안, 망 관리

이 정 일

2004년 2월 청주대학교 전산학과 (공학사)

2006년 8월 충북대학교 전기전산공학과 (공학석사)

2007년 3월~현재 충북대학교 컴퓨터공학과 박사과정

〈관심분야〉 정보보안, 네트워크 보안,

오 창 석

1978년 2월 연세대학교 전자공학과(공학사)

1980년 2월 연세대학교 전자공학과(공학석사)

1988년 8월 연세대학교 전자공학과(공학박사)

1982년~1984년 한국전자통신연구원 연구원

1985년~현재 충북대학교 전기전자컴퓨터공학부 교수

1990년~1991년 Stanford대학교 객원교수

2007년~현재 한국엔터테인먼트산업학회 회장

〈관심분야〉 컴퓨터네트워크, 뉴로컴퓨터, 정보보호