HTTP 트래픽의 클라이언트측 어플리케이션별 분류

종신회원 최 미 정*°, 학생회원 진 창 규*, 종신회원 김 명 섭**

Classification of Client-side Application-level HTTP Traffic

Mi-jung Choi*° *Lifelong Member,* Chang-gyu Jin* *Student Member,*
Myung-sup Kim** *Lifelong Member*

요 약

오늘날 많은 어플리케이션들이 방화벽에서 차단을 막기 위해 HTTP 프로토콜의 기본 포트인 80번 포트를 사용하고 있다. HTTP 프로토콜이 예전처럼 웹 브라우징에만 사용되는 것이 아니라 P2P 어플리케이션의 검색, 소프트웨어 업데이트, 네이트온 메신저의 광고 전송 등 다양한 어플리케이션에 사용되며 다양한 형태의 서비스를 제공하고 있다. HTTP 트래픽의 증가와 다양한 어플리케이션들이 HTTP 프로토콜을 사용함으로써 어떤 서비스들이 어떻게 HTTP를 이용하는지에 대한 파악이 중요해지고 있으며, 방화벽과 같은 장비에서 특정 어플리케이션의 트래픽을 차단하기 위해서는 HTTP 프로토콜 레벨이 아닌 어플리케이션 레벨의 분석이 필요하게 되었다. 따라서 본 논문에서는 HTTP 트래픽에 대해 HTTP 프로토콜을 사용하는 클라이언트측의 어플리케이션별로 분류하고 이를 서비스별로 그룹지어 HTTP 트래픽을 클라이언트측면에서 분류하는 방법을 제안하고자 한다. 제안한 방법론을 학내 네트워크에서 발생하는 트래픽에 적용함으로써 알고리즘의 타당성을 검증하였다.

Key Words : HTTP, Application-level Traffic Classification, Client-side, Service Group

ABSTRACT

Today, many applications use 80 port, which is a basic port number of HTTP protocol, to avoid a blocking of firewall. HTTP protocol is used in not only Web browsing but also many applications such as the search of P2P programs, update of softwares and advertisement transfer of nateon messenger. As HTTP traffics are increasing and various applications transfer data through HTTP protocol, it is essential to identify which applications use HTTP and how they use the HTTP protocol. In order to prevent a specific application in the firewall, not the protocol-level, but the application-level traffic classification is necessary. This paper presents a method to classify HTTP traffics based on applications of the client-side and group the applications based on providing services. We developed an application-level HTTP traffic classification system and verified the method by applying the system to a small part of the campus network.

Ⅰ. 서 론

HTTP(Hyper-Text Transfer Protocol)[6]는 1989년 팀버너스리에 의해 처음 설계되어 인터넷을 통한 월드와이드웹(World-Wide Web) 기반에서 전 세계적인 정보공유를 이루는데 큰 역할을 수행하였다. 인터넷을

※ 이 논문은 2009년 정부(교육과학기술부)의 재원으로 한국연구재단(2009-0090455)의 지원 및 2011년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(20110020518).
* 강원대학교 컴퓨터과학과 (mjchoi@kangwon.ac.kr, jinchanggyu@kangwon.ac.kr), (°:교신저자)
** 고려대학교 컴퓨터정보학과 (tmskim@korea.ac.kr)
논문번호 : KICS2011-08-346, 접수일자 : 2011년 8월 12일, 최종논문접수일자 : 2011년 10월 19일

통하여 가공되지 않은 데이터를 전송하기 위한 단순한 프로토콜로 시작하여, 데이터에 대한 전송과 요구·응답에 대한 수정 등 가공된 정보를 포함하는 프로토콜로 개선되었다. HTTP 프로토콜은 현재 단순한 웹 브라우징 이외에 파일 다운로드, 메신저 프로그램, P2P(peer to peer) 프로그램, 멀티미디어 동영상 재생 프로그램 등에서도 통신 수단으로 사용되고 있다. 예를 들면, 네이트온과 같은 메신저는 HTTP를 통하여 광고를 전송하고, P2P 프로그램 중 하나인 uTorrent는 파일 검색을 위해 HTTP를 사용하기도 한다. 또한 방화벽을 통과하기 위하여 기본 포트가 9493을 사용하는 파일구리와 같은 경우에는 TCP/80 포트를 이용하여 통신하는 등 많은 어플리케이션들이 통신을 하기 위하여 HTTP를 사용하고 있는 실정이다. 이는 HTTP 프로토콜의 클라이언트와 서버간의 요청/응답 파라다임이 많은 어플리케이션에 적합하며 HTTP 프로토콜의 구현이 쉽기 때문이다. 또한, 대부분의 엔터프라이즈 네트워크에서는 방화벽의 필터링을 통하여 프로토콜 레벨과 포트 레벨의 차단을 제공하지만 웹 연결을 위한 프로토콜인 HTTP와 80번 포트는 열어놓는다. 어플리케이션 프로그램들이 방화벽을 구애없이 통과하기 위해 HTTP 프로토콜을 이용하여 통신한다. 따라서, HTTP 기반의 트래픽을 제어하기 위하여 통신이 이루어지는 TCP/80번 포트로 분류하여 단순히 막는 것만으로는 정확한 어플리케이션을 제어할 수 없다. 더군다나 80번 포트를 이용한 악성코드 유포와 침입 공격도 이루어지고 있다. 불필요한 트래픽에 대한 차단을 위해서는 HTTP 트래픽에 대해 프로토콜 레벨이 아닌 HTTP 트래픽을 사용하는 프로그램이 무엇인지 분석하는 어플리케이션 레벨의 분석이 필요하다. HTTP 프로토콜을 사용하는 트래픽을 세부적으로 분석하여 어떤 어플리케이션 프로그램이 HTTP 트래픽을 얼마만큼 유발하는지 분석함으로써 추후 방화벽에서 트래픽을 차단하거나 QoS 보장을 위한 트래픽 엔지니어링에 사용할 수 있다.

HTTP 트래픽 분석에 대한 연구들[1-4]이 진행되어 네트워크 관리에 적용하고 있지만 클라이언트 측의 어플리케이션에 따른 완벽한 분석이 이루어지지 않고 있다. 또한 현재 수백개 이상의 어플리케이션들이 다양한 목적으로 HTTP 프로토콜을 사용하고 있으며, 이는 단순한 어플리케이션의 나열만으로는 HTTP 프로토콜을 사용하는 추세에 대한 분석이 어려우므로 HTTP 트래픽을 서비스별로 그룹짓고 각 서비스별로 어떤 어플리케이션들이 존재하는지 분석하는 것 또한 HTTP 트래픽의 정확한 파악을 위해 필요하다. 따라서 본 논문에서는 HTTP 트래픽의 클라이언트측의 서비스 그룹을 찾고 서비스 그룹별로 어플리케이션을 분류하는 체계에 대한 기준을 정립하고 그 방법을 제시하고자 한다. 즉, 클라이언트 측에서 트래픽을 서비스별로 그룹지어 분석하고 세부적인 어플리케이션 프로그램 수준까지 분석하고자 한다. 예를 들어, 사용자의 네이트온 프로그램에서 HTTP 프로토콜을 사용하여 통신을 했을 경우 서비스별 분류는 메신저 서비스가 되고, 어플리케이션 분류에서 이를 네이트온 메신저로 분류하겠다는 것이다.

본 논문은 다음과 같이 구성된다. 2장에서는 관련 연구로 HTTP 트래픽 분석을 수행한 기존 연구를 정리하고, 3장에서는 HTTP 트래픽의 어플리케이션들을 서비스 그룹으로 분류하는 기준을 제시한다. 4장에서는 HTTP 트래픽 분석을 위한 모니터링 시스템과 분석 알고리즘에 대해 기술하며, 5장에서는 분석 결과를 제시한다. 6장에서는 결론을 맺고 향후 과제에 대해 기술한다.

Ⅱ. 관련 연구

이 장에서는 HTTP 트래픽의 분석에 대한 기존 연구에 대해서 살펴본다. HTTP 트래픽 분석에 대한 기존 연구는 HTTP 프로토콜이 어떤 목적으로 사용되는지에 대한 분석과 어떤 내용을 전송하는데 사용되었는지에 초점을 맞추고 있다.

Wei Lie[1]의 논문은 HTTP 패킷의 헤더 필드를 조사하여 HTTP 트래픽을 단순한 웹 브라우징 이외에 웹 메일, 웹 어플리케이션, 파일 다운로드, 멀티미디어, 메신저 등의 서비스 그룹으로 분류 하였다. 그룹별로 분류하기 위해 HTTP 헤더의 URL, Content-Type, User-Agent, Host 등의 필드를 어떻게 조합하여 검사할지를 체험적(heuristic)으로 발견하여 검사하여 분류하였다. 같은 네트워크상에서 2006년 data와 2008년 data를 수집하여 비교 분석함으로써 HTTP 트래픽의 증가와 분포의 변화에 대한 설명도 추가하였다. 본 논문[1]은 HTTP 트래픽을 어플리케이션별로 묶을 수 있는 기본 분류 기준을 정립하는데 아주 유용하다. 본 논문을 참조하여 HTTP 패킷을 서비스별로 구분하기 위하여 HTTP 헤더에서 어떤 부분을 점검해야 하는지 참고할 수 있다. 현재는 더 다양한 서비스들과 어플리케이션들이 HTTP 프로토콜을 사용하여 본 논문의 분류 기준만으로 모든 어플리케이션들을 서비스 그룹별로 묶는 것이 불가능하다. 또한 본 논문에서는 어플리케이션의 정확한 명칭이 아닌 그룹명만 알 수

있어서 각 어플리케이션 프로그램별 제어를 수행하기 위한 분류 방법에 적용하기에는 부족하다.

Fabian Schneider[2]의 논문은 현재 구글 맵이나 구글 메일이 사용하고 있는 웹 서버인 AJAX 서버에서 생성하는 웹 어플리케이션의 특징이 기존의 웹 서버와 구동 방식과 달라 갖는 특징을 정리한 것이다. 기존의 HTTP 트래픽의 페이로드는 90%가 45KB 이하였다. 그러나 AJAX 어플리케이션들은 기존의 HTTP 트래픽의 페이로드 보다 연결당/세션당 더 많은 바이트를 전송한다. 또한 세션별 요구(request) 수에 있어서도 AJAX 어플리케이션은 더 많은 요구를 생성하고, 요구 요청 시간 간격(inter-request-time)에 있어서도 그 간격이 기존의 HTTP 트래픽보다 짧다. 즉 AJAX 어플리케이션들은 더 많은 세션이 더 bursty한 트래픽을 생성하는 것이다.

Schneider[2]의 논문은 특정 웹 서버에 대한 특징을 분석하면서 어떤 웹 트래픽의 특징을 조사해야 하며 어떤 통계적인 데이터를 분석하는 것이 의미가 있는지를 정리하는데 좋은 논문이다. 이 논문은 시그네쳐 분석 방법으로 분류가 되지 않은 HTTP 트래픽에 대한 분석을 통계적 분석 방법을 적용함으로써 분석률(completeness)이 증가하며 정확도(accuracy)가 향상될 수 있음을 알려준다. 그러나 본 논문은 AJAX 웹 서버에서 생성된 HTTP 트래픽에 대한 분석률만을 향상시키는 단점이 있다.

K. Kim[3]의 논문은 HTTP 트래픽을 HTTP가 전송하는 내용(contents)에 기반을 두어 분석을 수행했다. 이 논문은 HTTP 트래픽의 헤더가 아니라 페이로드의 내용을 분석하여, 쇼핑(shopping), 성인물(adult), 주식(stock), 커뮤니티(community), 게임, 음악, 영화, 웹메일, 교육, 뉴스&웹 등 총 10가지로 나누어 분석하고 있다. 논문 제목에서 나타나듯이 이 연구는 일본어로 된 홈페이지에 대해서만 수행한 것이다. 먼저 각 항목별로 분류 가능한 키워드 사전을 만드는 작업을 수행한 후, 각 분류별로 10~15개의 키워드를 추출하여 사전을 만들고, HTTP 트래픽만 따로 수집하여 페이로드 중에서 일본어로 특정 키워드만을 추출하여 사전의 키워드들과 비교 분석하여 HTTP 트래픽을 항목별로 나누었다. 키워드 매칭에 대해서는 HTML 페이지에서 첫 번째 나온 단어로 매칭한 경우와 가장 많이 사용된 키워드로 매칭된 경우의 분류 결과가 약간의 차이를 보였다.

특히 압축되지 않은 HTML 파일뿐 아니라 압축된 HTML, XML 파일에 대해서는 HTTP 헤더의 content-encoding 필드를 검색하여 압축 알고리즘에 따라 압축을 해제하여 압축된 내용에 대해서도 분류하였다. 일본의 큐수 대학의 700M 트래픽을 키워드가 가장 많이 나온 것을 기준으로 분류한 결과 뉴스와 웹이 33% 정도로 가장 많이 차지했고, 영화가 약 23%, 커뮤니티가 21%, 웹 메일이 13% 정도로 이 네 개의 내용별 분류가 전체의 90% 정도를 차지하는 결과를 보여주었다. 일본어라는 특정 언어지만 HTTP 트래픽의 페이로드를 분석하여 키워드를 기반으로 전송 내용별 분류 시도는 어느 정도의 정확도를 보였다.

K. Kim의 논문[3]은 Wei Lie의 논문[1]과 또 다르게 HTTP 트래픽을 내용을 기반으로 분류하는 방법을 제시하였으며, gzip이나 deflate로 압축되어 시그니쳐를 추출하기 어려운 페이로드에 대한 분석 방법으로 활용이 가능하다. 단지 시스템이 분류를 처리하는 오버헤드에 대한 언급이나 키워드 분석의 정확도에 대한 고려가 부족하여, 실시간 분류 시스템으로 적용가능성에 대해서는 판단이 어렵다.

현재 한국의 http://www.rankey.com 사이트[4]는 웹 사이트의 산업군에 대해 대분류, 중분류, 소분류의 분류 체계를 기반으로 정의하고 이에 따른 산업군의 카테고리별 웹 사이트의 순위 정보와 트렌트를 확인할 수 있다. 또한 사이트의 카테고리별 분류 없이 전체 사이트에 대한 순위와 트래픽 데이터도 확인할 수 있다. 순위를 추출하기 위해 인구통계학적 근거에 의해 선정된 패널들의 개인 PC에 에이전트 프로그램을 설치하게 하고 사이트 접속 정보를 수집한 것이다. 개인의 웹서핑 내역을 토대로 통계적 방법으로 순위를 측정한 것으로 결국은 샘플링 방법을 도입한 것이다. 또한 사이트 리스트도 직접 사이트 쪽에서 사전에 등록된 것으로 한정된 면이 있지만, 사이트의 순위 정보에 대한 정보는 서비스 제공자나 광고주 입장에서는 중요한 정보가 될 것이다.

Ⅲ. HTTP의 서비스별 분류 기준

이 장에서는 HTTP 프로토콜을 사용하는 서비스를 정의하고 클라이언트 측의 어플리케이션들을 각 서비스별로 그룹짓는 분류 기준을 설정한다. HTTP 프로토콜을 사용하는 어플리케이션 리스트를 나열한 연구는 찾아볼 수가 없다. 본 논문에서는 수집한 HTTP 트래픽을 분석한 결과 약 500여개의 어플리케이션이 HTTP 프로토콜을 사용하고 있는 것을 발견할 수 있었다. 본 장에서는 이 어플리케이션을 서비스별로 그룹짓는 분류 기준을 정리하고자 한다.

서비스 그룹은 크게 10가지로 나눠볼 수 있다. 표

1은 HTTP 트래픽의 서비스별 어플리케이션 구분 및 제공기능에 대한 설명이다.

서비스 그룹은 브라우징, 웹 어플리케이션, P2P, 메신저, 소프트웨어 업데이트, 보안, 모바일, 크롤러(crawler), 멀티미디어, 웹기반의 서비스로 나뉜다. 브라우징 서비스 그룹에는 다양한 브라우저 어플리케이션과 브라우저의 버전별 어플리케이션별 구분도 가능하다. 브라우저에서 프로그램을 다운받아 개인 PC에 설치하고 수행하는 툴바와 같은 다양한 웹 어플리케이션이 존재하며, P2P 프로그램은 찾고자 하는 파일을 검색하는데 HTTP 프로토콜을 사용한다. 메신저는 메신저 프로그램 내에 추가적인 광고나 정보를 보여주는데 HTTP 프로토콜을 사용하고, 다양한 소프트웨어가 업데이트에 HTTP 프로토콜을 사용한다. 인터넷 뱅킹과 인터넷 결재 시스템의 경우 보안을 위한 다양한 소프트웨어 설치에 HTTP 프로토콜을 사용하고 현재 증가하고 있는 스마트 장비들의 다양한 어플리케이션들이 HTTP를 사용하여 전송된다. 또한 구글이나 네이버와 같은 검색 사이트들이 웹 서버의 내용을 미리 저장하여 DB를 구축하기 위한 크롤러들도 HTTP를 사용하는 대표적인 서비스로 볼 수 있고, GOMTV나 멜론과 같은 멀티미디어 서비스를 제공하는 어플리케이션들도 자막이나 가사를 전달하는데 HTTP 프로토콜을 사용한다. 파일 전송에 있어서도 대용량의 파일은 ftp로 전송하나 소용량의 파일은 HTTP를 사용하여 전송하기도 한다.

표 1. 서비스그룹별 대표적인 어플리케이션들

서비스 그룹	대표적인 어플리케이션 목록	기능
Browser	MSIE(iExplorer), Firefox, Safari, Chrome, Opera	HTML, XML 데이터 전송
Web application	WordPress, NaverToolbar MSN Toolbar NaverMiniCalendar	웹(HTTP) 기반의 어플리케이션
P2P	uTorrent, BigFileSearch uTorrentApplication	파일검색
Messenger	Nateon, Tachy	광고,알림전송
S/W update	Windows update, Microsoft update, Noton update, Alyac update	소프트웨어 업데이트
Security	NoPhishingUI, win_security, KALogoutComponent	보안 모듈 설치를 위한 다운로드
Mobile	KaKaoTalkAndroid, iPhoneMelon, Android-YouTube, SAMSUNG_KIES	스마트폰태블릿 PC의 앱
Crawler	Naver_bot, bingbot, Daum_bot, Googlebot, baidu_robot, Yahoo!Slurp	웹페이지 검색 및 저장
Multimedia	GomTV, DaumPot, MelonPlayer, WMPlayer, AfeecaPlayer,Mnet_Player	멀티미디어 가사, 자막 등 데이터 전송
Web_based service	WSCLOUD, MeDCore, MicrosoftBIT, NeoMapa	웹을 기반한 서비스

현재 HTTP를 사용하는 가장 대표적인 어플리케이션은 당연히 웹브라우저이다. 웹브라우저도 인터넷 익스플로러 이외에 크롬, 파이어폭스, 사파리, 오페라 등 그 종류가 다양하다. 본 논문에서는 어플리케이션을 일일이 조사하여 서비스 그룹별로 구분 짓는 작업을 수행했다. 현재는 일일이 HTTP 트래픽을 수집하여 수작업으로 어플리케이션을 찾아내고 그 어플리케이션을 서비스 그룹으로 분류한다. 추후 이 어플리케이션을 자동적으로 추출하고 서비스 그룹으로 구분짓는 자동화에 대해서도 연구가 필요하다. 그러나 다양한 어플리케이션이 존재하고 이것을 자동으로 서비스별로 그룹짓는 것은 쉽지 않은 작업이 될 것이다.

Ⅳ. HTTP의 서비스별 분류 기준

이 장에서는 HTTP 트래픽을 서비스 그룹 및 그 그룹에 속하는 어플리케이션별로 분석하는 시스템의 구조와 세부 분류 알고리즘에 대해 설명한다.

4.1 모니터링 시스템

네트워크 트래픽을 모니터링하기 위한 대표적인 시스템에는 MRTG, Ntop 등이 있다. MRTG는 표준 네트워크 관리 프로토콜인 SNMP를 이용하여 트래픽 모니터링 및 관리를 위해서 널리 사용 중인 툴이다. 그래프를 포함한 웹페이지를 제공하여 모니터링의 용이성을 제공하지만 전체 트래픽 양의 변화 추이만을 제공할 뿐 상세한 트래픽 분석을 제공하는 데는 부족한 점이 많다. Ntop은 Flow 기반으로 실시간 네트워크 트래픽을 모니터링 해주는 시스템이다. 실시간 트래픽을 상세하게 분석하여 주지만, 과거 트래픽 정보를 확인할 수 없으므로 장기간의 트래픽을 분석하여 얻을 수 있는 트래픽 패턴 등의 정보를 얻을 수 없는 단점이 있다.

그림 1은 HTTP 트래픽 수집 및 분류 시스템의 전체 구조를 보여준다. 목표 네트워크 라우터를 통해 들어오고 나가는 모든 패킷을 미러링을 통해 패킷 수집기(Packet Collector)가 수집하고, 이를 플로우 생성기(Flow Generator)가 받아서 TCP/IP 헤더에서 5-tuple

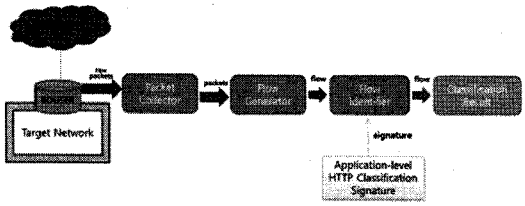


그림 1. HTTP 트래픽 분류 시스템 구조

정보(source IP address, source port, destination IP address, destination port, protocol)가 같은 패킷에 대해서는 같은 플로우(flow)로 묶는 작업을 수행 한다[5].

플로우 생성기가 5-tuple 정보를 기반으로 플로우를 생성하면서 시그니처 기반 분석을 통해 HTTP 플로우의 경우, 어플리케이션 프로토콜 부분에 HTTP 플로우임을 표시한다. HTTP는 어플리케이션 레벨의 프로토콜로 패킷 페이로드에서 HTTP 패킷에서 나타나는 시그니처를 추출하여 HTTP 트래픽을 분류한다. 페이로드에서 'HTTP', 'OPTIONS', 'GET', 'HEAD', 'POST', 'PUT', 'DELETE', 'TRACE', 'CONNECT' 등의 HTTP 헤더 필드에 나오는 시그니처들이 나오면 이를 HTTP 플로우로 표시하게 되고 플로우 식별자(flow identifier)는 이 정보를 바탕으로 HTTP 플로우만을 따로 추출하여 어플리케이션 레벨의 분류를 추가적으로 수행하여 그 분류 결과를 DB에 저장하게 된다.

4.2 분석 알고리즘

HTTP 프로토콜을 사용하는 어플리케이션을 분석하기 위해서는 HTTP의 페이로드 정보가 아닌 헤더 정보만을 분석한다. HTTP 헤더 네 가지 중에 가장 방대한 양을 가진 요청 헤더의 정보를 사용한다. 요청 헤더 정보에는 User-Agent, Method, Host, Referrer 정보 등 특정 어플리케이션을 분석할 수 있는 정보들이 있다. 이중에서 클라이언트 소프트웨어 버전의 정보를 가지고 있는 User-Agent[4]를 기준으로 어플리케이션을 분석한다. 또한 User-Agent 정보가 존재하지 않는 플로우의 경우 Host와 Method 등 헤더의 User-Agent 이외의 각각의 필드 정보를 통해서 어플리케이션을 파악해 낸다. 만약 이 단일 정보로 분석이 불가능 할 경우 Method와 Host, Content-type 등의 헤더 정보를 조합하여 어플리케이션을 분석해 낸다.

그림 2는 HTTP 트래픽의 어플리케이션별 분석 알고리즘을 나타낸다. 먼저 그림 2의 왼쪽 부분은 HTTP 플로우 분석을 위한 전처리(Preprocessing) 과정으로 이는 오프라인에서 수작업으로 이루어진다. HTTP 플

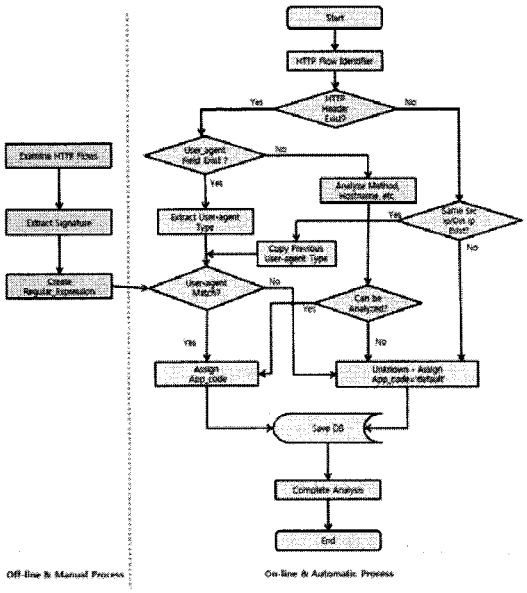


그림 2. HTTP 트래픽 분류 알고리즘

로우를 따로 모아 오프라인에서 어플리케이션을 찾음으로써 온라인상에서 실시간으로 HTTP 어플리케이션 분석을 수행할 수 있다. 먼저 HTTP 헤더의 User-Agent 필드를 분석하여 어플리케이션별 시그니처를 추출하고 각 어플리케이션별로 표 1에 정리한 서비스 그룹을 표시한다. 분석된 시그니처를 다음 그림 3과 같이 정규 표현식(regular expression)으로 나타낸다. 각 어플리케이션의 시그니처를 정규 표현식으로 나타냄으로써 단순한 스트링 매칭만으로는 분석할 수 없는 것을 효율적으로 매칭할 수 있다.

그림 3은 User-Agent 값이 약간씩 다른 인터넷 익스플로러 8.0(Internet Explorer 8.0)의 표기법이다. 이를 인터넷 익스플로러 정규 표현식을 사용함으로써 간단하게 표기할 수 있다. 생성된 정규 표현식에 각각의 어플리케이션 번호(App_Code)를 부여한다. 이 어플리케이션 번호는 실시간 처리시 User-Agent와 매칭될 경우 해당 플로우에 부여되고, 이를 통하여 후에 웹을 통해 어플리케이션별 분류 및 어플리케이션을 그룹핑한 서비스별 분석을 수행할 수 있다.

실시간 온라인 시스템은 그림 1의 플로우 생성기가 생성한 플로우 식별자(flow identifier)를 검사하

» Internet Explorer
· Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; InfoPath.2; InfoPath.1)
· Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; GT87.1; Net CLR 2.0.50727; Net4.0C; Net4.0E; NET CLR 3.0.45062152; NET CLR 3.5.30729; InfoPath.2)
정규표현식 : ^Mozilla\WW/(0-9)\WW.(0-9).*\WW(compatible; *MSIE *(0-9)\WW; *\WW)$

그림 3. MSIE User-Agent와 정규표현식

```
POST /game/object_information.php HTTP/1.1
Host: rts.gamekiss.com
Accept: */*
Accept-Encoding: gzip
Content-Length: 38
Content-Type: application/x-www-form-urlencoded

HTTP/1.1 200 OK
Date: Thu, 07 Jul 2011 02:59:55 GMT
Server: Apache/2.2.4 (Unix) PHP/5.3.3
X-Powered-By: PHP/5.3.3
Content-Length: 80
Connection: close
Content-Type: text/xml; charset=utf-8
```

그림 4. HTTP 요청/응답 헤더

여 HTTP 트래픽만 분석하게 된다. 먼저 HTTP 플로우에 HTTP 헤더가 존재하는지를 점검한다. HTTP 헤더가 존재하지 않는 경우에는 이전 플로우의 연속된 데이터이므로 헤더의 5-tuple의 정보를 확인하여, 이전 플로우와 근원지주소(srcIP), 목적지주소(dstIP), 포트번호가 같은 것을 찾아 같은 플로우로 그룹핑하고 이전 플로우의 User-Agent 값을 현재 플로우에 복사한 후, HTTP 헤더가 존재하는 플로우의 User-Agent를 분석하는 작업과 같은 작업을 수행한다.

헤더가 존재하는 플로우는 User-Agent 필드값을 추출한 후 전처리부분에서 미리 생성한 정규 표현식(regular expression)과 매칭을 한다. 매칭되는 값이 존재하는 분류 가능한 플로우의 정보에는 정규 표현식에 해당되는 어플리케이션 번호를 할당하고 DB에 저장을 하게 된다. User_Agent와 매칭이 되지 않았을 경우에는 어플리케이션 값을 'default'로 부여하고, 이런 플로우의 경우 추후에 오프라인에서 분석을 통해 정규표현식을 추가하는 작업이 이루어져야 한다. User-Agent가 존재 하지 않은 플로우의 경우에는 헤더의 Method, Host, Content 타입의 확인을 통하여 플로우의 어플리케이션별 분류를 수행한다.

그림 4는 User-Agent가 존재 하지 않는 플로의 요청헤더와 응답헤더이다. 그림 4의 요청 헤더에서 Method 정보를 이용하여 하이퍼텍스트 생성 언어에 포함되어 동작하는 스크립팅 언어인 PHP(Personal Hypertext Preprocessor)파일이라는 것을 확인 할 수 있다. 응답 헤더의 Content-Type을 통하여 전송되는 데이터가 xml 파일임을 알 수 있다. 이 두 가지 정보를 통하여 브라우징이라는 정보를 획득하고 이를 browser 서비스로 분류한다.

V. 분석 결과

5.1 데이터

HTTP 트래픽의 분석을 위하여 강원대학교 컴퓨터과학과의 255 Host 학내망 트래픽을 2011년 7월 7일 12시부터 7월 10일 12시까지 3일간 수집하였다. 컴퓨터과학과의 학내망은 10G로 스위치를 통하여 강원대학교 백본망에 연결되어 있다. 표 2는 3일간 수집한 트래픽의 플로우의 개수(Flow), 바이트(Byte) 양, 패킷의 개수(Packet)를 보여 준다.

표 2. 2011년 7월 7일 pm.12 ~ 7월 10일 pm.12시까지 수집한 데이터

Day	Traffic	Flow	Byte	Packet
7일 12시 ~ 8일 12시	Total_traffic	6,160,408	531,517,105,496	509,502,819
	HTTP_traffic	985,094 (15.99%)	57,938,752,730 (10.83%)	65,080,732 (11.43%)
8일 12시 ~ 9일 12시	Total_traffic	5,662,689	329,079,118,175	361,575,389
	HTTP_traffic	678,366 (11.98%)	32,321,015,856 (9.82%)	37,066,976 (10.25%)
9일 12시 ~ 10일 12시	Total_traffic	4,178,463	176,871,180,942	208,065,568
	HTTP_traffic	372,342 (8.91%)	13,589,868,560 (7.68%)	16,777,318 (7.92%)

그림 5는 표 2에서 제시한 3일간의 HTTP 트래픽을 분석한 결과를 보여준다. 전체 트래픽에서 HTTP 트래픽이 차지하는 비율을 요약해서 보여주는 화면이다. 그림 5에서 보듯이 HTTP 트래픽은 전체 트래픽의 플로우 기준으로는 약 11%, 바이트 기준으로는 9%, 패킷 기준으로 9%를 차지한다. 각 3일이 약간의 차이만 있을 뿐 전체 트래픽에서 차지하는 HTTP의 트래픽은 유사하게 나타났다.

그림 6은 HTTP 트래픽을 표 1에서 제시한 서비스 그룹별로 분류한 화면이다. 각 분류 화면들은 플로우 개수를 기준으로 내림차순 정렬하여 보여주고 있다. HTTP 트래픽은 약 95% 정도의 트래픽이 서비스 그룹과 세부 어플리케이션 항목으로 분석이 가능했다. 그 중에서 웹 브라우징이 80%이상을 차지하고 있었으며 P2P의 비율이 높았는데, P2P서비스는 파일을 다운받고 전송하기 위한 것도 있지만 대부분은 파일을 다운받기 위해 파일의 이름을 검색하는 부분에서 많이 사용되는 것을 확인할 수 있었다.

분석의 정확도는 일부 컴퓨터에 트래픽을 측정하는 에이전트(TMA: Traffic Measurement Agent)[7]를 설

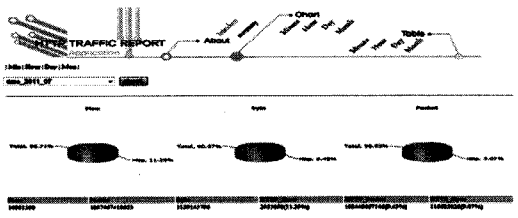


그림 5. HTTP 트래픽의 요약 정보

	Category	Flow	Byte	Packet
1	Browser	1740320 (85.49%)	76873038036 (74.31%)	90277374 (76.1%)
2	P2P	74554 (3.66%)	5128180342 (4.96%)	5598232 (4.71%)
3	Unknown	67982 (3.34%)	6320342850 (6.11%)	6651374 (5.61%)
4	Messenger	26614 (1.31%)	606371158 (0.59%)	811520 (0.68%)
5	Web_app	24276 (1.19%)	298330422 (0.29%)	760218 (0.64%)
6	Crawler	28710 (1.41%)	3279789956 (3.17%)	3506634 (2.96%)
7	Web_based_service	16288 (0.8%)	2709036892 (2.62%)	2724964 (2.3%)
8	SW_update	26668 (1.31%)	2998190254 (2.9%)	3229938 (2.72%)
9	Multimedia	18208 (0.89%)	3414829876 (3.3%)	3204224 (2.7%)
10	Security	6290 (0.31%)	237899486 (0.23%)	280344 (0.24%)
11	Mobile	5892 (0.29%)	1585757874 (1.53%)	1590204 (1.34%)

그림 6. HTTP 트래픽의 서비스 그룹별 분류

치하고 각 컴퓨터에서 생성한 트래픽에 대해 어떤 어플리케이션이 생성했는지를 로그를 남긴다. 이를 본 시스템에서 분류한 결과와 비교하여 그 결과의 정확도를 검증하였다.

만약 분석되지 않은 5%의 트래픽에 침입이나 보안을 위협하는 트래픽이 있다면 이는 전체를 분석하지 못한 것과 같은 결과를 낳는 것이므로 3장에서 말한 오프라인에서의 추가적인 분석을 통해 HTTP 프로토콜을 사용하는 어플리케이션을 추가하는 등의 분석률을 높이는 작업이 이루어져야 한다.

메신저 프로그램을 실행하였을 때 메신저 화면을 통하여 광고를 전송하는 역할로 HTTP 프로토콜을 사용했다. 크롤러의 경우는 여러 개의 로봇들이 많은 양의 트래픽은 아니지만 주기적으로 웹페이지 검색을 위한 사전 데이터 저장을 위해 발생했다. 스마트폰의 보급과 타블렛 PC 등의 보급으로 인하여 모바일 트래픽이 증가하기 시작하였는데, 모바일 트래픽의 경우에는 어플리케이션을 다운받거나, 웹브라우징을 사용하기 위하여 사용된다.

또한 멀티미디어 트래픽은 음악을 들을 때 가사를 전송하거나 영화를 볼 때 자막을 전송하는 목적으로 HTTP를 사용했으며, 멀티미디어 플레이어 안에서 광고를 전송하는 곳에서도 HTTP를 사용하였다. 이외에도 보안이나 클라우드 서비스와 같은 웹 기반 서비스 웹 어플리케이션들이 HTTP 프로토콜을 사용하는 것을 확인할 수 있었다.

HTTP 트래픽의 서비스 그룹별 분류를 통해 단순히 사용빈도가 높은 어플리케이션 목록을 나열하는 것을 넘어 서비스 그룹별 사용 분포도를 확인할 수 있었다. 그림 7과 그림 8을 통하여 각 서비스별 어플리케이션의 사용빈도 또한 파악이 가능하다.

	Name	Flow	Byte	Packet
1	MSIE	1496934 (86.01%)	62883094420 (81.54%)	74390284 (82.4%)
2	Chrome	164134 (9.43%)	7821703702 (10.17%)	9008830 (9.98%)
3	Web_unknown	35382 (2.03%)	4677483890 (6.08%)	4829492 (5.35%)
4	Firefox	21892 (1.26%)	1034788552 (1.35%)	1209228 (1.34%)
5	Safari	11338 (0.65%)	394749100 (0.51%)	514574 (0.57%)
6	AppleWebKit	10230 (0.59%)	234231052 (0.31%)	318592 (0.35%)
7	Opera	400 (0.02%)	7007430 (0.01%)	10274 (0.01%)

그림 7. 브라우저 서비스의 어플리케이션 분류

	Name	Flow	Byte	Packet
1	Googlebot	14562 (50.72%)	737054574 (22.47%)	932178 (26.58%)
2	Naver_bot	7274 (25.34%)	1135960028 (34.64%)	1334812 (33.21%)
3	Yahoo!Slurp	4718 (16.43%)	340971900 (10.4%)	383890 (10.95%)
4	bingbot	1810 (6.3%)	318939198 (9.72%)	317798 (9.06%)
5	Daum_bot	216 (0.75%)	703208020 (21.44%)	591134 (16.86%)
6	google_image_bot	122 (0.42%)	43507226 (1.33%)	46690 (1.33%)
7	baidu_robot	8 (0.03%)	140350 (0%)	224 (0.01%)

그림 8. 크롤러 서비스의 어플리케이션 분류

그림 7은 브라우저 그룹의 어플리케이션 리스트들을 보여준다. 그림 7에서 보듯이 MSIE(익스플로러)의 비중이 브라우저 중에서 80% 이상을 차지하고 있다. 그 이외에 크롬, 파이어폭스, 사파리 등의 웹브라우저들이 조금씩 사용되는 것을 볼 수 있다.

그림 8은 크롤러의 어플리케이션 리스트를 보여준다. googlebot이 가장 많이 차지하고 있으며 naver, yahoo, daum의 포털사이트들의 로봇들이 많이 차지하고 있다. Byte와 Packet의 경우 주요 포털사이트들이 비슷한 비율을 유지하고 있는 것을 볼 수 있었다.

Ⅵ. 결론 및 향후 연구

본 논문에서 제안한 어플리케이션별 분류 방법으로 플로우 개수를 기준으로 96%에 달하는 HTTP 트래픽의 클라이언트 프로그램을 분석할 수 있었다. 이를 HTTP 트래픽 분석이 필요한 엔터프라이즈 네트워크에 적용한다면 네트워크 운영자 측면에서는 더 효율적으로 HTTP 트래픽을 제어하는 것이 가능할 것으로 보인다.

향후 과제로는 처음으로는 현재의 시스템을 캠퍼스 전체로 확장하여 HTTP 트래픽을 분석하고, 다른 캠퍼스 네트워크에 적용할 예정이다. 이를 통하여 본 논문에서 제안한 시스템의 분석에 대한 정확도 및 성능

에 대한 신뢰도를 향상시킬 계획이다. 추가적으로 HTTP 트래픽 중에서 모바일 트래픽에 대한 세부적인 분석을 수행할 예정이다. 어떤 기기들의 어떤 어플리케이션들이 HTTP를 사용하고 있는지 세부적으로 분석하여 증가하는 모바일 트래픽의 추세를 분석할 예정이다.

참 고 문 헌

[1] Wei Li, Andrew W. Moore, Marco Canini, "*Classifying HTTP Traffic in the New Age*", ACM SIGCOMM'08, Seattle, USA, Aug., 17~22, 2008.

[2] Fabian Schneider, Sachin Agarwal, Tansu Alpcan, and Anja Feldmann, "*The New Web: Characterizing AJAX Traffic*", In Proceedings of Passive and Active Measurement Conference 2008 (PAM 2008), April 2008, Cleveland, OH.

[3] K. Kim, B. Lee, T. Kwon, N. Ryo, K. Okamura, and Y. Lee, "*Japanese Content classification of HTTP Traffic*," DICOMO 2009, Beppu, July, 8, 2009.

[4] 랭키닷컴, *http://www.rankey.com*

[5] 박진완, 박상훈, 김명섭, "Flow를 이용한 호스트 기반 트래픽 모니터링 및 분석", 2008 *한국통신학회 하계종합학술발표회 논문집*, July, 2008, pp.197-198.

[6] R.Fielding,et.al.,"*HypertextTransferProtocol--HTTP/1.1*", RFC 2616, Sep., 2004.

[7] 윤성호, 노현구, 김명섭, "TMA(Traffic Measurment Agent)를 이용한 인터넷 응용 트래픽 분류", 2008 *한국통신학회 하계종합학술발표회 논문집*, July, 2008, p.618.

최 미 정 (Mi-jung Choi) 종신회원

1998년 이화여자대학교 컴퓨터공학과 학사

2000년 포항공과대학교 컴퓨터공학과 석사

2004년 포항공과대학교 컴퓨터공학과 박사

2004년~2005년 프랑스 INRIA 연구소 박사후 연구원

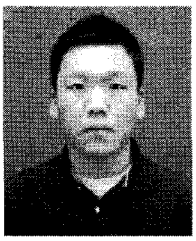
2005년~2006년 캐나다 워터루대학 컴퓨터과학부 박사후 연구원

2006년~2008년 포항공대 컴퓨터공학과연구 조교수

2008년 8월~현재 강원대학교 컴퓨터과학과 조교수

<관심분야> 트래픽 모니터링 및 분석, 미래 인터넷, M2M 네트워크 및 서비스 관리

진 창 규 (Chang-gyu Jin) 학생회원

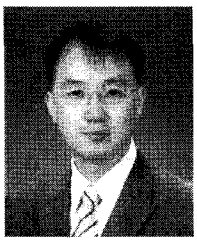


2011년 강원대학교 컴퓨터과학학사

2011년~현재 강원대학교 컴퓨터과학과 석사과정

<관심분야> 트래픽 모니터링 및 분석, 네트워크 관리 및 보안

김 명 섭 (Myung-sup Kim) 종신회원



1998년 포항공과대학교 전자계산학과 학사

2000년 포항공과대학교 컴퓨터공학과 석사

2004년 포항공과대학교 컴퓨터공학과 박사

2004년~2006년 Post-Doc.Dept. of ECE, Univ. of Toronto, Canada

2006년~현재 고려대학교 컴퓨터정보학과 부교수

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석, 멀티미디어 네트워크