네트워크 혼잡도에 강건한 통계 정보 기반 트래픽 분석 방법

안현민, 함재현, 정우석, 김명섭

고려대학 컴퓨터정보학과

{ queen26, jhham, hary5832, tmskim}korea.ac.kr

Robust Traffic Classification based on Statistical Information in Network congestion

요 약

응용 레벨 트래픽 분류는 안정적인 네트워크 운영과 자원 관리를 위해서 필수적으로 요구된다. 최근 기존의 포트 정보 기반, 페이로드 정보 기반의 분석 방법들의 단점을 보완한 통계 정보 기반 트래픽 분류 방법이 많이 연구되고 있지만 이러한 방법들은 실제 네트워크에서 트래픽 분류에 적용하기 힘들다. 네트워크가 혼잡할 경우 통계 정보들은 얼마든지 변형될 수 있기 때문이다. 본 논문에서는 네트워크 혼잡 상황에서도 강건한 통계 정보 기반 트래픽 분석 방법을 제안한다. 트래픽을 생성된 시퀀스에 맞춰 재 정렬 함으로써 네트워크가 혼잡하여 수집 시점에서 통계 정보가 변형되더라도 응용에서 의도한 트래픽 순서에 맞춰 정보를 얻음으로써 통계 정보의 일관성을 유지하고, 믿을 수 있는 분석 결과를 낼 수 있다.

1. 서론

네트워크의 고속화에 힘입어 많은 인터넷 기반 응용 프로그램들이 개발되고 서비스되고 있다. 이에 따라 네트워크의 효율적 운용과 관리를 위한 트래픽의 응용 계층 분석은 종량제 과금, CRM , S LA, 보안 분석 등의 여러 정책을 위해 필수적이다. 이를 위해서는 다양한 종류의 응용 레벨 트래픽을 정확하게 분류할 수 있는 방법과 고속 링크에서 발생하는 대용량의 트래픽을 실시간으로 처리하는 방법이 요구된다.

응용레벨 트래픽 분류에 대한 기존 연구들은 포트 정보나 페이로드 정보에 기반한 방법들을 제안하였다[1]. 이러한 방법들은 매우 효과적이고 정확하였다. 하지만 수많은 인터넷 기반의 응용들이 등장하고, 트래픽 분석을 피하는 여러 방법들이 개발되고 이용됨에 따라 기존의 방법들을 이용한 트래픽 분류는 힘들어졌다. 포트 정보 기반의 분석 방법은 다이나믹 포트를 사용하는 여러 응용들에 대처할 수 없으며, 실제 네트워크에서 가장 많이 사용되는 페이로드 기반의 분석 방법은 분석 속도가 매우 느리고 분석에 필요한 정보를 수작업을 통해 추출해야 하는 단점을 가지고 있으며, 암호화 된 트래픽을 분석하지 못한다.

최근 몇 년 간 암호화 된 트래픽 분류에 강하며 빠르고 정확한 분류 성능을 가진 플로우 통계 정보를 이용한 트래픽 분류 방법이[2-6] 많이 연구되었다. 하지만 이들은 실시간 트래픽 분류가 어려우며, 개별 응용 단위로 적용되는 여러 네트워크 관리 및 운영 정책에 적용하기 어려운 분석 결과를 낸다. 무엇보다도 네트워크가 혼잡할 시 TCP 세션에서는 패킷 재전송과 Out-of-order, 그리고 Cross-order 문제, UDP 세션에서는 Out-of-order 와 Cross-order 문제가 발생할 수 있어 트래픽의 통계 정보가 수시로 변형될 수 있는데 해당 연구들에서는 이에 대한 처리가 없으므로 트래픽 특징이 일관적이지 못해 실제 네트워크에서 트래픽 분류에 적용하기 힘들다.

본 논문에서는 기존 통계 정보 기반 트래픽 분석 방법의 단점들을 극복한 새로운 분석 방법을 제안한다. 제안하는 방법은 통계 정보를 수집하는 범위를 플로우의 처음 N 개 패킷으로 한정하고, 속성 추출 시 계산을 하지 않으며 트래픽 분석 시 필요한 계산 또한 선형 시간복잡도를 가지므로 실제 네트워크에서 실시간 분류에 적용할 수 있다. 트래픽 분석 단위를 응용 프로토콜이 아닌 개별 응용 프로그램으로 삼아 자세한 분석 결과를 냄으로써 개별 응용 단위로 적용되는 여러 네트워크 관리 및 운영 정책에 적용하기에도 적절하다. 마지막으로, 제안하는 방법은 패킷의 네트워크 계층 IP 헤더의 Identification, flag, fragmentation o ffset 필드와 전송 계층 프로토콜인 TCP 헤더의 Sequence 필드를 이용하여 TCP, UDP 에서 발생할 수 있는 여러 이상동작들을 해결하고 트래픽을 생성된 시퀀스에 맞춰 정렬함으로써 응용에서 의도한 트래픽 순서에 맞춰 정보를 얻을 수 있다. 이를 통해 통계 정보의 일관성

이 논문은 2012 년 정부(교육과학기술부)의 재원으로 한국연구재단(2012R1A1A2007483) 및 2013 년도 정부(미래창조과학부)의 재원으로 한국연구재단-차세대정보. 컴퓨팅기술개발사업(2010-0020728)의 지원을 받아 수행된 연구임.

을 유지하고, 믿을 수 있는 분석 결과를 낼 수 있다.

본 논문은 다음과 같은 순서로 구성된다. 2 장에서는 기존 통계정보를 이용한 트래픽 분석 방법론에 대해 기술하고, 3 장에서는 제안하는 트래픽 분석 방법을 자세하게 기술한다. 마지막으로 4 장에서는 결론 및 향후 연구에 대해 기술한다.

2. 관련연구

응용 트래픽 플로우의 통계적인 특성을 이용한 트래픽 분류 방법은 최근 몇 년간 많은 관심을 받으며 연구되었다. 그 대부분은 머신러닝(ML, Machine L earning) 알고리즘을 이용한다. 이러한 방법은 응용 별 인터넷 트래픽의 특징이 될 수 있는 항목(port number, flow duration, inter-arrival time, packet size)들을 머신러닝 알고리즘에 적용하여 트래픽을 분류한다. 이 방법은 최근 증가하고 있는 암호화된 트래픽의 분석에 용이하며, 패킷의 페이로드 정보를 분석하지 않기 때문에 개인정보 침해의 문제가 없고 트래픽을 빠른 속도로 분류할 수 있다는 장점을 가진다. 또한, 머신러닝의 고급 알고리즘을 이용함으로써 트래픽을 응용 별로 분류함에 있어 다른 방법에 비해 보다 높은 정확도를 제공한다는 것이다. 하지만 이러한 방법에는 세 가지의 단점이 있다.

첫째, 실시간 트래픽 분류가 어렵다. 이들은 플로우가 끝나고 난 후 통계 정보를 생성하기 때문에 실시간 트래픽 분류가 불가능하다.[2,3,7,8]. 이를 극복하기 위해 플로우의 처음 N 개 패킷만을 이용하여 트래픽을 분류하는 연구들이[4-6] 진행되어 왔으나, 속성(Feature) 추출 계산의 오버헤드와 Machine Learning(ML) 알고리즘의 높은 계산 복잡도가 high-speed b ackbone 네트워크상에서의 실시간 트래픽 분류에 적용하는 것을 어렵게 한다.

둘째, 응용 프로토콜을 기준으로 트래픽을 분석함으로써 분석 결과가 포괄적이다[2-7]. 해당 연구들은 동일한 응용 프로토콜을 사용하는 여러 응용들을 하나로 분류하기 때문에 개별 응용 단위로 적용되는 여러 네트워크 관리 및 운영 정책에 적용하기엔 결과가 너무 포괄적이다.

마지막으로, 트래픽 통계 정보의 비 일관성 문제로 인해 실제 네트워크에서 트래픽을 분석하기에는 무리가 따른다. 해당 방법들은 제한된 환경하에서 이루어진 학습으로 추출된 통계 정보를 이용하여 트래픽을 분석한다. 하지만 이러한 통계 정보는 네트워크의 혼잡도에 따라 얼마든지 변형될 수 있다. 네트워크가 혼잡할 시 TCP 세션에서는 패킷 재전송과 Out-of-order, 그리고 Cross-order 문제가 발생할 수 있으며 UDP 세션에서도 Out-of-order 가 발생할 수 있다. TCP 세션에서 일어나는 패킷 재전송은 패킷에서 에러가 발생하면(혹은 패킷이 분실되면) 수신측의 요청으로 인해 발생한다. T CP 와 UDP 모두에서 발생하는 Out-of-order 는 단일 플로우에서도 패킷이 전송되는 경로는 매번 다르기 때문에 먼저 전송되는 패킷이 후에 전송되는 패킷보다 긴 경로로 전송될 경우 발생한다. 마찬가지로 TCP 와 UDP 모두에서 발생하는 Cross-order 는 서로 반대방향으로 전송되는 패킷이 도착하기 전 전송 경로 중간에서 교차할 때, 교차 전과 후에 트래픽 수집 지점에 따라 패킷의 순서가 달라지는 문제이다[9]. 이러한 문제들은 통계적인 특징인 패킷의 개수와 크기, 그리고 순서와 방향에 영향을 끼치게 된다. 따라서 특징 추출 및 트래픽 분석 시에는 응용에서 처음 발생시킨 그대로의 순서대로 트래픽을 재정렬 해 주어야 항상 일관된 순서의 트래픽을 얻을 수 있다. 한정된 상황에서 학습 된 ML 알고리즘을 통한 트래픽 분류는 통계 정보가 혼잡도에 따라 계속 변하는 실제 네트워크에서 신뢰도 높은 결과를 내기 힘들다.

이러한 기존 연구들의 단점을 해결하기 위해 본 논문에서는 새로운 통계 정보 기반의 트래픽 분석 시스템을 제안한다. 제안하는 방법은 플로우의 초반 5 개 패킷을 사용하며 계산이 필요하지 않은 특징을 사용하고, 간단한 1 차식으로 유사도를 측정하기 때문에 high-speed b ackbone 네트워크에서도 실시간 트래픽 분류에 적용할 수 있다. 또한 트래픽을 개별 응용 단위로 분류함으로써 더욱 자세한 분류 결과를 제시할 수 있다. 마지막으로, 네트워크 계층 IP 헤더의 Identification, flag, f ragmentation o ffset 필드들과 전송 계층 TCP 헤더의 Sequence 필드를 이용하여 트래픽이 본래 생성된 순서대로 재정렬함으로써 특징의 일관성을 유지한다.

3. 트래픽 분석 시스템

본 장에서는 제안하는 트래픽 분석방법에 대해 설명한다. 그림 1 은 제안하는 통계 정보 기반 트래픽 분석 방법의 순서도이다. 제안하는 방법은 트래픽 재정렬부와 시그니쳐 생성부, 그리고 트래픽 분석부로 나눌 수 있으며 이는 크게 off-line t raining 단계와 on-line c lassification 단계로 나눌 수 있다. 트래픽 재정렬부가 두 단계 모두에서 관여하여 시그니쳐 추출시나 트래픽 분석 시에 트래픽 발생 순서로 정렬하기 때문에 통계적 특징을 일관성 있게 다룰 수 있다.

3.1 트래픽 재정렬부

트래픽 재정렬부는 off-line tr aining 단계와 on-line classification 단계 모두에 관여한다. o ff-line tr aining 단계에서는 수집 되어 있는 Ground-truth 트래픽을 재정렬하기 위해 트래픽 전체를 입력으로 받아 처리하며, on-line c lassification 단계에서는 실시간으로 입력되는 개별 플로우를 처리한다. 트래픽 재정렬부는 통계 정보가 변형되었을 때 이를 응용에서 의도한 순서에 맞춰 재정렬 한다. 또한, 각 플로우(Bi-directional f low)를 시그니쳐 생성 및 트래픽 분석에 사용되는 두 개의 단방향 플로우(Uni-directional f low, Uni-flow) 벡터로 변형한다.

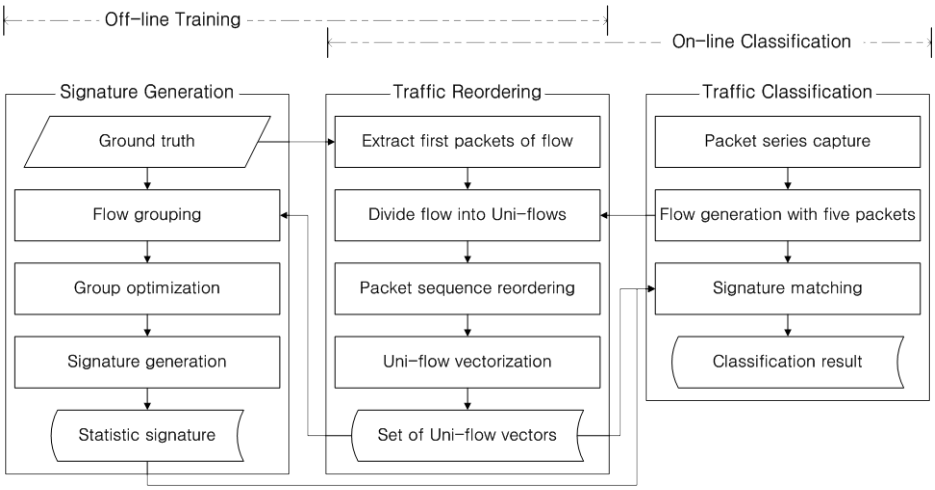


그림 1. 통계 정보 기반 트래픽 분석 시스템의 순서도

본 논문에서는 플로우를 5-tuple(Source IP, P ort, Destination I P, P ort, 전송 계층 프로토콜)이 동일한 양방향 패킷들의 순서 집합으로 정의하며, 5 -tuple 과 전송 방향이 동일한 패킷들의 순서 집합을 단방향 플로우라 정의한다. 전송 방향은 forward, b ackward 로 나뉘는데, client 와 server 의 기준이 명확한 TCP 의 경우 client 에서 서버로 전송하는 방향을 forward 방향, 그 반대되는 방향을 backward 방향이라 정의한다. UDP 의 경우에는 처음 패킷을 요청 패킷으로 간주하여 처음 패킷과 같은 전송방향을 forward 라 정의하고 반대방향을 backward 라 정의한다. 불규칙적으로 발생하여 통계 정보의 일관성을 저하시키는 TCP 플로우의 control 패킷들은 제외한다.

그림 2 는 혼잡한 네트워크에서 발생할 수 있는 여러 이상동작들로 인해 트래픽 수집지점에 따라 수집되는 트래픽 통계 정보의 상이함이 발생하는 예를 보인다. 패킷 a'는 패킷 a 의 분실로 인해 재전송 된 패킷이고, 점선 원으로 표시 된 부분이 Cross-order 발생 부분, 점선 마름모로 표시된 부분이 Out-of-order 가 발생한 부분이다. C1 부터 C4 는 각각 트래픽 수집 지점을 나타낸다.

이상동작이 발생하면 트래픽 수집 지점마다 수집되는 트래픽의 패킷 수와 순서, 해당 순서의 패킷 전송 방향에서 차이를 보인다. 패킷 수에서의 차이는 자주 사용되는 통계적 특징인 전체 패킷 크기에도 영향을 끼친다. 이 때문에 같은 응용의 동일 기능을 사용하여 생성된 트래픽이라 하더라도 트래픽 수집 지점이 달라 생성되는 통계 정보가 달라질 수 있으며, 동일한 지점의 트래픽 수집지점에서도 네트워크의 상황 변화에 따라 트래픽의 통계 정보가 달라질 수 있다. 매 상황마다 달라지는 트래픽의 통계 정보는 결국 통계 정보를 이용한 트래픽 분류 방법의 신뢰도를 낮추며, 실제 네트워크 상의 트래픽 분류에 적용키 어렵게 한다. 따라서 제안하는 방법에서는 트래픽이 수집된 순서를 응용에서 의도한 전송 순서로 재 정렬함으로써 항상 동일한 순서로 트래픽을 다루어 이를 해결한다.

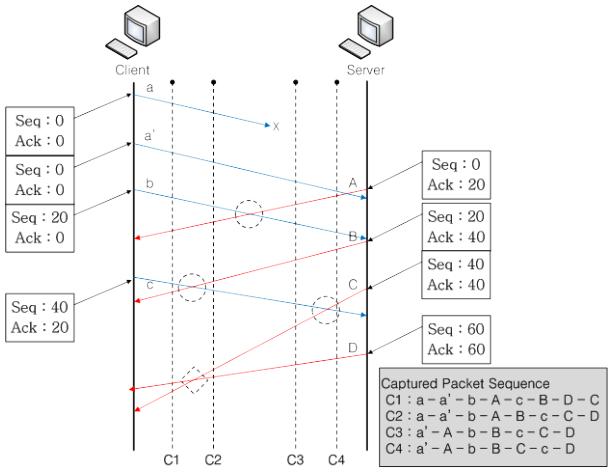


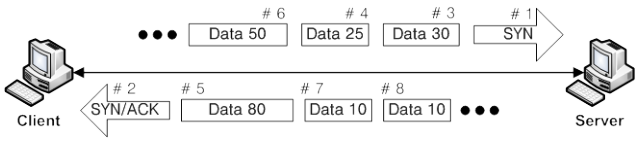
그림 2. 혼잡한 네트워크 상의 트래픽 수집 문제

TCP 의 패킷 재전송과 Out-of-order 문제는 양방향 플로우에서도 TCP 헤더의 Sequence 필드와 Acknowledge 필드 값의 비교로 해결할 수 있다. 정확한 정답 순서가 있기 때문이다. 하지만 나머지 문제는 다르다. O ut-of-order 문제를 해결하기 위해선 단방향 내에서의 순서를 찾아도 반대방향 패킷과의 순서 또한 고려해야 하는데 UDP 에서는 이를 확인할 수 있는 정보가 없기 때문에 정답을 알 수 없다. 또한, 두 전송 프로토콜 모두에서 발생 가능한 Cross-order 문제는 UDP 에서는 물론이고, TCP 에서도 정답을 알 수 없다. 양 호스트에서 패킷 전송 시, 상대 호스트의 전송이 끝나기를 기다리지 않고 전송함으로써 교차하는 두 패킷의 Sequence, Acknowledge 필드 값의 비교는 무의미하기 때문이다. 이 문제를 해결하기 위해 본 논문에서는 플로우를 두 개의 단방향 플로우로 나눈 뒤 트래픽을 재정렬하고, 통계 정보를 수집 및 이용한다. 단방향 플로우에서의 트래픽 정렬은 정확한 정답을 알 수 있기 때문에 어떠한 이상동작이 발생하더라도 응용에서 의도한 전송 순서대로 트래픽을 정렬할 수 있다.

먼저 수집되는 양방향 플로우 내 첫 5 개 패킷만

을 이용하여 두 단방향 플로우로 나눈다. 5 개 패킷에서 추출되는 통계 정보는 충분히 개별 응용을 분류해 낼 수 있다[10]. 그 후 단방향 플로우 내의 패킷들을 응용에서 생성한 순서에 맞춰 재정렬한다. 패킷 순서의 재정렬은 TCP 의 경우 TCP 헤더의 필드인 Sequence 값을 이용하며, UDP 의 경우 IP 헤더의 필드들을 이용한다. TCP 헤더의 Sequence 필드는 TCP 에서 데이터 스트림의 투명성을 위해 제공되는 값으로써 상대 호스트가 자신에게 전송된 패킷의 순서가 맞는지 확인하고 에러를 검출하기 위해 사용된다. 응용계층에서 생성된 데이터가 네트워크 계층, 전송계층을 거치며 패킷화 될 때 해당 패킷의 순서를 기록하는 필드이기 때문에 단방향 트래픽의 순서 정렬에 사용하기 적합하다. Sequence 필드가 없는 UDP 의 경우에는 IP 헤더의 Identification 필드와 flag 필드, fragmentation offset 필드를 이용하여 트래픽을 정렬한다. Identification 필드는 응용 계층에서 생성된 데이터가 네트워크 계층을 지날 때 부여되는 값으로 패킷의 순서를 나타낸다. 해당 값은 패킷 하나가 생성될 때 마다 1 씩 증가하며, 해당 값이 송신자의 주 기억장치에 유지되는 한 유일성을 보장한다. 하나의 패킷이 추가적으로 단편화 되었을 때 각 단편화 된 패킷들은 동일한 identification 필드 값을 가지게 되는데 이때, flag 필드와 fragmentation offset 필드를 이용하여 해당 패킷들의 순서 또한 정렬한다. fragmentation offset 필드는 전체 패킷 내 단편화 된 패킷의 상대적 위치를 나타낸다. flag 필드의 세 번째 비트는 해당 패킷이 마지막 단편일 때 0 값을 가지며, 해당 값이 1 일 경우에는 뒤에 단편화 된 패킷이 더 존재한다는 뜻이다. 단방향 플로우의 경우에는 데이터 스트림을 보장하지 않는 UDP 에서도 이와 같은 IP 헤더의 필드들을 이용하여 생성 순서에 맞춰 정렬할 수 있다.

양방향 플로우를 두 단방향 플로우로 나눈 뒤 순서를 재정렬 한 뒤에 각 단방향 플로우들을 벡터화하여 통계 정보를 추출한다. 즉 각 양방향 플로우들은 forward 벡터와 backward 벡터, 두 방향 벡터를 갖는다. 플로우 벡터화는 단방향 플로우 내 패킷의 순서와 그 크기를 이용한다. 그림 3 은 한 양방향 플로우를 벡터화 했을 때 생성되는 두 단방향 벡터를 나타낸다. #는 양방향 플로우 기준으로 전송 된 순서를 나타내며, 각 상자 안 Data 뒤의 숫자는 Data 크기를 나타낸다. TCP 플로우의 control 패킷들은 제거하고 남은 데이터 패킷들 중 처음 5 개 패킷을 이용하여 방향 별로 벡터를 생성한다.



forward vector = (30, 25, 50)
backward vector = (80, 10)

그림 3. forward, backward 플로우 벡터

3.2 시그니쳐 생성부

응용 프로그램의 시그니쳐란 다른 응용 프로그램과 구별할 수 있는 고유한 통계적 특징을 의미한다. 시그니쳐 생성부에서는 먼저 트래픽 재정렬부에 의해 재 정렬되고 벡터화 된 Ground-truth 트래픽을 이용하여 플로우들을 그룹핑하고, 각 그룹을 최적화한 뒤 시그니쳐를 생성한다. 플로우 그룹핑은 개별 프로세스 내에서 진행되고 그룹 최적화 후 남은 그룹에서 각각 하나의 시그니쳐를 생성하므로 제안하는 방법을 통하여 개별 응용의 시그니쳐를 추출할 수 있다.

플로우 그룹핑은 양방향 플로우를 기준으로 하여, 플로우와 그룹의 대표 단방향 벡터 각각의 유사도가 일정 threshold 내일 때 진행된다. 벡터의 유사도 비교는 방향 별로 이루어지며, 그룹은 두 개의 대표 단방향 벡터를 갖는데, 대표 단방향 벡터는 그룹에 속한 모든 플로우의 동일 방향 벡터의 요소의 평균값이다.

벡터의 유사도 비교는 벡터의 요소의 개수가 같아야 진행되며, 총 2 단계의 순서를 갖는다. 1 단계 비교에서는 벡터 내 각 요소 크기의 전반적인 유사도를 확인한다. 두 벡터 각 요소의 값의 차이가 모두 일정 threshold 보다 작을 때 1 단계 유사도 매칭을 종료하고, 2 단계 유사도 비교를 실행한다. 1 단계 비교로는 두 벡터가 충분히 유사한지를 살필 수는 있지만 유사 정도를 알 수는 없다. 때문에 1 단계에서 유사함이 확인 됐을 경우, 유사 정도 파악을 위해 City-block distance 로 유사도를 비교 한다. 존재하는 모든 플로우 그룹과 유사도 비교를 마친 플로우는 유사 정도가 가장 큰 그룹, 즉 City-block distance 가 가장 작은 그룹으로 그룹핑되고, 그룹의 대표 벡터가 재계산된다. 어떤 그룹과도 유사하지 않은 플로우가 있다면 새로운 그룹을 생성하고, 생성된 그룹의 대표 벡터값들을 해당 플로우의 벡터값으로 할당한다.

모든 플로우에 대해 플로우 그룹핑이 완료되면 그룹 최적화 단계를 실행한다. 그룹 최적화는 2 단계를 거쳐 진행되는데 그룹 내 Outlier 플로우 제거와 Outlier 그룹의 제거가 그것이다. 플로우 그룹핑 단계가 진행될 때 마다 각 플로우는 하나의 그룹으로 할당되며, 그룹의 대표 벡터값은 매번 재계산되어 달라진다. 따라서 한 때 동일 그룹으로 그룹핑되었으나, 그룹핑 단계가 끝났을 때 그룹의 대표벡터와의 유사도가 threshold 값을 넘어서는 Outlier 플로우들이 생기게 된다. 그룹 최적화의 1 단계에서는 그룹 내 모든 플로우와 대표 벡터와의 유사도 비교를 통해 Outlier 플로우를 판별하고 제거한다. 2 단계에서 제거하는 대상은 Outlier 그룹이다. 너무 적은 수의 플로우를 포함하는 그룹은 시그니쳐를 생성하기에 적합하지 않다. 응용의 특정 기능을 사용할 때 항상 발생하는 종류의 플로우라면 충분한 반복을 할 것이고 그러한 플로우들이 모여 이루어진 그룹

의 포함 플로우수가 적을 수 없기 때문이다. 때문에 특정 threshold 보다 적은 수의 플로우를 포함하는 그룹을 제거하여 오탐을 감소시킨다. 그룹 내 플로우 수 threshold 는 너무 작으면 Outlier 플로우들로 이루어진 그룹이 생성될 것이고 너무 크면 정상적인 플로우들을 포함하는 그룹 마저 제거할 수 있기 때문에 threshold 값은 신중히 결정되어야 한다.

그룹 최적화가 끝나고 남아있는 각 그룹에서 각각 하나의 시그니쳐를 생성한다. 시그니쳐는 총 4 개의 벡터로 구성되는데 forward 대표벡터 및 threshold 벡터와 backward 대표벡터 및 threshold 벡터이다. threshold 벡터는 시그니쳐 생성 단계에서 계산되는 값으로, 동일한 방향의 대표 벡터와 threshold 벡터의 합과 차 사이에 그룹 내 모든 플로우들의 벡터를 포함할 수 있도록 하는 가장 작은 값들을 가진 벡터를 의미한다. threshold 벡터는 요소별로 계산되며, 대표 벡터의 한 요소와 그룹 내 플로우 벡터들의 같은 순서의 요소 중 그 차가 가장 큰 값을 threshold 벡터의 요소로 삼는다. 모든 그룹에서 threshold 벡터들이 계산되고, 4 개의 벡터를 갖는 시그니쳐를 생성하게 되면 시그니쳐 생성부는 종료한다.

3.3 트래픽 분석부

트래픽 분석부에서는 생성 된 시그니쳐를 이용하여 온라인으로 트래픽을 분석한다. 온라인으로 수집되는 패킷들을 양방향 플로우 기준으로 5 개까지 기다린 뒤 재정렬 한다. 재정렬 후 플로우 벡터는 off-line training 에서 생성 된 시그니쳐와 매칭한다. 플로우의 단방향 벡터의 각 요소와 시그니쳐의 동일 방향 대표 벡터의 각 요소의 차가 같은 방향 threshold 벡터 각 요소의 값보다 작을 때 해당 단방향 벡터는 시그니쳐의 동일 방향 대표벡터와 유사한 것으로 판단하며, 두 단방향 벡터가 시그니쳐의 두 대표벡터와 모두 유사할 때 해당 플로우는 유사한 시그니쳐가 속한 응용으로 분류된다.

4. 결론 및 향후 연구

네트워크의 효율적 운용과 관리를 위한 트래픽의 응용 계층 분석의 중요성이 증가함에 따라 기존 방법들의 단점을 보완한 트래픽의 통계 정보 기반의 트래픽 분석 방법들이 많이 연구되고 있다. 하지만 기존의 방법들은 수집된 트래픽에 존재하는 이상동작에 대한 처리를 하지 않았다. 수집되는 트래픽의 순서는 네트워크 상황에 따라 발생하는 이상동작들로 인해 언제든 변할 수 있고, 이는 통계 정보에 지대한 영향을 끼친다. 따라서 본 논문에서는 어떤 때이든 응용에서 의도한 순서로 트래픽을 정렬하고 통계 정보를 수집 및 이용하는 새로운 통계 정보 기반 트래픽 분석 방법론을 제안하였다. 제안하는 방법은 수집된 트래픽을 응용에서 의도한 원래의 순서로 정확히 정렬 할 수 있어 어떤 이상동작이 발생하였더라도 정확한 통계 정보를 얻을 수 있는 트래픽 재정렬부와 시그니쳐를 생성하는 시그니쳐 생성부, 그리고 실제 트래픽을 분석하는 트래픽 분석부로 나뉜다.

향후 연구로는 본 논문에서 제안한 설계를 기반으로 실제 네트워크에서 운영되는 시스템을 구축할 계획이다. 또한, 트래픽 정렬에 있어 고려해야 할 다른 사항들을 연구할 것이다.

참고 문헌

[1] A. Callado, C. Kamienski, G. Szabo, B. Gero, J. Kelner, S. Fernandes, et al., "A Survey on Internet Traffic Identification," *IEEE Communications Surveys and Tutorials*, vol. 11, pp. 37-52, 2009.

[2] T. T. T. Nguyen and G. Armitage, "A Survey of Techniques for Internet Traffic Classification using Machine Learning," *IEEE Communications Surveys and Tutorials*, vol. 10, pp. 56-76, 2008.

[3] A. Dainotti, A. Pescape, and K. C. Claffy, "Issues and Future Directions in Traffic Classification," *IEEE Network*, vol. 26, pp. 35-40, January-February 2012.

[4] L. Bernaille, R. Teixeira, and K. Salamatian, "Early application identification," *in Proc. of ACM CoNEXT conference*, 2006.

[5] T. Bujlow, T. Riaz, and J. M. Pedersen, "A method for classification of network traffic based on C5.0 Machine Learning Algorithm," *in Proc. of International Conference on Computing, Networking and Communications (ICNC)*, pp. 237-241, 2012.

[6] C. Yin, S. Li, and Q. Li, "Network traffic classification via HMM under the guidance of syntactic structure," *Computer Networks*, vol. 56, pp. 1814-1825, April 2012.

[7] Y. Jin, N. Duffield, J. Erman, P. Haffner, S. Sen, and Z. L. Zhang, "A Modular Machine Learning System for Flow-Level Traffic Classification in Large Networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, pp. 1-34, 2012.

[8] S. Runyuan, Y. Bo, P. Lizhi, C. Yuehui, Z. Lei, and J. Shan, "Traffic classification using probabilistic neural networks," *in Proc. of International Conference on Natural Computation (ICNC)*, pp. 1914-1919, 2010.

[9] 안현민, 함재현, 김명섭, "Cross-order 문제를 해결한 PSS 시그니쳐 기반 응용 트래픽 분류," KNOM Review, Vol. 16, No. 2, Dec. 2013, pp. 1-11.

[10] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 23-26, 2006.