한국통신학회논문지 '13-08 Vol.38B No.08
http://dx.doi.org/10.7840/kics.2013.38B.8.615

페이로드 시그니쳐 자동 생성 시스템

박 철 신◆, 박 준 상*, 김 명 섭°

Automatic Payload Signature Generation System

Cheol-Shin Park◆, Jun-Sang Park*, Myung-Sup Kim°

요 약

페이로드 시그니쳐 기반 분석 방법에서 정확한 시그니쳐는 분석 성능을 높이는데 있어 필수적이다. 하지만 정확한 시그니쳐를 생성하기 위한 수동생성 방법에는 한계가 있다. 따라서 이를 극복 하기 위해 페이로드 시그니쳐를 자동생성하기 위한 페이로드 시그니쳐 자동 생성 시스템을 제안한다. 또한 프로토콜 필터를 이용한 응용의 프로토콜 인식을 통해 시그니쳐 자동 생성의 효율성을 향상 시키고, 프로토콜 별 응용의 페이로드 시그니쳐를 자동 생성하여 세분화된 분석에 적용 할 수 있는 페이로드 시그니쳐 자동 생성 방법을 제안하였다. 본 논문에서 제안한 시스템의 타당성을 검증하기 위해 수동 생성 시그니쳐와 자동 생성 시그니쳐의 비교 및 프로토콜 별 자동 생성 시그니쳐를 통해 시스템의 타당성을 보였다.

Key Words : Automated Signature Generation, Payload Signature, Signature Generation, Traffic analysis, Traffic Classification

ABSTRACT

Fast and accurate signature extraction is essential to improve the performance of the payload signature-based traffic analysis methods. However the slow manual process in extracting signatures make difficult to deal with the rapidly changing application in current Internet environment. Therefore, in this paper we propose a system automatically generating signatures from ground-truth traffic data. In addition, we improve the efficiency of signature extraction by recognizing the application protocol using a protocol filters and generating signatures automatically according to the application-specific protocol contents. In order to verify the validity of the system proposed in this paper, we compared the signatures automatically generated from our system with the signatures manually created for a few popular applications.

Ⅰ. 서 론

인터넷의 발전과 함께 네트워크를 이용하는 응용의 수가 급격하게 늘어가고 있다. 응용의 증가에 따라 네트워크 관리에 있어 응용의 분석은 필수적이며 응용 분석의 성능에 따라 네트워크 트래픽 분석 성능 또한 좌우된다. 따라서 네트워크 관리자는 자신의 네트워크를 효율적으로 운영하기 위해 다양한 방식의 네트워크 트래픽 분석 기법을 도입하여 사용하고 있으며, 이 중 하나로 포트기반의 분석 방법과 시그니쳐 기반의 분석 방법이 이용되고 있다.

포트기반 분석 방법은 IANA[1]에 정의된 Well-known Port기반의 네트워크 프로토콜을 사용하는 응용 인식에 효과 적이다. 또한 빠른 인식 속도를

※ 이 논문은 정부(교육과학기술부)의 재원으로 2010년도 한국연구재단-차세대정보컴퓨팅기술개발사업(20100020728) 및 2012년도 한국연구재단(2012R1A1A2007483)의 지원을 받아 수행된 연구임.
◆ 주저자 : 고려대학교 컴퓨터정보학과 네트워크 관리 연구실, iandyoy@korea.ac.kr, 준회원
° 교신저자 : 고려대학교 컴퓨터정보학과 네트워크 관리 연구실, tmskim@korea.ac.kr, 종신회원
* 고려대학교 컴퓨터정보학과 네트워크 관리 연구실, junsang_park@korea.ac.kr, 학생회원
논문번호 : KICS2013-03-128, 접수일자 : 2013년 3월 7일, 최종논문접수일자 : 2013년 7월 23일

확보 할 수 있는 장점을 가지고 있어 대용량 네트워크에서 현재 까지도 널리 사용되는 방식의 하나이다. 하지만 포트기반의 응용 분석은 과거의 단순한 형태로 하나의 응용 별 단일 프로토콜을 사용하는 응용 분석에 적합 하다. 현대의 Passive FTP와 같이 단일 프로토콜 하에 둘 이 상의 포트를 사용 하는 응용이나 혹은 방화벽을 피하기 위해 Well-known Port를 활용하는 응용, 임의의 포트를 설정 할 수 있는 기능을 제공하는 응용, 다양한 프로토콜을 하나의 응용에서 이용하는 멀티미디어 응용 등 현대의 복잡한 구조를 갖는 응용에 대해서는 높은 신뢰성을 확보하기 힘들다. 따라서 이런 한계를 극복하고 보다 높은 분석 성능을 확보 하기 위한 방법의 하나로 시그니쳐를 이용한 시그니쳐 기반 페이로드 분석 방법이 대표적으로 이용되고 있다. 기계학습 기반 분석 방법은 트래픽의 특징 값을 이용해 페이로드 기반 분석의 한계인 암호화 트래픽의 분석이 가능 하지만 논문에서 제안하는 동일한 응용 계층 프로토콜을 이용하는 응용에 대해서는 분류가 어려운 문제가 발생된다.

페이로드 분석 방법은 패킷의 페이로드 내에 응용만의 특징을 갖는 의미 있는 특징 값을 추출하여 응용을 식별할 수 있는 시그니쳐로 정의 한 후 패킷의 페이로드 내에 응용의 시그니쳐 포함 여부를 판단하여 응용을 분석하는 방법이다. 페이로드 기반 분석 방법은 분석률과 정확도 측면에서 가장 높은 분석 성능을 보이기 때문에 트래픽 분석 방법 중 현재까지 가장 광범위 하게 이용되고 있다. 하지만 페이로드 시그니쳐 기반 분석 방법에 있어 높은 정확도의 시그니쳐를 생성하는 것에 다음과 같은 단점을 가지고 있다. 페이로드 시그니쳐의 수동 생성 시 해당 응용의 프로토콜을 분석할 수 있는 전문성 확보 및 응용의 출현, 변화 등의 대처가 힘들다. 이러한 페이로드 시그니쳐 기반 분석의 한계점이 있음에도 불구하고 분석률과 정확도 측면에서 큰 장점으로 작용하기 때문에 단점을 극복하기 위한 많은 연구들이 진행 되고 있다.

본 논문에서는 페이로드 시그니쳐의 수동 추출에 한계 점을 극복하기 위해 프로토콜 필터 기반 페이로드 시그니쳐 자동 생성 시스템을 제안 하고자 한다. 또한 이미 연구된 다양한 자동 생성 시스템들의 한계점으로 지적할 수 있는 하나의 응용에서 다중 프로토콜 사용시 추출 효율 문제를 프로토콜 필터를 통해 극복 할 수 있는 방법을 제안 한다. 마지막으로 최종 추출 된 시그니쳐를 통해 좀더 세분화 된 트래픽 분석의 가능성을 제시 하고, 다양한 방법으로 수집된 트래픽을 통해 시스템의 활용성을 높일 수 있는 방법을 제안한다.

본 논문은 다음과 같은 순서로 기술한다. 2장에서는 다양한 페이로드 시그니쳐 생성 방법들에 대해 살펴보고, 3장에서는 페이로드 시그니쳐 자동 생성 시스템에 대해 정의하고 설명한다. 4장에서는 프로토콜 필터의 정의와 구조에 대해 설명한다. 5장에서는 제안한 자동 생성 시스템의 타당성을 증명하기 위한 실험 결과를 기술한다. 마지막으로 6장에서는 결론 및 향후 연구에 대하여 기술한다.

Ⅱ. 관련 연구

자동화된 시그니쳐 기반 분석은 주로 웜 감지를 위한 Intrusion Detection System(IDS)에 사용되어 왔다. Scheirer[2]의 연구에서는 웜 시그니쳐 생성을 위한 방법으로 웜의 공통적 특징을 나타내는 Instruction code와 같은 Hex값을 이용하여 특정 위치(Break point)가 나타날 때까지 여러 가지 슬라이딩 윈도우(Sliding-window)알고리즘을 사용하여 비트 패턴을 찾아 내는 방식을 제안 했다. 하지만 페이로드 시그니쳐 생성에서는 특정 비트 패턴을 갖는 Break point가 존재 하지 않으므로 이 방법을 그대로 적용 하기 힘들다. 따라서 이 방법을 응용하여 다양한 응용을 분석할 수 있는 페이로드 시그니쳐 생성 방법이 제안 되었다.

TS Choi[3]의 연구에서는 높은 성능의 시그니쳐 기반 분석 방법을 제시하였으나 시그니쳐를 수동으로 추출 해야 하는 한계를 가지고 있다. 이러한 문제점을 극복하기 위해 여러 논문[4-7]에서는 페이로드 시그니쳐를 정의 하고 시그니쳐를 자동으로 생성 할 수 있는 시스템을 정의 하였다. Mingjiang Ye[5]이 논문에서는 Single이라는 단위의 서브 스트링을 정의하고 Single의 발생 빈도수 및 임계값을 이용하여 최종 공통 스트링을 추출하는 시그니쳐 자동 추출 시스템을 제안 하였다. Cheng[6]의 논문에는 페이로드를 3종류의 비트(1bit, 4bit, 8bit) 단위로 자른 후 같은 오프셋에서 두 비트 시퀀스 간의 공통 비트 시퀀스를 찾아 시그니쳐로 활용하였다. Szabó[7]의 논문에서는 본 논문에서 사용되고 있는 바이오 인포매틱스의 유전자 분석 알고리즘 기법의 하나인 LCS(Longest Common Subsequence)와 유사한 형태의 Motif finding and multiple sequence alignment 기법을 이용 하였다. LCS와 마찬가지로 가장 큰 커버리지를 갖는 가장

작은 공통 스트링을 추출 하여 이것을 응용의 시그니쳐로 사용 하는 방법을 제안하였다.

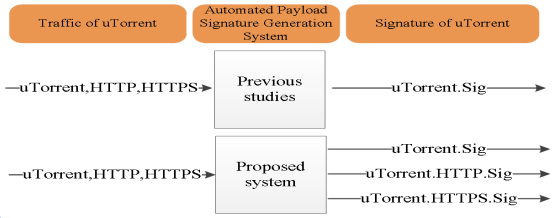

Fig. 1. Conceptual diagram of the automatic signature generation of multi-protocol-based application

위에서 언급한 논문들은 모두 하나의 프로토콜을 하나의 분류 단위로 정의 하고 이 분류 단위를 바탕으로 응용 단위의 시그니쳐를 추출 하는 것에 초점을 맞추고 있다. 최근 응용의 트래픽 변화를 보면 Figure 1의 uTorrent의 트래픽 구성과 같이 하나의 응용에서 uTorrent, HTTP, HTTPS와 같이 서로 다른 프로토콜을 사용하는 응용이 증가 하고 있다. 선행연구에서 제안되었던 시그니쳐 자동생성 시스템은 한 응용에서 다양한 프로토콜을 이용하는 응용에 대해서 다양한 프로토콜의 트래픽 특성을 고려하지 않고 하나의 프로토콜에서 발생된 트래픽으로 정의한 후 시그니쳐 생성을 위한 시퀀스로 나열하고 이를 통해 공통 서브 스트링을 찾아 내는 방식을 이용한다. 하지만 이런 경우 각 프로토콜에 의해 발생된 트래픽 특성의 변화에 따라 공통 스트링의 발생 빈도가 줄어 들어 추출 효율의 문제가 발생될 수 있으며 또한 발생 빈도가 빈번한 프로토콜의 키워드가 시그니쳐로 추출 될 확률이 높아지는 문제가 있다. 따라서 본 논문에서는 이런 한계를 극복하기 위해 프로토콜 필터를 이용한 트래픽 그룹핑을 통해 하나의 응용에서 발생되는 다양한 프로토콜 별 응용 시그니쳐를 자동 생성 할 수 있는 페이로드 시그니쳐 자동생성 시스템을 제안한다. 또한 자동생성 시스템에 의해 생성된 시그니쳐와 수동생성 시그니쳐의 비교 및 프로토콜 필터를 적용 한 시그니쳐를 보임으로서 시스템의 타당성을 검증하고, 시그니쳐 추출 대상 트래픽을 다양화 하여 시스템의 활용성을 높일 수 있는 방안을 제시한다.

Ⅲ. 시그니쳐 자동생성 시스템

본 장에서는 프로토콜 필터 그룹핑 방법론을 적용한 페이로드 시그니쳐 자동생성 시스템을 소개 한다. 제안하는 시스템은 Figure 2에서 보여지는 바와 같이 총 5개의 모듈로 하나의 프레임워크를 이룬다. 각 모듈은 독립 프로세서로 운영 될 수 있으며 각 단계별 출력을 독립적으로 이용 할 수 있도록 구성되어 있다. 각 모듈의 이름과 주요 역할은 다음과 같다.

- AGT (Application Ground Truth):응용별 정답지 생성
- FTG (Flow Type Grouping):그룹핑 알고리즘을 이용한 트래픽의 기능별 분류
- SGE (Signature Extractor):공통 스트링을 갖는 응용 시스니쳐 추출
- FFS (Flow Filter by Signature):시그니쳐 발생 플로우 제거
- SGV (Signature Verifiers):생성된 시그니쳐 검증 및 분석

이렇게 독립적으로 실행 가능한 모듈의 구성은 각 단계의 분업을 가능하게 하고 각 기능별 출력 결과를 통해 좀더 상세한 검증이 가능 하다는 장점을 가지고 있다.

3.1. AGT (Application Ground Truth)

시그니쳐를 생성하고 생성된 시그니쳐의 검증을 위한 정답지 트래픽을 생성하는 모듈이다. 입력은 응용에서 발생된 트래픽 트레이스와 응용의 이름을 기본으로 한다. 본 논문에서는 KU-MON에서 수집된 양방향의 트래픽 트레이스와, 정답지 생성을 위해 각 호스트에 설치된 TMA(Traffic Measurement Agent)로부터 발생된 트래픽의 응용정보를 수집하는 TMS(Traffic Measurement Server)정보를 주로 활용하였다[4]. 또한 시스템의 활용도를 높이기 위해 트래픽 분석에 광범위하게 사용되고 있는 Wireshark[8]나 Microsoft Network Monitor[9]의 트래픽 포멧(pcap,cap)을 지원하도록 하였다. 2장에서 언급된 선행 연구의 다양한 자동생성 시스템은 해당 시스템의 자체 포멧을 이용하거나, 공통적으로 사용되는 PCap[10] 형식만 지원하기 때문에 활용성 측면에 있어 한계가 있다. 다양한 트래픽 포멧의 지원은 페이로드 시그니쳐 자동생성 시스템이 특정 트래픽에 종속되는 것을 방지할 수 있으며 시스템의 활용에 있어 필수적이다. 이렇게 다양한 입력을 바탕으로 응용의 이름 별로 구분된 플로우 집합을 생성한다. 이 집합은 FTG, FFS, SGV 모듈에서 입력으로 사용된다.

3.2. FTG (Flow Type Grouping)

페이로드 시그니쳐 자동 생성 시스템에서 가장 핵심 모듈로서 정답지 트래픽을 시그니쳐가 효과적으로 추출될 수 있는 그룹으로 세분화하는 그룹핑을 수행

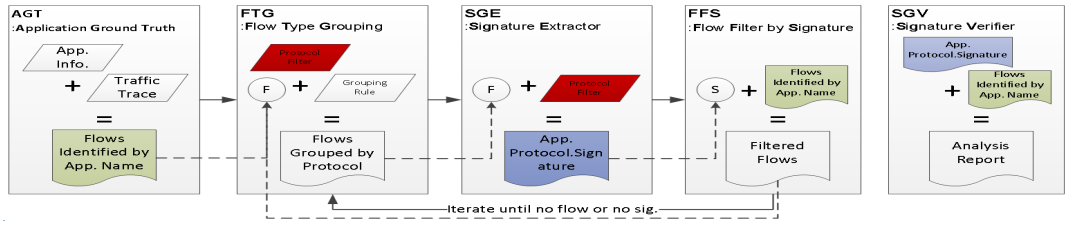

Fig. 2. Automated Payload Signature Generation Framework Structure

한다. 그룹핑의 성능에 따라 페이로드 시그니쳐 자동 생성 시스템의 성능을 좌우 한다고 해도 과언이 아니다. 본 논문에서 제안한 프로토콜 기반의 페이로드 시그니쳐 생성을 위해 다음과 같은 방법을 이용한다. Figure 3과 같이 한 응용의 트래픽을 3~6단계의 각 그룹핑 과정을 거치도록 하였다. 각 그룹핑 단계를 살펴 보면 선행 논문[4]에서 제안되었던 1~4단계의 그룹핑 알고리즘을 다음과 같이 적용 하였다. 한 응용의 정답지 트래픽을 TCP와 UDP 플로우로 분류한 후 TCP 플로우에 대하여 SSIP(Same Server IP Port), PSD(Packet Size Distribution) 그룹핑 방법을 적용한다. 또한 SSIP그룹핑의 임계 값(TH)을 넘지 못하는 플로우에 대하여 HTTP 그룹핑을 하였다. 하지만 이렇게 분류 한다 하더라도 분류된 트래픽에서 특정 프로토콜의 키워드가 시그니쳐로 추출 되는 한계가 있다. 따라서 본 논문에서는 이 단계를 6단계로 세분화 하였고 5~6단계의 과정을 프로토콜 필터라 정의하였다. 플로우 그룹핑 과정에서 프로토콜의 특징을 적용한 필터를 사용함으로써 좀더 명확하고 세분화된 트레이스를LCS기반의 SGE(Signature Extractor)의 입력으로 사용 하수 있도록 하였다. 이렇게 세분화된 트래픽 그룹핑은 선행연구[4-7]에서 제시된 시그니쳐 보다 세분화된 응용 별 프로토콜 시그니쳐를 생성 할 수 있는 장점을 가질 수 있다.

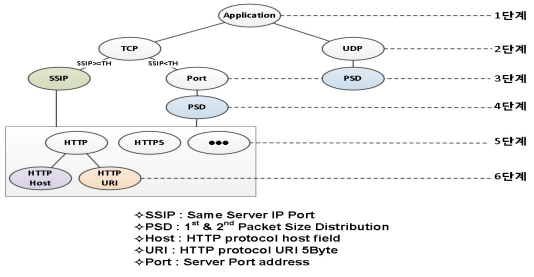

Fig. 3. Flow Grouping Diagram

3.3. SGE (Signature Extractor)

LCS 알고리즘 기반으로 트래픽으로 부터 공통 서브스트링을 추출하기 위한 모듈이다. LCS기반의 추출 시스템의 약점으로 지적되고 있는 계산 복잡도 문제는 풀 수 없는 한계점으로 지적되고 있다. 이 문제는 실제로 시그니쳐를 생성할 때 대량의 트래픽을 순차적 입력으로 사용 할 경우 최종 생성된 시그니쳐의 결과를 확인하는데 굉장히 많은 시간이 소요 되게 한다. 하지만 본 논문에서는 이 문제에 대해 성능을 향상 시킬 수 있는 방법을 프로토콜 필터에서 찾고자 한다. 프로토콜 필터를 적용 할 경우 LCS의 입력으로 사용되는 트래픽에 대해 필터를 통해 시그니쳐 발생 확률이 낮은 부분을 제거 하고 다시 필드 추출 과정을 거쳐 LCS의 입력으로 이용되는 데이터의 양을 줄여 시그니쳐의 확인을 빠르게 하였다. 이를 통해 계산복잡도의 문제를 체감 속도를 빠르게 함으로서 단점을 보완 할 수 있다.

3.4. FFS (Flow Filter by Signature)

이 모듈은 입력으로 SGE(Signature Extractor)에서 생성된 시그니쳐와 FTG(Flow Type Grouping)의 트래픽 트레이스를 이용한다. SGE모듈에서 생성된 시그니쳐를 이용하여 그룹핑된 트래픽에서 시그니쳐를 통해 분석 할 수 있는 모든 플로우를 제거 한다. 이런 기능을 바탕으로 SGE모듈과 함께 사용 함으로서 시그니쳐 생성에서 제외된 플로우를 바탕으로 시그니쳐를 추가 생성 할 수 있도록 하였다. 시그니쳐가 생성되지 않거나, 더 이상 시그니쳐를 생성할 플로우가 존재 하지 않을 때까지 반복 수행 하여 하나의 응용에서 추출 가능 한 시그니쳐를 모두 생성해 낼 수 있게 하는데 목적이 있다. 또한 SGV 모듈과 함께 사용하여 여러 시그니쳐가 동일 플로우를 분석 할 경우 중복 분석되는 시그니쳐를 제거 할 수 있도록 한다.

3.5. SGV (Signature Verifiers)

이 모듈은 위에서 언급한 다른 모듈과는 달리 추출의 목적이 아닌 생성된 시그니쳐의 검증에 목적을 두고 있다. 이를 위해 정답지 트래픽에서 자동생성 시스템에 의해 생성된 시그니쳐를 통해 얼마나 분석 될 수 있는 지를 확인 할 수 있는 분석 보고서를 제공한다. 보고서에는 분석율과 오분류율을 각각 Flow count,

Packet count, Byte Size로 실제 값과 백분율로 표기 한다. 이를 통해 생성된 시그니쳐의 정확도를 확인 할 수 있도록 하여 사용자가 시그니쳐의 활용 여부를 생성 시점에서 즉시 결정 할 수 있도록 하였다.

Ⅳ. 시그니쳐 자동생성 시스템

최근 들어 하나의 응용에서 발생되는 트래픽 특성이 다양한 프로토콜을 사용하는 형태로 변해 가고 있으며 이에 따라서 트래픽 분류 체계 또한 다차원 적인 방법으로 분류 할 수 있는 방법[11]들이 연구 되고 있다. 이런 상황에서 하나의 응용에 대한 시그니쳐를 생성할 때 어떤 프로토콜을 사용하는지를 파악하고 이를 이용해 프로토콜 별로 정확한 시그니쳐를 생성하는 것은 시그니쳐 자동생성 시스템에서 피할 수 없는 조건이다. 따라서 본 장에서는 본 논문에서 제안한 페이로드 시그니쳐 자동생성 프레임 워크에 주요 부분인 프로토콜 필터의 정의 및 적용 방법에 대해 설명하고자 한다.

프로토콜 필터는 공개된 프로토콜(HTTP, FTP, SMTP, POP3등) 정의 문서를 바탕으로 프로토콜을 인식 할 수 있는 방법을 정의한 모듈이다. 프로토콜 필터는 그룹핑 모듈(FTG)과 추출 모듈(SGE)에서 활용 된다. Figure 4와 같이 각 파트로 구성되어 다음과 같이 각각의 특징을 갖는다.

그룹핑 모듈(FTG)에 적용되는 프로토콜 인식 파트의 필터특징은 프로토콜의 인식에 초점을 맞추고 있으며, 추출 모듈(SGE)에 적용 되는 프로토콜 특징 분석 파트의 필터특징은 공통스트링을 효율적으로 생성하기 위한 방법에 초점을 맞추고 있다.

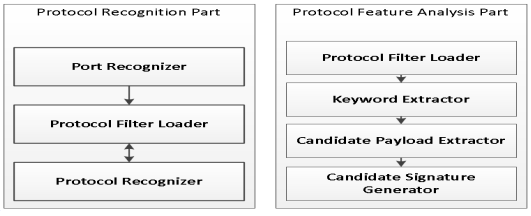

Fig. 4. Configuration of Protocol Filter Parts

4.1. 프로토콜 인식

그룹핑 모듈(FTG)에 입력으로 주어진 플로우가 어떤 프로토콜에 기반하고 있는지를 최대한 빠른 시간 내에 인식 해야 하는 것이 프로토콜 인식 파트의 목적이다. 빠르고 정확한 인식을 위해서는 포트기반 인식 및 페이로드 기반 인식 이렇게 두 가지 요소를 병행하여 사용한다.

입력으로 들어오는 여러 플로우에 대하여 Figure 3에서 보여지는 3단계의 SSIP 그룹핑 정책에 따라 모두 포트 별로 구분된다. 이렇게 분리된 포트 번호를 이용하여 Port Recognizer 는 프로토콜 별 대표 포트 테이블과 맵핑 한다. 포트와 맵핑된 프로토콜 필터가 있는 경우 정확한 인식을 위해 해당 프로토콜 필터를 Protocol Filter Loader가 로드 한 후 로드 된 필터를 이용하여 Protocol Recognizer는 페이로드 검사를 통해 Port Recognizer 가 인식한 프로콜이 맞는지 다시 한번 최종 확인한다. Port Recognizer 에 의해 인식되지 않거나 Protocol Recognizer를 통한 인식이 실패한 경우 빠른 인식을 위해 페이로드 검사를 통한 인식 속도가 가장 빠른 프로토콜을 우선순위로 순차적으로 검사한다.

4.2. 프로토콜 특징 분석

특징 분석 파트는 추출 모듈(SGE)의 입력을 최소화하여 추출 효율을 향상 시키는 것에 목적이 있다.

선행 연구에서 제시한 시그니쳐 자동 생성 시스템에 의해 추출된 시그니쳐는 많은 부분에서 공개 프로토콜 고유의 키워드 들이 포함되어 있다. 이런 키워드 들은 해당 프로토콜을 사용하는 다양한 응용에서 반복적으로 나타날 수 밖에 없는 한계를 가지고 있다. 이렇게 추출된 시그니쳐에서 키워드들을 응용의 시그니쳐로 정의 할 경우 트래픽 분석 시 시그니쳐 충돌이 발생 될 수 있다. 이런 문제점을 프로토콜 특징 분석 파트에서는 다음과 같이 해결한다.

프로토콜의 인식이 끝난 플로우에 대해 Protocol Filter Loader는 해당 프로토콜 필터를 로드하고 Keyword Extractor를 통해 키워드를 인식 시킨다. 프로토콜의 키워드 중 <Keyword> ::= <Value> 또는 <Keyword> <Value> </Keyword> 같은 구조를 가지고 있는 필드에 대해 키워드와 값을 분리 한 후 테이블을 생성한다. 이후 Candidate Payload Extractor를 통해 전체 트래픽 중 시그니쳐 발생 확률이 낮은 부분을 제거 한다. 이렇게 해서 만들어진 최종 테이블을 바탕으로 LCS 기반의 Candidate Signature Generator 는 같은 키워드를 갖는 동일 필드에 대해 값에 해당 되는 내용만을 공통스트링으로 추출 하기 위한 입력으로 사용한다. 이런 과정은 Candidate Signature Generator 의 LCS 계산 복잡도 문제에 따른 추출 속도의 저하 문제를 입력

데이터를 최소화 함으로써 속도 문제를 향상 시키기 위해 필요하다. 또한 키워드 단독으로 시그니쳐로 생성 되는 것을 방지 하기 위해 값을 대상으로 공통스트링 추출 한 후 해당 필드 키워드와 재 조합 하여 최종 추출 시그니쳐를 원래 형태로 복원 후 출력 한다. 이렇게 생성된 시그니쳐는 키워드를 통해 시그니쳐의 발생 위치를 확인 할 수 있는 장점을 가진다. 이런 과정을 HTTP프로토콜에 적용하는 경우 "User-Agent, GET, POST"와 같은 프로토콜의 키워드가 시그니쳐로 생성되는 것을 방지하여 HTTP프로토콜을 이용하는 다른 응용으로 오 분류되는 문제를 해결할 수 있다.

V. 실험 및 성능 평가

본 장에서는 본 논문에서 제안한 페이로드 시그니쳐 자동 생성 시스템의 타당성을 증명하기 위해 수작업으로 추출한 시그니쳐와 자동생성 시그니쳐와의 비교 및 프로토콜 필터를 적용하여 생성된 시그니쳐 생성하여 페이로드 시그니쳐 자동 생성 시스템의 타당성을 증명한다.

Table 1. Traffic Trace of Ground Truth Generated by AGT

Trace Set	Time	Size	Size of Ground Truth	App.
SET 1	13 Hour	35.2GB	1.11GB	127
SET 2	1 Day	90.9GB	1.19GB	155
SET 3	1 Day	82.1GB	919MB	165
SET 4	1 Day	56.8GB	898MB	151

5.1. 실험환경

실험 환경은 두 가지로 나뉜다. 시그니쳐 수동 추출과 자동추출을 비교하기 위해 호스트로부터 직접 수작업으로 수집한 정답지 트래픽을 이용하였고, 프로토콜 필터 적용을 통한 추출 성능 향상 실험을 위해 학내망에 연결된 KU-MON을 통해 자동으로 수집된 트래픽을 사용하였다. Table 1과 같이 수집된 트래픽은 총 265GB이며, AGT(Application Ground Truth)모듈의 입력으로 사용해 총 4.2GB의 정답지 트래픽을 생성하였다. 이를 통해 생성된 응용의 정답지는 1일기준 약 150~160여개의 응용에 총 251개의 응용의 정답지를 생성 하였다.

5.2. 수동 추출 시그니쳐 VS 자동 추출 시그니쳐

이 실험은 전문가에 의해 추출된 응용의 시그니쳐와 자동 추출 된 시그니쳐의 비교를 통해 자동 추출 시스템의 실용성을 검증했다. Table 2의 수동 추출에 의한 시그니쳐는 실제 트래픽 분석 장비에 사용되는 시그니쳐로 게임 사이트에 로그인 할 때 발생되는 트래픽을 분석하여 추출한 시그니쳐 이다. 또한 자동 추출은 본 논문에서 제시한 시스템에 동일한 트래픽 트레이스를 적용하여 최종 생성된 시그니쳐를 나타내고 있다. 수동 추출 시그니쳐는 정확도 100%의 시그니쳐로 자동 추출에 의해 생성된 시그니쳐가 수동 추출에 의해 추출된 시그니쳐보다 다소 복잡하지만 충분히 활용 가능한 시그니쳐를 추출 한 결과를 볼 수 있다. 또한 동일 트래픽을 이용한 분석에서 수동 추출에 의한 분석결과와 동일한 분석률을 나타냈다.

Table 2. Manually VS Automatically generated

App	Manually generated	Automatically generated
AION	Host: aion.plaync.co.kr	Host:aion.plaync.co.kr
	login.plaync.co.kr	login.plaync.com
Cyphers	Host: cyphers.nexon.com	^GET/cyphers/id/login?err=wrong&targetUrl=*HTTP/1.1 Host:cyphers.nexon.com
	user.cyphers.nexon.com	user.cyphers.nexon.com
	/Ajax/Default.aspx?_vb=GetPasswordHashKey&_cs	^GET/Ajax/Default.aspx?_vb=GetPasswordHashKey&_cs=*HTTP/1.1 *.nexon.com*
	Referer: http://cyphers.nexon.com	Host:*cyphers*.co* Referer:http://cyphers.nexon.com/
	Host: auth01.nexon.com	Host:auth01.nexon.com
Blade & Soul	Host: bns.plaync.co.kr	Host:bns.plaync.com
	/common/support/loginLib	^GET/common/support/loginLib*HTTP/1.1
	Host: static.plaync.co.kr	Host: static.plaync.co.kr
	bnslauncher.plaync.com	bnslauncher.plaync.com

5.3. 자동 추출 시그니쳐 VS 프로토콜 필터 적용

이 실험은 선행연구에서 제시된 자동 추출 시그니쳐와 본 논문에서 제안된 프로토콜 필터를 적용한 자동 생성 시그니쳐의 비교를 통해 프로토콜 필터를 적용한 자동 추출 시스템의 성능 향상 및 제안한 프로토콜 필터의 타당성을 보이기 위한 실험이다. Table 3은 제안한 시스템에 HTTP 프로토콜 필터를 적용한 결과를 선행연구의 결과 비교표 이다. 추출 결과를 보면 국내에서 대표적으로 사용되는 메신저인 nateon과 대표적인 p2p응용인 utorrent의 경우 자체 프로토콜과 HTTP프로토콜을 병행하여 사용 하고 있는 것을 추출된 시그니쳐를 통해 알 수 있다. 선행 연구의 두 응용

Table 3. Signatures Generated by Automated Payload Signature Generation Framework

Previous studies		Proposed system	
App	Signature	App	Signature
nateonmain	^NCPT1*0 ^RCON1dpl.nate.com5004 ^GET /exndr.jsp HTTP/1.1 NateOn/4.1.4.0(2010)	nateonmain	^NCPT1*0 ^RCON1dpl.nate.com5004
		nateonmain.http	^GET /exndr.jsp HTTP/1.1 User-Agent:NateOn/4.1.4.0(2010) Host:cyxso.cyworld.com
utorrent	[1]^\x13BitTorrent protocol [1]^GET/@peer_ip@User-Agent: uTorrent/@HTTP1\.1 [1]utorrent\.com uTorrent [1]^\x64\x31\x3A.\x64\x32\x3A\x69\x64\ x32\x30\ x3A@\x3A@\x31\x3A\x79\x31\x3A.\x65	utorrent	^\x13BitTorrent protocol * WHB ^Go away, we're not home ^\x64\x31\x3A.\x64\x32\x3A\x69\x64\x32\x30\x3A@\x3A@\x31\x3A\x79\x31\x3A.\x65 ^*12:complete_agoi*e1:md11:upload_onlyi3e12:ut_holepunchi4e11:ut_metadatai2e6:ut_pexi1ee13:metadata_sizei32374e1:pi21081e4:reqqi255e1:v15:??Torrent2.2.16:yourip4:
		utorrent.http	^GET/announce?info_hash=*&peer_id=-*WHB&port=*&uploaded=*&downloaded=*&left=*&corrupt=*&key=*&numwant=*&compact=*&no_peer_id=*&ipv6=2002%3aa398%3adb15%3a%3aa398%3adb15 HTTP/1.1 Host:*t*.com:2710 User-Agent:uTorrent/2210(25203)

에 대한 시그니쳐의 경우 보여지는 시그니쳐 만으로는 어떤 프로토콜을 이용 하고 있는지 알 수 없을 뿐만 아니라 시그니쳐의 발생 위치를 알 수 없어 다양한 페이로드 시그니쳐 기반 트래픽 분석 시스템에 적용 시 시그니쳐를 재 분석해야 하는 문제가 발생한다. 반면에 제안 시스템의 시그니쳐의 경우 각 응용의 프로토콜별 시그니쳐의 생성을 통해 응용의 프로토콜별 트래픽 분석이 가능하고, HTTP 프로토콜 필터 적용을 통해 추출된 시그니쳐에 대해 HTTP Header필드를 재 조합 함으로써 발생 부분을 명확하게 할 수 있었다. 이를 통해 실제 시그니쳐를 이용한 트래픽 분석에 시그니쳐 적용의 용이성 및 분석의 효율성을 가질 수 있도록 하였다. 마지막으로 추출 실험 결과 페이로드 기반 분석의 한계로 지적된 암호화 트래픽에서는 시그니쳐를 추출 할 수 없었으며 이 것은 트래픽 분석에 있어 단점으로 작용 할 수 있다. 하지만 추출된 시그니쳐를 이용해 분석된 트래픽의 정보를 이용한 IP 상관관계 기반 분석을 활용하여 보완 할 수 있다. 또한 프로토콜 필터를 이용한 시그니쳐 생성으로 추출 대상의 최소화를 통해 개인정보를 배제한 시그니쳐를 생성할 수 있었다.

Ⅵ. 결론 및 향후 과제

본 논문에서는 프로토콜 별 특징을 활용한 페이로드 시그니쳐 자동 생성 프레임 워크를 제안 하였다. 이를 통해 멀티 프로토콜을 사용하는 응용에 대한 시그니쳐 생성 성능을 향상 시킬 수 있는 방법을 제시하였다. 또한 LCS기반의 시그니쳐 자동 생성 시스템의 취약점인 계산 복잡도 문제를 입력데이터의 크기를 최소화 함으로서 체감 속도를 높이는 방안을 제시 하였다. 자동 생성 시스템의 활용성을 확보하기 위해서 다양한 응용을 통해 수집된 트래픽을 이용하여 시그니쳐를 생성 할 수 있도록 하였다, 마지막으로 제안된 시스템의 타당성을 검증하기 위한 시그니쳐의 추출 성능을 비교 평가하고 추출된 시그니쳐를 보였다.

향후 연구로써는 더 많은 공개 프로토콜에 대한 확장된 필터를 적용하여 시스템의 실용성을 확보할 수 있는 방법을 연구하고자 한다. 또한 LCS알고리즘의 입력에 대해 “similarity function”을 적용하여 생성 효율을 향상 시킬 수 있는 방법을 연구하고자 한다.

References

[1] IANA, *IANA port number list*, Retrieved 3, 2, 2013, from http://www.iana.org/assignments/port-numbers

[2] W. Scheirer and M. Chuah. *Comparison of three sliding-window based worm signature generation schemes*, Lehigh Univ. Technical Report LU-CSE-05-025, 2005.

[3] T. S. Choi, C. H. Kim, S. H. Yoon, J. S. Park, H. S. Chung, B. J. Lee, H. H. Kim, and T. S. Jeong, “Rate-based internet accounting system

using application-aware traffic measurement," in *Proc. APNOMS 2003*, pp. 404-415, Fukuoka, Japan, Oct. 2003.

[4] J.-S. Park, J.-W. Park, S.-H. Yoon, H.-S. Lee, and M.-S. Kim, "Development of signature generation and update system for application-level traffic classification," *J. KIPS*, vol. 17C, no. 1, pp. 99-108, Feb. 2010.

[5] M. Ye, K. Xu, J. Wu, and H. Po. "AutoSig-automatically generating signatures for applications," in *Proc. IEEE CIT '09*, vol. 2, pp. 104-109, Xiamen, China, Oct. 2009.

[6] C. Mu, X.-H. Huang, X. Tian, Y. Ma, and J.-L. Qi, "Automatic traffic signature extraction based on fixed bit offset algorithm for traffic classification," *J. China Univ. Posts Telecommun.*, vol. 18, no. 2, pp. 79-85, Dec. 2011.

[7] G. Szabó, Z. Turányi, L. Toka, S. Molnár, and A. Santos, "Automatic protocol signature generation framework for deep packet inspection," in *Proc. ICST VALUETOOLS '11*, pp. 291-299, Cachan, France, May 2011.

[8] Wireshark, *Wireshark*, Retrieved 3, 2, 2013, from http://www.wireshark.org/.

[9] Microsoft, *Microsoft Network Monitor 3.4*, Retrieved 3, 2, 2013, from http://www.microsoft.com/en-us/download/details.aspx?id=4865.

[10] 1. TCPDUMP & LiBPCAP, *LiBPCAP*, Retrieved 3, 2, 2013, from http://www.tcpdump.org. 2. WinPcap, *WinPcap*, Retrievd 3, 2, 2013, from http://www.winpcap.org.

[11] J.-H. Kim, S.-H. Yoon, and M.-S. Kim, "Research on traffic taxonomy for internet traffic classification," in *Proc. APNOMS 2011*, pp. 21-23, Taipei, Taiwan, Sep. 2011.

박 철 신 (Cheol-Shin Park)

2007년 고려대학교 컴퓨터 정보학과 졸업

2011년~현재 려대학교 컴퓨터 정보학과 석사과정

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

박 준 상 (Jun-Sang Park)

20008년 고려대학교 컴퓨터 정보학과 졸업

2010년 고려대학교 컴퓨터 정보학과 석사

2010년~현재 고려대학교 컴퓨터 정보학과 박사과정

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

김 명 섭 (Myung-Sup Kim)

1998년 포항공과대학교 전자계산학과 졸업

2000년 포항공과대학교 컴퓨터공학과 석사

2004년 포항공과대학교 컴퓨터공학과 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