응용계층 트래픽 분류를 위한 응용프로그램 선정 및 시그니처 추출 방법

정재윤 [1], 최영락 [1], 리건 [1], 홍원기 [1,2]

포항공과대학교 컴퓨터공학과 [1], 포항공과대학교 정보전자융합공학부 [2]

{dejavu94, dkby, gunine, jwkhong}@postech.ac.kr,

요 약

인터넷 트래픽은 급증하는 응용프로그램 수와 더불어 계속해서 증가 하고 있다. 이러한 인터넷 트래픽을 효과적으로 관리 하기 위하여 응용계층에서 트래픽을 구분함으로써 네트워크 관리자에게 응용프로그램들의 네트워크 사용 현황 등 보다 상세한 정보를 제공하기 위한 연구가 진행되어 왔다. 하지만 이러한 분류 방법 및 기준은 네트워크 상황과 네트워크 사용자 구성에 따라 매우 다르기 때문에 다른 연구의 분류 기준들을 그대로 네트워크에 적용할 경우 만족할 만한 결과를 얻기 어렵다. 또한 분류 대상이 되는 응용프로그램 선정 역시 대부분 다른 기관에서 조사된 인기도를 기반으로 하고 선정하고 있지만 분석할 네트워크에서도 이들이 인기 응용프로그램이라는 확신은 할 수 없다. 특히 최근 급증하는 모바일 응용프로그램들은 단기간에 트래픽 양이 급증 하지만 그 수명이 짧아 이들을 효과적이고 빠르게 파악하고 시그니처를 추출해 내야만 한다. 본 연구에서는 원격 서브넷 주소를 사용한 그룹핑 방법을 사용하여 현재 네트워크에서 많이 사용되는 응용프로그램 트래픽을 확인하고 시그니처를 추출하는 방법을 제시한다.

1. 서론

급증하는 네트워크 트래픽과 단말, 그리고 응용프로그램은 네트워크에 더욱 많은 부담을 주고 있다[1]. 네트워크 관리자는 이러한 네트워크 상태를 정확히 파악하고 진단해야 하며, 트래픽 분석 연구는 네트워크 관리자에게 다양한 트래픽 정보를 제공하는 것을 목적으로 한다. 특히 트래픽이 어떤 응용프로그램으로부터 생성되었는지 확인하는 응용계층의 트래픽 분석연구는 네트워크 수준에서 관련 정보를 제공하지 않는다는 점에서 많은 연구가 진행되어 왔다.

트래픽을 응용프로그램에 따라 구분하기 위해서는 분류 기준을 선정 해야 한다. 포트 넘버 기반의 분류 방법부터 패이로드의 특정 바이트 패턴을 사용하는 방법까지 다양한 방법이 사용되고 있으나 이러한 분류 기준을 생성해 내는 과정은 여전히 수작업에 의존하고 있다 특히 최근 수많은 모바일 응용프로그램들이 새로 생겨나도 또 사라지고 있으며 그 유행 주기도 매우 짧다는 특징을 보이고 있다. 따라서 관리자는 더욱 빈번하게 이들 트래픽의 분류 기준을 찾아내야 하지만 다수의 응용프로그램이 섞여있는 트래픽에서 각각의 응용프로그램 트래픽을 확인하는 일은 거의 불가능하다.

다음의 일반적인 트래픽 분석 연구과정을 살펴보면

(1) 시그니처를 추출 하고자 하는 응용프로그램을 선정한다.
(2) 해당 응용프로그램 트래픽을 수집한다
(3) 수집된 트래픽으로부터 포트 넘버, 공통 바이트 패턴 등 시그니처를 추출한다.
(4) 추출한 시그니처를 트래픽 분석 시스템에 적용한다.
(5) 정답 트래픽과 분석된 트래픽을 비교하여 정확도를 검증한다.

순서로 연구가 진행된다. 본 연구는 (1)-(3) 단계의 문제인 분석 대상 응용프로그램의 선정, 응용프로그램 트래픽 확인, 시그니처 추출 문제를 해결 하고자 한다.

트래픽을 분석하기 위해 네트워크 관리자는 응용프로그램의 시그니처를 확보해야만 한다. 이를 위해 가정 먼저 네트워크 사용자가 많이 사용할만한 응용프로그램을 (1)의 단계에서 선정하게 된다. 이 과정에서 현재는 다른 기관에서 조사된 응용프로그램 사용 순위나 앱스토어 등 마켓의 인기 순위를 기반으로 분석 대성을 선정하고 있다. 하지만 이런 방법으로 응용프로그램을 선정 할 경우 실제 분석 하고자

본 연구는 한국연구재단을 통해 교육과학기술부의 세계수준의 연구중심대학육성사업(WCU)(R31-2010-000-10100-0)과 2011 년도 정부(교육과학기술부)의 재원으로 한국연구재단-차세대정보컴퓨팅기술개발사업(No. 2011-0020518)의 지원을 받아 수행된 연구임

하는 네트워크의 상황과 정확히 맞지 않는 경우가 대부분이다. 이는 상용 트래픽 분석 도구를 적용하는 과정에서도 동일하게 발생한다. 따라서 분석할 네트워크에서 사용되는 응용프로그램이 무엇인지 직접 확인하는 작업이 필요하지만 관리자에게는 대략의 트래픽 trace 외에는 특별한 정보가 제공되지 않는 것이 대부분이다. 다음 (2)의 단계는 특정 응용프로그램의 트래픽을 수집하기 위해 실제 응용프로그램을 직접 PC 나 스마트폰에 설치하여 관리자가 직접 트래픽을 발생시킴으로서 정확한 정답 트래픽을 생성한다. 이는 수집되는 정답 트래픽이 매우 제한적일 뿐만 아니라 많은 시간과 노력을 필요로 하는 작업이다. (3)의 단계 역시 네트워크 관리자가 직접 트래픽 패턴 및 패킷의 페이로드를 분석하여 시그니처를 추출하는 작업을 수행한다. LCS 알고리즘 등을 사용한 시그니처 자동 생성 연구 등[2][3]이 수행되었으나 추출된 시그니처가 어떤 응용프로그램에 대한 것인지는 여전히 관리자가 직접 작성 해야만 한다.

본 연구에서는 다수의 응용프로그램 트래픽이 혼재하는 트래픽 trace 로부터 주요 응용프로그램 트래픽을 확인하고 해당 시그니처를 추출하는 방법을 제시한다. 이는 네트워크 관리자 및 트래픽 분석 연구에서 빈번하게 겪는 문제로써 본 연구에서는 서브넷 기반의 트래픽 그룹핑 방법을 사용하여 문제를 해결 한다.

2. 관련 연구

네트워크 트래픽 분류 및 분석 연구에서 시그니처 생성 및 추출은 수작업에 의존하고 있기 때문에 이를 자동화 하려는 연구가 진행되었다. 일반적으로 시그니처라 함은 응용프로그램 트래픽이 고유하게 갖고 있는 패킷의 바이트 패턴을 뜻한다. 따라서 Longest Common Substring 을 찾아 이를 시그니처로 활용하는 연구[2]와 여러 개의 공통 substring 을 모두 찾아 사용하는 방법[3]이 제안되었다. 하지만 동일한 응용프로그램 트래픽끼리 그룹핑을 하는 작업과 시그니처와 응용프로그램을 맵핑 하는 작업은 여전히 관리자의 수고를 필요로 한다.

또한 최근 모바일 트래픽 분석 연구에서는 단말별, 응용프로그램 별 트래픽 특성을 분석하였다. 실제 연구는 각 트래픽의 특성을 비교 분석하는 것이었으나 응용프로그램 트래픽 분석 기술 사용하여 각 응용프로그램 트래픽을 구분하는 작업이 선행되었다[4][5]. 이들 연구에서는 마켓의 인기 순위를 참고하여 분석 대상 응용프로그램을 선정하였다. 하지만 이들 연구는 실제 네트워크를 사용하는 응용프로그램들과 서로 다를 수 있고 유행의 주기가 매우 짧은 모바일 응용프로그램의 특성을 고려하지 못하고 있다.

또한 네트워크 사용자 구성과 용도에 따라 네트워크에서 발생하는 트래픽의 종류가 매우 다르기 때문에 다른 연구나 시스템에서 사용되는 분류 기준을 그대로 분석할 네트워크에 적용 할 수 없다.

본 연구는 분석 대상 트래픽만을 사용하여 응용프로그램 사용 상태를 파악하고 주요 응용프로그램 목록 선정 할 수 있는 방법을 제시한다. 이는 관리자에게 분석 대상 응용프로그램을 선정 할 수 있는 기준을 제공한다. 정답 트래픽을 기반으로 시그니처를 추출하는 것에 비해 응용프로그램의 분류 기준을 아주 정확하게 찾아낼 수는 없지만, 본 연구는 트래픽 이외에 아무런 정보도 제공받지 못한 관리자로 하여금 네트워크 분석을 시작할 때 적용 할 수 있는 분석 방법론을 제시한다.

3. 본론

그림 1은 제안하는 응용프로그램 트래픽 확인 시스템과 POSTECH 트래픽 수집 시스템을 보여주고 있다. Optical Tap 을 사용하여 미러링 된 트래픽으로부터 플로우 정보와 패킷 정보를 추출한다. 트래픽 분석 엔진은 추출된 트래픽 정보로부터 응용프로그램 단위로 트래픽을 구분하고 추후 분산 처리가 가능하도록 확장 할 계획이다. 본 연구에서 제안하는 응용프로그램 트래픽 확인은 원격 주소의 서브넷 단위로 트래픽 양의 랭킹을 계산하는 Unknown Traffic Ranker 와 서브넷 단위의 그룹핑된 트래픽으로부터 시그니처를 추출하는 Classifier Extractor, 추출된 시그니처를 관리하는 Classifier Manager 로 구성된다.

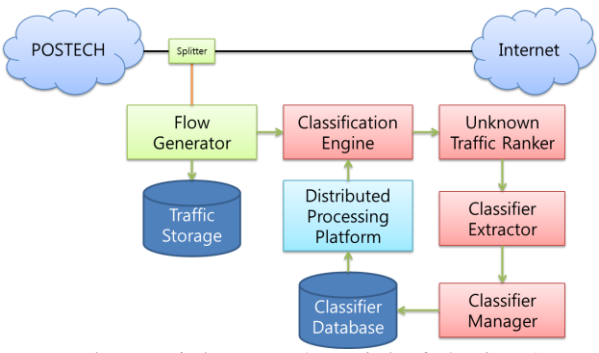

그림 1 응용프로그램 트래픽 확인 시스템

3.1. 응용프로그램 트래픽 확인

본 연구에서 분석 대상이 되는 응용프로그램 선정은 네트워크에서 각 응용프로그램의 사용량을 기반으로 찾아내는 것을 목표로 한다. 본 연구에서는 원격 호스트를 서브넷 단위로 그룹핑 하여 트래픽을 구분하고 이들은 같은 응용프로그램 트래픽이라고 가정한다. 실제 하나의 서브넷에 많아야 3~4 개의 응용프로그램 트래픽이 확인되었고

이는 관리자가 쉽게 구분해 낼만한 정도이다.

서브넷 단위의 트래픽 그룹핑은 서버-클라이언트 기반의 응용프로그램 트래픽 확인에 매우 효과적이다. 각 기업의 서버들은 그들이 소유하고 있는 서브넷에 배포 되어 있고 IP 주소를 고정적으로 사용하고 있기 때문에 트래픽이 여러 서브넷으로 흩어지지 않기 때문에 동일한 응용프로그램 트래픽들만이 하나의 서브넷에서 확인된다.

반면 P2P 응용프로그램은 서버가 아닌 일반 PC 환경에서 동작하는 것이 대부분이다. 하지만 PC 사용자가 사용하는 응용프로그램 수는 매우 제한적이고, P2P 트래픽은 유사한 트래픽이 빈번하게 나타난다는 특징을 갖고 있다. 따라서 관리자는 하나의 서브넷에서 유사하게 반복되는 P2P 응용프로그램 트래픽을 확인 할 수 있으며 동시에 다른 서브넷에서도 같은 트래픽을 확인 할 수 있다.

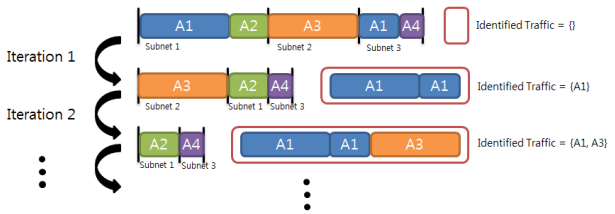

그림 2 응용프로그램 트래픽 확인 과정

```
While (completeness < th_comp || can extract signature)
    Pick n heaviest destination subnets in unknown traffic
    For each destination subnet
        Pick m flows
        Extract common substring in payload of m flows
        End if
    End for
    Classify traffic using extracted signatures
End while
```

그림 3 서브넷 단위 트래픽 그룹핑

그림 2는 제안하는 응용프로그램 트래픽 확인 과정을 나타내고 있다. 본 연구에서는 분석되지 않은 트래픽 양이 많은 서브넷부터 시그니처를 추출함으로써 분석 할 네트워크에서 주로 사용되는 트래픽 순서로 응용프로그램을 확인 할 수 있다. 그림 2에서 A*n* 은 각각의 응용프로그램 트래픽을 나타내고 트래픽 양이 많을수록 그 길이가 길다. 가장 많은 트래픽이 발생한 subnet1 으로부터 A1 응용프로그램 트래픽이 확인되었다면 그 다음 iteration 에서는 subnet2 가 분석되지 않은 트래픽이 가장 많은 트래픽이 발생한 subnet 이 된다. 비록 첫번째 iteration 에서 A2 트래픽은 확인하지 못하였으나 3 번째 iteration 에서 다시 subnet1 을 분석하게 되므로 전체 트래픽에 대해 빠짐 없이 응용프로그램을 확인 할 수 있다.

그림 3은 iterative 한 방법으로 응용프로그램 트래픽을 확인하는 알고리즘을 기술 하고 있다. 알 수 없는 트래픽이 가장 많은 n 개의 서브넷을 선택하고 각각의 서브넷에서 m 개 플로우로부터 패이로드 정보를 포함한 트래픽 정보를 분석한다. 이로부터 각 응용프로그램 트래픽이 무엇인지 확인하고 시그니처를 추출하게 된다.

3.2. 시그니처 추출

제안하는 응용프로그램 확인 방법은 서브넷 단위로 트래픽을 우선적으로 그룹핑 하기 때문에 시그니처 역시 서브넷 단위로 추출되게 된다. 시그니처로 패이로드의 공통적인 바이트 패턴, 원격 서버 주소, 포트 넘버 등 다양한 포맷을 사용 할 수 있으나 본 연구에서는 패이로드의 공통 바이트 패턴을 기준으로 기술 한다.

시그니처 추출은 네트워크 관리자가 직접 플로우별 패이로드 스트림을 확인하고 추출하는 방법을 사용하였다. 플로우 패이로드 스트림에서 시그니처를 추출하는 직업은 그렇게 복잡하지 않으며 향후 자동 추출 알고리즘을 적용할 계획이다.

응용프로그램 트래픽의 패이로드 스트림은 크게 형식이 있는 프로톨과 형식을 알수 없는 프로토콜로 구분 할 수 있다. HTTP 가 형식이 있는 프로토콜의 대표적인 예이다. HTTP 뿐만 아니라 대부분의 응용프로그램들 역시 자체적인 응용계층의 프로토콜을 사용하고 있으나 프로토콜이 공개되어 있지 않아 파악하기 어렵다. 본 연구에서는 시그니처를 서브넷 단위로 추출하고 적용하기 때문에 관리자는 현재 서브넷의 트래픽이 어떤 응용프로그램의 트래픽인지만 확인한다면 단순히 해당 그들의 공통점을 찾는 것으로 시그니처를 대신 할 수 있다. 예를 들어 어떤 서브넷에서 HTTP 트래픽 다수가 'Host: facebook.com\r\n'이라는 HTTP-host 값을 갖고 있다면 이 서브넷은 Facebook 의 서버들이 존재하는 서브넷이며 위 공통 substring 은 해당 서브넷에서 Facebook 트래픽을 구분할 수 있는 기준이 된다. 시그니처 적용 역시 서브넷 단위로 적용하기 때문에 관리자는 현재 서브넷 안에서 구분 가능한 어떠한 기준을 선택 하더라도 문제가 없다. 따라서 관리자는 현재 서브넷에서 트래픽이 구분 가능하다면 어떠한 기준도 시그니처로 정의 할 수 있다.

3.3. 시그니처 적용

앞 장에서 설명한 바와 같이 제안하는 방법은 서브넷 단위로 트래픽을 분석하여

응용프로그래밍을 확인하는 방법을 사용하기 때문에 시그니처 역시 서브넷 단위로 추출되게 된다. 이렇게 추출된 시그니처는 크게 local 시그니처와 global 시그니처로 구분된다. Local 시그니처는 시그니처가 추출된 서브넷에만 적용하며 반대로 global 시그니처는 모든 서브넷에 대해 적용하게 된다. 실제 하나의 원격 서브넷에서 관찰 되는 트래픽의 종류가 3~4 가지 정도이기 때문에 수백개의 시그니처를 모두 적용하는 것은 비효율적이다.

관리자가 시그니처를 추출하고 응용프로그램 이름을 확인하는 단계에서 시그니처를 local 로 적용할 것인지 global 로 적용 할 것인지 결정하게 된다. 응용프로그램이 웹브라우져나 P2P 응용프로그램일 경우 트래픽이 다수의 서브넷에서 동시에 발견되는 경향이 크기 때문에 global 로 적용하는 것이 유리하다. 하지만 서버-클라이언트 기반의 응용프로그램이나 웹 기반 응용프로그램일 경우 local 시그니처로 적용하는 것이 분석 속도나 정확도 측면에서 유리하다. 특히 제안하는 방법은 응용프로그램 트래픽을 확인 및 추출 하기 위해 불확실한 추론을 해야만 하는 네트워크 관리자가 하나의 서브넷에만 적용하면 되는 시그니처를 정의하면 되기 때문에 잘못 추출된 시그니처 적용에 대한 위험부담을 줄일 수 있다. 뿐만 아니라 P2P 시그니처 등은 다른 서브넷에서 추출된 시그니처와 비교 함으로써 보다 정확한 시그니처를 찾아 낼 수 있다.

4. 검증

제안하는 방법의 효율성을 보여주기 위하여 본 논문에서는 응용프로그램 트래픽을 정확하게 확인할 수 있는지를 실험하였다. 또 추출한 시그니처를 다른 시스템에서 사용하고 있는 시그니처와 비교 함으로써 응용프로그램 단위의 트래픽 수집 없이 응용프로그램 트래픽을 정확히 찾아 낼 수 있는지 확인하였다.

응용프로그램 확인을 위한 샘플 트래픽으로서 2012 년 3 월 1 일 오후 6 시(dataset1), 오후 7 시(dataset2) 각 1 분에 대하여 POSTECH 기숙사 지역 트래픽을 수집했으며 트래픽 양은 각각 2,651 MB, 833 MB 이다.

4.1. 응용프로그램 트래픽 확인

그림 4은 트래픽 양 순서로 상위 10,000 개의 서브넷에 대하여 시그니처를 추출하고 그에 따라 트래픽을 분류한 트래픽의 양을 보여주고 있다. Dataset1 은 약 1000 개의 상위 서브넷에서 전체 트래픽의 약 70% 정도를 분석해 내고 있음을 보여준다. Dataset2 역시 약 100 개의 상위 서브넷에서 전체 트래픽의 약 70% 정도를 분석해 내고 있다. 이는 서브넷 단위로 트래픽을 분석하고 적용하는 방법이 효과적으로 응용프로그램 트래픽을 찾아내고 이것이 전체 트래픽의 대부분을 차지하고 있음을 보여준다.

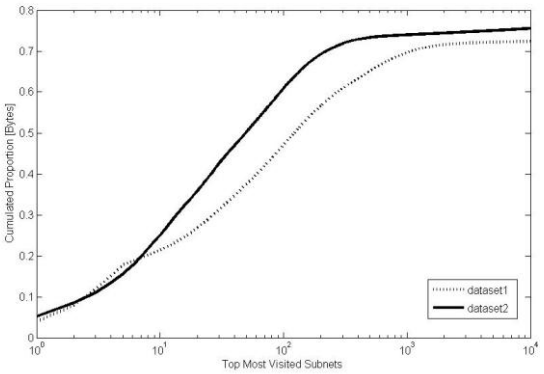

그림 4 트래픽 양 상위 10,000 개 서브넷에 대한 분석률

4.2. 시그니처 비교

제안하는 방법을 사용하여 상위 약 200 개의 서브넷으로부터 91 종류의 응용프로그램을 확인하였으며 111 개의 시그니처(local 95 개; global 16 개)를 확보하였다. 이는 전체 표본 트래픽 약 3.5 GB 중 76.2%를 구분해 낼 수 있는 정도이다. 네트워크의 트래픽이 어떤 응용프로그램 트래픽으로 구성되어 있는지 전혀 모르는 상태에서 접근하는 방법으로서 대다수의 응용프로그램 트래픽을 확인 할 수 있음을 보여준다.

표 1. 응용프로그램 확인과 시그니처 추출 결과 일부

분류	응용프로그램	시그니처
HTTP	Daum Cloud	User-Agent: DaumCloud.*\r\n.*android
	iTunes	User-Agent: iTunes-iPhone.*\r\n
	NateOn	User-Agent: NateOn.*\r\n
P2P	BitTorrent	\x13BitTorrent protocol;^d1:rd2:id20:;^d1:ad2:id20:;
	PandoraTV	^\x0ePando protocol; PeerSvrHost=p2prtmp1.pandora.tv
SSL	Google SSL	google\.com
	Facebook SSL	facebook.com; fbcdn-sphotos-a.akamaihd.net

표 1은 서브넷을 기반의 트래픽 그룹핑을 사용해 확인한 응용프로그램과 시그니처의 일부를 보여주고 있다. Daum 의 Cloud 응용프로그램과 같이 일부 응용프로그램 트래픽은 단말에 대한 정보를 포함하고 있다. (e.g. 모바일 프래임워크 이름, 단말 모델 등) P2P 응용프로그램의 대표적인 BitTorrent 의 경우 전체 110,853 의 모니터링된 서브넷 중 63,471 개의 서브넷에서 발견되어 global 시그니처를 적용 하여 매우 효과적으로 트래픽을 분류 해 낼

수 있다. 또한 SSL 트래픽의 경우 'hello 패킷'의 해더에서 추론 가능한 substring 을 사용하였으며 시그니처가 다른 응용프로그램 트래픽에도 존재할 가능성이 매우 높은 형태를 하고 있다(e.g. 'google\.com'). 본 연구의 서브넷 단위의 시그니처 추출 및 적용은 추출된 시그니처의 추출 및 적용 범위를 제한함으로써 해당 시그니처의 오탐율을 줄일 수 있겠다.

제안하는 방법은 응용프로그램으로부터 직접 발생시킨 트래픽이 아니고 구분하지 못한 트래픽으로부터 직접 응용프로그램 트래픽을 확인하는 방법이기 정답 트래픽으로부터 추출된 시그니처와 본 연구의 시그니처를 비교해 보았다. 본 연구에서는 비교 대상 연구 및 시스템으로 Clear Foundation 의 L7-filter[6]와 PacketLogic[7]의 시그니처를 선정하였다.

표 2를 통해 POSTECH 네트워크에서 확인된 응용프로그램 트래픽과 현재 PacketLogic (2012. 4. 2 업데이트 기준) 과 l7-filter (2009. 5. 28 업데이트 기준)가 분류 할 수 있는 응용프로그램 트래픽을 비교해 보았다. 제안하는 방법이 100% 정확하게 응용프로그램 트래픽들을 구분 할 수 있는 것은 아니지만 교내에서 사용되는 주요 응용프로그램 트래픽들을 확인 할 수 있다. PacketLogic 의 경우 최근까지 지속적으로 시그니처가 업데이트 되고 있으며 등록된 시그니처도 630 여개 이상이다. 하지만 대다수가 현재 교내 트래픽에서 발견되지 않고 있으며 또한 Web 기반의 응용프로그램 트래픽(대부분의 모바일 앱)은 따로 구분하지 않는 것으로 여겨진다. L7-filter 는 응용프로그램 프로토콜 단위로 트래픽을 구분하기 때문에 응용프로그램 트래픽을 구분하는데 어려움이 있으며 시그니처 업데이트도 지속적으로 이루어지지 않고 있다.

표 2. POSTECH 네트워크의 응용프로그램 트래픽과 PacketLogic 및 L7-filter 의 구분 가능한 응용프로그램 비교

응용프로그램	트래픽량 (MB)	Pro-posed	Packet-Logic	L7-filter
BitTorrent	1,424.01	O	O	O
Web Browser for Windows	719.11	O	X	X
Pandora TV	113.29	O	O	X
Kpeer	77.804	O	O	X
Apple Core Media Player for iPhone	67.087	O	X	X
CoFile File Share	18.623	O	X	X
Web Browser for iPad	17.119	O	X	X
Melon Music Streaming	12.263	O	X	X

5. 결론 및 향후 연구

본 연구에서 제안하는 응용프로그램 확인 및 시그니처 추출 방법은 네트워크 관리자가 현재 네트워크에 존재하는 대부분의 응용프로그램 트래픽들을 확인 할 수 있었다. 또한 다른 연구나 시스템과 네트워크 사용자 분포나 사용 지역이 달라 이들이 확인 할 수 없었던 응용프로그램 트래픽을 대부분 찾아내는 방법을 제시함으로써 관리자가 자신의 네트워크에 최적화된 트래픽 시그니처를 확보 할 수 있음을 보여주었다.

향후 연구로서 서브넷 단위로 그룹핑된 트래픽으로부터 시그니처를 자동 추출하는 알고리즘을 연구할 계획이다. 또한 SSL 트래픽등 암호화된 트래픽에 대한 트래픽 확인 연구 역시 진행되어야 하겠다. Condor 나 MapReduce 등을 사용한 분산 처리 방법을 서브넷 단위로 적용하여 분석 시간을 줄이는 연구도 진행할 계획이다.

6. 참고 문헌

[1] Cisco, "Visual Networking Index 2011-2016", Cisco, Feb. 14, 2012.

[2] B.C. Park, Y.J. Won, M.S. Kim, and J.W. Hong. "Towards Automated Application Signature Generation for Traffic Identification", Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS) 2008, Salvador, Brazil, Apr. 7-11, 2008, pp. 160-167.

[3] M. Ye, K. Xu, J. Wu, and H. Po, "AutoSig-Automatically Generating Signatures for Applications", IEEE 9th International Conference on Computer and Information Technology, Xiamen, China, Oct. 11-14, 2009.

[4] S.W. Lee, J.S. Park, H.S. Lee, and M.S. Kim, "A Study on Smart-phone Traffic Analysis", Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2011, Taipei, Taiwan, Sep. 21-23, 2011.

[5] J.Y. Chung, Y.R. Choi, B.C. Park, and J.W. Hong, "Measurement Analysis of Mobile Traffic in Enterprise Networks", Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2011, Taipei, Taiwan, Sep. 21-23, 2011.

[6] L7-filter, Clear Foundation, http://l7-filter.clearfoundation.com/.

[7] PacketLogic, PROCERA, http://www.proceranetworks.com/