Torrent Hunter: 토렌트 트래픽 완전 분석을 위한 트래픽 분류 시스템

(Torrent Hunter: Traffic Classification System for Torrent Traffic Perfection Analysis)

심 규 석, 윤 성 호, 정 우 석, 이 성 호, 김 명 섭

고려대학교 컴퓨터정보학과

{kusuk007, sungho_yoon, hary5832, tmskim}@korea.ac.kr

요 약

오늘날 P2P 서비스의 이용률이 증가되고, P2P 서비스 클라이언트가 다양해짐에 따라 P2P 서비스에 의한 네트워크 자원 소모가 증가하고 있다. 특히, P2P 서비스의 대부분은 토렌트인데, 서비스의 특성상 네트워크 관리자가 분류하기 어렵다. 토렌트 트래픽을 네트워크 관리자가 분류하지 못하면 해당 네트워크 망의 자원 대부분이 토렌트 서비스에 소모되고, 해당 네트워크 망의 목적 서비스는 원활하게 이루어지지 않는다. 따라서 본 논문은 토렌트 서비스를 완벽하게 분류하고, 클라이언트 별로 분석하기 위한 메커니즘을 제안한다. 또한 분석률 뿐만 아니라 향후 실시간 서비스에 본 메커니즘을 적용하기 위해 처리속도 향상에 대한 방법을 제안한다. 본 논문에서 제안한 시스템을 통해 28 개의 트래픽 트레이스에 대해 100% 분석률을 보인다.

Keywords: Torrent, P2P, Traffic Classification, Network Management, Analysis

1. 서론

초고속 인터넷의 보급과 인터넷 기반의 서비스가 다양해짐에 따라 네트워크 관리의 중요성이 강조되고 있다. 네트워크 사용자 측면에서는 고품질 서비스의 안정적인 제공에 대한 요구가 증가하고, 네트워크 제공자 측면에서는 망 관리 비용을 최소화하면서 다양한 고품질의 서비스를 제공하기 위한 요구가 증가하고 있다. 하지만 급증하는 네트워크 트래픽에 비해 한정적인 네트워크 자원은 네트워크의 부담을 가중시킨다. 네트워크 관리자는 망의 안전성과 신뢰성을 확보하기 위해 네트워크 장비의 대역폭을 증가시키기 위해 네트워크 장비의 확충과 성능 향상하는 방법이 있지만, 이러한 방법은 비용과 기술적인 측면에서 무리가 있다. 따라서 효과적인 네트워크 자원 활용을 위해 기존에 발생해왔던 네트워크 트래픽의 응용레벨에서 분석하고, 네트워크 소모량이 시간에 따라 높은 서비스를 파악하여 사용자의 이용패턴을 분석한다. 이러한 분석을 통한 QoS(Quality of Service)와 같은 다양한 네트워크 관리 정책의 수립하여, 네트워크 자원 활용의 질을 보다 높이기 위한 연구가 진행 되고 있다.

네트워크 관리의 목적은 네트워크 자원을 최대한 활용하여 해당 네트워크의 목적에 맞는 서비스를 원활하게 이루어질 수 있도록 하는 것이다. 이를 위해 네트워크 관리자들은 적절한 네트워크 정책을 수립하여 관리 대상 네트워크에 적용한다[3,4]. 네트워크 정책을 수립하기 위해서는 일

본 연구는 2013 년 BK21 플러스 사업 및 2012 년 정부(교육과학기술부) 의 재원으로 한국연구재단(2012R1A1A2007483)의 지원을 받아 수행된 결과임.

반적으로 발생하는 트래픽에 대해서 전수 조사 하여 트래픽 발생 응용에 대해서 분석이 필요하다. 트래픽 분석을 통해 네트워크 환경에 맞게 정책을 수립한다. 트래픽을 분석하기 위해 가장 중요한 단계는 분석 대상 트래픽의 공통된 특징을 찾아내고, 다양한 응용에서 발생하는 트래픽을 구분할 수 있어야 한다.

오늘날 네트워크 환경에서 가장 많이 발생하는 트래픽은 P2P 응용이라 해도 과언이 아니다. 현재까지 P2P 응용의 종류는 다양해지고, 점차 트래픽 발생량도 증가하고 있다. 그러나, 학내 또는 회사 네트워크와 같은 공용 네트워크에서 무분별한 P2P사용은 네트워크 활용의 주목적을 잃게 한다. 학내 네트워크의 주 목적은 인터넷 강의, 학교 포털 사이트, 도서관 도서 검색 서비스 등 학교, 학문에 주목적을 두고, 네트워크 자원을 활용해야 한다. 그러나 P2P 서비스가 다양해지고, 증가하면서 주목적의 서비스들이 오히려 침해 받고 있다. 따라서 각 네트워크 환경의 주목적 서비스들에 대한 실용도 높은 서비스 제공을 위해 P2P 트래픽의 네트워크 관리 정책이 시급하다.

P2P 서비스 중 가장 많고 다양하게 트래픽을 발생시키는 응용은 토렌트 서비스이다. 2011년 유럽에서 Up/Down Stream 발생량을 Byte기준으로 토렌트 서비스가 가장 많이 발생하였다고 발표되었다. 또한, 토렌트 응용은 Bittorrent, utorrent, Deluge등 점차 다양해지고 있기 때문에 각 응용에 대한 네트워크 자원 정책 수립이 쉽지 않다[8].

따라서 본 논문은 학내 망에서 발생하는 토렌트 응용에 대한 트래픽 수집 및 Signature를 추출한다. 추출된 Signature를 이용한 학내 망 토렌트 응용 트래픽에 대해 100% 탐지를 목표로 하고, 각 토렌트 클라이언트 별로 구분한다. 또한 토렌트 응용 트래픽 분석 결과를 바탕으로 학내 망의 대역폭을 조절을 통해 효율적인 학내 망 트래픽 관리가 가능하게 된다.

본 논문의 구성은 2장에서 관련연구에 대해 언급하고, 3장에서 토렌트 정답 트래픽 수집과정과 4장에서 수집된 트래픽을 이용한 시그니처 추출 및 분석에 대해 기술한다. 5장에서 분석 시스템의 속도향상을 위한 연구와 6장에서 분석 결과에 대해 설명하고, 마지막으로 7장에서 결론 및 향후 연구를 언급한다.

2. 관련 연구

오늘날 토렌트 서비스의 트래픽을 분석 중요성이 증가함에 따라 지속적으로 연구가 진행되고 있다. 트래픽 분석 방법들은 트래픽 분석 시 사용하는 트래픽 특징을 기준으로 포트기반 분석, 페이로드 기반 분석, 통계정보 기반 분석, 행위기반 분석 등으로 구분된다. 또한, 분석 단위 기준으로 패킷 기반 분석, 플로우 기반 분석으로 구분될 수 있다.

그러나 현재의 네트워크는 응용의 다양성, 트래픽의 암호화 등 여러 가지의 한계로 하나의 분석 방법으로 모든 트래픽을 분석하기는 어렵다. 따라서 본 논문에서는 페이로드, 헤더, 통계 정보 기반 트래픽 분석 방법을 적용한 멀티 레벨 분석 시스템을 제안한다.

포트기반 분석은 Internet Assigned Number Authority (IANA)[2]에서 지정한 포트 정보를 이용한 트래픽 분석 방법으로 포트 번호와 대응하는 서비스인 HTTP(80), telnet(23), e-mail(25,110), FTP(20,21)를 기준으로 분석한다. 따라서 적은 메모리 사용으로 매우 빠르게 분석할 수 있는 장점을 가진다. 본 논문의 목적인 토렌트 분석에서 토렌트의 시드를 찾기 위해 지원하는 Tracker정보는 고정된 IP Address 또는 포트 번호를 사용하기 때문에 충분히 적용할 수 있다. 하지만, 토렌트 와 같은 P2P 응용에서 데이터 전송이 이루어 지는 연결은 포트 번호를 사용자가 설정하거나 매 실행 시 임의의 포트 번호를 사용하기 때문에 본 논문에서 적용하기 어려운 점이 있다[5].

이러한 문제를 해결하기 위해 패킷의 페이로드 내에서 응용마다 가지는 특정 스트링(Signature)의 포함 유무를 통해 트래픽을 분석하는 페이로드 기반 분석 방법이 제안되었다[6]. 트래픽의 내용을 확인하는 작업이기 때문에 분석률과 정확도는 매우 높지만, 각 응용의 특정 스트링(Signature)을 생성하고 관리하는데 많은 시간이 소비된다. 또한, 암호화 트래픽, 높은 계산 복잡도, 패킷 단편화 등과 같은 많은 한계점을 가지고 있다. 따라서, 본 논문에서 토렌트 트래픽 분석 시스템은 페이로드 기반 트래픽 분석 방법을 마지막으로 적용시킨다.

페이로드 기반 트래픽 분석 방법의 단점을 해결하기 위해 트래픽의 내용을 보지 않고 패킷 및 윈도우 크기, 패킷 간 시간 간격 등과 같은 통계적 특징만을 이용한 통계 기반 분석 방법이 제안

되었다[7]. 이 방법론은 패킷의 헤더 정보를 통해 통계 정보를 생성하므로 기존 트래픽 분류 방법론들의 한계점들을 보완할 수 있다. 또한 페이로드 기반 트래픽 분석 방법보다 계산 복잡도가 낮기 때문에 더 신속한 결과를 얻을 수 있다. 하지만 모든 응용의 트래픽에서 특정한 특징을 나타내는 것은 매우 어려운 일이기 때문에 분석 응용 다양성에 대한 한계점이 존재한다. 따라서 본 논문에서는 특징이 나타나는 토렌트 트래픽에 대해서 신속하게 분석하고, 나머지 트래픽에 대해서 페이로드 기반 트래픽 분석 방법을 적용한다.

기존 토렌트 트래픽 분류 방법에 대한 연구로는 토렌트 클라이언트 별로 트래픽을 분류하는 연구결과가 있다[8]. 본 방법은 토렌트 클라이언트 별 시그니처를 이용하여 토렌트 트래픽을 분류하고, 전송 프로토콜을 이용한 트래픽 분류 분석률을 보인다. 하지만 각 클라이언트 별로 모든 토렌트 트래픽을 분류하지 못한다.

본 논문은 헤더, 통계, 페이로드 시그니처를 기반 분석방법으로 토렌트 트래픽을 분류하고, 분석률 100%를 목표로 한다. 정확도와 신속성을 검증하기 위해 처리속도가 빠른 헤더기반 분석방법과 통계 기반 분석방법으로 분석을 실시하고, 나머지 트래픽에 대해 페이로드 기반 분석 방법으로 분석한다. 또한 클라이언트 별 토렌트 트래픽이 다르기 때문에 다양한 클라이언트로부터 수집하여 실험한다. 본 논문에서 제안하는 방법론은 모든 토렌트 클라이언트에서 발생하는 트래픽을 100% 분석이 가능하다.

3. 트래픽 수집

네트워크 트래픽의 응용 레벨 분석에 있어서 가장 중요한 단계는 트래픽 수집단계이다. 특정 응용을 분석할 때 수집된 트래픽에 noise가 많다면, 정확한 시그니처를 수집하지 못하고 정확한 분석결과를 기대하기 힘들다. 따라서 본 논문에서는 정확한 트래픽 수집을 위해 TMA[1]를 이용한다.

토렌트 트래픽 수집 자는 TMA에서 트래픽이 수집될 수 있도록 실행 시킨 후, 토렌트 응용을 실행시키며 트래픽을 수집한다. 수집된 토렌트 트래픽은 CAP 형태의 파일에서 PCAP형태의 파일로 변환하고, 변환된 PCAP 형태의 파일에서 순수 토렌트 트래픽을 남기기 위한 비정상 트래픽을 제거한다. 본 논문에서는 비정상 트래픽을 다음과 같이 기준 한다. 첫 번째로 학내 망에서 학내 망으로 전송되는 패킷에 대해서는 비정상 트래픽으로 판단하여 제거한다. 두 번째는 학내 망 외에서 학내 망 외로 전송되는 패킷에 대해서 비정상 트래픽으로 판단하여 제거한다. 세 번째는 TCP 또는 UDP 트래픽만을 대상으로 하기 때문에 그 외의 L4 Protocol에 대해서 비정상 트래픽으로 판단하여 제거한다.

다음과 같은 비정상 트래픽 제거 과정을 마치면 L4 Protocol이 TCP 또는 UDP을 사용하는 정상적인 트래픽만이 존재하게 된다. 정상 트래픽이 생성된 후, 마지막 단계로 순수한 토렌트 트래픽만을 분석하기 위해 TMA를 사용한다. TMA는 해당 호스트에서 실행중인 프로세스들이 사용하는 소켓정보를 주기적으로 수집하여 지정된 서버로 제공하는 역할을 한다. TMA가 제공하는 정보는 Process name, IP address(local, remote), Port number(local, remote), State(start, continue, end, server), Protocol, Path 이다. 다음과 같은 정보가 각 종단 호스트에서 수집되고 수집된 정보와 위에서 생성한 정상 트래픽을 비교한다. 토렌트 트래픽만이 선택되어 토렌트 트래픽으로만 이루어진 정답 트래픽이 생성된다.

4. 토렌트 분석 시스템

토렌트 트래픽 분석 시스템은 다음과 같이 4 단계로 구성된다. 먼저 PCAP형태로 수집된 정답 트래픽을 Binary 형태의 패킷 형태로 변환한다. 변환된 패킷은 플로우 형태로 생성된다. 플로우는 5-tuple(Source IP Address, Source Port Number, Destination IP Address, Destination Port Number, L4 Protocol Number)이 같은 패킷의 집합을 말한다.

그림 은 토렌트 분석 시스템은 전체 과정을 나타낸다. 토렌트 분석 시스템은 다음과 같이 트래픽 분석 부와 시그니처 추출 부로 나누어 진다. 트래픽 분석 부는 재정렬된 플로우를 입력으로

받고, 토렌트 분석 결과와 분석되지 않은 트래픽을 출력한다. 분석되지 않은 트래픽은 Cor-relation 과정을 거쳐서 분석된 트래픽에 한해 분석 결과에 추가되고, 다시 분석되지 않은 트래픽은 시그니처 추출 부로 들어가 잠재적인 시그니처를 추출하게 된다. 추출된 시그니처는 다시 분석부의 시그니처로 추가된다.

4.1 토렌트 트래픽 전처리

토렌트 분석 시스템은 먼저 토렌트 트래픽 전 처리과정을 거친다. 수집된 Packet을 플로우로 생성하고, 생성된 플로우를 분석하기 위해서는 패킷 재정렬을 해야 한다. 재정렬을 하는 이유는 플로우를 분류 할 때, 수집된 플로우내의 패킷의 크기와 전송방향, 전송순서, 그리고 수집된 시간 등의 특징을 사용하여 분류하는데, 수집된 통계 정보를 그대로 사용하여 트래픽을 분류하는 것에는 한계가 존재한다. TCP 세션 에서 발생하는 패킷 역전 문제, 패킷 재전송에 의한 패킷 중복 문제가 발생하기 때문이다[9]. 따라서 패킷들을 재정렬 한 뒤에, 다양한 시그니처 정보를 이용하여 트래픽을 분류할 수 있다. 이러한 전처리 과정을 마친 후 분석 시스템은 재정렬된 플로우 가 입력되고, 토렌트 응용 트래픽에 대한 분석 결과를 출력하게 된다.

응용 트래픽 분류를 위한 시그니처에는 여러 가지 종류가 있다. 그 중 Payload 시그니처는 일반적으로 다른 시그니처에 비해 보다 정확하고 높은 분석률을 보여 준다. 하지만 Payload 시그니처는 시그니처의 개수가 증가 할수록 매칭하는 과정에서 부하가 커져 분석시간이 증가하는 단점 또한 포함하고 있다. 따라서 본 시스템에서는 헤더 시그니처와 통계 기반 시그니처로 먼저 트래픽을 분류하고, 나머지 트래픽을 페이로드 기반 시그니처로 분석함으로써 분석 시스템의 부하를 감소한다.

4.2 토렌트 트래픽 분석 시스템

그림 1은 토렌트 분석 시스템의 개요이다. 3장에서 정의한 방법으로 생성된 토렌트 정답지 트래픽에 Traffic Identifier을 통해 헤더 기반, 통계 기반, 페이로드 기반 시그니처 순으로 분석시스템이 실행된다. Traffic Identifier에서 트래픽 분석이 완료되면 Identified 플로우 checker 에서 토렌트 트래픽 분석율을 평가한다.

분석율을 평가 한 뒤, 미 분류 플로우들에 대한 처리 방법을 정하게 된다. 처리 방법은 크게 두 가지가 있다. 첫 번째로 미 분류 플로우의 페이로드 내용을 확인 후 토렌트 응용 고유의 시그니처를 추출해 시그니처 리스트에 추가 하는 방법이 있다. 이러한 방법을 시그니처 추출이라고 정의한다. 두 번째 방법으로는 Cor-relation 방법이 있다. Cor-relation 방법은 시그니처 추출이 불가능한 미 분류 플로우를 분석된 플로우와의 상관관계를 통해 분석하는 방법이다.

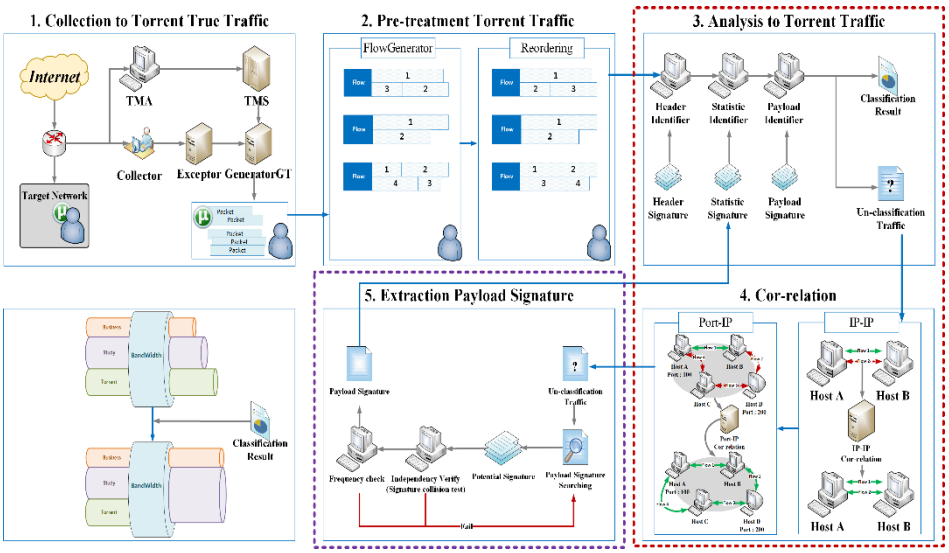

그림 1. 토렌트 트래픽 분류 및 시그니처 추출 과정

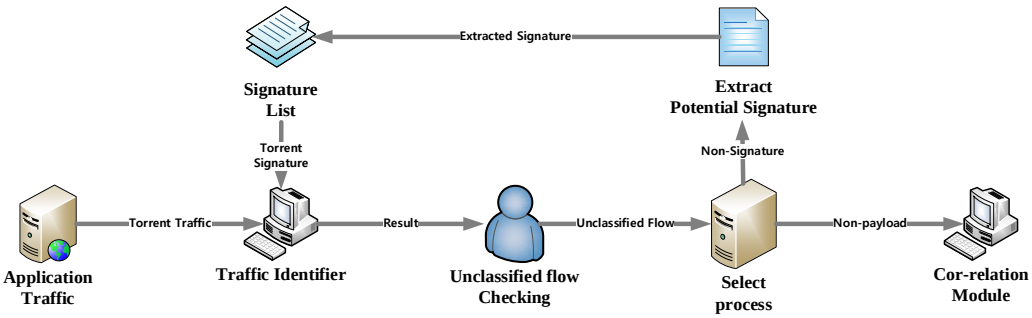

그림2. 미 분류 Traffic 처리 방법

4.3. Cor-relation 분석 방법

Correlation 분석 방법은 플로우간 IP 혹은 Port의 상관 관계를 통해 플로우를 분석하는 방법이다. Correlation 분석을 하는 이유는 응용 트래픽 분석에 있어 시그니처 만으로는 높은 분석률을 얻어내는 데에 한계가 있기 때문이다. 또한 실제로 시그니처로 분류 할 수 없는 플로우에 대해서 높은 트래픽 분석률을 보였다. Correlation은 크게 IP-IP correlation과 Port-IP correlation으로 나뉜다. IP-IP correlation분석된 플로우의 start host와 end host의 IP주소를 바탕으로, 분석되지 않은 플로우 중 동일한 2-tuple set을 갖는 플로우가 존재하는지 검사한다. 만약 동일한 2-tuple set을 갖는 플로우가 존재한다면 해당 플로우는 IP-IP correlation을 통해 분석된 플로우로 가정한다.

그림3에서 1번 플로우는 IP-IP correlation에 의해 분석 된 플로우이다. 1번플로우는 2번와 플로우가 동일한 2-tuple set을 갖기 때문에 1번 플로우와 상관관계를 통해서 2번 플로우 또한 분석 가능하게 된다.

Port-IP correlation은 특정 응용 트래픽을 발생시키는 서버와 호스트간의Port-IP상관 관계를 통해 플로우를 분석하는 방법이다. 그림을 통해 쉽게 확인이 가능하다. 그림의 2번과 3번 그리고 4번 플로우는 미 분류 플로우이다. 그러나 Port-IP correlation을 적용하게 되면, 1번 플로우와의 상관관계에 의해 분석 가능하게 된다. 2,3,4번 플로우 모두 동일한 서버IP와 Port정보를 갖고 있다. 이 때, 1번 플로우는 응용 시그니처를 통해 분석되었다고 가정하면 2번과 3번 플로우 역시 1번과 동일한 응용의 플로우라고 판단 할 수 있다.

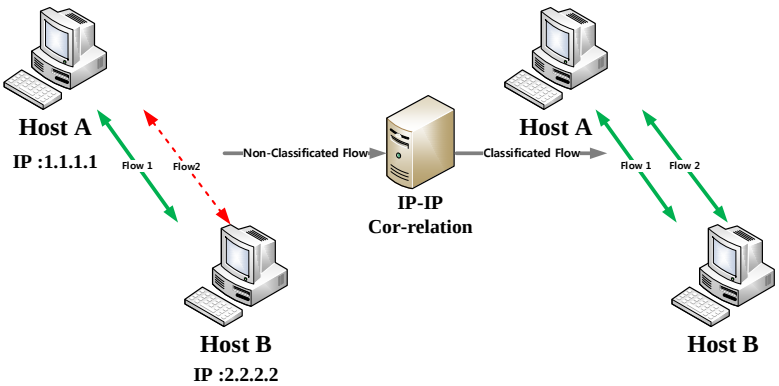

그림3. IP-IP Cor-relation

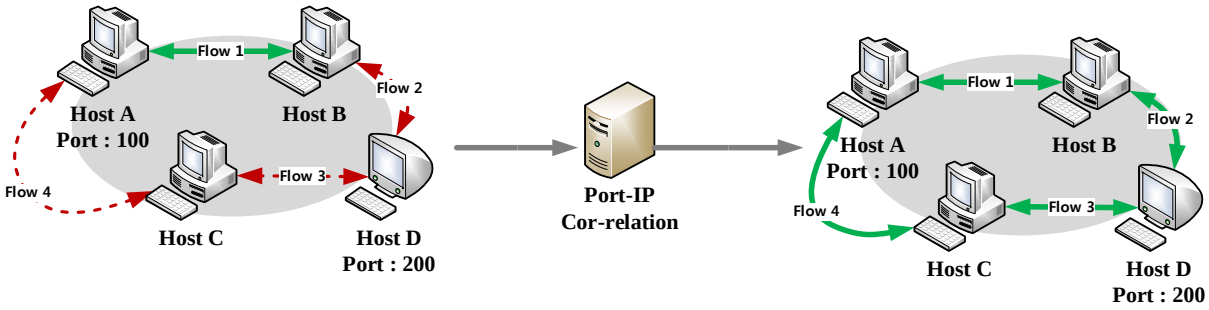

그림4. Port-IP Cor-relation

Correlation관계를 통하면 기존의 시그니처만 으로는 분석 할 수 없던 다수의 플로우들 또한 분석 가능하게 된다. Correlation 알고리즘은 다음 그림과 같다. Correlation관계를 통하면 기존의 시그니처만 으로는 분석 할 수 없던 다수의 플로우들 또한 분석 가능하게 된다. Correlation 알고리즘은 다음 그림과 같다. IP-IP correlation은 시작점 IP와 도착지 IP를 비교 후 같은 tuple set을 가질 경우 Correlation 으로 분석 된 것으로 정의한다. Port-IP correlation 또한 마찬가지로 방법으로 서버와 호스트간의 IP와 Port를 비교해준다.

<IP-IP cor-relation>

Srcaddr : source IP address
Dstaddr : destination IP address
Set : Apply Flow classification Code

1: **if**((F_i.Srcaddr==F_j.Srcaddr) && (F_i.Dstaddr==F_j.Dstaddr))
2: **Set** F_i as cor-related Flow
3: **else if**((F_i.Srcaddr==F_j.Dstaddr)&&(F_i.Dstaddr==F_j.Srcaddr))
4: **Set** F_j as cor-related Flow

알고리즘1. IP-IP cor-relation

<Port-IP cor-relation>

Srcaddr : source IP address
Dstaddr : destination IP address
Set : Apply Flow classification Code
SrcPort : source Port number
DstPort : destination Port number

1: **if**((Fi.Dstport==Fj.Srcport)&&(Fi.Dstaddir= Fj.Srcaddr))
2: **Set** Fi as cor-related Flow
3: **else if**((Fi.Srcport==Fj.Srcport)&&(Fi.Srcaddir= Fj.Srcaddr))
4: **Set** Fi as cor-related Flow
5: **else if**((Fi.Srcport==Fj.Dstport)&&(Fi.Srcaddir= Fj.Dstaddr))
6: **Set** Fi as cor-related Flow
7: **else if**((Fi.Dstport==Fj.Dstport)&&(Fi.Dstaddir= Fj.Dstaddr))
8: **Set** Fi as cor-related Flow

알고리즘2. Port-IP cor-relation

4.4 페이로드 시그니처 추출

모아진 토렌트 트래픽 트레이스를 완벽하게 분석하기 위해서는 토렌트 응용을 확실하게 분류할 수 있는 시그니처가 필요하다. 페이로드 시그니처는 이러한 관점에서 가장 효율적인 시그니처이기 때문에 페이로드 시그니처를 추출하는 과정이 필요하다. 페이로드 시그니처를 추출하기 위해서는 그림과 같은 선행 분석과정을 통해 선별된 미 분류 플로우에 대한 분석 과정이 필요하다.

플로우의 페이로드 컨텐츠에서 시그니처를 선별 및 추출해 낼 때 가장 중요한 것은 해당 시그니처 스트링의 개별성과 빈도수이다. 페이로드 내용 중 시그니처로 추출 가능 하다고 판단되는 스트링을 potential signature 라고 정의 했을 때, potential signature는 첫째로 개별적 이여야 한다. 개별적의 의미는 해당 시그니처가 분석하고자 하는 응용(토렌트)의 플로우 만을 분석하는 시그니처여야 한다는 것이다. 만약 potential signature가 다수의 응용에 매칭되는 시그니처라면 분석과정에서 응용간 충돌이 발생하게 되고 결국 정확한 응용 분류를 할 수 없게 된다. 따라서 potential siganture는 개별적으로 한 가지 응용만 분석 할 수 있어야 한다. 두번째로 potental signature는 다양

한 플로우에서 빈번하게 발생해야 한다. 만약 potental signature가 특정 플로우에서 특별한 경우에만 발생하는 내용이라면, 해당 시그니처는 특정 응용을 분석 해내는 일반적인 시그니처라고 판단할 수 없다. 따라서 potential signature는 다양한 플로우에서 빈번하게 발생하는 내용이어야 한다.

시그니처 추출은 다음과 같은 단계를 통해 추출된다. 처음 플로우의 패킷 페이로드 부분에서 활용 가능한 Potential signature 스트링을 추출한다. 추출된 시그니처 트래픽 분석 프로그램에 적용시켜 다른 시그니처와 충돌이 일어나는지를 검사하면서 동시에 Potential signature의 개별성과 고유성을 평가한다. 조건이 충족되면 분석 결과를 바탕으로 해당 시그니처의 발생 빈도수를 파악해본다. 매칭 빈도수 또한 만족 된다면, potential signature 에서 Payload signature로 확정시키고 시그니처 리스트에 추가한다. 시그니처를 추출 할 수 없는 미 분류 플로우에 대해 서는 플로우 간 Correlation 방법을 이용하여 분석 할 수 있다.

5. 분석 속도 향상

일반적인 응용 분류 시스템에서는 플로우에 대해 응용 시그니처가 매칭 되면 매칭을 종료하는 부분 매칭(Partial-matching)방식을 사용한다. 시그니처가 매칭 될 경우 페이로드의 처음부터 매칭 지점까지 탐색을 한다. 만약 시그니처가 매칭 되지 않는 경우 페이로드의 전체를 탐색하여 불필요한 탐색을 수행하고, 분석시간이 증가하게 된다.

본 절에서는 불필요한 탐색 범위를 줄이기 위해 시그니처가 페이로드에 매칭되는 패킷의 순서, 전송방향 그리고 범위를 활용하여 시그니처를 매칭 유형별로 분류하는 방법에 대해 기술한다.

5.1. Offset value

본 절에서는 시그니처를 매칭 유형 별로 6가지 유형으로 나누기 위한 알고리즘의 파라미터 값으로 사용되는 시그니처의 매칭 위치 정보(Matching Offset value)에 대해 설명한다.

첫 번째 변수는 Packet Offset이다. Packet Offset은 시그니처가 매칭 되는 패킷의 순서가 고정적(Fixed)일 경우 해당 패킷의 순서 값을 의미한다. 두 번째 변수는 Direction이다. Direction은 시그니처가 매칭 되는 패킷의 전송방향이 서버를 향하는지 클라이언트를 향하는지를 나타내는 값이다. udp의 경우 수집된 첫 번째 패킷의 전송방향이 서버를 향한다고 판단하여 이후 패킷의 전송 방향을 결정한다. forward일 경우 서버를 향하고, backward일 경우 클라이언트를 향하는 방향이라고 정의한다.

세 번째 변수는 First Offset이다. First Offset은 시그니처와 페이로드의 매칭이 시작되는 위치 값이 Fixed일 경우 해당 위치 값을 나타낸다. 네 번째 변수는 First Range이다. First Range는 시그니처와 페이로드가 매칭이 시작되는 위치 값이 not Fixed일 경우 매칭이 시작되는 여러 위치 값들 중 최소값을 나타낸다. 다섯 번째 변수는 Last Offset이다. Last Offset은 시그니처가 매칭 된 마지막 위치 값이다. 마지막 변수는 Depth이다. Depth는 시그니처가 매칭 되는 범위응 나타내는 정보이다. Last Offset에서 First Offset 또는 First Range를 뺀 값이 된다. 변수 Packet Offset, Direction, First Offset는 not fixed 경우는 사용하지 않는다. First Range는 First Offset이 not fixed인 경우에만 사용한다.

5.2. 분석 속도 향상

표1은 시그니처의 매칭 유형별 분류를 적용시켜 트래픽을 분석하여 기존의 트래픽 분석 방법과 비교한 결과이다. 전체적으로 시그니처의 매칭 유형별 분류를 적용시킨 트래픽 분석 방법은 기존의 트래픽 분석 방법과 비교 했을 때, 분석률은 동일했으며, 분석 시간 측면에서는 평균적으로 약 25%의 분석 시간 단축을 보였고, 분석 시간 단축의 최댓값은 41%로 측정되었다. 결과적으로 시그니처의 매칭 유형을 분류하여 트래픽 분석에 적용한 것이 적용하지 않은 것과 비교 했을 때 탐색 범위가 축소되었음에도 분석률이 동일한 것은 불필요한 탐색 범위만을 축소시켰다고 분석할 수 있다. 또한, 실험 트레이스마다 분석시간의 축소 폭이 차이나는 부분은 해 당 트래픽에 매칭되는 시그니처와 시그니처의 매칭 유형이 각각 달라 축소된 탐색 범위가 달랐기 때문이다.

결과적으로 실험을 통해 본 절에서 제안하는 시그니처의 매칭 유형에 따른 분류가 효과적이라는 것을 증명하였다. 표1은 본 절에서 제안하는 시그니처 매칭 유형에 따른 분류 시 결과이다.

표1. 매칭 유형에 따른 분류 후 분석 속도 향상 결과

Trace ID	시그니처 매칭 유형 미 분류		시그니처 매칭 유형 분류	
	Completeness	**Time**	**Completeness**	**Time**
006_UT_FP02	94.64	0.59	94.64	0.49
006_UT_FP03	98.57	0.23	98.57	0.28
006_UT_FP04	99.52	0.18	99.50	0.14
006_UT_FP05	99.32	0.17	99.32	0.12
006_BT_FP02	89.77	0.31	89.77	0.24
006_BT_FP03	96.67	0.19	96.67	0.15
006_BT_FP04	98.36	0.21	98.36	0.17
006_BT_FP05	99.01	0.14	99.01	0.11
008_UT_FP06	85.61	0.02	85.61	0.02
008_UT_FP07	99.28	0.25	99.28	0.19
008_BT_FP08	85.11	0.12	85.11	0.10
008_BT_FP09	96.91	0.14	96.91	0.10
008_UT_FP10	84.23	0.08	84.23	0.07
008_UT_FP11	94.24	0.03	94.24	0.02
008_UT_FP13	85.96	0.22	85.96	0.17

6. 실험 및 결과

분석 실험은 본 논문의 목적인 완벽한 토렌트 응용 분석을 검증하기 위한 것을 목표로 한다. 실험 방법은 4장에서 정의한 토렌트 분석 시스템을 이용해 최종적으로 그림과 같이 진행한다. 실험을 위해 3장에서 정의한 방법을 통해 토렌트 트래픽을 수집하고 Traffic identifier을 통해 페이로드 시그니처 리스트와 매칭시킨다. 시그니처 매칭을 완료하고 남은 미분석 플로우에 대해서는 다시 Correlation 과정을 적용 시킨다. Correlation과정까지 끝내면 토렌트 응용 분석률을 평가한다. 트래픽이 완벽히 분석 되었다면 분석 과정을 끝내고 분석 결과가 불완전 할 경우 4-1절에서 정의한 시그니처 추출과정을 거쳐 시그니처를 추가 한 뒤 분석 과정을 다시 반복한다.

6.1. 실험 트래픽 정보

토렌트 응용 트래픽을 완벽히 분석하는 것을 목적으로 하는 본 논문에서는 실험을 통해 정확하고 확실한 분석을 위해 다양한 토렌트 트래픽을 수집했다. 트래픽은 모두 28개의 파일로 종료는 모두 영상과 mp3파일이다. 28개의 파일 모두 실제 네트워크 상에 유통중인 토렌트 파일을 받아서 전 처리과정을 거친 후 그림의 분석 시스템에 적용시켰다. 또한 다양한 경우의 트래픽이 존재하기 때문에 토렌트 클라이언트, 옵션, 다양한 호스트에서 트래픽을 수집하였다. 토렌트 트래픽 파일 정보는 표2와 같다.

6.2. 시그니처 정보

표1의 모든 토렌트 응용 트래픽을 분석 했을 때 추출된 페이로드 시그니처는 총 32개로 표3과 같다. 위의 32개의 시그니처는 모두 4-3절에서 정의한 시그니처 추출방법으로 추출된 페이로드 시그니처로 각 시그니처는 토렌트 응용 트래픽의 종류에 따라 다양하게 나타난다.

표2. 토렌트 트래픽 정보

Trace ID	Size (MB)	Duration (min)	Flow	Packet ($x10^4$)	Byte (MB)
UT_01	119	24	10,874	210	1,715
UT_02	42	6	3,988	101	805
UT_03	355	7	2,996	150	1,277
UT_04	46	9	2,961	114	93
BT_01	119	18	4,946	199	1,705
BT_02	42	8	3,329	94	798
BT_03	355	10	3,603	150	1,281
BT_04	46	5	2,619	13	1,157
UT_05	20	1	278	5	33
UT_06	961	21	2,505	106	1,115
BT_05	899	5	1,760	53	427
BT_06	1,490	3	2,262	62	652
UT_07	1,520	1	1,363	104	914
UT_08	25	1	330	3	27
UT_09	954	5	3,204	100	810
VZ_01	119	6	806	125	880
FD_01	890	3	758	90	856
QB_01	1,020	18	2,632	105	861
UT_10	788	9	3,198	82	744
UT_11	788	7	2,599	92	801
UT_12	788	9	3,124	79	705
UT_13	788	3	126	88	799
UT_14	788	3	131	85	829
UT_15	788	3	127	85	828
UT_16	788	3	131	73	703
UT_17	788	4	109	68	665
UT_18	788	4	102	67	649
UT_19	788	5	106	71	682

표3. 토렌트 Payload Signature 정보

Sig ID	Payload Sig	Protocol	Port
1	.*Bittorrent protocol.*	TCP	N/A
2	.*Bittorrent protocol.*	UDP	N/A
3	.*d1:ad2:id20.*	TCP	N/A
4	^d1:ad2:id20.*	UDP	N/A
5	.*d1:rd2:id20.*	TCP	N/A
6	^d1:rd2:id20.*	UDP	N/A
7	.*find_node.*UT.*	UDP	N/A
8	.*info_hash.*get_peer.*	UDP	N/A
9	.*5:peers.*	TCP	N/A
10	.*User-Agent:.*utorrent.*	TCP	N/A
11	.*Host: com-utorrent.*	TCP	N/A
12	.*User-Agent: BTWebClient.*	TCP	N/A
13	.*//announce.info_hash=.*peer_id=.*	TCP	N/A
14	.*//scrape.info_hash=.*	TCP	N/A
15	.*d5:added.*	TCP	N/A
16	.*d5:added.*	UDP	N/A
17	.*Host:.*utorrent\.com*.“	TCP	N/A
18	.*Host:.*bittorrent\.com*.“	TCP	N/A
19	.*bittorent.com.*	UDP	N/A
20	.*tracker.*	UDP	N/A
21	.*BT.*announce.*	UDP	N/A
22	.*UT.*announce.*	UDP	N/A
23	.*d2:ip6:.*1:rd2:id20.*	UDP	N/A
24	^d2:ip6:.*1:rd2:id20.*	UDP	N/A
25	^.....BT.....*	UDP	N/A
26	^.....UT.....*	UDP	N/A
27	.*\x00\x00\x04\x17\x27\x10\x19\x80\x00\x00\x00.*	UDP	N/A
28	^BT-SEARCH.*	UDP	N/A
29	.*Bittorrent.*	UDP	1900
30	.*utorrent.*	UDP	1900
31	.*Host:.*deluge-torrent\.org	TCP	N/A
32	.*User-Agent:Deluge.*	TCP	N/A

표 4. 토렌트 분류 결과

Trace ID	Total Flow	Identified (Signature)		Identified (Cor-relation)		Completeness (%)
		Flow	%	Flow	%	
UT_01	10,874	10,483	96	391	4	100
UT_02	3,988	3,957	99	31	1	100
UT_03	2,996	2,986	99	10	1	100
UT_04	2,961	2,944	99	17	1	100
BT_01	4,946	4,675	94	271	6	100
BT_02	3,329	3,306	99	23	1	100
BT_03	3,603	3,583	99	20	1	100
BT_04	2,619	2,607	99	12	1	100
UT_05	278	258	92	20	8	100
UT_06	2,505	2,492	99	13	1	100
BT_05	1,760	1,676	95	84	5	100
BT_06	2,262	2,222	98	40	2	100
UT_07	1,363	1,237	90	126	10	100
UT_08	330	320	96	10	4	100
UT_09	3,204	2,760	86	444	14	100
VZ_01	806	804	99	2	1	100
FD_01	758	695	91	63	9	100
QB_01	2,632	2,619	99	13	1	100
UT_10	3,198	3,193	99	5	1	100
UT_11	2,599	2,593	99	6	1	100
UT_12	3,124	3,115	99	9	1	100
UT_13	126	122	96	4	4	100
UT_14	131	128	97	3	3	100
UT_15	127	124	97	3	3	100
UT_16	131	126	96	5	4	100
UT_17	109	104	95	5	5	100
UT_18	102	95	93	7	7	100
UT_19	106	103	97	3	3	100

6.3. 분석 결과

토렌트 트래픽 분석 결과는 표4과 같다. 28개의 토렌트 트래픽 트레이스에 대해 5-2절의 시그니처와 4-3절에서 정의한 Cor-relation분석 방법을 적용 시켰을 때 모두 100%의 응용 분석률을 보였다. 표4의 분석결과를 보면 대부분의 트레이스에서 페이로드 시그니처로 90%이상의 분석률을 보였고 시그니처로 분석되지 않은 부분은 Cor-relation을 통해 분석되었다.

결과적으로 페이로드 시그니처와 Cor-relation방법을 통해 완벽한 토렌트 응용 트래픽 분석이 가능하다는 것을 확인 할 수 있다. 페이로드 시그니처와 Cor-relation이라는 방법을 통해 토랜트 응용 트래픽을 100% 분석 할 수 있다는 점에서 의미가 있다.

토렌트 응용 분석의 최종 목표는 분석 결과를 바탕으로 학내 망 대역폭 제어를 통한 학내 네트워크의 효율적 관리이다. 따라서 향후 연구에서는 추출된 시그니처와 Cor-relation분석 방법을 실시간 응용 분석 시스템에 적용시키는 것을 목표로 할 계획이다. 다양한 트래픽에서 토렌트 응용만을 정확히 분석 하고 시그니처 간 충돌이 일어나지 않는지 판단하고 응용 분석을 얼마나 빠르게 할 수 있는지는 향후 연구 목표 달성을 위해 연구해야 할 중요한 논점이다.

7. 결론 및 향후 연구

본 논문에서 제안하는 시스템은 토렌트 트래픽을 100% 분석을 목표로 하였다. 본 시스템은 헤더, 통계, 페이로드 기반 트래픽 분석 시스템 순으로 분석하고, 분석되지 않은 토렌트 트래픽에 대해 Cor-relation 방법과 페이로드 시그니처 추출 단계를 거침으로써 모든 토렌트 트래픽을 분류한다. Cor-relation 단계를 거쳐 분석된 트래픽과 분석되지 않은 트래픽의 상관관계(IP-IP, Port-IP)를 분석하여 분석되지 않은 트래픽을 분석한다. 두 번째 단계는 잠재적 Payload 시그니처를 추출하여 다시 분석단계를 거쳐 분석한다. 또한 현재 상용되고 있는 토렌트 클라이언트의 종류가 다양하기 때문에 토렌트 트래픽 수집 또한 다양한 클라이언트에서 발생하는 트래픽을 수집하였고, 다양한 환경에서 수집하였다. 6장에서 실험을 통해 본 논문의 목표인 모든 클라이언트, 환경에서 토렌트 응용 트래픽 100% 분석함으로써 증명한다.

향후 계획으로는 토렌트 응용 트래픽 뿐만 아닌 다른 P2P 서비스 응용을 분석하여 많은 트래픽이 발생하는 응용에 대한 관리를 해야 한다.

8. 참고 문헌

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

[2]IANA port number list. Available: http://www.iana.org/assignments/service-names-portnumbers/service-names-port-numbers.xml

[3]H. Kim, K. C. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: myths, caveats, and the best practices," in Proceedings of the 2008 ACM CoNEXT conference, 2008, p. 11.

[4]A. Dainotti, A. Pescape, and K. C. Claffy, "Issues and Future Directions in Traffic Classification," Ieee Network, vol. 26, pp. 35-40, Jan-Feb 2012.

[5]A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," in Passive and Active Network Measurement, ed: Springer, 2005, pp. 41-54.

[6]Y. Wang, Y. Xiang, W. L. Zhou, and S. Z. Yu, "Generating regular expression signatures for network traffic classification in trusted network management," Journal of Network and Computer Applications, vol. 35, pp. 992-1000, May 2012.

[7]N. F. Huang, G. Y. Jai, H. C. Chao, Y. J. Tzang, and H.Y. Chang, "Application traffic classification at the early stage by characterizing application rounds," Information Sciences,

vol. 232, pp. 130-142, May 2013
[8]권재범, 유종현, 심규석, 김명섭, "토렌트 프로토콜의 응용 별 분석", 통신망운용관리 학술대회 (KNOM 2014), 충남대학교, 대전, May. 15-16, 2014, pp.120-121.
[9]S.K Lee, H.M Ahn, and M.S Kim, "Packet Out-of-order and Retransmission in Statistics-based Traffic Analysis," Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2014, Hsinchu, Taiwan, Sep. 17-19, 2014.
[10]유종현, 권재범, 이수강, 김명섭, "토렌트 응용 설정에 따른 트래픽 전송 방식에 대한 연구", 통신망 운용관리 학술대회 (KNOM 2014), 충남대학교, 대전, May. 15-16, 2014, pp.118-119.
[11]J.S Park, S.H Yoon, M.S Kim, "Performance Improvement of the Payload Signature based Traffic Classification System using Application Traffic Temporal Locality," Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2013, Hiroshima, Japan, Sep. 25-27, 2013.
[12]S.H Yoon, M.S Kim, "An Efficient Method to Maintain the Header Signature for Internet Traffic Identification," Proc. of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2013, Hiroshima, Japan, Sep. 25-27, 2013.

심 규 석
2014년 : 고려대학교 컴퓨터 정보학과 졸업
2014년~현재: 고려대학교 컴퓨터정보학과 석사과정
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

윤 성 호
2009년 : 고려대학교 컴퓨터 정보학과 졸업
2011년 : 고려대학교 컴퓨터 정보학과 석사
2011년~현재 : 고려대학교 컴퓨터정보학과 박사과정
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

정 우 석
2009년 ~ 현재: 고려대학교 컴퓨터 정보학과 졸업
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

이 성 호
2010년 ~ 현재 : 고려대학교 컴퓨터 정보학과
<관심분야> 네트워크 관리 및 보안

김 명 섭
1998년 : 포항공과대학교 전자 계산학과 졸업
2000년 : 포항공과대학교 컴퓨터 공학과 석사
2004년 : 포항공과대학교 컴퓨터 공학과 박사
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