네트워크 트래픽 기반 인터넷 응용의 동작형태 분석*

박진완, 윤성호, 박준상, 김명섭
고려대학교 컴퓨터정보학과
e-mail : {pakjw84, hiption, runtoyou, tmskim}@korea.ac.kr

Behavior Analysis of Internet Applications based on Network Traffic

Jin-Wan Park, Sung-Ho Yoon, Jun-Sang Park, Myung-Sup Kim
Dept. of Computer and Information Science, Korea University

요 약

네트워크 트래픽의 응용 별 분류는 최근 학계의 중요한 이슈 중 하나이다. 기존의 전통적인 트래픽 분류 방법으로 대표되는 well-known 포트 기반 분류 방법 및 페이로드 시그니쳐 기반 분류 방법의 구조적 한계점을 극복하기 위한 새로운 대안으로써, 트래픽의 상관관계를 통한 분류 방법이 제안되었다. 본 논문에서는 트래픽 상관관계에 대한 정형화된 식이나 룰을 찾는데 유용한 정보를 제공하기 위해 인터넷 응용 별 트래픽을 동작형태의 관점에서 분석하였다. 학내 망에서 자주 사용되는 인터넷 응용을 선정하고, 이들이 실행 초기에 발생시키는 트래픽을 플로우와 패킷 단위로 분석한 내용을 기술하였다. 특히, 인터넷 응용이 발생시키는 플로우 중 페이로드가 존재하는 첫 플로우를 first talk 라 정의하였으며, 이에 대한 상세한 분석 내용을 기술하였다.

1. 서론

네트워크의 발달과 개인 및 기업의 인터넷 의존도가 높아지면서 SLA(Service Level Agreement)[1] 및 QoS(Quality of Service)[2] 등의 다양한 분야에서 트래픽 분류에 대한 중요성이 날로 높아지고 있다. 분류 기준 또한 다양한 분야에 적용하기 위해 상세 분류에 해당하는 응용 별 분류를 요구하기 때문에 네트워크 트래픽의 응용 별 분류는 최근 학계의 중요한 이슈 중 하나이다.

인터넷 응용 별로 트래픽을 분류하기 위한 기존의 전통적인 방법들 중 대표적인 방법으로 well-known 포트 기반 분류 방법과 페이로드 시그니쳐 기반 분류 방법이 있다. Well-known 포트 기반 분류 방법은 IANA[3]에서 지정한 포트 번호를 이용하는 방법이다. 이 방법은 비교적 단순하게 트래픽을 분류할 수 있다는 장점을 가지지만, well-known 포트 번호를 다른 목적으로 이용하거나 포트 번호를 동적으로 할당하는 최근의 인터넷 응용에 대해서는 정확한 분류가 어렵다. 페이로드 시그니쳐 기반 분류 방법은 특정 인터넷 응용에서 발생시킨 페이로드를 분석하여 다른 인터넷 응용과 구분 지을 수 있는 substring, 즉 시그니쳐를 추출한 후, 이 시그니쳐를 통해 트래픽을 분류한다. 이 방법은 시그니쳐를 추출한 인터넷 응용에 대해서 높은 정확도를 보이지만, 패킷의 암호화나 둘 이상의 응용이 동일한 시그니쳐를 갖는 등의 문제로 시그니쳐를 추출하지 못하는 인터넷 응용에 대해서는 분류가 어렵다는 단점을 가지고 있다.

이러한 기존의 문제점들을 해결하기 위해 최근에는 트래픽의 상관관계를 통해 트래픽을 분류하고자 하는 연구가 주목을 받고 있다[4-5]. 트래픽 상관관계를 이용한 방법은 주소체계(IP, 포트, 프로토콜), 트래픽 발생 시점, 발생 형태 등의 특성을 바탕으로 트래픽 사이에 연관성을 찾아 트래픽을 분류하는 방법이다. 하지만 아직 정형화된 식이나 룰이 없어 실제 네트워크 트래픽에 적용하기가 어렵다.

이에 본 논문에서는 트래픽 상관관계에 대한 정형화된 식이나 룰을 찾는데 유용한 정보를 제공하기 위해 인터넷 응용 별 트래픽을 동작형태의 관점에서 분석하였다. 인터넷 응용 별 트래픽 분석은 학내 망에서 외부 인터넷 망으로 오가는 트래픽을 대상으로 플로우와 패킷 단위로 이루어지며, 분석 대상이 되는 인터넷 응용은 학내 망에서 자주 사용되는 10 개의 응용들로 선정하였다. 선정된 인터넷 응용의 실행 초기 트래픽에 대해 전반적인 특성을 기술하였으며, 인터넷 응용이 발생시키는 플로우 중 페이로드가 존재하는 첫 플로우를 first talk 라 정의하고, first talk 에 대해 상세히 분석하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 선정된 인터넷 응용과 응용 별 트래픽 수집 방법을 기술하고, 3 장에서는 수집된 응용 프로세스 별 초기 트래픽 및 first talk 에 대한 분석 내용을 기술한다. 마지막으로 4 장에서는 결론 및 향후 연구를 제시한다.

2. 인터넷 응용 별 트래픽 수집 방법

2 장에서는 선정된 인터넷 응용에 대한 설명과 응용 별 트래픽 정보를 수집하는 방법에 대해 기술한다.

* 이 논문은 2007 년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임.(KRF-2007-331-D00387)

2.1 인터넷 응용의 선정

본 논문에서 분석한 인터넷 응용은 표 1 과 같다.

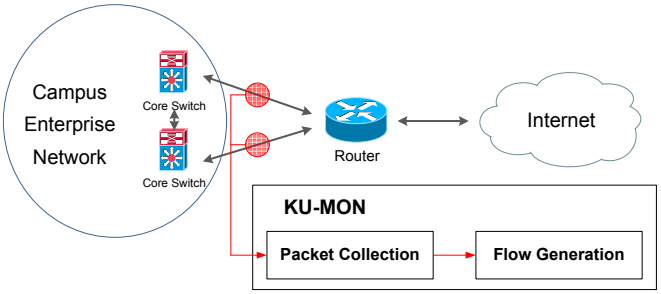
<표 1> 선정된 인터넷 응용

Application Type	Application	Main Process
P2P Messenger	Msn Messenger	msnmsgr.exe
	NateOn Messenger	nateonmain.exe
	Skype	skype.exe
P2P File Sharing	Fileguri	filegurimain.exe
	Pruna	pruna.exe
P2P Streaming	Afreeca TV	afreecaplayer.exe
	Gom player	gom.exe
Audio Streaming	Gorealra	gorealra.exe
Web Disk	Exfile	exfiledown2.exe
Gaming	Starcraft	starcraft.exe

응용의 선정은 학내 망에서 자주 사용되는 응용 중 임의로 10 개를 선정하였다. 선정된 응용은 주로 P2P 기술을 이용하여 다양한 서비스를 제공하는 응용들이다. 응용은 서비스를 제공하기 위해 다수의 프로세스를 생성하기도 한다. 본 논문에서는 인터넷 응용의 프로세스 중 중요한 기능을 하는 프로세스에 의해 발생된 트래픽에 대해서만 분석한다.

2.2 트래픽 수집 장소

트래픽의 수집은 학내 망에서 인터넷으로 오가는 모든 트래픽을 대상으로 수집하였다. 그림 1 은 트래픽을 수집한 장소를 나타내고 있다. 본 학내 망의 인터넷 접속점은 인터넷으로 향하는 라우터와 그 하단에 두 대의 코어 스위치로 구성되어 있다. 트래픽 수집은 본 연구실에서 개발한 KU-MON[6]이라는 시스템을 통해 이루어지며, 라우터와 두 대의 코어 스위치 사이에서 트래픽을 수집한다. 모든 패킷을 1 분 단위로 수집하며, 수집된 패킷들을 통해 1 분 단위의 플로우 정보를 생성한다. 플로우는 일반적으로 사용되는 5-tuple 정보(source IP, source port, destination IP, destination port, protocol)가 동일한 단방향 패킷들의 집합으로 정의한다. 그리고 플로우의 패킷 중 페이로드가 존재하는 패킷에 대해서는 먼저 발생되는 순서로 최대 10 개까지 패킷의 페이로드 정보를 저장한다.

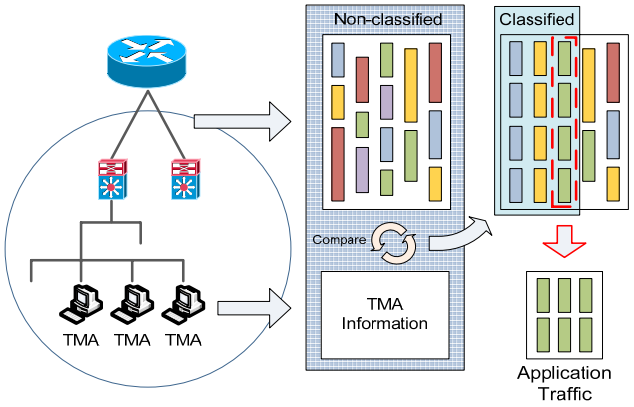


(그림 1) 트래픽 수집 장소

2.3 프로세스 별 트래픽 분류

특정 프로세스에서 발생하는 트래픽을 분석하기 위해서는 우선 프로세스 별 트래픽을 수집하는 작업이 선행되어야 한다. 이를 위해 본 논문에서는 학내 망의 종단 호스트에 TMA(Traffic Measurement Agent)[7]를 설치하고 이를 이용하여 트래픽을 프로세스 별로 분류한다.

그림 2 는 TMA 를 이용한 트래픽 분류 방법을 나타낸다. TMA 는 해당 호스트의 현재 활성화된 소켓 정보를 토대로 TMA 정보(process name, path, source IP, source port, destination IP, destination port, protocol)를 제공해 준다. TMA 정보를 이용하여 학내 망에서 수집된 플로우와 패킷이 어떤 프로세스에 의해 발생되었는지 판단할 수 있다.



(그림 2) 프로세스 별 트래픽 분류

본 연구에서는 프로세스 별 트래픽을 수집하기 위해 먼저 특정 호스트에 TMA 를 설치하고 선정된 인터넷 응용을 실행시킨다. 그런 다음 학내 망에서 수집된 플로우와 패킷에서 실험 호스트의 해당 프로세스가 발생시킨 플로우와 패킷을 분류해 낸다.

3. 응용 프로세스 별 초기 트래픽 분석

이 장에서는 인터넷 응용의 메인 프로세스가 실행 초기에 발생시킨 플로우와 패킷에 대해 분석한 내용을 기술한다.

실험 방법은 표 1 에 나타나 있는 10 개의 인터넷 응용들을 학내 망의 특정 호스트에서 30 회씩 시작과 종료를 반복하여 실행시키고, 해당 트래픽을 분석한다. 본 연구에서 초기 트래픽이란 프로세스가 실행되고 발생되는 첫 플로우의 시작시간을 기준으로 플로우의 시작 시간이 10 초 이내인 플로우들과 그 플로우에 속하는 모든 패킷들을 말한다.

3.1 프로세스 별 초기 트래픽의 개요

<표 2> 프로세스 별 초기 트래픽

σ = Standard Deviation

Process Name (.exe)	Prot	In/ Out	Flow		Pkt		Byte	
			average	σ	average	σ	average	σ
afreeca player	TCP	In	60.4	6.5	1593.0	105.9	1365325.4	149597.7
		Out	60.0	6.6	1483	70.6	858086	20172.4
exfile down2	TCP	In	8.0	0.0	46.1	4.8	15587.6	6798.7
		Out	8.0	0.0	51.3	2.7	9796.7	171.2
fileguri main	TCP	In	7.5	1.0	152.9	12.9	143362.7	11027.6
		Out	7.5	1.0	104.1	9.8	23912.0	2633.1
gom	TCP	In	27.8	1.4	238.3	20.3	149090.1	22847.3
		Out	27.8	1.4	188.2	13.3	21735.7	1380.4
gorealra	TCP	In	20.7	1.9	200.4	23.5	149185.0	21249.7
		Out	20.7	1.9	205.5	22.1	55243.4	7571.5
msnmsgr	TCP	In	11.3	1.2	194.8	111.4	159314.3	165184.4
		Out	11.3	1.2	145.0	60.6	27868.0	4521.3
	UDP	In	1.0	0.0	1.0	0.0	66.0	0.0
		Out	1.1	0.3	1.1	0.3	74.3	22.8

nateon main	TCP	In	20.1	0.5	163.4	4.6	92153.1	1332.0
		Out	20.1	0.5	152.9	5.9	19233.8	536.4
pruna	TCP	In	25.6	2.9	234.2	41.7	130399.4	50079.2
		Out	26.1	2.9	236.2	28.6	60393.5	5097.2
skype	TCP	In	3.0	0.0	24.8	2.4	3275.9	334.5
		Out	3.0	0.0	28.4	2.9	2902.6	347.8
	UDP	In	33.8	5.3	53.7	6.7	12654.5	2328.1
		Out	36.9	5.9	53.2	6.3	10404.3	2405.7
starcraft	TCP	In	1.4	0.5	45.9	8.9	7041.9	3684.0
		Out	1.4	0.5	32.2	6.7	2839.2	453.9
	UDP	In	1.0	0.0	8.0	0.0	512.0	0.0
		Out	1.0	0.0	3.0	0.0	192.0	0.0

표 2 는 프로세스의 초기 플로우 정보를 분석한 내용이다. 30 회의 실험을 플로우 개수, 패킷 개수, 바이트 크기로 평균과 표준 편차 값을 계산하였다. 이 때 계산 값은 소수점 두 번째 자리에서 반올림하였다. 그리고 평균과 표준편차를 protocol, inbound(외부 인터넷 망에서 학내 망으로 들어오는 트래픽)/outbound(학내 망에서 외부 인터넷 망으로 나가는 트래픽)로 나누어 표현하였다.

초기 트래픽을 분석한 결과는 다음과 같다.

1) 프로세스의 초기 바이트 크기는 outbound 보다 inbound 가 크다. 이는 인터넷 응용이 실행 시 기본적으로 필요한 데이터를 서버에 요청하여 다운받는 서버/클라이언트의 형태를 가지기 때문이다. 조사한 인터넷 응용들은 메신저, P2P 파일공유, 스트리밍 등 각기 다른 종류에 해당하는 응용이기에 메인 트래픽들은 각기 다른 형태의 모습을 나타낼 수 있지만, 초기 트래픽에 대해서만큼은 서버/클라이언트 형태의 모습을 나타낸다.

2) 표준 편차를 통해 프로세스가 매번 실행될 때마다 발생시키는 초기 트래픽의 양이 다른 것을 확인할 수 있다. 이는 두 가지 원인으로 분석된다. 하나는 응용을 실행할 때마다 특정 기능을 위해 발생되는 플로우가 매번 정확하게 동일하지 않기 때문이다. 실제 패킷의 페이로드를 통해 확인해 본 결과 전송되는 내용이 매번 차이가 있었다. 또 다른 하나는 초기 트래픽을 10 초 이내라는 시간 단위로 정의하였기 때문이다. 서버 혹은 클라이언트에 해당하는 호스트의 CPU 처리 속도나 네트워크 환경에 따라 패킷이 발생하는 시간이 달라진다. 그 결과로 인해 어떤 플로우는 초기 트래픽에 해당될 수도 있고 아닐 수도 있다.

3) Inbound 의 표준 편차가 outbound 의 표준 편차보다 크다. 이는 각 프로세스가 서비스를 제공하기 위해 서버로 요청 메시지를 보내는 내용은 매번 비슷하지만 그에 따른 응답 메시지의 내용은 매번 차이가 크다는 것을 나타낸다.

3.2 First talk 의 정의

본 논문에서는 first talk 라는 용어를 제시한다. First talk 란 프로세스가 발생시키는 플로우 중 페이로드가 존재하는 첫 번째 플로우를 말한다. 대부분의 인터넷 응용은 서버로의 접속을 시작으로 트래픽을 발생시키므로 first talk 는 대부분 outbound 트래픽에 속한다.

지금부터는 선정된 응용 프로세스의 first talk 에 대한 IP 와 포트, 패킷의 크기 분포에 대해 분석한다.

3.3 First talk 의 IP 와 포트

본 논문에서는 학내 망 내의 호스트를 local 호스트, 그 외 학내 망 밖에 있는 호스트를 remote 호스트라 지칭한다.

<표 3> 프로세스 별 first talk 의 IP 와 포트

Process Name (.exe)	Prot	Local Port	Remote IP	Remote Port
filegurimain	TCP	1025 – 5000	xxx.xxx.48.131	80
gom	TCP	1025 – 5000	xxx.xxx.77.18	80
gorealra	TCP	1025 – 5000	xxx.xxx.138.61	80
nateonmain	TCP	1025 – 5000	xxx.xxx.253.91	5004
pruna	TCP	1025 – 5000	xxx.xxx.58.116	80
starcraft	TCP	1025 – 5000	xxx.xxx.0.54 ~ xxx.xxx.0.64	6112
			xxx.xxx.0.72 ~ xxx.xxx.0.74	
exfiledown2	TCP	1025 – 5000	xxx.xxx.196.10	80
			xxx.xxx.196.12	
			xxx.xxx.196.14	
			xxx.xxx.196.18	
			xxx.xxx.196.18	5612
msnmsgr	TCP	1025 – 5000	207.xxx.28.93	1863
			65.xxx.239.140	
			65.xxx.165.179	443
afreecaplayer	TCP	1025 – 5000	121.xxx.76.75	4004
			121.xxx.76.74	
			222.xxx.54.59	
			218.xxx.31.51	
skype	UDP	16658	118.xxx.160.187 220.xxx.154.162 …	58094 34744 …

표 3 은 프로세스 별 first talk 의 프로토콜, local 포트, remote IP, remote 포트에 관한 내용을 정리한 표이다. First talk 에 대한 실험을 학내 망의 특정 호스트에서만 실시하였기 때문에 local IP 정보는 일정하므로 생략하였다.

Local 포트는 TCP 의 경우 운영체제에서 할당하는 동적 포트를 사용하고, UDP 의 경우에는 고정된 포트를 사용한다.

Filegurimain, gom, gorealra, nateonmain, pruna 는 항상 동일한 remote IP 와 포트로 접속을 하지만, 나머지 프로세스들은 실행 시마다 다르다. Remote IP 를 살펴보면 starcraft 와 exfiledown2 는 일정한 IP 범위 내의 서버에 접속하지만, msnmsgr, afreecaplayer, skype 는 그렇지 않다. 또한 Remote 포트는 skype 를 제외한 모든 응용에서 1 개 또는 2 개의 포트 번호가 관측되었다.

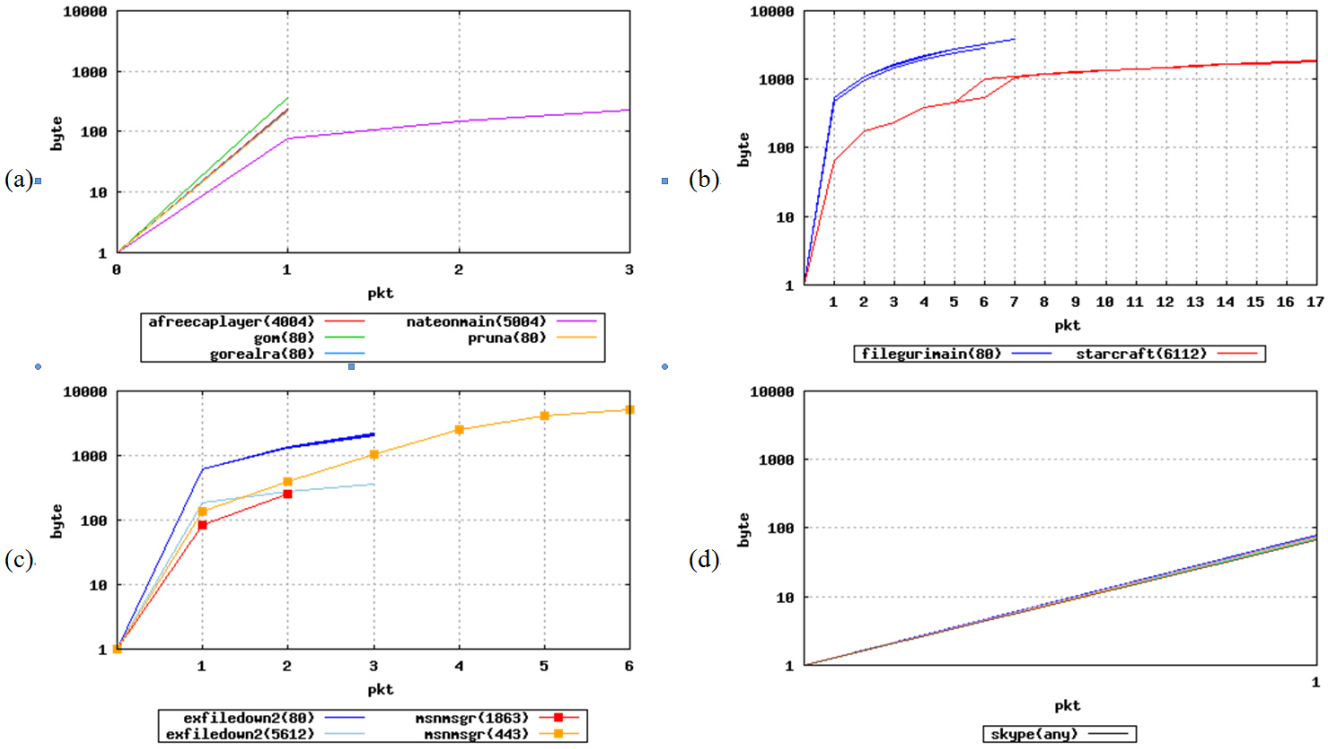
3.4 First talk 의 패킷 크기 분포

그림 3 은 프로세스/포트 별 first talk 의 패킷 크기 분포를 나타낸다. 3.4 절에서 말하는 패킷은 페이로드가 존재하는 패킷을 말한다.

총 4 가지 종류의 그래프가 다음과 같이 나타났다.

(a) 패킷 개수와 크기가 모두 동일한 경우
(b) 패킷 개수와 크기가 약간씩 다른 경우
(c) Remote 포트가 여러 개이고 포트 별 패킷 개수와 크기가 동일한 경우
(d) Remote 포트가 여러 개이고 포트 별 패킷 개수와 크기가 약간씩 다른 경우

(c)의 경우에 속하는 프로세스는 exfiledown2 와 msnmsgr 로써 first talk 가 두 개의 remote 포트를 가지는 경우이다. 포트 번호와 패킷의 크기가 전혀 다른 것으로 보아 서로 다른 기능을 위한 플로우로 유추할 수 있다. 다른 종류의 first talk 가 나타난 것은 응용의



(그림 3) 프로세스 별 first talk 의 패킷 크기에 대한 CDF

여러 기능들이 스레드로 처리되어 CPU 처리 순서에 따라 플로우의 발생 순서가 달라지는 것이 원인이라 판단된다.

(d)의 경우인 skype 는 remote IP 와 포트의 쌍이 매번 바뀌지만, first talk 의 패킷 개수는 동일하고, 크기 또한 일정한 범위 안에 속하는 것으로 관측되었다.

3.5 First talk 의 분석 결과

First talk 는 다음과 같은 특징을 가진다.

첫째, first talk 는 여러 개가 될 수 있다. 하나의 프로세스가 여러 가지 기능을 제공하기 위해 스레드를 사용함으로써 first talk 가 달라질 수 있다.

둘째, TCP 일 경우는 remote 포트, UDP 일 경우에는 local 포트가 동일한 first talk 에 대하여 패킷 크기 분포가 동일하거나 약간씩 차이를 보인다. 이는 first talk 의 패킷 크기 분포가 프로세스의 first talk 를 분류할 수 있는 시그니쳐로 사용될 수 있다는 것을 의미한다.

본 논문의 분석 결과는 특정 네트워크의 특정 호스트에 대한 트래픽을 분석하였고, 인터넷 응용 별로 설정이나 환경을 동일하게 한 상태에서 실험을 행하여 제한적인 요소가 있다. 하지만, 각 프로세스의 트래픽을 실제 분석하면서 인터넷 응용이 가지는 동작형태의 특징이 존재한다는 것을 파악할 수 있었다.

4. 결론

트래픽 상관관계를 이용한 트래픽 분류 방법론은 아직 정형화된 식이나 룰이 없어 실제 네트워크에 적용하기 힘들다. 이에 본 논문에서는 트래픽 상관관계에 대한 정형화된 식이나 룰을 찾는데 유용한 정보를 제공하기 위해 인터넷 응용 별 트래픽을 동작형태의 관점에서 분석하였다. 응용 별 초기 트래픽의 전반적인 특성에 대해 분석하였으며, first talk 에 대한 정의와 이에 대한 분석 결과를 제시하였다. 분석 결과를 통해 first talk 의 패킷 크기 분포가 동작형태 시그니쳐로 사용될 수 있는 가능성을 보였다.

향후 연구에서는 패킷 크기 분포가 시그니쳐로 사용될 수 있는 지의 여부에 대해 연구하고, 분석한 10 개의 응용 외에 많은 응용들을 분석하여 동작형태의 다양한 특징을 찾는데 노력을 기울이겠다.

참고문헌

[1] TM Forum, "Service Level Agreement Management Handbook," GB917 v1.5, Jun., 2001.

[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services," IETF RFC2475, Dec., 1998.

[3] IANA port number list, IANA, http://www.iana.org/assignments/port-numbers.

[4] Myung-Sup Kim, Young J. Won, and James Won-Ki Hong, "Application-Level Traffic Monitoring and an Analysis on IP Networks," ETRI Journal, Vol.27, No.1, Feb., 2005, pp. 22-42.

[5] T. Karagiannis, K.P apagiannaki and M.F aloutsos, "BLINC: Multilevel Traffic Classification in the Dark," in Proc. of ACM SIGCOMM, Aug., 2005.

[6] 박상훈, 박진완, 김명섭, "Flow 기반 실시간 트래픽 수집 및 분석 시스템", 정보처리학회 추계학술대회, 목포대학교, 전주, Nov. 9-10, 2007, pp. 1061.

[7] 윤성호, 노현구, 김명섭, "TMA(Traffic Measurement Agent)를 이용한 인터넷 응용 트래픽 분류", 통신학회 하계종합학술발표회, 라마다플라자호텔, Jul. 2-4, 2008, pp. 618.