(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2013-0054511
(43) 공개일자 2013년05월27일

(51) 국제특허분류(Int. Cl.)
H04L 12/26 (2006.01) *H04L 12/24* (2006.01)
H04L 12/22 (2006.01) *H04L 29/06* (2006.01)
(21) 출원번호 10-2011-0119909
(22) 출원일자 2011년11월17일
심사청구일자 2011년11월17일

(71) 출원인
고려대학교 산학협력단
서울 성북구 안암동5가 1
(72) 발명자
김명섭
충청남도 연기군 조치원읍 신흥리 푸르지오1차 107-401
윤성호
경상북도 포항시 북구 우현동 금호어울림 103동 702호
(뒷면에 계속)
(74) 대리인
특허법인엠에이피에스

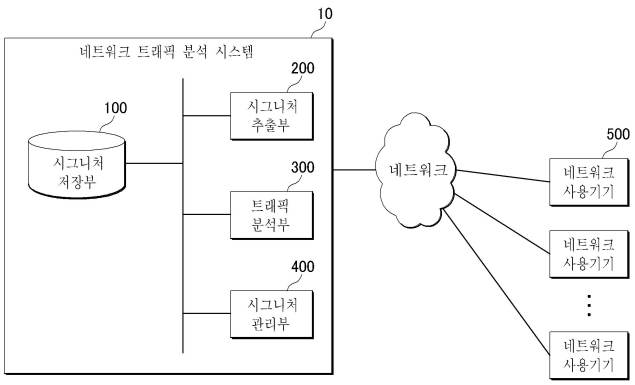
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 **네트워크 트래픽 분석 시스템 및 방법**

(57) 요 약

본 발명은 시그니처를 저장하는 시그니처 저장부; 네트워크 트래픽으로부터 상기 시그니처를 추출하여 상기 시그니처 저장부에 저장하는 시그니처 추출부; 상기 시그니처 저장부에 저장되어 있는 시그니처를 사용하여 네트워크 트래픽을 분석하는 트래픽 분석부; 및 상기 시그니처의 적어도 하나 이상의 특성을 기초로, 트래픽 분석에 유용한 시그니처는 유지시키고 유용하지 않은 시그니처는 삭제하여, 상기 시그니처 저장부에 저장되어 있는 시그니처를 최소량으로 유지하는 시그니처 관리부;를 포함하는 네트워크 트래픽 분석 시스템을 제공한다.

대 표 도 - 도1



(72) 발명자

박준상

전라남도 무안군 현경면 수양촌길 88-3

김지혜

서울특별시 강서구 가양1동 동신대아아파트 105동 101호

특허청구의 범위

청구항 1

네트워크 트래픽 분석 시스템에 있어서,

네트워크 트래픽으로부터 시그니처를 추출하는 시그니처 추출부;

상기 시그니처 추출부에 의해 추출된 상기 시그니처가 저장되는 시그니처 저장부;

상기 시그니처 저장부에 저장되어 있는 시그니처를 기초로 네트워크 트래픽을 분석하고, 상기 시그니처에 상기 트래픽의 분석 이력을 저장하는 트래픽 분석부; 및

상기 분석 이력을 기초로 도출되는 상기 시그니처의 적어도 하나 이상의 특성을 기초로 상기 시그니처 저장부에 저장되어 있는 시그니처의 삭제 여부를 결정하여 상기 시그니처 수를 최소량으로 유지하는 시그니처 관리부;를 포함하되,

상기 분석 이력은 트래픽의 크기, 지속 시간, 호스트, 및 사용 시간 중 적어도 하나 이상을 포함하는 네트워크 트래픽 분석 시스템.

청구항 2

제 1 항에 있어서,

상기 트래픽 분석부는 분석 대상 네트워크의 트래픽을 수집하여, 상기 트래픽을 발생시킨 응용에 따라 분류하고, 상기 분류된 트래픽을 수량적으로 측정하는 것이되,

상기 시그니처 추출부는 상기 네트워크 트래픽을 응용별로 분류하여 상기 응용에 대한 정보를 포함하는 시그니처를 생성하며,

상기 트래픽 분석부는 상기 트래픽 분석시 상기 시그니처에 포함되어 있는 상기 응용 정보를 사용하는 네트워크 트래픽 분석 시스템.

청구항 3

제 1 항에 있어서,

상기 시그니처 추출부는 상기 응용 트래픽을 서버별로 재구성하여 상기 시그니처를 추출하며, 상기 시그니처는 상기 서버의 헤더에서 추출한 IP 주소, 포트 번호, 및 전송 계층 프로토콜 정보를 키로 하는 해시 자료 구조를 사용하여 상기 시그니처 저장부에 저장되며,

상기 트래픽 분석부는 수집된 네트워크 트래픽의 헤더 정보에서 추출한 IP 주소, 포트 번호, 및 전송 계층 프로토콜 정보를 해시 키로 하여 상기 시그니처 저장부에서 상기 해시 키가 동일한 시그니처를 검색하고, 상기 검색된 시그니처의 응용을 통해 상기 트래픽의 응용을 식별함으로써 상기 트래픽을 분석하는 것인 네트워크 트래픽 분석 시스템.

청구항 4

제 1 항에 있어서,

상기 시그니처 추출부는 정답지 트래픽(Ground Truth Traffic)을 실시간으로 수집하여 상기 시그니처를 추출하는 네트워크 트래픽 분석 시스템.

청구항 5

제 1 항에 있어서,

상기 시그니처 특성은 시그니처 발생 형태 정보를 포함하며, 상기 시그니처 관리부는 상기 정보를 기초로 일회성 시그니처를 삭제하는 네트워크 트래픽 분석 시스템.

청구항 6

제 1 항에 있어서,

상기 시그니처 특성은 시그니처 사용 주기 정보를 포함하며, 상기 시그니처 관리부는 상기 정보를 기초로 상기 시그니처가 트래픽 분석에 사용되지 않은 시간이 응용별 최대 유휴 시간 이상인 시그니처를 삭제하는 네트워크 트래픽 분석 시스템.

청구항 7

제 1 항에 있어서,

상기 시그니처 특성은 시그니처가 분석한 트래픽의 통계적 특성 정보를 포함하며, 상기 시그니처 관리부는 상기 정보를 기초로 전체 트래픽에 큰 영향을 미치는 우세한 트래픽을 분석해내는 우세한 시그니처를 유지하되,

상기 우세한 시그니처는 상기 분석 이력에 기록된 트래픽의 다양한 특성 중 적어도 하나가 우세하다고 판단될 경우 우세한 시그니처로 결정되는 것이며,

상기 특성이 우세한지 여부는 상기 시그니처에 해당되는 응용으로 분석된 트래픽의 평균과 표준 편차에 기초하여 판단되는 것인 네트워크 트래픽 분석 시스템.

청구항 8

제 1 항에 있어서,

상기 시그니처 특성은 시그니처 사용 빈도 정보를 포함하며, 상기 시그니처 관리부는 상기 정보를 기초로 여러 호스트에 사용된 인기 시그니처를 유지하되,

상기 인기 시그니처는 상기 시그니처의 응용을 사용하는 호스트 수 대비 상기 시그니처를 사용하는 호스트 수 비율이 임계값을 초과한 경우인 네트워크 트래픽 분석 시스템.

청구항 9

네트워크 트래픽 분석 방법에 있어서,

(a) 네트워크 트래픽을 응용별로 분류하여 상기 응용에 대한 정보를 포함하는 시그니처를 추출하는 단계;

(b) 분석 대상 네트워크의 트래픽을 수집하고, 상기 시그니처에 포함되어 있는 상기 응용 정보를 사용하여 상기 트래픽을 발생시킨 응용에 따라 분류하고, 상기 분류된 트래픽을 수량적으로 측정하여 네트워크 트래픽을 분석하고, 상기 시그니처에 상기 트래픽의 크기, 지속 시간, 호스트, 및 사용 시간 중 적어도 하나 이상을 포함하는 트래픽 분석 이력을 저장하는 단계; 및

(c) 상기 분석 이력을 기초로 도출되는 상기 시그니처의 적어도 하나 이상의 특성을 기초로 상기 시그니처의 삭제 여부를 결정하여, 상기 시그니처 수를 최소로 유지하는 단계;를 포함하는 네트워크 트래픽 분석 방법.

청구항 10

제 9 항에 있어서,

상기 시그니처는 상기 응용 트래픽을 서버별로 재구성하여 헤더에서 추출한, 상기 서버의 IP 주소, 포트 번호, 및 전송 계층 프로토콜 정보를 포함하는 네트워크 트래픽 분석 방법.

청구항 11

제 9 항에 있어서,

상기 시그니처 특성은 시그니처 발생 형태, 시그니처사용 주기, 시그니처가 분석한 트래픽의 통계적 특성, 및 시그니처 사용 빈도 정보를 포함하며,

상기 (c) 단계는 상기 정보를 기초로 일회성 시그니처, 및/또는 트래픽 분석에 사용되지 않은 시간이 응용별 최대 유휴 시간 이상인 시그니처를 삭제하고,

전체 트래픽에 큰 영향을 미치는 우세한 트래픽을 분석해내는 우세한 시그니처, 및 여러 호스트에 사용된 인기

시그니처를 유지하는 네트워크 트래픽 분석 방법.

명 세 서

기 술 분 야

[0001] 본 발명은 네트워크 트래픽 분석에 관한 것으로서, 보다 상세하게는 네트워크 트래픽 분석에 사용되는 시그니처를 효과적으로 관리하는 네트워크 트래픽 분석 시스템 및 방법에 관한 것이다.

배 경 기 술

[0002] 네트워크와 인터넷이 발달함에 따라 효율적이고 안전하게 네트워크를 관리하기 위해 네트워크 트래픽 분석의 중요성이 커지고 있다. 네트워크 수요를 만족시키기 위해 대역폭을 관리하기 위해서도, 악성적인 공격을 감지, 방어하기 위해서도 트래픽 분석은 중요하다.

[0003] 가장 원시적인 방법은 포트 정보만을 사용하는 것으로, 간단하고 빠르지만, 정확성이 매우 낮다. 따라서 정확한 트래픽 분석을 위한 연구가 많이 제안되고 있다.

[0004] 예를 들어, 페이로드(payload) 기반 시그니처를 추출하여 트래픽 분석에 활용하는 방법이 있다. 하지만 페이로드 시그니처는 분석 오버헤드가 높고, 페이로드가 없는 트래픽이나 암호화된 트래픽에는 적용하기 어렵다는 난점이 있다.

[0005] 헤더 기반 분석 방법은 포트 번호와 함께 호스트 정보를 함께 사용한다. 특정 응용을 제공하는 서버의 헤더 정보(IP 주소, 포트 번호, 전송 계층 프로토콜 등)를 사용하여 트래픽을 분석하는 것이다. 헤더 시그니처를 사용하면 빠르고 정확하게 트래픽을 분석할 수 있는 장점이 있다.

[0006] 하지만 추출되는 시그니처의 양이 많아진다는 단점이 있다. 따라서 헤더 시그니처의 장점을 살리되, 시그니처의 수는 최소한으로 줄이기 위한 네트워크 트래픽 분석 시스템 및 방법이 필요하다. 즉, 트래픽 분석에 도움이 되는 시그니처는 유지하되, 트래픽 분석에 많이 사용되지 않는 시그니처는 삭제할 수 있는 네트워크 트래픽 분석 시스템 및 방법이 요구된다.

[0007] 이와 관련하여 한국등록특허 제10-0615080호("10-0615080")에는 네트워크 데이터의 프로토콜 헤더로부터 TCP 세션 기반 통계적 정보를 추출하고 가공하는 가공 단계; 데이터 가공을 통해 얻은 탐지척도에 기반하여 의사결정 나무 알고리즘을 이용해서 나무 모형을 생성하는 생성 단계; 나무모형을 탐지규칙으로 탐지 패턴을 자동 생성하는 정형화단계 및; 각 네트워크의 세션이 끝날 때 정보를 탐지규칙과 비교하여 매칭되면 알람을 울리고, 이에 해당되는 규칙을 살펴봄으로 공격의 특징을 분류하는 실험 및 분석 단계를 포함하는 것을 특징으로 하는 컴퓨터 네트워크상에서 악성 봇과 웜에 대한 규칙기반 탐지패턴을 자동 생성하는 구성이 개시되어 있다.

[0008] 한편, 한국등록특허 제10-0937217호("시그니처 최적화 시스템 및 방법")는 네트워크를 통해 입력되는 데이터 플로우로부터 공격 의심되는 적어도 하나의 패킷을 이용하여 시그니처를 생성하는 시그니처 생성수단; 네트워크를 통해 입력되는 플로우를 저장하고, 플로우 중 시그니처 생성수단을 통해 생성된 시그니처에 대응하는 플로우를 추출하는 패킷 수집부; 패킷 수집부를 통해 추출된 플로우가 공격 징후를 포함하는지 확인하는 검증수단; 및 시그니처 생성수단을 통해 생성된 시그니처를 검증수단을 통해 확인된 공격 징후 별로 최종 시그니처를 생성하는 시그니처 최적화수단;을 포함하는 구성이 개시되어 있다.

발명의 내용

해결하려는 과제

[0009] 본 발명은 전술한 네트워크 트래픽 분석 문제를 해결하기 위한 것으로서, 그 목적은 네트워크 트래픽 분석에 유용한 최소의 시그니처만을 유지하여 분석 성능과 효율성이 높은 네트워크 트래픽 분석 시스템 및 방법을 제공하는 것이다.

과제의 해결 수단

[0010] 상기와 같은 목적을 달성하기 위한 본 발명의 제 1 측면에 따른 네트워크 트래픽 분석에 사용되는 시그니처를 관리하는 네트워크 트래픽 분석 시스템은, 상기 시그니처를 저장하는 시그니처 저장부; 네트워크 트래픽으로부터 상기 시그니처를 추출하여 상기 시그니처 저장부에 저장하는 시그니처 추출부; 상기 시그니처 저장부에 저장

되어 있는 시그니처를 사용하여 네트워크 트래픽을 분석하는 트래픽 분석부; 및 상기 시그니처의 적어도 하나 이상의 특성을 기초로, 트래픽 분석에 유용한 시그니처는 유지시키고 유용하지 않은 시그니처는 삭제하여, 상기 시그니처 저장부에 저장되어 있는 시그니처를 최소량으로 유지하는 시그니처 관리부;를 포함하는 것을 특징으로 한다.

[0011] 상기와 같은 목적을 달성하기 위한 본 발명의 제 2 측면에 따른 네트워크 트래픽 분석에 사용되는 시그니처를 관리하는 네트워크 트래픽 분석 방법은, (a) 네트워크 트래픽으로부터 상기 시그니처를 추출하는 단계; (b) 상기 시그니처를 사용하여 네트워크 트래픽을 분석하는 단계; 및 (c) 상기 시그니처 목록을 최소량으로 유지하는 단계;를 포함하는 것을 특징으로 한다.

발명의 효과

[0012] 본 발명은 네트워크 트래픽 분석 시스템 및 방법에 있어, 오버헤드는 낮추면서 정확하고 빠르게 네트워크 트래픽을 분석하는 효과를 얻는다.

[0013] 즉, 포트 번호만을 사용하는 방법보다 정확하고 페이로드 시그니처를 사용하는 방법보다 오버헤드가 낮고 빠르며, 네트워크 트래픽 분석에 유용한 최소의 시그니처만을 유지하므로 분석 성능은 더욱 향상되고 자원은 절약된다.

[0014] 또한 트래픽 수집시 발생할 수 있는 패킷 손실 및 TCP/IP 조각화 문제를 고려하지 않아도 된다.

[0015] 또한 페이로드가 없거나 암호화된 경우에도 트래픽을 분석할 수 있다.

[0016] 또한 시그니처 추출, 트래픽 분석, 시그니처 관리를 실시간으로 수행할 수 있다는 장점이 있다.

도면의 간단한 설명

[0017] 도 1은 본 발명에 따른 네트워크 트래픽 분석 시스템의 구조를 도시함.

도 2는 도 1의 시스템이 사용하는 트래픽 데이터의 개념을 도시함.

도 3은 도 1의 시스템이 트래픽 헤더에서 추출하는 데이터의 구조를 도시함.

도 4는 본 발명에 따른 네트워크 트래픽 분석 방법의 흐름을 도시함.

도 5는 시그니처 추출 단계의 흐름을 도시함.

도 6은 트래픽 분석 단계의 흐름을 도시함.

도 7은 시그니처 관리 단계의 흐름을 도시함.

도 8은 비정상 트래픽의 특성 및 P2P트래픽의 특성을 분석한 그래프를 도시함.

도 9는 발생 형태 분석 단계의 흐름을 도시함.

도 10은 응용별 시그니처 최대 유휴 시간을 분석한 그래프를 도시함.

도 11은 사용 주기 분석 단계의 흐름을 도시함.

도 12는 시그니처별 트래픽 분석 성능을 분석한 그래프를 도시함.

도 13은 통계적 특성 분석 단계의 흐름을 도시함.

도 14는 응용별 시그니처 사용률을 분석한 그래프를 도시함.

도 15는 사용 빈도 분석 단계의 흐름을 도시함.

발명을 실시하기 위한 구체적인 내용

[0018] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0019] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐

아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.

[0020] 먼저, 도 1에서 도 3을 참조하여 본 발명에 따른 네트워크 트래픽 분석 시스템(10)을 설명한다.

[0021] 도 1은 본 발명에 따른 네트워크 트래픽 분석 시스템(10)을 나타낸 블록도이다.

[0022] 도 1을 참조하면, 본 발명에 따른 네트워크 트래픽 분석 시스템(10)은 하나 이상의 네트워크 사용 기기(500)와 네트워크를 통해 연결되어 있다. 여기서 네트워크는 근거리 통신망(Local Area Network; LAN), 광역 통신망(Wide Area Network; WAN) 또는 부가가치 통신망(Value Added Network; VAN) 등과 같은 유선 네트워크나 이동통신망(mobile radio communication network) 또는 위성 통신망 등과 같은 무선 네트워크 등 모든 종류의 네트워크로 구현될 수 있다. 네트워크 사용 기기(500)도 컴퓨터, 휴대용 단말 등 기기의 종류에 제한을 두지 않으며, 서버인지 클라이언트인지에 상관 없이 네트워크를 사용하는 모든 기기를 포함한다. 또한 네트워크 트래픽 분석 시스템(10)을 구성하는 방법에도 제한이 없다. 도면에는 단일 시스템으로 도시되어 있으나, 여러 네트워크 사용 기기(500)에 분산되어 구성될 수도 있다.

[0023] 네트워크 트래픽 분석 시스템(10)은 시그니처를 저장하는 시그니처 저장부(100), 네트워크 트래픽으로부터 시그니처를 추출하는 시그니처 추출부(200), 시그니처를 활용하여 네트워크 트래픽을 분석하는 트래픽 분석부(300), 시그니처 저장부(100)에 저장되어 있는 시그니처를 최소량으로 유지하는 시그니처 관리부(400)를 포함한다.

[0024] 이때 네트워크 트래픽 분석이란 분석 대상 네트워크의 트래픽을 수집하여 분석 기준에 맞게 트래픽을 분류하고 분류된 트래픽을 수량적으로 측정하는 것을 의미한다. 본 발명에 따른 네트워크 트래픽 분석 시스템(10)이 채택하고 있는 분류 기준은 트래픽을 발생시킨 응용 프로그램이다(이하 응용이라 함). 즉, 네트워크 트래픽을 해당 트래픽을 발생시킨 응용을 기준으로 분류하고 응용별로 트래픽량을 측정한다. 분석된 응용 트래픽은 네트워크 관리 및 보안에 중요한 자료로 사용될 수 있다. 예를 들어, 특정 응용에서 발생하는 트래픽의 대역폭을 조절하거나 차단하는 등 네트워크 자원을 관리하기 위해서는 응용 트래픽을 실시간 분석하는 것이 중요하다.

[0025] 응용 트래픽을 측정하기 위해, 시그니처 추출부(200)는 네트워크 트래픽을 응용별로 분류하여 응용에 대한 정보를 포함하는 시그니처를 생성하며, 트래픽 분석부(300)는 트래픽 분석시 시그니처에 포함되어 있는 응용 정보를 사용하여 트래픽을 응용별로 분류한다.

[0026] 이때 트래픽은 서버 별로 재구성된다. 설명을 위해 네트워크 트래픽 분석 시스템(10)이 사용하는 트래픽 데이터의 개념을 도시하고 있는 도 2를 참조한다. 도면에서 원은 IP 주소, 사각형은 포트를 나타낸다.

[0027] 종래 기술들은 트래픽 즉, 패킷을 플로우(flow) 단위로 조합하여 사용한다. 플로우는 동일한 5-튜플(5-tuple: 소스 IP 주소, 목적지 IP 주소, 소스 포트, 목적지 포트, 전송 계층 프로토콜)을 가지는 단방향 패킷들의 집합이다. 하지만 네트워크에서 발생하는 응용 트래픽은 특정 서버를 중심으로 발생되기 때문에 플로우를 서버 기준으로 재조합하면 응용의 발생 형태와 플로우간의 관계를 좀더 효과적으로 파악할 수 있다.

[0028] 따라서 본 발명은 응용 서버 기준으로 플로우를 재조합한 번치(bunch)를 사용한다. 번치는 특정 서버의 3-튜플(3-tuple: IP 주소, 포트, 전송 계층 프로토콜)를 포함하는 모든 플로우들의 집합이다. 도 2의 우측 도면에 도시된 것과 같이, 번치는 하나의 서버 노드와 하나 이상의 클라이언트 노드로 구성된다. 대부분의 응용 서버는 하나의 서버 포트와 다수의 클라인언트 포트를 통해 트래픽을 발생시킨다. 이러한 관점에서 볼 때, 번치 단위의 트래픽은 특정 서버에서 발생한 응용 트래픽을 더욱 명확하게 나타낸다.

[0029] 도 3에서 보는 바와 같이, 응용 서버별로 재구성된 트래픽의 헤더에서 본 발명에 따른 네트워크 트래픽 분석 시스템(10)은 IP 주소, 포트 번호, 전송 계층 프로토콜을 추출한다. 즉, 서버 노드의 3-튜플이 헤더 시그니처로 추출된다. 이 3-튜플은 시그니처 저장부(100)가 사용하는 해시(hash) 자료 구조의 키(key)로 사용되기 위한 것이다.

[0030] 즉, 시그니처 추출부(200)는 시그니처 생성시 생성된 시그니처를 응용 서버별로 재구성된 트래픽의 헤더에서 추출한 <IP 주소, 포트 번호, 전송 계층 프로토콜 정보>를 키로 하여 시그니처 저장부(100)에 저장하고, 트래픽 분석부(300)는 시그니처 저장부(100)에서 트래픽 분석에 사용할 시그니처를 검색하기 위해 <IP 주소, 포트 번호, 전송 계층 프로토콜 정보>를 키로 사용한다. 즉, 분석할 트래픽에서 시그니처 추출부(200)가 시그니처를 생성할 때와 같은 방법으로 <IP 주소, 포트 번호, 전송 계층 프로토콜 정보>를 추출하고, 시그니처 저장부(100)에 저장되어 있는 시그니처들 중 이와 동일한 키를 갖는 시그니처가 있는지 검색한다. 이 키가 분석 대상 트

래픽과 동일한 시그니처를 발생시킨 응용은 분석 대상 트래픽을 발생시킨 응용과 동일할 것이므로, 3-튜플을 키로 하여 검색된 시그니처에 포함된 응용 정보를 통해 트래픽을 분석하는 것이 가능하다.

[0031] 이렇게 헤더 정보를 이용하여 트래픽을 분석할 경우 종래의 포트 번호만을 사용하는 방법이나 페이로드 시그니처를 사용하는 방법에 비해 여러가지 이점이 있다. 예를 들어, 포트 번호만을 사용하여 트래픽을 분석하는 경우보다 정확도가 높으며, 트래픽 수집시 발생할 수 있는 패킷 손실 및 TCP/IP 조각화(fragmentation) 문제를 고려하지 않아도 된다. 헤더 정보를 통해 트래픽을 분석하기 때문에 페이로드(payload)가 없거나 암호화된 경우에도 트래픽을 분석할 수 있으며, 페이로드 시그니처보다 분석 오버헤드가 덜하고 자원을 덜 사용한다. 트래픽의 헤더 정보는 고정된 위치에 존재하기 때문에 추출과 분석이 용이하다. 이에 더해, 본 발명에 따른 네트워크 트래픽 분석 시스템(10)은 전술한 바와 같이 추출된 헤더 정보를 해시 자료 구조의 키로 사용하기 때문에 빠르게 동일한 시그니처를 찾을 수 있다는 장점도 갖는다.

[0032] 이렇게 헤더 기반 시그니처를 사용하면 정확하고 빠르게 트래픽을 분석할 수 있지만, 헤더 정보를 사용하기 때문에 많은 양의 시그니처가 추출된다. 따라서 최적의 시그니처를 유지할 수 있는 관리 방법이 필요하다. 전술한 헤더 시그니처의 장점을 활용하면서도 단점은 최소화하는 것이 본 발명의 목적이다. 그 방법은 분석에 도움이 되는 시그니처는 목록에 유지하고, 분석에 도움이 되지 않으면서 자원만 낭비하고 있는 시그니처는 삭제하는 것이다.

[0033] 이를 위해 시그니처의 사용 이력이 이용된다. 즉, 시그니처는 자신이 분석한 트래픽의 크기, 지속 시간, 호스트, 사용 시간 등 트래픽 분석 이력을 저장하며, 이 정보는 시그니처 관리부(400)가 시그니처를 유지, 관리하는 데 사용된다. 이렇게 응용이 발생시키는 트래픽의 발생 형태와 특성, 그리고 사용 이력을 고려하여 트래픽 분석에 활용하는 데 최적인 헤더 시그니처만을 목록에 유지하기 때문에, 본 발명에 따른 네트워크 트래픽 분석 시스템(10)은 오버헤드는 줄이면서 분석률(completeness), 정확도(accuracy) 등 시그니처 분석 성능은 높게 유지할 수 있다.

[0034] 시그니처 관리에 대한 자세한 설명은 이하 도 4에서 도 15를 참조하여 설명한다.

[0035] 도 4는 본 발명에 따른 네트워크 트래픽 분석 방법의 흐름을 도시하고 있다.

[0036] 본 발명에 따른 네트워크 트래픽 분석 방법은 네트워크 트래픽으로부터 시그니처를 추출하는 단계(S410), 시그니처를 사용하여 네트워크 트래픽을 분석하는 단계(S420), 시그니처 목록을 최소량으로 유지하는 단계(S430)을 포함한다.

[0037] 도 4에는 이 단계들이 순차적으로 이루어지는 것처럼 도시되어 있지만, 이 단계들은 동시에 수행될 수도 있으며, 또한 많은 경우 실제 네트워크 환경에서 실시간으로 이루어질 수 있다. 인터넷 응용은 사용자의 요구에 따라 매번 수정되고 새로 출현하기 때문에 실시간 시그니처 추출 및 관리가 필수적인데, 본 발명에 따른 네트워크 트래픽 분석 시스템(10)은 시그니처 추출, 트래픽 분석, 시그니처 관리를 실시간으로 수행할 수 있다는 장점이 있는 것이다.

[0038] 도 5는 시그니처 추출 단계(S410)의 흐름을 도시하고 있다.

[0039] 시그니처 추출 단계(S410)에서 시그니처 추출부(200)는 정답지 트래픽(Ground truth Traffic)을 실시간으로 수집하여 시그니처를 추출한다. 정답지 트래픽을 생성하는 방법은 이미 많은 연구를 통해 제안되었으므로, 자세한 설명은 생략한다. 예를 들어, 정답지 트래픽은 실제 트래픽을 발생 시키는 호스트에 설치된 에이전트를 통해 수집될 수 있다.

[0040] 전술한 바와 같이, 정답지 트래픽에서 특정 응용을 서비스하는 서버의 헤더 정보가 시그니처로 추출된다. 이를 위해서는 플로우의 서버를 구분해야 하는데, TCP 플로우의 경우 SYN, ACK 플래그를 통해 서버를 쉽게 판별할 수 있지만, UDP 플로우는 서버 판별이 모호하다. 따라서 UDP 플로우인 경우에는 두 종단 호스트에 발생한 첫 번째 패킷을 서비스 요청으로 가정하여 첫 패킷의 목적지 호스트를 서버로 판별한다.

[0041] 도면에 도시되어 있는 추출 방법은 플로우를 번치로 재조합하는 부분(line:4~6)과 생성된 번치의 서버 노드에서 시그니처를 추출하는 부분(line:7~9)으로 구성된다. 추출된 시그니처와 기존에 추출된 시그니처가 동일한 3-튜플을 가지고 서로 다른 응용에서 추출된 경우 충돌이라 정의하며, 충돌이 발생하면 기존의 시그니처와 새로 추출된 시그니처 모두 삭제한다

[0042] 도 6은 트래픽 분석 단계(S420)의 흐름을 도시하고 있다.

[0043] 트래픽 분석 단계(S420)에서 트래픽 분석부(300)는 트래픽의 서버 3-튜플을 추출(line:5)하여 동일한 3-튜플을 가지는 헤더 시그니처를 찾는다(line:6). 3-튜플을 추출하는 방법은 앞서 설명한 시그니처 생성 단계(S410)에서 서버를 구분한 방법과 동일하다. 동일한 시그니처를 찾았을 경우, 해당 시그니처의 응용으로 트래픽을 분석한다(line:7). 이때 3-튜플 정보를 키로 사용하는 해시 자료 구조를 사용하기 때문에 빠르게 동일한 시그니처를 찾을 수 있다. 또한, 시그니처의 분석 이력을 기록하여 시그니처 관리 단계(S430)에서 활용한다

[0044] 도 7은 시그니처 관리 단계(S430)의 흐름을 도시하고 있다.

[0045] 전술한 바와 같이 시그니처를 관리한다는 것은 각 시그니처의 특성을 파악하여 트래픽 분석에 유용한 시그니처는 목록에 유지 시키고 그렇지 않은 시그니처는 목록에서 삭제하는 것을 의미한다. 이를 위해 시그니처 관리부(400)는 발생 형태, 사용 주기, 트래픽 통계적 특성, 사용 빈도를 고려한다.

[0046] 트래픽 분석 단계(S410)에서 입력 데이터는 현재의 시그니처 목록(HSbefore)이고 출력 데이터는 최적화된 시그니처 목록(HSafter)이다. 시그니처 관리부(400)는 추출된 시그니처의 특성을 파악하여 비정상 트래픽(예: 단방향 통신, P2P)을 분석하는 시그니처(isAbnormal)와 해당 응용을 기준으로 일정 기간 사용되지 않은 시그니처(isTimeout)를 삭제한다(line:5). 단, 해당 시그니처로 분석된 트래픽이 우세한 특징을 가지고(isDominant), 여러 호스트에 의해 사용된 경우(isPopular), 삭제 대상에서 제외된다(line:6)

[0047] 각 세부 관리 방법은 이하 도 8에서 도 15를 사용하여 설명한다. 도 8, 9는 발생 형태, 도 10, 11은 사용 주기, 도 12, 13은 트래픽 통계적 특성, 도 14, 15는 사용 빈도 정보를 시그니처 관리에 어떻게 사용하는지 도시한 도면이다.

[0048] 도 8은 비정상 트래픽의 특성 및 P2P트래픽의 특성을 분석한 그래프를 도시하며, 도 9는 발생 형태 분석 단계의 흐름을 도시한다.

[0049] 비정상적인 트래픽(포트 스캔, NetBIOS-SMB)과 P2P 응용에서 발생한 트래픽은 발생 형태의 특성 때문에 많은 시그니처가 추출되지만, 추출된 시그니처의 대부분은 분석에 일회적으로만 사용되기 때문에 분석 성능을 떨어뜨리고 메모리 오버헤드를 증가시키므로, 삭제하는 것이 바람직하다.

[0050] 먼저 도 8을 통해 비정상 트래픽(위)과 P2P 응용(아래)의 특성을 살펴보자.

[0051] 비정상과 정상 트래픽의 시그니처 당 분석된 플로우 수(fps)와 시그니처 당 분석된 클라이언트 수(cps)를 CDF(Cumulative Distribution Function)로 나타내면, 도면과 같이 비정상 트래픽에서 추출한 대부분의 시그니처(96.16%)는 오직 하나의 플로우만 분석한다. 즉, 정상적인 양방향 통신이 아니 단방향 통신에서 발생된 트래픽을 분석한다. 이에 반해 정상 트래픽에서 추출한 시그니처는 소수의 시그니처(10.01%)만이 하나의 플로우를 분석한다.

[0052] 이러한 특성을 이용하여 비정상 트래픽에서 추출한 시그니처를 판별할 수 있지만, 좀 더 정확한 판별을 위해 추가적인 특성을 사용할 수 있다. 비정상 트래픽에서 추출한 시그니처로 분석된 트래픽의 고유한 클라이언트 수(cps)를 조사한 결과 시그니처의 cps가 대부분 1이었다. 즉, 해당 시그니처로 분석된 플로우를 앞서 설명한 번치로 재구성하였을 때, 오직 하나의 클라이언트 노드로 구성된다.

[0053] 따라서 특정 시그니처가 분석한 플로우의 개수가 1이고, 해당 시그니처로 분석된 클라이언트 수가 1이면, 비정상 시그니처로 판별할 수 있다. 즉, 단일 호스트에서 발생된 단방향 플로우를 나타내는 것은 비정상 시그니처로 판별할 수 있다

[0054] 또한 P2P 시그니처로 분석된 트래픽의 플로우 대부분(80%)은 오직 하나의 패킷으로 구성되었다. 따라서 오직 하나의 패킷으로 구성된 트래픽은 P2P 응용의 검색 또는 파일 전송에 관련된 것이므로 해당 시그니처를 P2P 시그니처로 판별할 수 있다.

[0055] 이러한 특성을 이용해 도 9의 방법은 비정상 시그니처(line:4~7), P2P 시그니처(line:8~10)를 판별한다. 비정상 시그니처를 판별하기 위해 해당 시그니처로 분석된 플로우 개수를 확인(line:5)하고 단방향 플로우이면 해당 트래픽을 발생 시킨 클라이언트 개수를 확인(line:6)하고, 해당 트래픽이 특정 호스트에서 독점적으로 발생한 것이면 비정상 시그니처로 판별한다. P2P 시그니처는 해당 시그니처로 분석된 트래픽의 플로우 당 패킷 수를 확인(line:9)하여 1이면 P2P 시그니처로 판별한다.

[0056] 도 10은 응용별 시그니처 최대 유휴 시간을 분석한 그래프를 도시하며, 도 11은 사용 주기 분석 단계의 흐름을 도시한다.

[0057] 시그니처 목록의 폭발적인 증가를 막기 위해서는 오랜 기간 사용하지 않은 시그니처를 목록에서 제거해야 한다. 하지만 모든 응용에 동일한 유휴 시간 기준값을 적용하면 사용 주기가 긴 응용의 시그니처는 지속적으로 목록에서 삭제되어 버릴 것이다. 따라서, 응용의 사용 주기를 반영하여 기준값을 설정하고 시그니처를 관리해야 한다

[0058] 도 10은 응용별 시그니처의 사용 주기를 확인하기 위해 몇가지 응용을 선정하여 시그니처를 추출하고 실제 발생 트래픽을 분석하고, 분석에 사용된 시그니처에 사용 이력을 기록하여 시그니처 별 최대 유휴 시간을 분석한 그래프이다. 정확한 실험을 위해 2회 이상 사용된 시그니처만 대상으로 하였다. 즉, 일회성 시그니처는 실험 대상에서 제외시켰다.

[0059] 그 결과 웹(IE)이나 메신저(nateon) 트래픽을 분석한 시그니처의 유휴 시간은 P2P 응용(torrent)에서 발생한 트래픽을 분석한 시그니처의 유휴 시간보다 상대적으로 길었다. 이를 근거로 시그니처 사용 이력 정보를 활용하여 응용별로 최대 유휴 시간을 정할 수 있음을 알 수 있다. 즉, 응용별로 가장 긴 유휴 시간을 갖는 시그니처에 기록된 유휴 시간을 응용별 최대 유휴 시간을 정하는 데 사용할 수 있다.

[0060] 하지만 네트워크의 환경에 따라 각 응용에 적합한 유휴 시간 기준값이 다를 것이므로, 본 발명의 일실시예에서는 각 응용의 최대 유휴 시간 값을 유동적으로 구하기 위해 해당 응용 시그니처 중 가장 긴 유휴 시간을 α배한 값을 사용할 수 있다. 즉, α배의 이유는 응용 별 유휴 시간 기준을 응용의 사용량에 따라 유동적으로 변화시키기 위함이다. 여기서 α값은 예를 들어 1.1이 될 수 있다.

[0061] 도 11의 방법은 입력된 시그니처의 최대 유휴시간과 해당 응용의 최대 유휴 시간을 비교하여 일정 기간 이상 사용하지 않은 시그니처를 판별한다. 입력된 시그니처의 응용을 확인(line:4)하고, 현재 시간을 기준으로 유휴 시간을 계산한다(line:5). 입력된 시그니처의 유휴 시간이 해당 응용에서 추출된 시그니처들의 최대 유휴 시간 α배와 비교(line:6)하여 크면 참을 반환(line:7)하고 그렇지 않으면 거짓을 반환(line:8)한다. 즉, 해당 응용의 최대 유휴 시간 보다 오랜 기간 사용하지 않은 시그니처를 삭제 대상 시그니처로 판별한다.

[0062] 도 12는 시그니처별 트래픽 분석 성능을 분석한 그래프를 도시하며, 도 13은 통계적 특성 분석 단계의 흐름을 도시한다.

[0063] 트래픽 특성을 연구한 논문들에 따르면, 소량의 우세한(dominant) 몇몇 트래픽이 전체 트래픽에 큰 영향을 미친다. 즉, 전체 트래픽에서 소수의 유력(heavy hitter) 플로우가 트래픽의 많은 바이트 비율을 차지한다.

[0064] 마찬가지로 헤더 시그니처에도 유력 시그니처가 존재 한다. 즉, 트래픽 분석에 있어 우세한 트래픽을 분석하는 우세한 헤더 시그니처가 존재한다. 따라서 이러한 우세한 시그니처들을 시그니처 테이블에 오래 유지시켜야 트래픽 분석 성능을 향상시킬 수 있을 것이다.

[0065] 도 12는 우세한 시그니처의 존재를 확인하기 위하여 일정 기간 동안 시그니처를 추출하고 해당 시그니처로 분석된 트래픽의 비율을 조사한 결과를 보여준다. 우세한 시그니처를 확인하기 위하여 분석한 트래픽의 비율이 높은 순서로 시그니처 아이디를 부여하였다.

[0066] 그 결과, 일부 시그니처(73.56%) 만이 트래픽 분석에 사용되었고, 또한 몇몇 소량의 시그니처(15.17%)가 분석된 트래픽의 대부분(90%)을 분석하였음을 알 수 있다. 즉, 우세한 트래픽을 분석하는 우세한 시그니처가 존재함을 확인할 수 있다

[0067] 본 관리 방법에서는 우세한 시그니처를 판별하기 위해 해당 시그니처가 분석한 트래픽의 다양한 특성(예를 들어, 플로우의 크기, 플로우의 지속 시간)을 사용하며, 하나 이상의 측면에서 우세한 특성을 가지면 해당 시그니처를 우세한 시그니처로 판별한다.

[0068] 또한, 응용별로 상이한 특성을 반영하기 위해 응용별 평균 트래픽 특성을 기준으로 각 트래픽 특성을 파악한다. 우세한 특성을 판단하기 위해 각 특성의 평균(mean)과 표준편차(standard deviation)를 사용한다. 즉, 어떤 특성이 우세한 특성인지 여부는 해당 응용으로 분석된 전체 트래픽의 평균과 표준편차 3배의 합보다 해당 특성의 값이 큰지 여부로 판별된다. 이 방법은 종래에 제안된 방법이므로 왜 이러한 수식을 사용하는지에 대한 설명은 생략한다.

[0069] 도 13의 방법은 앞에서 예로 든 플로우의 크기(i가 0일 때), 플로우의 지속 시간(i가 1일 때)이라는 2가지 측면의 트래픽 특성을 파악하기 위해 반복 구조(line:4~11)를 사용한다. 우선, 시그니처가 입력되면, 해당 시그니처가 추출된 응용을 확인한다(line:6). 확인된 응용으로 분석된 전체 트래픽의 해당 특성에 대한 평균(mean) 과 표준 편차(standard deviation)을 구한다(line:7~8). 해당 시그니처의 특성과 해당 시그니처가 속한 응용의 전

체 특성을 비교하여 우세한 시그니처를 분석한다(line:9~10).

[0070] 도 14는 응용별 시그니처 사용률을 분석한 그래프를 도시하며, 도 15는 사용 빈도 분석 단계의 흐름을 도시한다.

[0071] 시그니처는 해당 응용 서버를 대표하는 것이기 때문에 해당 응용을 사용할 때 자주 사용되는 것이어야 한다. 따라서, 해당 시그니처로 분석된 트래픽을 발생시킨 호스트의 개수 대비 해당 응용으로 분석된 트래픽을 발생시킨 호스트 개수의 비율로 계산되는 시그니처의 사용률이 임계값 β 이상일 때 해당 시그니처를 삭제 대상에서 제외하는 것이 필요하다. 즉, 인기있는 시그니처는 시그니처 목록에 유지하자는 것이다.

[0072] 도 14는 응용별 시그니처의 사용률을 확인하기 위해 몇몇 응용을 선정하고 사용률의 분포를 분석한 그래프이다. 그 결과, 단일 호스트에서 트래픽이 발생하는 P2P 응용의 경우 대부분 시그니처의 사용률이 0인 것에 반해, 메신저 응용의 경우 사용률이 상대적으로 높은 시그니처가 존재함을 확인할 수 있다.

[0073] 도 15의 방법은 시그니처가 입력되면 해당 시그니처가 추출된 응용을 확인한다(line:4). 확인된 응용을 통해 해당 응용을 사용한 호스트의 개수를 구한다(line:5). 이 응용을 사용한 총 호스트의 개수와 해당 시그니처를 사용한 호스트의 개수의 비율(line:6)이 특정 기준값(β)을 넘으면 해당 시그니처는 인기있는 시그니처로 판별된다.

[0074] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

[0075] 본 발명의 범위는 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

[0076] 10: 네트워크 트래픽 분석 시스템

100: 시그니처 저장부(100)

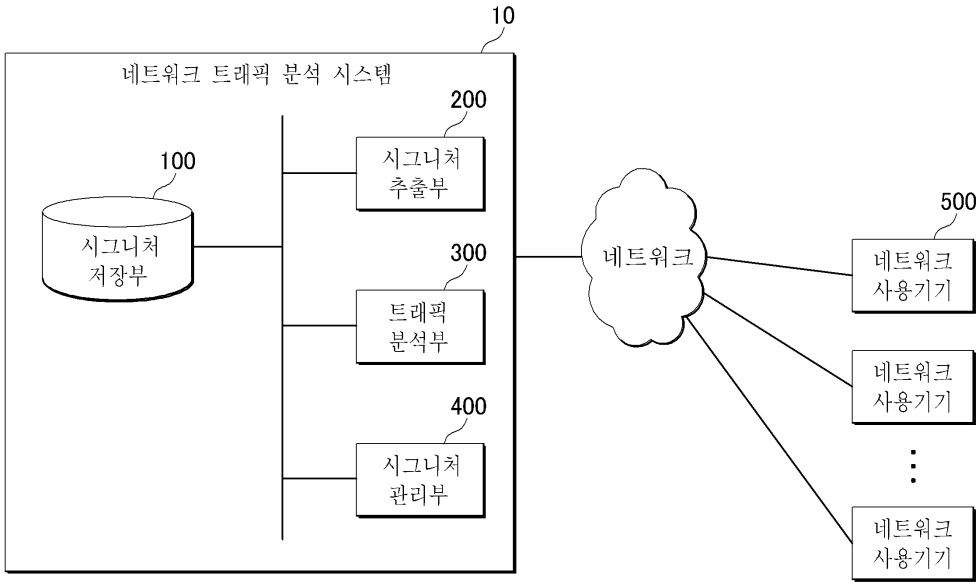
200: 시그니처 추출부(200)

300: 트래픽 분석부(300)

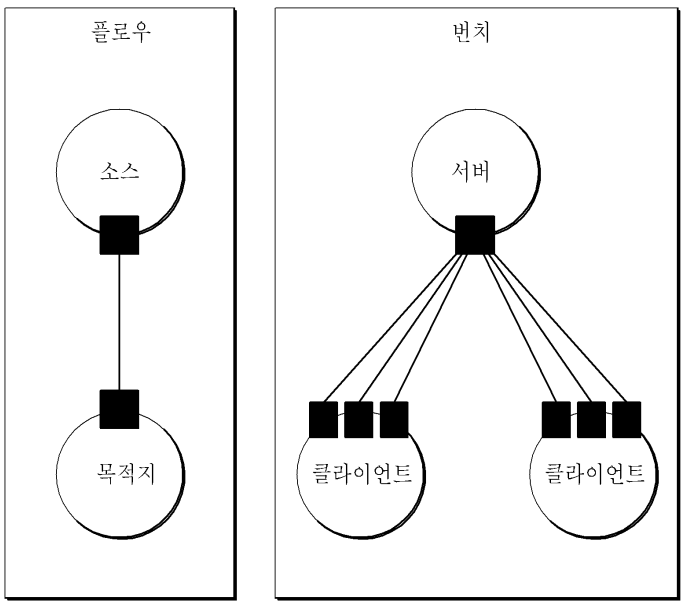
400: 시그니처 관리부(400)

도면

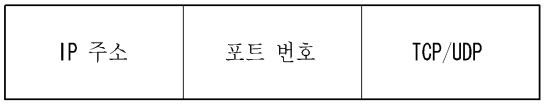
도면1


10
네트워크 트래픽 분석 시스템
200
시그니처
추출부
100
시그니처
저장부
300
트래픽
분석부
400
시그니처
관리부
네트워크
500
네트워크
사용기기
네트워크
사용기기
네트워크
사용기기

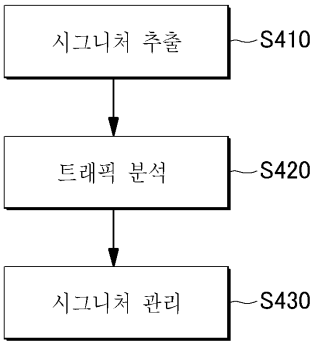
도면2


플로우
소스
목적지
번치
서버
클라이언트
클라이언트

도면3



도면4



도면5

Algorithm 1. extractSignature

```
1:  Input: Flow={f1,...,fn}, HS= ø
2:  Output:Bunch={b1,...,bk }, HS={hs1,...,hsm}
3:  void extractSignature (...){
4:      for( i=1; i≤n ; i++){
5:  makeBeunch(fi);
6:      }
7:  for( i=1; i≤k ; i++){
8:  HS=HS∪{getServer(bi)};
9:      }
10: }
```

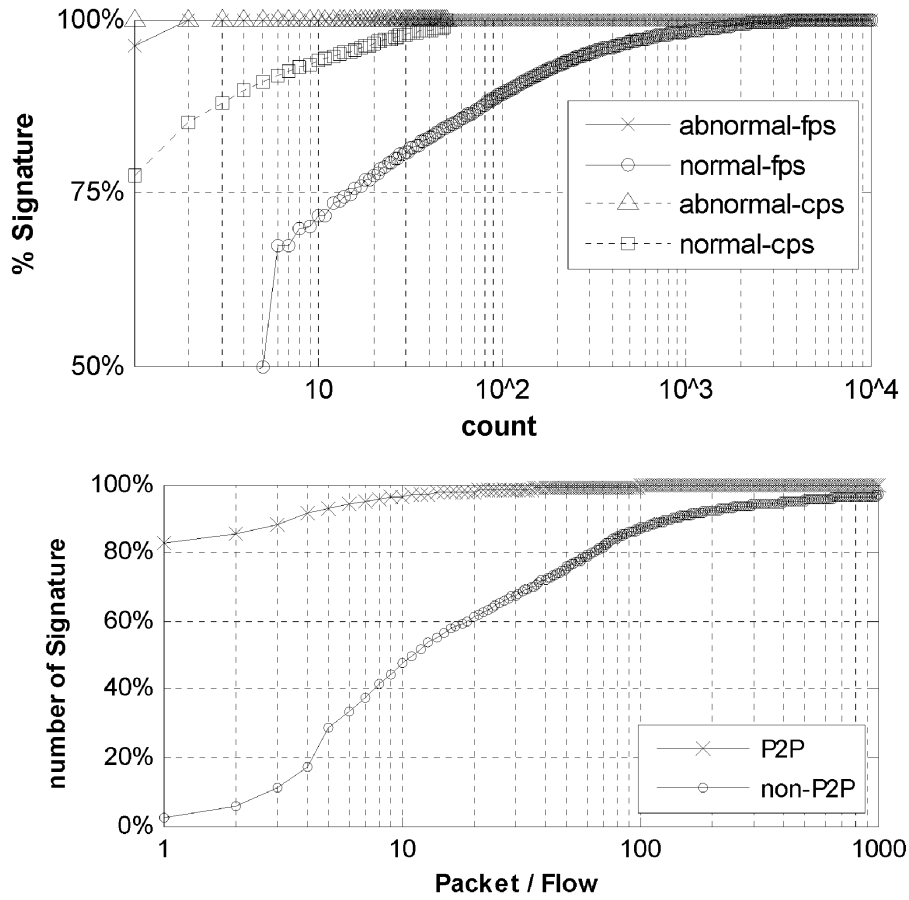
도면6

```
Algorithm 2. identifyTraffic
Input: Traw={t1,...,tk}, HS={hs1,...,hsm}
Output: Tidentified={t1,...,tk}
void identifyTraffic (...){
    for( i=1; i≤k ; i++){
hstemp=getServer(ti);
        if(code =foundHS(hstemp))
          ti.setCode(code);
   }
}
```

도면7

```
Algorithm 3. maintainSignature
1:  Input: HSbefore={hs1,...,hsm}
2:  Output: HSafter={hs1,⋯,hsn}, m≥n
3:  void maintainSignature (...){
4:  for( i=1; i≤m ; i++){
5:  if(isAbnormal(hsi)||isTimeout(hsi))
6:  if(!isDominant(hsi)&&!isPopular(hsi))
7:  HS=HS-{hsi};
8:  }
9:  }
```

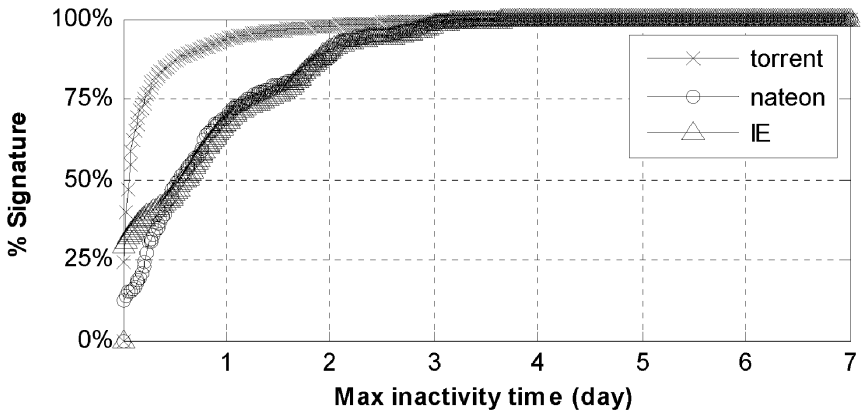
도면8


100%
75%
50%
% Signature
10
10^2
10^3
10^4
count
abnormal-fps
normal-fps
abnormal-cps
normal-cps
100%
80%
60%
40%
20%
0%
number of Signature
1
10
100
1000
Packet / Flow
P2P
non-P2P

도면9

```
Algorithm 3-1 isAbnormal()
1:   Input: hs
2:   Output:True or False
3:   bool isAbnormal (...){
4:   // check abnormal
5:        if(getIdentfiedFlow(hs)==1)     //one-way flow
6:   if (getIdentifiedClient(hs) ==1)    //exclusive
7:             return True;
8:   // check P2P
9:   if (getIdentifiedPacket(hs)
     /getIdentifiedFlow(hs)==1);
10:       return True;
11:    return False;
12:  }
```

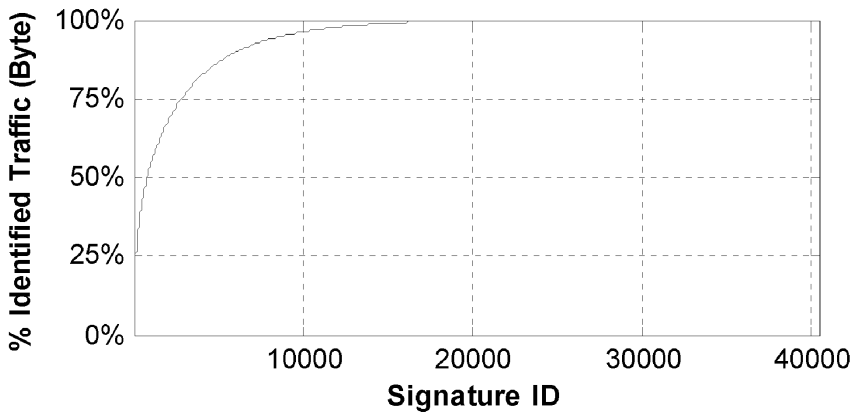
도면10



도면11

```
Algorithm 3-2. isTimeout()
1:   Input: hs
2:   Output:True or False
3:   bool isTimeout (...){
4:   application = getApp(hs));
5:   inactivityTime=currentTime-getLastTime(hs);
6:   if(inactivityTime
       >getMaxInactivityTime(application)*α)
7:           return True;
8:       return False;
9:   }
```

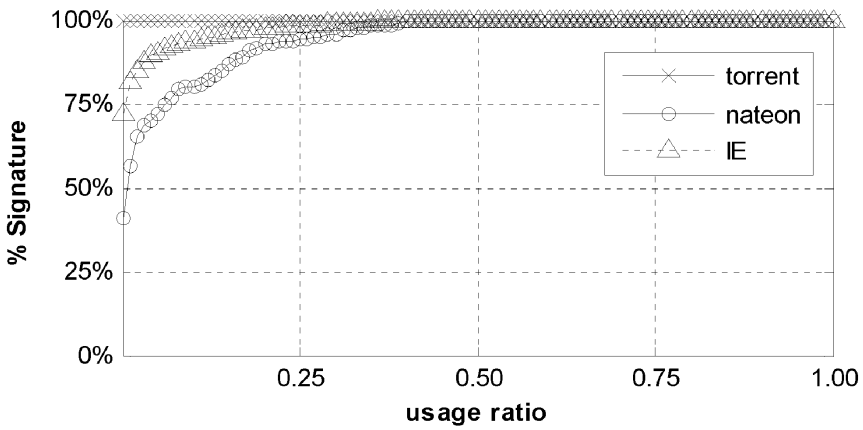
도면12



도면13

```
Algorithm 3-3. isDominant()

1:   Input: hs

2:   Output:True or False

3:   bool isDominant (...){

4:        for( i=0; i<2 ; i++){

5:   // i means 0:size 1:duration

6:   application = getApp(hs));

7:            mean = getMean(application, i);

8:   std = getStd(application, i);

9:   if (getProperty(hs,i) >mean+std*3)

10:             return True;

11:    }

12:    return False;

13:  }
```

도면14



도면15

```
Algorithm 3-4. isPopular()
1:   Input: hs
2:   Output:True or False
3:   bool isPopular (...){
4:   application = getApp(hs));
5:   appClient=getAppClient(application);
6:        if(getClient(hs)/appClient >β)
7:   return True;
8:        return False
9:   }
```