# NAT 탐지 알고리즘의 실험적 분석[1)]

황성현, 이영석
충남대학교 전기정보통신공학부 컴퓨터전공
kzeus@hanmail.net
yslee@cnu.ac.kr

# Experimental Analysis of NAT Detecting Algorithms

Sunghyun Hwang, Youngseok Lee
Dept of Computer Science and Engineering, Chungnam National University

요 약

NAT는 IP주소 부족의 단기적인 해결방법으로 홈네트워크에서 널리 사용되고 있다. ISP는 트래픽 과금 등에 사용하기 위하여 NAT를 사용하는 IP 주소와 NAT에 연결된 호스트 수를 탐지하려고 한다. NAT 탐지 방법으로는 IP 패킷 헤더의 TTL 필드를 이용하는 것과 ID 필드를 이용하는 것이 있다. 본 논문에서는 NAT(Network Address Translator) 탐지를 위해 제안되어 널리 사용되는 ID 필드를 이용한 방법을 실험적으로 분석한다. 실험결과 서브넷의 계층이나 운영체제의 다양성으로 인하여 완벽하게 NAT 이용 호스트를 탐지하기 어렵다는 것을 검증하였다.

## 1. 서론

부족해지는 IPv4(Internet Protocol version 4) 주소 부족 문제를 해결하기 위해 단기적인 해결방안으로 제안된 NAT(Network Address Translator)[1]는 RFC 1918[2]의 사설주소 영역을 이용하여 공인 IP 주소로 접근할 수 있도록 해주는 기술이다[3]. 일반적으로, NAT는 홈 네트워크 또는 사용자 네트워크에 설치되어 사설 주소들을 공인 IP 주소의 송신자 또는 수신자로 변환한다. NAT는 IPv4 주소 고갈에 대한 단기적인 해결방안[1]을 제공하고, 부가적으로 NAT에 연결되어 있는 내부 사설 IP 주소를 보호하는 기능을 제공한다.

ISP입장에서는 NAT에 많은 컴퓨터들이 연결되어 인터넷 트래픽을 증가시킬 수 있기 때문에 NAT 탐지 및 NAT에 연결된 호스트 개수를 파악하려고 한다. 몇몇 ISP는 NAT를 사용하는 IP 공유기 사용자에 대한 회선 제공을 중단하겠다는 발표도 하였다[6]. 특히, 특정 ISP에서는 데이터 공개를 통해 IP 공유기 사용자가 다른 일반 사용자에 비해 2배 이상의 통화량을 발표시킨다는 내용을 발표하였다[7].

최근에 들어 NAT 장치 탐지에 대한 몇몇 연구들이 진행되어 왔다. 하지만, NAT를 사용하는 IP 주소를 탐지하고, NAT에 연결되어 있는 인터넷 호스트 수를 산정하는 방법은 쉽지 않다[4]. 지금까지 널리 알려지 NAT 탐지 방법은 IP 헤더의 TTL 필드를 이용하는 방법과 IP 헤더의 ID필드를 이용하는 것이다. TTL필드를 이용한 NAT 탐지 방법은 사용자 호스트로부터 관찰되는 지점까지의 경로 상에 존재하는 라우터 수만큼 TTL 값이 감소되는 특징을 이용하여 사용자가 추가한 NAT의 존재를 TTL 값의 변화로 탐지할 수 있다. 반면, ID 값을 이용한 방법은 각 호스트에서 사용하는 ID의 값이 서로 다른 점을 이용하여 찾아낼 수 있다.

본 논문에서는 대표적으로 사용되는 ID 필드 기반의 NAT 장치 탐지 알고리즘을 실험적으로 분석한다. 본 논문의 구성은 다음과 같다. 2장에서 NAT 장치 탐지 알고리즘에 대한 관련연구를 분석하고, 3장에서는 사용된 NAT 탐지 알고리즘에 대해 설명한다. 4장에서는 NAT 트래픽 측정 실험과 실험결과에 대해 분석하고, 5장에서 결론을 맺는다.

## 2. 관련연구

NAT 장치 및 NAT 장치에 연결된 호스트의 개수를 탐지하기 위한 연구로는 Peter Phaal이 제안한 IP 헤더의 TTL(Time To Live)값을 이용하는 방법[8]과 Steven M.Bellovin의 IP 헤더 IP ID값을 이용하는 방법[4]이 있다.

### 2.1 IP TTL을 이용한 NAT 장치 탐지

IP 헤더의 TTL값은 IP 패킷이 네트워크에 있을 수 있는 최대 홉 수를 나타내는 것으로서, TTL값이 0이 된 패킷은 버려지게 된다[8].

TTL값은 최대 255까지 지정될 수 있지만, Windows

1) 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구 결과로 수행되었음 (IITA-2005-(C1090-0502-0020))

XP는 128, Red Hat Linux 9.0은 64와 같이 OS별로 서로 다르게 사용되어 있다[10]. NAT 장비는 사설 주소를 공인주소로 변환하기 위해 IP 패킷 헤더를 처리해야 한다. 따라서, NAT 장비를 지나가는 IP 패킷의 TTL값은 1이 감소하고, 이를 이용하면 NAT 장치를 탐지할 수 있다[9]. 그림 1처럼 호스트 B와 C가 NAT 라우터에 연결되어 있고, 호스트 A는 공인 IP 주소로 이용하고 있다고 하자. 이 때, 라우터 R에서 호스트 A, B, C의 IP 헤더 TTL값을 모니터링 해보면, 각각 128, 127, 127이 관찰된다. 이 방법은 sFlow에 적용되어 있다[7].

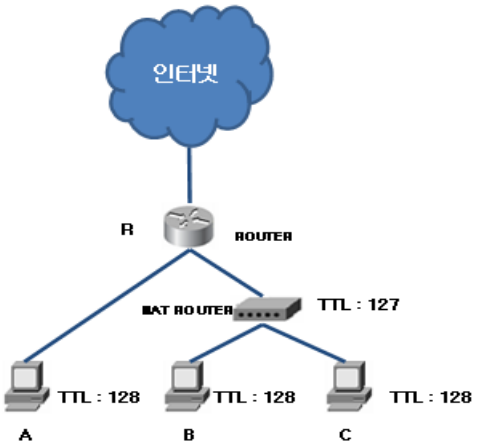



그림 1. TTL을 이용한 NAT 탐지

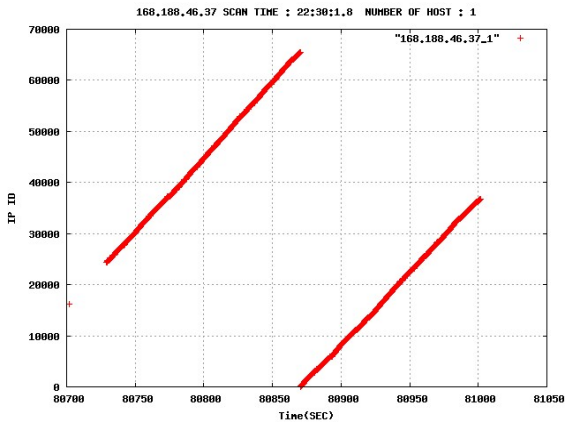
그러나, TTL을 이용한 NAT 장치 탐지는 NAT에 연결되어 있는 호스트의 수는 정확하게 탐지할 수 없다[8]. 또한 NAT장치를 탐지하고자 하는 네트워크 영역의 에지 라우터 또는 모든 스위치들에 NAT탐지를 위한 별도의 소프트웨어를 설치해야 하는 비용이 들게 된다.

### 2.2 IP ID를 이용한 NAT 장치 탐지

IP ID값은 파편화된 IP 패킷을 재조립하기 위해 송신자에 의해 부여되는 16비트 값[9]이다. Steven M.Bellovin은 IP ID값이 모든 패킷에 의하여 선형적으로 증가한다는 것을 이용한 NAT 탐지 방법을 제안하였다[4]. 즉, 호스트에서 생성되는 IP 패킷의 ID 값은 일정한 사이클을 반복하며 연속적으로 증가한다. 즉, 그림 3처럼 1부터 65535까지 생성이 되고, 65535가 되면 다시 1부터 65535까지 반복적으로 생성된다.

이와 같은 특징을 이용하여 연속적인 IP ID의 패턴을 비교하여 하나의 IP 주소에서 2개 이상의 연속적인 IP ID의 패턴이 발견되면, NAT가 특정 IP 주소에 대해 사용된다는 것을 알 수 있다.

하지만, 복잡한 네트워크 환경에서 지나가는 모든 패킷을 분석하여 IP ID의 연속적인 패턴을 탐지하고자 하게 되면, 실제로는 IP ID가 그림 3과 같이 정형화된 형태로 나타나지가 않게 된다.




(그림 3) IP ID의 연속적이고 주기적인 생성

Steven M.Bellovin이 제안한 IP ID를 이용한 호스트 탐지 알고리즘은 다음과 같다.

- 마지막에 도착한 패킷의 IP ID와 새로 도착한 IP ID 패킷간의 시간 간격은 *timelim(300 초)*내에 있어야 한다.
- 새로 도착한 IP ID가 바로 앞에 마지막으로 도착한 IP ID보다 1만큼만 크면 앞서 도착한 IP ID에 새로 도착한 IP ID가 하나의 연결된 패턴으로 분류된다.
- IP ID가 마지막에 도착한 IP ID와 *gaplim(64)* 범위에 있더라도, 지금 당장 구분 짓지는 않고 *OutOfOrder*로 분류한다.
- 또한 IP ID가 충분히 가깝더라도 앞에서 나타난 적이 있으면 *Dup*으로 분류한다.

위와 같은 방법으로 분류된 흐름들은 다시 서로 비교하여 각각이 *gapfac(70)* • *gaplim(64)*내에 있고, *timefac(5)* • *timelim(300)* 시간내에 있으면서 충분히 가까운 것 끼리 다시 합친다. 각각의 연속적인 패턴에 대한 분석 및 조합이 끝나고 나면 *fsize(50)* 보다 적은 흐름은 버리고 나머지 연속적인 흐름 패턴의 개수로 호스트의 개수로 판별하게 된다.

## 3. NAT 장치 탐지 및 트래픽 측정 실험

### 3.1 탐지 및 측정 프로그램 구현

NAT 장치 탐지 모듈은 Steven M.Bellovin이 제안한 [4] IP ID를 이용하여 NAT에 연결된 호스트의 개수를 탐지하는 알고리즘을 참고하여, 아래와 같은 알고리즘을 적용하여 pcap을 이용 구현하였다.

- 패킷 캡쳐를 시작한 후 IP ID가 65000보다 크거나 0이면 버림
- 앞서 도착한 패킷보다 현재 도착한 패킷의 IP ID가 1만큼 커야 하고, 시간의 차이가 1초 이내이면 앞선 패킷이 속한 리스트에 현재 도착패킷을 포함.
- 상기 2조건을 만족하지 않지만, 패킷간 시간차가 5초 이내이고, IP ID의 격차가 200이내이면 리스트에 포함
- 모든 패킷을 각각의 연속적인 리스트로 분류한 후 서로 간의 격차는 한계범위가 존재하지 않고, 가장

가까운 것을 리스트끼리 연결

NAT 장치 탐지내역은 탐지 시간별, IP 주소별 사후 분석이 가능하도록 모두 jpg파일로 생성하였으며, 트래픽은 IP 헤더의 Total Length를 합산하여 트래픽 양을 측정하였다.

3.2 탐지 및 측정환경

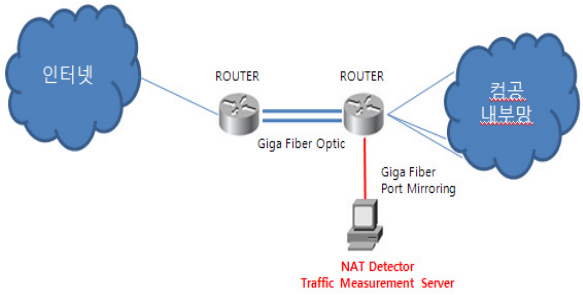



그림 4. NAT 탐지 및 측정 실험 환경

측정 환경은 충남대학교 컴퓨터공학과 망으로 한정하였으며, 충남대학교 컴퓨터공학과 서브넷과 캠퍼스망 백본 네트워크 사이의 1Gbps 링크에서 트래픽을 수집하고 분석하였다. 매 5분간의 패킷을 수집하여 분석 NAT 장치유무를 판별하고 NAT 장치로 판별되는 IP 주소와 그렇지 않은 IP 주소간의 트래픽 양을 별도로 측정하였다. 실험은 2007년 4월 2일 00시부터 23시 55분까지 진행하였다.

## 4. 측정결과

4.1 단일 호스트 IP 주소 탐지결과 분석

측정결과 단일호스트의 경우 그림 5와 같이 선형적으로 증가하는 패턴은 외부망으로 전송되는 트래픽에서 상대적으로 많이 관찰되었다 (Outbound Packet, 그림 5 : 41MB, 그림 6 : 57 KB, 그림 7 : 203 KB). 단일 호스트로 분석된 선형 패턴의 생성된 그래프를 육안으로 보았을 때는 NAT에 연결된 호스트로 보이는 그래프는 없었다.

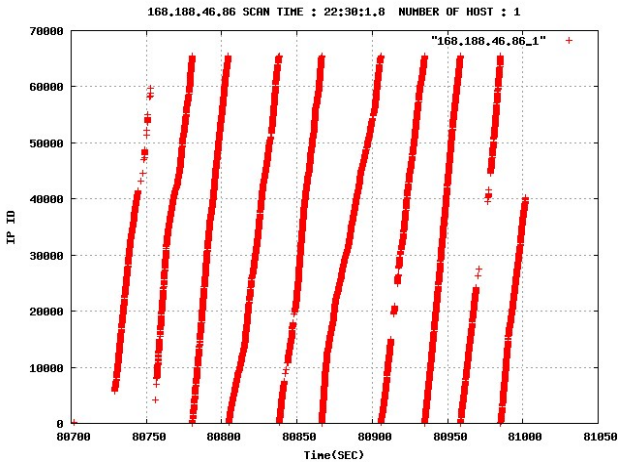



그림 5. 단일 호스트 a

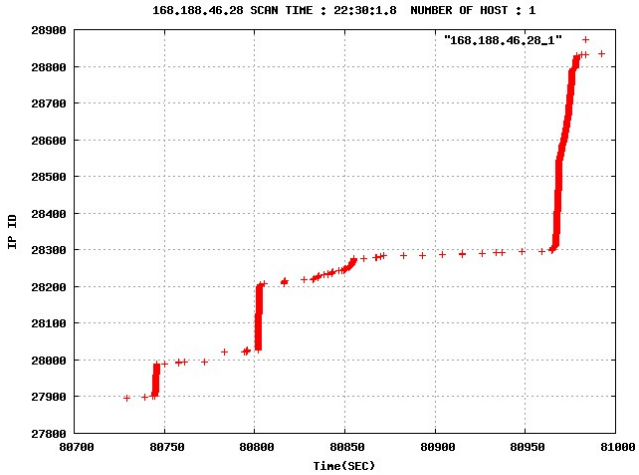



그림 6. 단일 호스트 b

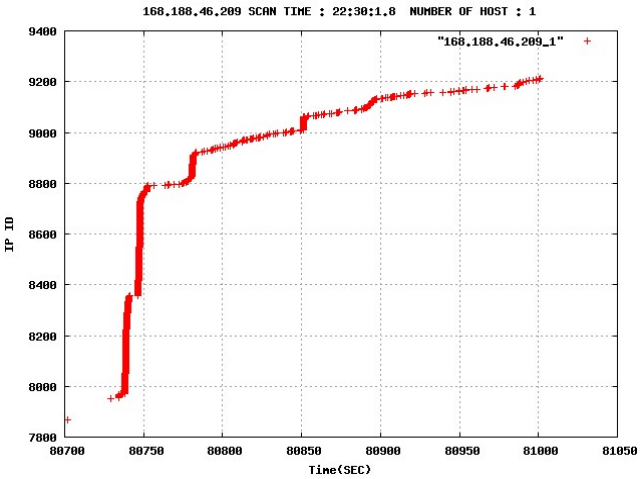



그림 7. 단일 호스트 c

4.2 NAT 이용 IP 주소 탐지결과 분석

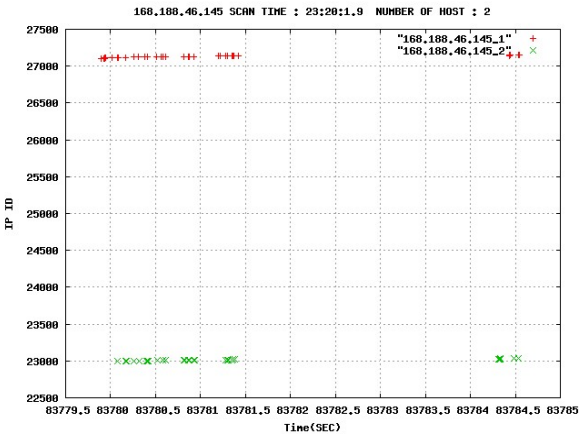



그림 8. NAT에 연결된 호스트 a

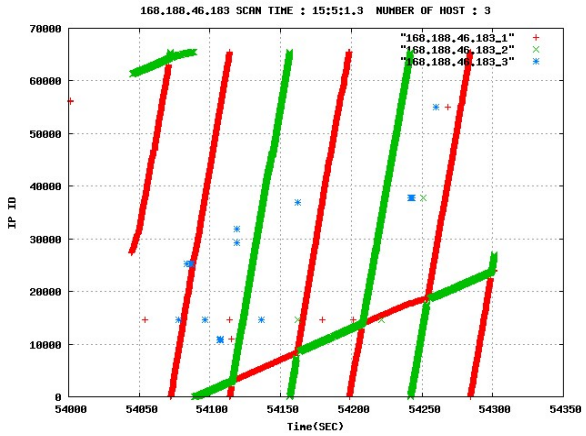



그림 9. NAT에 연결된 호스트 b

측정결과 그림 8과 같이 육안으로 보기에도 정확히 호스트의 수가 식별되는 것이 있는 것도 있지만, 그림 9처럼 육안 상으로 구별해내는 것이 어려운 경우도 있었다. IP ID의 리스트가 어떤 인접한 시간대에서 서로 만나게 되었을 때 나타나는 것으로 단순 선형 회귀분석을 통해 해당 지점은 개선이 가능하나, 단순 선형 회귀분석은 IP ID의 패턴을 추적하기가 어렵다. NAT 장치가 아닌 여러 대의 호스트가 NAT를 사용하는 것처럼 탐지가 되는 원인은 Red Hat 9.0 Linux 커널 2.4이상에서 랜덤하게 사용하는 IP ID 값 때문이다.

그림 10은 Windows XP 컴퓨터 2가 Red Hat 9.0에 공유기를 통해 FTP 접속을 하여 파일을 동시에 다운로드 했을 때 Red Hat 9.0 리눅스의 IP ID 패턴이다. 그림 10에서와 같이 IP ID를 기반으로 분석을 하였을 때 대상 Network에 Linux Kernel 2.4 이상의 시스템이 있을 경우 NAT장치 탐지오류는 증가한다. 특히 NAT 장치 탐지를 기반으로 트래픽을 측정할시 해당 시스템이 파일 서버인 경우 트래픽 측정 결과의 오차는 커지게 된다.

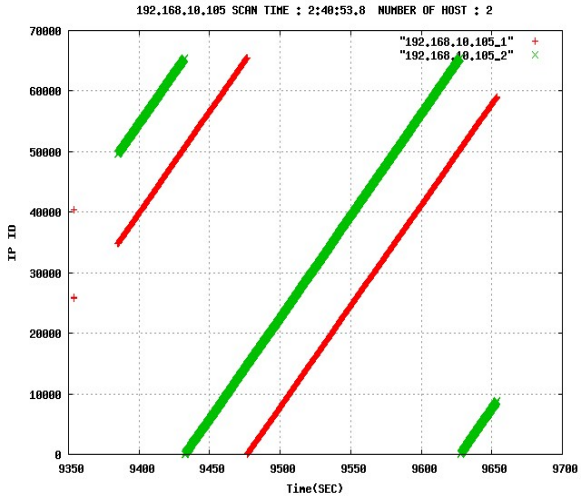



그림 10. Linux의 랜덤한 IP ID 생성

## 5. 결 론

본 논문에서는 NAT 탐지 알고리즘의 장단점을 충남대학교 캠퍼스 망 실험을 통하여 정확도와 한계점을 보였다. TTL을 조작할 수 있기 때문에 정확도가 떨어지면, IP ID를 이용한 방법은 Linux와 같은 랜덤 IP 사용에 따른 오류가 발생할 수 있다는 것을 알 수 있었다. 향후 연구에서는 passive OS fingerprint[9]를 이용하여 IP ID 뿐만 아니라 TTL, Window Size, DF flag, ToS값 등을 활용하여 OS 타입을 확인하고 복합적으로 NAT를 탐지함으로서 탐지 정확도를 향상시키도록 하여야한다.

## 참 고 문 헌

[1] K. Egevang and P. Francis,"The IP Network Address Translator(NAT)",IETF, RFC 1631, May 1994.

[2] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot and E. Lear,"Address Allocation for Private Internets", IETF RFC1918, Feb 1996.

[3] Cisco Press,"Building Cisco Remote Access Networks", Aug 1999.

[4] S. M. Bellovin,"A Technique for Counting Natted hosts", Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement, 2002.

[5] P. Srisuresh and K.Egevang,"Traditional IP network address translator(traditional NAT)", IETF RFC, Jan 2001.

[6] 디지털 타임스, "KT, IP 공유기 사용금지 '초강수'", 2004. 8.

[7] 아이뉴스24, "IP 공유기 '허용' 방안, 갑론을박, 국회 간담회", 2005. 3.

[8] Peter Phaal,"Detecting NAT Devices using sFlow", http://www.sflow.org/detectNAT/.

[9] J.Postel,"Internet protocol", IETF, RFC 791, Oct 1981.

[10] Honeynet Project, "Know your enemy: Passive fingerprinting",http://project.honeynet.org/papers/finger. March 2002.