통계 시그니쳐 기반의 응용 트래픽 분류

준회원 박 진 완*, 윤 성 호*, 박 준 상*, 학생회원 이 상 우**, 종신회원 김 명 섭***°

Statistic Signature based Application Traffic Classification

Jin-wan Park*, Sung-ho Yoon*, Jun-sang Park* *Associate Members*,
Sang-woo Lee** *Student Member*, Myung-sup Kim***° *Lifelong Member*

요 약

오늘날의 네트워크에서는 다양한 응용의 등장으로 인해 트래픽이 복잡 다양해지고 있다. 이러한 상황 속에서 트래픽의 응용 별 분류에 대한 중요성은 날이 갈수록 증가하고 있다. 트래픽의 응용 별 분류에 대한 요구에 따라 기존에도 많은 연구가 이루어졌었다. 포트 기반의 분류, 페이로드 기반의 분류, 머신러닝 기반의 분류 방법들이 제안되었는데 아직 트래픽을 완벽하게 분류해내는 방법론은 개발되지 않은 실정이다. 최근 연구 중에는 플로우의 통계 정보를 이용한 방법론이 많이 연구되고 있다. 본 논문에서는 통계 시그니쳐를 통한 응용 트래픽 분류 방법론을 제안하고자 한다. 플로우 중 첫 N개의 패킷의 페이로드 크기와 방향을 이용하여 통계 시그니쳐를 생성하고, 이를 이용하여 응용 트래픽을 분류한다. 그리고 검증 시스템을 통해 본 분류 방법론이 높은 정확도의 분류 방법론이라는 것을 보인다.

Key Words : Traffic Classification, Statistic Signature, Application Traffic

ABSTRACT

Nowadays, the traffic type and behavior are extremely diverse due to the appearance of various services and applications on Internet, which makes the need of application-level traffic classification important for the efficient management and control of network resources. Although lots of methods for traffic classification have been introduced in literature, they have some limitations to achieve an acceptable level of performance in terms of accuracy and completeness. In this paper we propose an application traffic classification method using statistic signatures, defined as a directional sequence of packet size in a flow, which is unique for each application. The statistic signatures of each application are collected by our automatic grouping and extracting mechanism which is mainly described in this paper. By matching to the statistic signatures we can easily and quickly identify the application name of traffic flows with high accuracy, which is also shown by comprehensive excrement with our campus traffic data.

Ⅰ. 서 론

오늘날의 네트워크에서는 다양한 응용의 등장으로 인해 트래픽이 복잡 다양해지고 있다. 이러한 상황 속에서 네트워크의 효과적인 운용과 관리를 위해 네트워크 트래픽 분석은 필수적인 요소가 되었다. 네트워크의 의존도가 앞으로 계속 증대됨에 따라 트래픽 분석의 중요성은 계속 커질 전망이다[1]-[2].

※ 이 논문은 2007년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2007-331-D00387)
* 고려대학교 컴퓨터정보학과 석사과정({jinwan_park, sungho_yoon, junsang_park}@korea.ac.kr)
** 고려대학교 컴퓨터정보학과 학사과정(sangwoo_lee@korea.ac.kr)
*** 고려대학교 컴퓨터정보학과 조교수(tmskim@korea.ac.kr)(°:교신저자)
논문번호 : KICS2009-08-371, 접수일자 : 2009년 8월 21일, 최종논문접수일자 : 2009년 10월 15일

엔터프라이즈 네트워크나 캠퍼스 네트워크에서는 네트워크 자원의 효율적인 운용 및 관리를 위해서 다양한 정책들을 마련하고 있다. 예를 들면, 학교나 공공 기관에서 P2P 트래픽, 게임 트래픽 등 네트워크 자원을 많이 차지하거나 업무와 관련이 없는 트래픽들을 제한하는 정책들이 있다. 이러한 정책들을 펼치기 위해서는 빠르고 정확한 응용계층 트래픽 분류가 필수적인 요소이다[1]-[5]. 응용계층 트래픽 분류란 네트워크 패킷을 수집하여 해당 패킷을 발생시킨 응용계층의 정체를 알아내고 이를 기준으로 트래픽을 분류하는 것을 말한다[1],[7]. 실시간의 정확한 응용계층 트래픽 분류는 응용기반의 트래픽을 모니터링하고 제어하는 다양한 분야 (응용 별 종량제 과금, 응용기반 트래픽 제어, CRM, SLA 지원, 응용계층 트래픽 보안 등)에서 신뢰성을 결정하는데 중요한 역할을 한다.

이러한 응용계층 트래픽 분류의 중요성에 의해 현재까지 많은 연구들이 진행되어 왔다. 기존의 분류 방법들로는 포트 기반 분석[5]-[6], 페이로드 시그니쳐 기반 분석[8], 머신러닝 기반 분석[2]-[3], 플로우 상관관계 기반 분석[1],[4] 등 다양한 방법론들이 존재한다. 이러한 다양한 방법론들이 있지만, 새로운 응용 프로그램의 등장, 응용 프로그램의 업데이트 등 응용의 변화에 따라 트래픽이 변하고 있기 때문에 오늘날의 응용 별 트래픽 분류는 더욱 더 힘들어지고 있다. 따라서, 보다 정확한 응용 별 트래픽 분류를 위하여 다양한 분류 방법론에 대한 지속적인 연구가 필요하다.

최근 연구[12]-[14]에서는 기존의 방법론들의 단점을 극복하고 빠르고 정확한 응용계층 트래픽 분류를 위해 패킷들의 통계 정보를 이용한 분류 방법을 연구 중이다. 통계 정보를 이용한 분류 방법은 패킷 크기, 패킷 간의 시간, 윈도우 크기 등에 대한 통계적 분포를 바탕으로 통계 시그니쳐를 생성해 내고, 이를 바탕으로 트래픽을 분석하는 방법론이다. 이 방법론은 최근 증가하고 있는 암호화된 트래픽의 분석에 용이하며, 패킷의 페이로드 정보를 분석하지 않기 때문에 트래픽을 빠른 속도로 분류할 수 있다는 장점을 가진다. 하지만, 통계 정보를 생성하기 위해 플로우가 끝날 때까지 기다려야 한다는 점이 실시간 트래픽 분류를 어렵게 하고 있다.

본 연구에서는 플로우의 페이로드가 존재하는 첫 N개의 패킷의 페이로드 크기와 방향을 이용하여 트래픽을 분류하는 방법론을 제시한다. 본 방법론은 3가지의 장점을 가진다.

첫째, 분류를 행한 트래픽에 대해서는 높은 정확도를 보인다. 비록 페이로드 크기와 방향 만으로 모든 트래픽을 분류할 수는 없지만, 분류를 행한 트래픽 만큼은 정확하게 분류해 낸다. 네트워크 트래픽 제어와 같은 분야에서는 트래픽을 분류하지 못하는 것보다 잘못 분류하는 것이 더 큰 문제를 야기시킨다. 따라서, 본 시스템은 이러한 분야에 활용될 수 있다.

둘째, 기존에도 패킷의 크기와 방향을 통해 트래픽을 분류하는 연구[9]-[10]가 진행되어 왔지만, 실험에 사용된 응용의 종류가 다양하지 않아, 해당 방법론에 대한 실제 적용 가능성이 크게 나타나지 않았다. 본 논문에서는 실제 네트워크에서 사용되는 다양한 응용의 통계 시그니쳐를 생성하고 이를 검토한다.

셋째, 실시간 트래픽 분류에 적합한 방법론이다. 본 방법론에서는 플로우의 첫 N개의 페이로드 패킷에 대해서 검사를 행하고, 페이로드 정보를 분석하는 것이 아니라 페이로드의 크기와 방향만을 이용하므로 빠른 속도로 트래픽을 분류할 수 있다.

본 논문은 다음과 같은 순서로 구성된다. 2장에서는 트래픽 분류와 관련된 기존 연구들을 살펴보고, 3장에서는 본 연구에서 사용한 분류 기준과 트래픽 트레이스에 관해 설명한다. 4장에서는 통계 시그니쳐의 정의와 페이로드 크기 및 방향에 대한 트래픽 분류 가능성을 살펴보고, 5장에서는 이를 기반으로 한 분류 방법론을 기술한다. 6장에서는 검증 시스템을 통해 본 연구에서 개발한 분류 방법론을 평가한다. 마지막으로 7장에서는 결론을 맺고 향후 연구에 대해 언급한다.

Ⅱ. 관련 연구

응용 별 트래픽 분류는 다양한 응용들이 존재하는 오늘날의 인터넷 환경에서는 분명히 쉬운 일은 아니다. 과거의 인터넷에서는 포트 번호 1024 이하를 사용하는 HTTP, telnet, e-mail, FTP, SMTP의 응용들이 대부분의 인터넷 트래픽을 차지하고 있었기 때문에 IANA[5]에 정의된 포트 정보 기반의 분석으로 신뢰성과 정확성이 높은 분석 결과를 도출할 수 있었다. 그 후 스트리밍 응용 프로그램 및 passive FTP와 같이 하나의 응용 프로그램이 둘 이상의 세션을 형성하고, 이들 중 데이터 세션의 포트가 동적으로 생성됨에 따라 포트 기반의 분석은 더 이상 높은 신뢰성과 분석률을 제공할 수 없게 되었다.

이를 보완하기 위한 방법으로 응용계층 프로토콜 내용을 참조하여 동적으로 생성되는 포트 정보를 얻어내어 분석하는 방법인 mmdump[15], SM-MON[16]에서 소개되었다. 이러한 방법의 장점은 동적 세션에 대한 포트 정보를 프로토콜 내용 참조를 통해 정확하게 알 수 있어 분석한 결과의 신뢰성이 높다는 것이다. 그러나 이 방법은 RTSP, MMS, SIP와 같이 응용 프로토콜이 공개되었거나 알려진 응용 트래픽의 분석에만 사용 가능하고 전체 인터넷 트래픽에 적용할 수 없다는 문제점이 있다. 현재 인터넷 트래픽의 상당수를 차지하고 있는 P2P 프로그램을 비롯한 많은 응용 프로그램들은 응용 프로토콜 및 포트 정보를 공개하지 않아 응용 프로토콜 내용 참조를 통한 동적 포트 번호 추출 방법은 거의 사용할 수 없다.

인터넷 트래픽의 응용 프로그램 별 분류는 위와 같은 어려운 점을 해결하기 위해 여러 가지 방법들이 제시되고 있다. 기존의 방법론들은 크게 3가지로 구분할 수 있는데, 이들은 시그니쳐 기반 분석[7], 트래픽 상관관계 기반 분석[1],[4], 머신러닝 기반의 분석[2]-[3]이다.

첫째, 시그니쳐 기반 분석 방법[7]은 특정 응용 프로그램에서 발생시킨 트래픽을 분석하여 다른 응용 프로그램과 구분 지을 수 있는 시그니쳐라고 하는 특정 응용만의 특징을 추출하고 이를 통해 트래픽을 분류하는 방법이다. 이 방법은 시그니쳐를 추출한 응용에 대해서는 높은 정확도를 보이지만, 시그니쳐를 찾는 작업이 수작업으로 이루어져 응용 프로그램의 변화에 적절히 대처하지 못한다라는 단점을 가지고 있다. 또한, 시그니쳐를 확인하기 힘든 응용 프로그램들은 분류를 하지 못한다. 예를 들어, 패킷의 페이로드 분석을 통해 다른 응용과 구분 지을 수 있는 substring으로 트래픽을 분류하는 페이로드 시그니쳐 기반 분석은 암호화된 패킷에 대해서는 트래픽을 분류할 수 없다.

둘째, 트래픽 상관관계 기반 분석 방법[1],[4]은 주소체계(IP 주소, 포트 번호, 프로토콜), 트래픽의 발생 시점, 발생 형태 등의 특성을 바탕으로 트래픽 플로우들 사이에 연관성을 가중치로 표현하고 가중치의 임계값을 적용하여 트래픽을 분류하는 방법이다. 이 방법의 장점은 트래픽의 분류에 있어 응용들이 가지는 특징을 분석에 활용하여 분석률을 높일 수 있다는 것이다. 그러나 응용 별 특징의 활용에 대한 명확한 알고리즘이 없이 시행착오를 통해 최적의 분석률을 보이는 임계값을 찾기 때문에 실제 인터넷 트래픽에 적용하였을 경우 분석 결과에 대한 신뢰성을 보장하기 어렵다.

셋째, 머신러닝 기반의 분석 방법[2]-[3]은 응용 별 인터넷 트래픽의 특징이 될 수 있는 항목(port number, flow duration, inter-arrival time, packet size)들을 머신러닝의 classification, clustering 기법을 이용하여 트래픽을 분류하는 방법이다. Classification 기법에서는 Bayesian Network, Decision Tree가 주로 사용되고[3], clustering 기법에서는 EM(Expectation Maximization)[2]을 활용한 연구가 발표되었다. 이 방법의 장점은 머신러닝의 고급 알고리즘을 이용함으로써 트래픽을 응용 별로 분류함에 있어 다른 방법에 비해 보다 높은 분석률을 제공한다는 것이다. 그러나 제한된 범위의 응용 트래픽에 대하여 트래픽 데이터를 수집하고 분석하였다는 점이 모든 인터넷 트래픽에 적용하였을 경우 분석의 정확성이 떨어질 수밖에 없는 단점을 갖고 있다. 또한, 분석률이 높은 classification 기법의 경우 모든 트래픽이 훈련된 응용들로만 구분이 되기 때문에 새로운 응용이 나올 경우 유연하게 대처하지 못하는 단점을 가지고 있다.

본 연구에서 개발하는 통계 시그니쳐 기반의 분류 방법론은 시그니쳐 기반 분석에 해당한다. 본 연구에서와 마찬가지로 패킷의 크기와 방향을 이용한 분류 방법[9]-[10]이 기존에 제안되었지만, 다양한 응용에 대한 분류 결과가 미흡하여 실제 적용 가능성을 검증하기 힘들었다. 본 연구에서는 기존 연구에서보다 간단하면서도 빠르고 정확한 분류 방법을 제안하고, 실제 학내 네트워크에 적용하여 그 타당성을 검증한다.

Ⅲ. 분류 기준 및 트래픽 트레이스

본 장에서는 본 연구에서 사용한 트래픽 분류 기준과 시그니쳐를 생성하고 분류 대상이 되는 트래픽에 관해 설명한다.

3.1 분류 기준

트래픽 분류에서 가장 선행되어야 하는 과정은 트래픽 분류 기준을 정하는 것이다. 응용 트래픽 분류와 관련된 많은 논문에서 응용 프로토콜을 기준으로 트래픽을 분류하였지만, 본 논문에서는 응용 프로세스를 기준으로 하여 트래픽을 분류한다.

응용 프로세스 기준의 트래픽 분류는 기존의 응용 프로토콜 기준의 분류에 비해 상세한 분석이기

에 좀 더 다양한 분야에 활용될 수 있다는 장점을 가지고 있다.

3.2 트래픽 트레이스

본 연구에서 사용된 트래픽은 고려대학교(세종캠퍼스) 학내 망에서 인터넷으로 오가는 모든 트래픽을 대상으로 한다. 플로우는 5-tuple(source IP, destination IP, source port, destination port, protocol)을 기준으로 양방향의 플로우로 정의한다. 본 연구에서 분류에 사용되는 특징은 패킷의 방향을 포함하기 때문에 두 호스트 간 통신에서 서로 향하는 모든 트래픽을 수집해야 한다. 특히, 외부 네트워크와 연결되는 구간이 여러 곳인 네트워크에서는 모든 트래픽을 한 곳으로 모으는 작업이 선행되어야 할 것이다. 본 연구에서의 대상 네트워크인 학내 망은 하나의 라우터를 통해 인터넷과 연결되므로 모든 인터넷 트래픽을 수집하기 위해 하나의 라우터에서 미러링을 통해 수집하였다.

시그니쳐를 생성하기 위한 training date set은 프로세스를 기준으로 각각 플로우와 플로우에 해당하는 페이로드 패킷(페이로드가 존재하는 패킷)으로 이루어졌다. 이렇게 하나의 플로우와 거기에 따른 패킷들이 연속적으로 저장된 형태를 flow with pkt 이라고 부른다. 응용 계층의 트래픽을 분류하는 것이 목적이므로 전송 계층에만 영향을 받는 TCP 컨트롤 패킷(SYN, Keep-Alive, 데이터가 없는 ACK)은 제외하였다.

Training data set은 정확하게 프로세스 별로 수집하는 것이 필요하다. 이를 위해 ground truth를 이용한다. Ground truth는 분류된 결과를 검증하기 위해서도 사용이 된다. 즉, ground truth는 정확한 알고리즘에 의해 어떤 트래픽이 어떤 응용에 의해 발생되었는지 알아야 한다.

본 연구에서는 ground truth를 생성하기 위해 학내 망 내에 있는 일부 말단 호스트들에 TMA(Traffic Measurement Agent)[11]를 설치하고 이를 통해 해당 데이터를 생성한다. Agent를 통한 ground truth 생성 방법은 특정 분류 방법을 통해 분류한 결과를 사용한 것[4]보다 높은 신뢰성을 보장해 준다.

그림 1은 TMA를 통한 ground truth 생성 방법에 관한 내용이다. 트래픽은 특정 네트워크에서 인터넷으로 향하는 모든 트래픽을 수집할 수 있는 위치에서 모든 패킷을 수집한다. 수집된 패킷을 이용하여 Flow Generator가 플로우를 생성한다. TMS (Traffic Measurement Server)는 네트워크 내의

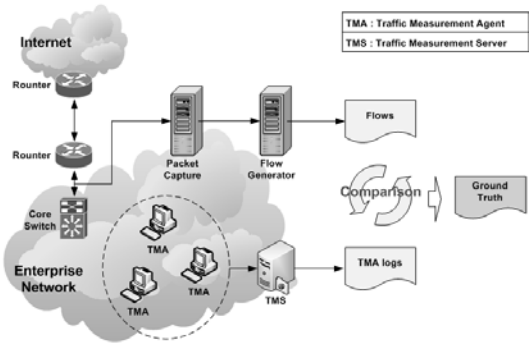


그림 1. TMA를 이용한 ground truth 생성 방법
Figure 1. Ground truth generation method using TMA

TMA가 설치된 여러 말단 호스트들로부터 TMA로그를 수집한다. TMA 로그 정보에는 해당 호스트에서 사용 중인 응용 프로세스의 이름과 열려진 포트 정보가 해당된다. 이를 통해 생성된 플로우와 비교하는 작업을 거치면 정확한 ground truth를 생성할 수 있다.

Ⅳ. 패킷 크기 분포

본 장에서는 통계 시그니쳐의 정의와 패킷의 페이로드 크기와 방향이 트래픽을 분류할 수 있는지에 대한 가능성을 검토한다.

4.1 통계 시그니쳐의 정의

통계 시그니쳐란 패킷의 헤더 정보(패킷 크기, 윈도우 크기 등)나 캡쳐 정보(패킷 캡쳐 시간 등)를 기반으로 하여 다른 응용 프로그램과 구별할 수 있는 응용 프로그램의 고유한 통계적 특징이다.

본 논문에서 이용한 통계 시그니쳐는 패킷의 페이로드 크기와 방향이다. 패킷의 페이로드 크기는 페이로드가 존재하는 패킷의 페이로드 크기만을 의미한다. 방향은 양수와 음수로 표현되며, TCP의 경우 양수는 클라이언트에서 서버로 향하는 패킷, 음수는 서버에서 클라이언트로 향하는 패킷을 의미한다. UDP는 서버/클라이언트의 구분이 명확하지 않기 때문에, 양수/음수의 의미는 단지 방향이 서로 반대라는 것만 표현할 수 있다. 따라서, UDP의 경우에는 첫 패킷을 양수로 표현하고 뒤에 이어지는 패킷은 첫 패킷을 기준으로 방향이 같으면 양수, 다르면 음수로 표현한다.

4.2 패킷의 페이로드 크기 분포

일반적으로 플로우의 초기 몇 개의 패킷은 응용

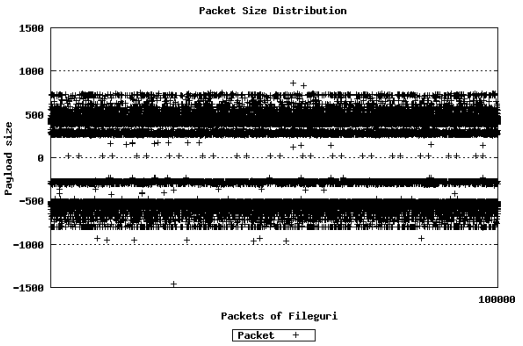


그림 2. 파일구리의 패킷 크기 분포
Figure 2. Packet size distribution of fileguri

프로그램에 의해 미리 정해진 정보를 전달하는 용도로 사용된다. 그러므로 플로우의 첫 N개의 패킷에 해당하는 페이로드 크기는 응용 프로그램마다 다를 가능성이 크다.

그림 2는 학내 망에서 자주 사용되는 응용 프로그램 중 파일구리라는 응용에 대해 모든 페이로드 패킷의 페이로드 크기 분포를 나타내고 있다. 가로축은 패킷의 수를 나타내며 총 100,000개의 패킷에 대해 표현하였다. 세로축은 4.1절에서 정의한 페이로드 크기와 방향을 나타내는 것으로 하나의 패킷은 -1460에서 +1460까지의 값을 갖는다. 다른 페이로드 크기보다 250~600, -600, -500~-700 정도의 크기가 대부분의 비중을 차지한다는 것을 살펴볼 수 있다. 이는 응용마다 특정한 패킷의 크기를 사용한다는 것을 알 수 있는 자료가 된다.

하지만, 모든 응용이 파일구리처럼 특정한 페이로드 크기를 갖는 것은 아니다. 그 대표적인 응용이 HTTP를 사용하는 인터넷 익스플로러이다. HTTP 플로우는 페이로드 크기 분포가 다양하기 때문에 페이로드 크기만으로는 트래픽 분류의 기준으로 삼을 수가 없다. 따라서, 본 논문에서는 HTTP 플로우를 제외하고 다른 응용에 대해서만 트래픽을 분류한다.

V. 분류 방법

본 장에서는 앞에서 살펴본 특징을 이용하여 트래픽을 분류하는 방법을 살펴본다. 본 연구의 분류 방법은 시그니쳐를 생성하는 단계와 트래픽을 분류하는 단계로 이루어져 있다. 그림 3은 전체적인 개요이다. 왼쪽의 시그니쳐 생성 단계는 트래픽 분류에 필요한 응용 프로그램마다의 통계 시그니쳐를

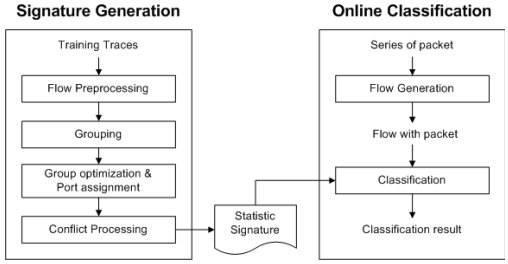


그림 3. 분류 방법론의 개요
Figure 3. Overview of classification methodology

생성하는 단계이며, 오른편의 실시간 분류 단계는 통계 시그니쳐를 통해 온라인 트래픽을 실시간으로 분석하는 단계이다. 시그니쳐 생성 단계와 실시간 분류 단계는 각각 5.1절과 5.2절에서 설명된다.

시그니쳐 생성 단계는 네트워크 내의 여러 호스트로부터 수집된 트래픽을 입력으로 한다. 먼저 수집된 트래픽을 플로우 단위의 그룹핑을 하기 위해 전처리 작업을 거친다. 전처리가 끝난 플로우들은 페이로드 크기와 방향을 이용하여 그룹핑을 한다. 그룹핑 된 그룹들은 최적화 작업과 포트 할당이라는 작업을 거치고, 마지막으로 각 그룹들 사이의 충돌을 해결함으로써 통계 시그니쳐를 생성한다.

실시간 분류 단계는 패킷들을 플로우로 만들고 시그니쳐 생성 단계에서 생성된 시그니쳐를 이용하여 트래픽을 분류한다.

5.1 시그니쳐 생성 단계

본 절에서는 통계 시그니쳐의 생성 방법에 대해 단계별로 기술한다.

5.1.1 플로우 전처리

플로우 전처리 단계는 시그니쳐 생성 단계의 입력인 training traces를 그룹핑하기 위한 형태로 만드는 작업을 행한다. Training traces의 플로우들은 몇 가지 요구 사항들이 있다. 첫째, 분류하고자 하는 응용의 트래픽이 모두 포함되어 있어야 한다. 둘째, 각 응용 별로 일정 개수 이상의 플로우가 있어야 한다. 너무 적은 양의 플로우로 생성된 시그니쳐는 신뢰성 하락과 응용 내의 분류 가능한 트래픽의 양을 줄이는 결과를 초래할 수 있다. 셋째, 응용 별 트래픽은 여러 호스트에서 수집되어야 한다. 어떤 응용이 하나의 호스트에 의해서 수집된 것이라면 트래픽의 특징이 해당 호스트의 의존적인 특징이 될 수 있다. 예를 들어, 오늘날의 다양한 응용 프로그램들은 자신의 포트 정보를 프로그램 내에서 개

별적으로 조절할 수 있는데, 사용자에 의해서 조절된 포트는 비록 그 호스트에서는 시그니쳐로 사용할 수 있지만, 다른 호스트에서 발생한 트래픽을 분류하기에는 시그니쳐로서 부족하다.

본 연구에서는 학내 망의 ground truth를 통해 타깃 프로세스들의 트래픽을 수집한다. 각 프로세스마다 1,000개 이상의 플로우를 수집하며, 프로세스 내 플로우의 선택은 각 프로세스마다 임의로 선택한다.

5.1.2 플로우의 표현

4.2절에서 패킷의 페이로드 크기와 방향을 통해 트래픽을 분류할 수 있다는 가능성을 보였다. 시그니쳐를 생성하고 이를 통해 트래픽을 분류하기 위해서는 플로우에 해당하는 첫 N개의 패킷의 페이로드 크기와 방향을 어떻게 표현할 것인지에 대해 결정해야 한다.

플로우의 첫 N개의 페이로드 패킷의 페이로드 크기와 방향을 표현하기 위해 벡터로 표현하는 방법을 사용한다. 플로우를 f라 하고, f의 i번째 페이로드 패킷의 페이로드 크기를 $s_i(f)$라 표현한다. $s_i(f)$는 패킷의 방향에 따라 양수와 음수 중 하나의 값을 갖는다. τ가 플로우를 N차원의 벡터로 표현하는 함수라고 정의하면, 식 1과 같이 정의할 수 있다.

$$\tau(f) = (s_1(f), ..., s_N(f)) \qquad \text{(식 1)}$$

그룹핑과 플로우의 분류를 위해 본 연구에서는 $\tau(f)$간의 거리를 기준으로 삼는다. 두 개의 플로우 f와 f'의 거리는 *city-block distance*를 이용하여 계산한다. d를 N차원에 ce 두 벡터간의 거리라고 한다면, 식 2와 같이 표현된다.

$$d(\tau(f), \tau(f')) = \sum_{i=1}^{N} |s_i(f) - s_i(f')| \qquad \text{(식 2)}$$

보통 N차원의 공간에서 두 벡터간의 거리는 *Euclidean distance*를 사용한다. 그럼에도 불구하고 *Euclidean distance* 대신 *city-block distance*를 사용하는 이유는 계산상의 속도 문제 때문이다. 두 개의 거리를 구하는 공식은 식에서 알 수 있듯이 서로 비례한다. 하지만, *city-block distance*는 *Euclidean distance*에 비해 계산식이 간단하므로 많은 양의 데이터를 처리해야 하는 트래픽 분석에서는 처리 속도에서 상당한 이점을 가질 수 있다. 특히, 실시간 트래픽 분류에서는 데이터의 처리 속도가 더욱 빨라야 하므로 *city-block distance*를 사용하였다.

5.1.3 그룹핑 알고리즘

그룹핑 되는 각 그룹은 표 1과 같이 총 5가지의 속성을 가진다. Process code는 특정 응용을 나타내는 것으로 프로그램을 용이하게 하기 위해 임의로 정한 정수이다. Transport protocol은 응용이 사용하는 전송계층의 프로토콜로써 TCP, UDP 등이 존재한다. Representative value(대표값)는 그룹을 대표하는 값으로 플로우의 벡터 표현과 같이 벡터로 표현된다. Dimension of representative value(대표값의 차원)은 대표값의 차원을 의미하는 것이며, 차원은 곧 페이로드 패킷의 개수를 의미하므로 1에서 N까지의 값을 가질 수 있다. 본 연구에서는 N의 값을 최대 5로 정하였다. Distance threshold(거리 임계값)는 대표값으로부터 가장 먼 *city-block distance*를 의미한다. 특정 그룹의 대표값과 플로우가 거리 임계값 내에 속하게 되면 같은 그룹에 해당된다. 초기 거리 임계값은 50이다.

여러 프로세스의 플로우들을 입력으로 받아 그룹핑 작업을 행한다. 그림 4는 그룹핑 알고리즘의 의사코드(pseudocode)이다.

모든 플로우는 순차적으로 그룹핑 작업의 입력으로 들어오는데, 하나씩 플로우가 들어올 때마다 가장 먼저 행하는 작업은 입력으로 들어온 플로우를 그룹들과 거리를 계산하기 위해 벡터로 표현하는 작업이다. 그런 다음, 플로우와 거리가 가장 가까운 그룹(*minDistGroup*)을 찾는 작업을 행한다. 이 작업은 플로우가 그룹의 모든 속성과 일치한 그룹들 중 *city-block distance*가 가장 가까운 그룹을 찾는 작업이다. *minDistGroup*을 찾으면 *minDistGroup*의 대표값을 재설정한다. 대표값은 해당 그룹의 모든 플로우의 벡터를 산술평균을 통해 계산한 벡터가

표 1. 그룹의 속성들
Table 1. Attributes of group

Attributes of group	Process code
	Transport protocol
	Representative value
	Dimension of representative value
	Distance threshold

```
1: procedure Flow Grouping(flow)
2:   transforms a flow into vector representation
3:
4:   search minDistGroup in all groups
5:
6:   if founded minDistGroup then
7:     insert the flow into minDistGroup
8:     recalculate representative value of minDistGroup
9:   else
10:    make new group
11: end procedure
```

그림 4. 그룹핑 알고리즘의 의사코드
Figure 4. Pseudocode of grouping algorithm

된다. 그리고 *minDistGroup*이 존재하지 않으면 새로운 그룹을 생성한다.

5.1.4 그룹의 최적화와 포트 할당

그룹의 최적화 작업은 그룹에 속하지 않는 플로우의 제거와 모든 그룹의 초기 거리 임계값을 최소화하는 작업이다. 모든 그룹은 그룹핑에 사용된 초기 거리 임계값으로 거리 임계값들이 모두 같이 설정되어 있다. 그림 5는 이해를 돕기 위해 2차원 그룹의 최적화를 보여준다. 그림 5의 왼쪽 그림에서 보는 바와 같이 그룹핑 과정에서 대표값의 변화에 의해 최종 그룹에는 속하지 않는 플로우들이 발생한다. 이 플로우들은 최종적으로 그룹에 속하지 않으므로 그룹에서 제거하는 작업이 필요하다. 그림 5는 그룹핑이 끝난 그룹이 대표값(마름모의 중심)으로부터 가장 멀리 떨어진 플로우들을 기준으로 거리 임계값(마름모의 크기)을 줄이는 것을 보여준다.

그룹의 최적화 작업이 종료되면 좀 더 정확한 트래픽 분류를 위해 그룹핑 된 각 그룹에 포트를 할당한다. 포트는 여러 응용 프로그램을 구별하는 기준이 된다. HTTP, FTP, SSH 등 전통적인 응용 뿐만 아니라, 고정된 포트를 사용하는 응용이 다수 존재한다. 따라서, 포트가 특징이 되는 그룹에는 포트를 할당하는 휴리스틱한 기법을 사용한다. 포트가 할당되는 조건은 그룹 안의 모든 플로우가 동일한 서버 포트를 가졌을 때이다. TCP의 경우 3-handshake 과정을 통해 서버의 구별이 명확하지만, UDP의 경우 그렇지 않다. 그래서 UDP의 경우에는 플로우의 첫 번째 패킷을 기준으로 첫 번째 패킷을 발생시킨 호스트를 클라이언트로, 첫 번째 패킷을 받는 호스트를 서버로 정한다. 이는 대부분의 UDP 플로우도 TCP와 같이 클라이언트가 먼저 서버로 패킷을 보내는 특징을 이용한 것이다.

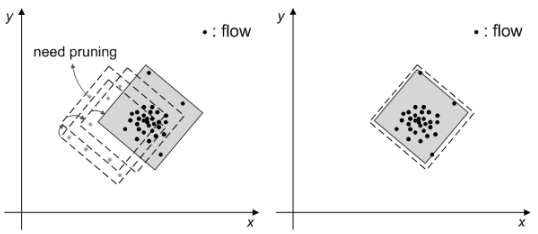


그림 5. 그룹의 최적화
Figure 5. Optimization of groups

5.1.5 충돌 처리

그룹핑 된 모든 그룹은 각각 프로세스의 통계 시그니쳐가 된다. 즉, 트래픽 분류의 기준이 되는 것으로 그룹끼리의 충돌은 실제 분류 결과에서 FP (False Positive)를 발생시킨다. 충돌이란 어떤 플로우가 하나의 그룹에만 속하지 않고 여러 개의 그룹에 속하는 것을 의미한다. 그림 6은 이러한 충돌을 어떻게 해결하는지를 보여주는 그림이다.

충돌이 많을수록 분류 결과에서 FP가 증가되고 정확도(accuracy)가 낮아진다. 이를 해결하기 위해 충돌을 발생시키는 플로우를 그룹에서 제외하는 방법이 있다. 제외시키는 가장 기본적이고 쉬운 방법은 그룹의 거리 임계값을 낮춰 충돌을 발생시키는 플로우를 제거하는 것이다. 이 방법은 FP를 FN (False Negative)로 변경한다. 즉, 플로우가 어떤 응용에 해당하는지 명확하게 판단하지 못한다면 분류하지 않겠다라는 것이다. 많은 경우에서 트래픽이 어떤 트래픽인지 알지 못하는 것보다 트래픽을 잘못 분류한 경우가 더 큰 문제를 일으킨다. 또한, 현

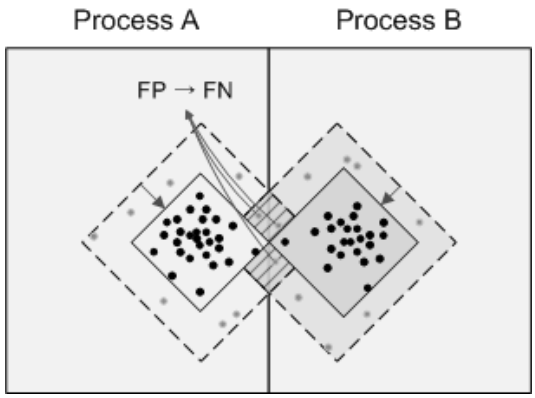


그림 6. 그룹 사이의 충돌 처리
Figure 6. Conflict process between groups

재 트래픽 분류 분야에서 많은 연구가 이루어지는 멀티레벨 분류 방법론[17]처럼 다양한 분류 방법론을 적용하여 트래픽을 분류하는 경우, FN이 발생하더라도 FP가 없는 경우가 다른 방법론과 쉽게 융화될 수 있다.

하지만, 이렇게 거리 임계값을 낮추는 방법의 문제는 충돌을 발생시키지 않는 플로우 또한 거리 임계값이 낮아짐에 따라 그룹에서 제거되는 현상이 나타난다는 것이다. 이러한 현상을 해결하기 위해서는 그룹의 모양이 원, 마름모와 같이 일정한 모양 대신 다른 복잡한 도형이 그려져야 한다. 본 연구에서는 이러한 문제가 발생하더라도 *city-block distance*에 의해 결정되는 마름모의 형태를 유지한다. 이렇게 하는 가장 큰 이유는 트래픽을 분류할 때 빠른 속도로 분류하기 위해서이다. 복잡한 모양일수록 해당 그룹에 속하는지를 판단하기 위해 복잡한 계산이 필요하기 때문이다. 그래도 FN을 최소화시키기 위해 그룹의 거리 임계값을 줄일 때 두 그룹 중 플로우의 손실이 적은 그룹의 거리 임계값을 줄인다.

최종적으로 그룹이 완성이 되면, 하나의 그룹은 하나의 통계 시그니쳐로서 트래픽 분류 단계에서 트래픽을 분류하는데 사용된다.

5.2 실시간 트래픽 분류 단계

실시간 트래픽 분류 단계에서는 실제 학내 망의 트래픽을 수집하고 해당 트래픽을 실시간으로 분류한다. 패킷들이 들어오면 먼저 플로우를 생성한다. 플로우는 일반적으로 사용되는 5-tuple(source IP, destination IP, source port, destination port, rcetocol)을 기준으로 양방향의 플로우이다. 통계 시그니쳐와의 비교를 위해 각 플로우는 해당되는 페이로드 패킷의 수에 따라 1부터 5차원까지의 벡터로 표현된다. 최대 5개의 패킷에 대한 페이로드 크기와 방향으로만 분류를 행하므로 실시간 트래픽 분류에 적합하다.

VI. 평 가

본 장에서는 통계 시그니쳐 기반의 트래픽 분류의 타당성을 검증하기 위해 검증 네트워크인 학내 망의 트래픽을 분류한 결과에 대해 설명한다.

6.1 평가 요소

검증을 위한 ground truth 데이터는 3.2절에서 설명한 바와 같이 TMA라는 agent를 통해 생성된다.

Coverage, completeness, accuracy와 같이 총 3가지의 평가 요소가 있다. Coverage는 트래픽 분류기가 얼마나 많은 종류의 응용을 분류할 수 있는가에 대한 척도로써, 응용 프로그램의 개수와 시그니쳐의 개수로 나타낸다. Completeness는 시그니쳐를 통해 전체 트래픽 중 얼마나 많은 양의 트래픽을 분석했는지에 대한 척도로써, 트래픽의 양을 나타낸다. 마지막으로 accuracy는 분류 결과와 ground truth를 비교하여 결과를 얻는 분류 정확도를 나타낸다. Accuracy는 precision과 recall과 같은 값들로 표현된다. Completeness와 accuracy는 플로우, 패킷, 바이트 단위로 분류 결과가 제시되어야 좀 더 상세한 정보를 얻을 수 있다.

6.2 검증 결과

통계 시그니쳐 기반의 트래픽 분류 시스템의 검증을 위해 하루 동안 학내 망의 모든 트래픽을 대상으로 실험을 하였다. 분류 시스템의 coverage는 표 2에 나타나 있다.

표 3은 completeness와 overall accuracy를 플로우, 패킷, 바이트 단위로 보여준 표이다. 높은 accuracy에 비해 낮은 completeness를 확인할 수 있다. 이는 두 가지 원인에 의해 분석된다. 우선, coverage가 낮아서 나타나는 문제이다. 트래픽을 많이 발생시키는 응용을 등록하지 못해서 completeness가 낮아지게 된다. 두 번째는 패킷의 페이로드 크기와 방향으로 트래픽을 분류 가능한 트래픽에 대해서는 매우 정확하게 분류하지만, 분류하기가 힘든 트래픽은 시그니쳐를 생성할 때 충돌 처리로 인해 일부러 분류를 하지 않기 때문에 나타나는 현상이다.

표 2. Coverage
Table 2. Coverage

	Coverage
Process	75
Signature	2,789

표 3. Completeness와 overall accuracy
Table 3. Completeness and overall accuracy

	Completeness	Overall Accuracy
Flow	18.99%	97.20%
Packet	30.31%	97.89%
Byte	31.52%	98.11%

표 4. 응용 별 precision과 recall
Table 4. Precision and recall by application

Application	Precision	Recall
softforum	100.00%	98.39%
outlook	100.00%	72.84%
nateon	100.00%	3.51%
fileguri	99.98%	86.85%
windows	99.91%	34.89%
torrent	98.49%	42.57%
skype	78.40%	76.05%

표 4는 응용 별 precision과 recall 값을 표현하고 있다. Skype를 제외한 나머지 프로세스의 precision 결과는 높게 나타났다. Skype의 precision이 낮게 나온 원인은 windows의 svchost라는 프로세스와의 충돌(FP) 때문이다. 즉, 시그니쳐를 생성할 시 어떤 프로세스에 의해서 발생할 수 있는 모든 플로우를 입력으로 할 수 없기 때문에 시그니쳐 생성 단계에서 충돌 처리를 하여도 다음과 같은 결과가 나온다. 하지만, 전체적으로 보았을 때 높은 precision의 결과가 도출된 것으로 보아 본 분류 방법론의 정확도는 매우 높은 것으로 판단할 수 있다.

Recall 값은 프로세스마다 많은 차이를 보이는데 대체적으로 낮은 값을 보인다. 이는 페이로드의 크기 및 방향만으로는 프로세스의 모든 플로우를 분류할 수 없다는 것을 뜻한다. 따라서, 이를 분류할 수 있는 다른 시그니쳐를 필요로 한다. 추후 연구 과제에서 이를 진행할 계획이다.

기존에 페이로드 시그니쳐 기반의 분류 방법은 현재 개발된 트래픽 분류 방법 중 높은 정확도로 인해 널리 사용되고 있는 방법이다. 본 방법론의 결과는 페이로드 시그니쳐 기반의 분류 방법론과 비교하였을 때 비슷한 정확도의 결과를 도출하고 있으며, 분류기의 성능적인 측면을 보았을 때 페이로드를 비교하지 않고 크기와 방향만을 이용하므로 트래픽을 빠르게 분류할 수 있다. 또한, 기존의 페이로드 크기와 방향을 이용한 방법론[9]-[10]에서 사용된 클러스터링 기법들보다 간단한 그룹핑 방법의 사용을 통해 빠른 시그니쳐의 생성과 트래픽 분류를 통해 높은 정확도의 분류 결과를 도출한다.

Ⅶ. 결론 및 향후 과제

트래픽의 통계 정보 중 페이로드 크기와 방향, 그리고 포트 정보를 이용하여 다양한 응용 프로그램을 높은 정확도로 분류할 수 있다는 것을 실험을 통해서 확인할 수 있었다. 기존의 연구들보다 상세한 분류 기준으로 여러 가지의 응용들을 직접 분류하여 본 방법론의 유용성을 증명하였다. 또한, 본 분류 방법론은 페이로드 내의 정보를 이용하지 않고, 플로우의 초기 몇 개의 패킷 만을 이용하여 트래픽 분류를 행하므로 실시간 트래픽 분류에 아주 적합하다.

하지만, FN의 처리 문제 등 앞으로 해결해야 할 과제들이 남아있다. 통계 시그니쳐 기반의 분류 방법론을 통해 앞으로 좀 더 정확하고 많은 종류의 트래픽을 분류할 수 있는 시스템을 개발할 계획이다.

참 고 문 헌

[1] 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, pp.22-42, Feb., 2005.

[2] Jeffrey Erman, Martin Arlitt, Anirban Mahanti, "Traffic Classification Using Clustering Algorithms", *Proc. of SIGCOMM Workshop on Mining network data*, Pisa, Italy, pp.281-286, Sep., 2006.

[3] Andrew W. Moore and Denis Zuev, "Internet Traffic Classification Using Bayesian Analysis Techniques," *Proc. of the ACM SIGMETRICS*, Banff, Canada, Jun., 2005.

[4] Thomas Karagiannis, Konstantina Papagiannaki, and Michalis Faloutsos. "BLINC: Multilevel Traffic Classification in the Dark," *Proc. of SIGCOMM 2005*, Philadelphia, PA, Aug., 22-26, 2005.

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

[6] Jian Zhang and Andrew Moore, "Traffic Trace Artifacts due to Monitoring Via Port Mirroring," *Proc. of the IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services (E2EMON) 2007*, Munich, Germany, May., 21, 2007.

[7] Liu, Hui Feng, Wenfeng Huang, Yongfeng Li, Xing "Accurate Traffic Classification", *Networking, Architecture, and Storage, 2007.* NAS 2007. International Conference

[8] Risso, F. Baldi, M. Morandi, O. Baldini, A. Monclus, P. Lightweight, Payload-Based Traffic Classification: An Experimental Evaluation. In *proceeding of*

Communications, 2008. ICC '08. IEEE International Conference, 2008.

[9] L.Bernaille, R. Teixeira, I. Akodkenou, A.Soule, and K.Salamatian. "Traffic classification on the fly". *SIGCOMM Comput.Commun.* Rev., 2006.

[10] Bernaille, L., Teixeira, R., Salamatian, K.: Early application identification. In: *CoNext 2006.* Conference on Future Networking Technologies., 2006.

[11] Byung-Chul Park, Young J. Won, Myung-Sup Kim, James W. Hong, "Towards Automated Application Signature Generation for Traffic Identification," *Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS) 2008*, Salvador, Bahia, Brazil, pp.160-167, April, 7-11, 2008.

[12] Rentao Gu, Minhuo Hong, Hongxiang Wang, and Yuefeng Ji, "Fast Traffic Classification in High Speed Networks" *Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2008*, LNCS 5297, Beijing, China, pp.429-432, Oct., 22-24, 2008.

[13] Ying-Dar Lina, Chun-Nan Lua, Yuan-Cheng Laib, Wei-Hao Penga and Po-Ching Lina, "Application classification using packet size distribution and port association" *Proc. of the Journal of Network and Computer Applications*, In Press, Corrected Proof, Available online, March, 20. 2009.

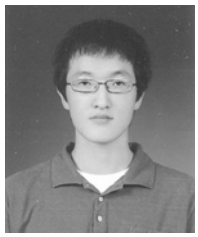
[14] Huifang Feng, Yantai Shu, "Statistical Analysis of Packet Interarrival Times in Wireless" *Proc. of the Wireless Communications, Networking and Mobile Computing, 2007.* WiCom 2007. International Conference, Shanghai, China, pp.1888-1891, Sept., 21-25, 2007.

[15] Jacobus van der Merwe, Ramon Caceres, Yang-hua Chu, and Cormac Sreenan "mmdump - A Tool for Monitoring Internet Multimedia Traffic," *ACM Computer Communication Review*, 30(4), October, 2000.

[16] Hun-Jeong Kang, Myung-Sup Kim, and James Won-Ki Hong, "Streaming Media and Multimedia Conferencing Traffic Analysis Using Payload Examination," *ETRI Journal*, Vol.26, No.3, pp.203-217, Jun., 2004.

[17] Y.J. Won, B.C. Park, H.T. Ju, M.S. Kim, and J. W. Hong. A hybrid approach for accurate application traffic identification. In *IEEE/IFIP E2EMON*, April, 2006.

박 진 완 (Jin-wan Park) 준회원

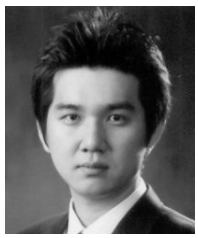


2009년 고려대학교 컴퓨터정보학과 학사

2009년~현재 고려대학교 컴퓨터정보학과 석사과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

윤 성 호 (Sung-ho Yoon) 준회원



2009년 고려대학교 컴퓨터정보학과 학사

2009년~현재 고려대학교 컴퓨터정보학과 석사과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

박 준 상 (Jun-sang Park) 준회원

2008년 고려대학교 컴퓨터정보학과 학사

2008년~현재 고려대학교 컴퓨터정보학과 석사과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

이 상 우 (Sang-woo Lee) 학생회원

2003년~현재 고려대학교 컴퓨터정보학과 학사과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

김 명 섭 (Myung-sup Kim) 종신회원

1998년 포항공과대학교 전자계산학과 학사
1998년~2000년 포항공과대학교 컴퓨터공학과 석사
2000년~2004년 포항공과대학교 컴퓨터공학과 박사
2004년~2006년 Post-Doc., Dept. of ECE, Univ. of Toronto, Canada
2006년~현재 고려대학교 컴퓨터정보학과 조교수
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석, 멀티미디어 네트워크