ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr☆

LDA를 활용한 네트워크 위협 시그니처 추출기법☆

Extraction of Network Threat Signatures Using Latent Dirichlet Allocation

이 성 일[1] 이 수 철[2] 이 준 락[3]* 염 흥 열[4]*
Sungil Lee Suchul Lee Jun-Rak Lee Heung-youl Youm

요 약

인터넷 웜, 컴퓨터 바이러스 등 네트워크에 위협적인 악성트래픽이 증가하고 있다. 특히 최근에는 지능형 지속 위협 공격 (APT: Advanced Persistent Threat), 랜섬웨어 등 수법이 점차 고도화되고 그 복잡성(Complexity)이 증대되고 있다. 지난 몇 년간 침입탐지시스템(IDS: Intrusion Detection System)은 네트워크 보안 솔루션으로서 중추적 역할을 수행해왔다. 침입탐지시스템의 효과적 활용을 위해서는 탐지규칙(Rule)을 적절히 작성하여야 한다. 탐지규칙은 탐지하고자 하는 악성트래픽의 핵심 시그니처를 포함하며, 시그니처를 포함한 악성트래픽이 침입탐지시스템을 통과할 경우 해당 악성트래픽을 탐지하도록 한다. 그러나 악성트래픽의 핵심 시그니처를 찾는 일은 쉽지 않다. 먼저 악성트래픽에 대한 분석이 선행되어야 하며, 분석결과를 바탕으로 해당 악성트래픽에서만 발견되는 비트패턴을 시그니처로 사용해야 한다. 만약 정상 트래픽에서 흔히 발견되는 비트패턴을 시그니처로 사용하면 수많은 오탐(誤探)을 발생시키게 될 것이다. 본고에서는 네트워크 트래픽을 분석하여 핵심 시그니처를 추출하는 기법을 제안한다. 제안 기법은 LDA(Latent Dirichlet Allocation) 알고리즘을 활용하여, 어떠한 네트워크 트래픽에 포함된 시그니처가 해당 트래픽을 얼마나 대표하는지를 정량화한다. 대표성이 높은 시그니처는 해당 네트워크 트래픽을 탐지할 수 있는 침입탐지시스템의 탐지규칙으로 활용될 수 있다.

☞ 주제어 : LDA, 네트워크 위협탐지, 침입탐지시스템, 시그니처

ABSTRACT

Network threats such as Internet worms and computer viruses have been significantly increasing. In particular, APTs(Advanced Persistent Threats) and ransomwares become clever and complex. IDSes(Intrusion Detection Systems) have performed a key role as information security solutions during last few decades. To use an IDS effectively, IDS rules must be written properly. An IDS rule includes a key signature and is incorporated into an IDS. If so, the network threat containing the signature can be detected by the IDS while it is passing through the IDS. However, it is challenging to find a key signature for a specific network threat. We first need to analyze a network threat rigorously, and write a proper IDS rule based on the analysis result. If we use a signature that is common to benign and/or normal network traffic, we will observe a lot of false alarms. In this paper, we propose a scheme that analyzes a network threat and extracts key signatures corresponding to the threat. Specifically, our proposed scheme quantifies the degree of correspondence between a network threat and a signature using the LDA(Latent Dirichlet Allocation) algorithm. Obviously, a signature that has significant correspondence to the network threat can be utilized as an IDS rule for detection of the threat.

☞ keyword : LDA, network threat detection, intrusion detection system, signature

1. 서 론

지난 10여 년간 스마트폰의 대중화에 따라 모바일 인터넷 시대가 열렸다. 인터넷 사용량이 급격히 증가하였으며, 이에 따라 인터넷 웜, 컴퓨터 바이러스 등 네트워크에 위협적인 악성트래픽 또한 증가하고 있다. 특히, 최근에는 지능형 지속 위협 공격 (APT: Advanced Persistent Threat), 랜섬웨어 등 신종 공격이 소개되며, 그 수법이 점차 고도화되는 등 복잡성(Complexity)이 증대되고 있다.

[1] Infrastructure Protection Division, National Security Research Insititute, Daejon, 34044, Korea
[2] Dept. of Computer Science and Information Engineering, Korea National University of Transportation, Uiwang, Kyunggi, 16106, Korea
[3] Dept. of Humanities and Social Sciences, Kangwon National University, Samcheok, Kangwon, 25913, Korea.
[4] Dept. of Information Security, Soonchunhyang University, Asan, Chungnam, 31538, Korea
* Corresponding author (jrlee@kangwon.ac.kr, hyyoum@sch.ac.kr)
[Received 23 July 2017, Reviewed 8 September 2017, Accepted 17 October 2017]
☆ 이 성과는 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2017R1C1B5017028). 이 연구는 2016년도 강원대학교 대학회계 학술연구조성비로 연구하였음(관리번호-620160030).

대학, 기업체, 정부기관은 보안솔루션으로서 통합 위협 관리시스템(UTM: Universal Threat Management)[2], 웹방화벽(WAF: Web Application Firewall)[3] 혹은 방화벽[1]과 함께 네트워크를 보호하기 위하여 침입탐지시스템(IDS: Intrusion Detection System)을 설치한다. 특히, 지난 몇 년간 침입탐지시스템은 네트워크 보안 솔루션으로서 중추적 역할을 수행해왔다. 침입탐지시스템은 망에 흐르는 악성트래픽을 탐지하고, 로깅(logging)한다. 추가적으로 inline으로 설치되어 네트워크 위협 차단 기능까지 포함하는 경우 침입방지시스템(IPS:Intrusion Prevention System)이라 부른다 [4].

침입탐지시스템은 악성트래픽을 탐지하기 위해 탐지규칙(rule)을 사용한다. 탐지규칙은 침입탐지시스템이 탐지하고 보고하기로 하는 공격의 형태를 미리 정의한 일정한 규칙에 따라 기술한다. 예를 들어, SNORT[5]의 탐지규칙의 형태는 (그림 1)과 같다.

```
alert tcp $HOME_NET any → any any (msg: “example rule”;
content: “FF FE”; sid:1000004; rev: 1;)
```

(그림 1) SNORT 탐지규칙 예시
(Figure 1) SNORT rule example

(그림 1)의 탐지규칙은 시그니처(signature) 문자열 “FF FE”를 포함하는 패킷이 있을 경우 침입탐지시스템으로 하여금 탐지(alert)하도록 한다. 상용 망에 설치되는 침입탐지시스템은 대개 수천 개의 최신 탐지규칙을 탑재하고 있다. 따라서 탐지규칙은 침입탐지시스템이 효과적으로 악성트래픽을 탐지하기 위해서 가장 중요한 역할을 담당한다. 쉽게 유추할 수 있듯 침입탐지시스템의 운영에 있어 가장 중요한 일은 적절한 악성트래픽의 시그니처를 추출하여 탐지규칙을 작성하는 일이다.

그러나 악성트래픽을 탐지하는데 적절한 시그니처를 추출하는 일은 상당히 어려운 일이다. 전통적으로 탐지규칙은 보안전문가 혹은 보안 관리자(CERT: Computer Emergency Response Team)가 악성트래픽을 분석하고 악성트래픽 시그니처를 추출하여 작성하여 왔다. 그렇다보니 작성된 탐지규칙의 적절성 여부는 거의 전적으로 보안 관리자의 노하우에 의존할 수밖에 없다. 그럼에도 불구하고 부적절하게 작성된 탐지규칙이 유발하는 부작용은 상당하다. 예를 들어, 망에서 흔히 발견되는 비트패턴을 탐지규칙의 시그니처로 사용하는 탐지규칙은 침입탐지시스템으로 하여금 정상트래픽을 악성트래픽으로 잘못 탐지하게 한다. 이를 오탐(誤探)이라고 하며, 오탐의 결과로서 침입탐지시스템의 성능저하를 유발하고 심지어는 침입탐지시스템을 설치한 네트워크를 마비시키기도 한다[6].

본고에서는 공격에 대한 사전 지식 없이 악성트래픽을 분석하고 해당 악성트래픽에 대한 핵심 시그니처를 추출하는 기법을 제안한다. 제안 기법은 LDA(Latent Dirichlet Allocation) 알고리즘[9]을 활용하여 네트워크 트래픽에 포함된 시그니처가 해당 트래픽을 얼마나 대표하는지를 정량화(quantification)한다. 대표성이 높은 시그니처는 해당 네트워크 트래픽을 탐지할 수 있는 침입탐지시스템의 탐지규칙으로 활용될 수 있다.

제안 기법의 기본 아이디어는 다음과 같다. 악성트래픽에 대한 핵심 시그니처를 추출함에 있어 다음과 같은 질문에 답할 수 있어야 한다. “악성트래픽에 포함된 수많은 시그니처 중에 어떤 것을 탐지규칙으로 사용할 것인가?” 악성트래픽을 대표할 수 있는 시그니처는 해당 악성트래픽이 수행하는 핵심악성행위에 관련된다. 악성트래픽은 대부분 컴퓨터 프로그램이므로 핵심악성행위 또한 프로그래밍 코드로서 정의된다. 이러한 프로그래밍 코드는 네트워크를 통하여 컴퓨터 간에 전송되므로 네트워크 패킷에 포함된다. 따라서 악성트래픽의 핵심악성행위를 정의하는 코드의 시그니처는 대표성이 높다고 볼 수 있으며 정상/악성 트래픽에서 통상적으로 발견되는 시그니처는 대표성이 낮다고 볼 수 있다. 이렇듯, 트래픽과 시그니처의 연관의 정도(the degree of correspondence)를 정량화 할 수 있는 알고리즘이 LDA다. 일반적으로 LDA는 자연어처리(NLP: Natural Language Processing) 분야에 활용되어 왔던 기술로 문서를 구성하는 단어의 관계에 대한 숨겨진 구조(hidden structure)를 모델링하는데 활용되어 왔다.

본고의 구성은 다음과 같다. 2장에서는 관련연구를 요약하고 3장에서는 제안기법을 이해하기 위한 LDA 관련 배경지식을 간략히 설명한다. 4장에서는 제안 기법을 기술하고, 5장에서 제안 기법의 성능을 평가한다. 마지막으로 6장에서 결론을 맺는다.

2. 관련 연구 동향

지난 10여 년간 학계에서는 탐지규칙 시그니처 추출과 관련한 연구가 진행되어왔다. Polygraph [10]는 다형성 인터넷 웜에 대응하기 위해 복수개의 불변의 콘텐츠 문자열을 결합하여 시그니처를 생성한다. Polygraph의 우수한 성능이 입증되었음에도 불구하고 Roberto의 연구 [11]는

공격자 입장에서 의도적으로 공격트래픽에 노이즈를 포함함으로써(일명: 데이터 오염 공격), Polygraph와 같은 탐지규칙 시그니처 추출 알고리즘을 무력화 할 수 있음을 보였다. Honeycyber [12]에서는 시그니처 추출을 위해서 LCS: Longest Common Subsequence 알고리즘을 사용한다.

추출한 시그니처의 탐지규칙으로서의 효용성을 정량화하기 위한 방법이 몇 가지 제안되었다. PCA(Principal Component Analysis) [13] 등 통계적인 방법을 활용할 수 있다. 예컨대, [14]에서는 확률론에 기반을 둔 탐지규칙 시그니처 효용성 정량화 휴리스틱(Heuristic) 알고리즘을 제안하였으며, [15][16]에서는 엔트로피(Entropy)를 기반으로 탐지규칙 효용성을 정량화 하였다.

탐지규칙 시그니처 관련 연구의 다른 갈래는 취약점 시그니처(Vulnerability signature)를 사용하는 것이다 [17]. 취약점 시그니처는 취약점이 악용(exploit) 될 수 있는 가능한 모든 방법들에 대한 의미론적인 정보*를 탐지규칙 시그니처로 활용하는 것이다. 이를 위해서는 악성프로그램에 대한 실행명령어 수준의 이해가 수반되어야 한다. 취약점 시그니처는 기존의 탐지규칙 시그니처처럼 악성트래픽에 대한 시그니처라기 보다는 해당 악성트래픽이 악용하는 취약점에 대한 시그니처로 높은 탐지 정확도를 가진다. 다만, 취약점이 악용될 수 있는 경로는 무한하여 이를 시그니처로 활용하기 위해서는 튜링머신(Turing Machine)을 필요로 한다. 튜링머신 탐지규칙을 생성하는 것은 컴퓨터과학에서 NP완비로 불리는 극한의 난제이다. 따라서 대부분의 경우 정규식 기반 시그니처 (PCRE: Perl Compatible Regular Expression)을 활용하게 되는데 이는 시그니처 기반 탐지규칙을 여러 개 합쳐 놓은 것과 수학적으로 동일하다.

본 연구와 가장 밀접한 연구는 LARGen[6]이다. LARGen에서는 LDA알고리즘을 활용하여 탐지규칙을 자동 생성하는 GUI기반 시스템을 제안하였다. LDA는 자연어처리 분야에서 PLSA(Probabilistic Latent Semantic Analysis)[7]와 같이 토픽모델링(Topic Model)[8]기법으로 널리 활용되어 왔다. 탐지규칙 시그니처 자동생성 분야에 LDA기법을 활용한 경우는 LARGen이 처음이다. LDA는 텍스트를 특징(Feature)으로 하는 자율(Unsupervised) 기계학습방법으로 LARGen에서는 네트워크 패킷이 이진법으로 기술된 텍스트라는 점에 착안, LDA알고리즘을 적용하여 탐지규칙 자동생성 분야에서 LDA의 활용가능성을 입증하였다. 그러나 정확도 측면**에서의 LDA의 우수성은 널리 알려져 있으나 알고리즘의 복잡도가 높은 편이어서 시급성을 다투는 보안대응(Response)분야에서는 알고리즘 수행성능이 문제가 되는 경우가 많다. [6]에서는 이를 해결하기 위해 출력 가능한(printable) ASCII문자열만을 고려하여 LDA알고리즘을 적용하였다. 그러나 네트워크 위협 시그니처는 출력가능하지 않은 HEX문자열을 포함한다. 본 연구는 LARGen의 후속연구로서 LDA수행성능을 고려하고 전체 ASCII문자셋을 시그니처로 추출할 수 있는 새로운 전집(corpus)구성기법을 적용하였다.

* 악성 프로그램의 모든 실행경로를 표현한 유한상태기계 (FSM: Finite State Machine)

** [6]에서 정확도라고 함은 오탐율(False Positive)을 의미함.

3. Latent Dirichlet Allocation

3.1 이론적 배경

LDA알고리즘[9]은 일반 텍스트문서들을 유사한 주제(topic)를 가진 군집(cluster)으로 클러스터링하기 위한 방법으로 활용되어 왔다. 예컨대, 포털 사이트 검색창에 검색어를 입력하면 관련된 인터넷 페이지가 검색되는데 이때 검색결과는 검색어와 관련된 주제를 다루는 인터넷 문서의 클러스터라고 볼 수 있다. 여기서는 제안기법을 설명하기에 앞서 LDA알고리즘의 이론적 배경을 간략히 살펴본다.

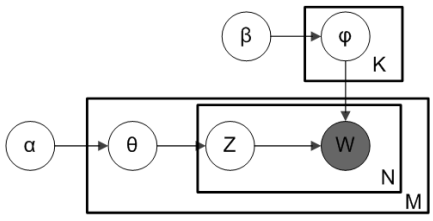
LDA는 확률적 문서 생성 모델(Probabilistic Document Generative Model)이다. 확률적 문서 생성 모델에서 기본이 되는 가정은 일반적으로 문서는 여러 가지 주제를 함축할 수 있다는 것이다. 예컨대 본 논문의 주제는 “침입탐지시스템”, “탐지규칙”, “시그니처” 등등 이다. 이를 수학적으로 정의하면 다음과 같다.

- 단어(word) w는 이산 자료의 기본단위로 단어집(vocabulary) $V=\{1,2,3,...,v\}$의 인덱스로 나타낼 수 있다.
- 문서 $i=\{w_1,w_2,\cdots,w_{N_i}\}$는 N_i개의 단어의 집합으로 w_{N_i}는 문서 i의 N_i번째 단어를 의미한다.
- 전집 $D=\{1,2,3,...,M\}$는 M개의 문서의 집합을 의미한다.

LDA에서 문서는 아래의 확률적 문서 생성 과정에(Probabilistic Document Generative Process)에 따라 생성된다고 가정한다.

- $\theta_i \sim Dir(\alpha)$를 선택한다. 여기서 θ_i는 문서 i의 주제의 분포이며 $Dir(\alpha)$는 인자 α에 대한 디리클레분포를 의미한다.
- $\phi_k \sim Dir(\beta)$를 선택한다. 여기서 ϕ_k는 주제 k의 단어의 분포이며 $Dir(\beta)$는 인자 β에 대한 디리클레분포를 의미한다.
- 단어 $w_{i,j}, j \in \{1,...,N_i\}, i \in \{1,...,M\}$의 주제 $z_{i,j}$와 단어 $w_{i,j}$는 다음과 같은 생성 과정을 따른다.
  - $z_{i,j} \sim Multinomial(\theta_i)$를 선택한다.
  - $w_{i,j} \sim Multinomial(\phi_{z_{i,j}})$를 선택한다.

LDA 생성모델은 (그림 2)와 같이 나타낼 수 있다.



(그림 2) LDA 확률적 문서 생성 모델
(Figure 2) LDA Probabilistic document generative model

3.2 확률분포의 추론

LDA의 목적은 문서에 함축된 주제를 찾는 것이다. "LDA문제를 푼다"고 함은 관찰 가능한 값인 전집 D와 인자값(α, β)과 주제의 개수(K)로부터 관찰 불가능한 확률분포 (θ, ϕ)를 추론(estimation)하는 것이다. 수식으로 표현하면 전집 D가 주어졌을 때 숨겨진 변수(hidden variable)에 대한 사후분포(posterior distribution)를 구하는 것이다.

$$p(\theta, \phi | D, \alpha, \beta) = \frac{p(\theta, \phi, D | \alpha, \beta)}{p(D | \alpha, \beta)} \qquad (1)$$

이 문제는 NP-완비문제로 알려져 있다. 그러나 근사해를 구할 수 있는 방법들이 몇 가지 학계에 소개되었다. 특히, EM(Expectation Maximization) 알고리즘을 활용하는 Gibbs샘플링방법[19]이 자주 사용된다. Gibbs샘플링을 통한 확률분포의 추론은 [19]을 참고하기 바란다.

3.3 시그니처 추출에 대한 LDA알고리즘 활용 근거

탐지규칙 시그니처 추출에 LDA알고리즘을 활용하는 데 대한 주요한 근거는 네트워크 트래픽과 일반 텍스트 문서가 상당히 유사하다는 점이다. 예컨대 네트워크 플로우(flow)는 일반 텍스트문서에 대응시킬 수 있으며, 플로우를 구성하는 이진 문자열은 일반 텍스트문서에서의 단어들로 대응시킬 수 있다. 나아가 네트워크 플로우로 구성된 네트워크 트래픽은 전집에 대응시킬 수 있다.

네트워크 플로우도 여러 개의 주제를 포함할 수 있다. 예컨대 e메일을 통하여 전파되는 멜리사(Melisa)웜을 생각해보자. 멜리사 웜을 전송하는 네트워크 플로우는 SMTP(Simple Mail Transfer Protocol) 등의 e메일 프로토콜을 포함한다. 만약 e메일이 웹 메일이라면, HTTP(Hyper Text Transfer Protocol) 또한 포함한다. 멜리사 웜의 실행이미지 또한 네트워크 플로우를 통해서 전송된다. 따라서 멜리사 웜을 전파하는 e메일은 멜리사 웜, SMTP, HTTP 등에 관련된 주제(프로토콜 또는 실행이미지)를 포함한다고 하겠다.

네트워크 트래픽에서의 주제들은 관련한 이진 문자열을 포함한다. 예컨대 HTTP는 "WWW", "HTTP", "./GET" 등의 이진 문자열을 포함한다. 이 문자열들은 대개 HTTP 트래픽을 분류하거나 탐지할 때의 기준이 된다.

일반화하면 LDA알고리즘이 올바르게 네트워크 트래픽을 클러스터링 한다고 가정하자. 탐지하고자 하는 악성트래픽은 특정 클러스터에 속하게 된다. 해당 클러스터가 가진 여러 개의 주제 중에 "악성트래픽" 주제에 밀접한 관련이 있는 단어(이진 문자열)를 탐지규칙의 시그니처로 활용할 수 있다.

4. LDA기반 탐지규칙 시그니처 추출

4장에서는 제안하는 LDA기반 탐지규칙 시그니처 추출 기법을 기술한다. 네트워크 트래픽에 LDA알고리즘을 적용하기 위해서는 LDA의 입력(input)형태로 네트워크 트래픽을 가공하는 전처리과정이 선행되어야 한다. 따라서 전처리 과정을 먼저 기술하고 제안 기법을 기술한다.

4.1 전처리 과정

4.1.1 전집의 구성

서술편의를 위해 LDA의 입력으로 한 개의 악성 플로우 $F_{악성}$과 $M-1$개의 정상 네트워크 플로우를 가정

하자. 총 M개의 플로우는 $F=\{F_1, F_2, ..., F_M\}$으로 표기한다. 제안 기법의 최종 목표는 악성 플로우 $F_{악성}$을 탐지할 수 있는 시그니처를 추출하는 것이다.

네트워크 트래픽에 LDA알고리즘을 적용하기 위해서는 수식 (1)의 전집 D를 구성해야 한다. (그림 3)은 주어진 네트워크 플로우들의 집합 F로부터 M개의 문서들을 구성하는 과정을 도식한다. 제안 기법에서는 트래픽 데이터 셋에서 64초 동안 발생한 모든 패킷 중 송신지 IP주소, 수신지 IP주소, 송신지 포트번호, 수신지 포트번호, 프로토콜 등 5-튜플이 같은 것을 동일 네트워크 플로우로 묶는다. 각 네트워크 플로우에 속한 패킷들의 헤더를 제외한 페이로드 부분을 단순 연쇄(concatenation)하여 문서 i를 구성한다. (그림 3)에서의 각 행이 문서 i를 구성하며 각 네트워크 플로우를 구성하는 패킷의 개수는 다르다는 점에 유의하라. 연쇄 과정을 통해 각 네트워크 플로우는 하나의 긴 HEX문자열을 형성한다.

전처리과정: 플로우에 속한 패킷 payload concatenation

$F_{악성}$'s 1st packet payload	$F_{악성}$'s 2nd packet payload	...	$F_{악성}$'s last packet payload
F_2's 1st packet payload	F_2's 2nd packet payload	...	F_2's last packet payload
⋮	⋮	...	⋮
F_M's 1st packet payload	F_M's 2nd packet payload	...	F_M's last packet payload

(그림 3) 전처리 과정 : 네트워크 플로우로 LDA문서 생성
(Figure 3) Pre-processing: LDA document generation using a network flow

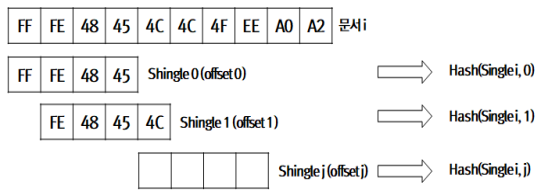
4.1.2 네트워크 플로우(문서)의 단어 구성

전통적인 텍스트문서에서 단어는 일반적으로 공백이나 구두점 같은 분리기호에 의해 분리되는 이어진 문자들로서 정의된다. 그러나 네트워크 트래픽에서의 단어 구성에 있어 동일한 정의를 적용할 수 없다. 왜냐하면 긴 HEX문자열을 단어로 구분하는 분리기호에 대한 정의를 명확하게 할 수 없기 때문이다.

[6]에서는 출력 가능한 ASCII문자열만을 고려한 네트워크 플로우(문서)의 단어 구성방법을 제안하였다. 출력 가능한 ASCII문자들의 연속구성(sequence)을 "단어"로 정의하였기 때문에 공백, 구두점 등 일반 텍스트에서 사용되는 분리기호를 이용해 문서를 구성할 수 있었다. 또한, 출력 가능한 ASCII문자열만을 고려하였기 때문에 수행 복잡도 문제를 큰 폭으로 해소하였다.

그러나 네트워크 패킷 페이로드는 ASCII코드표에 포함된 모든 문자를 포함할 수 있다. 예컨대 이진 실행코드 형태로 악성코드가 전파될 수 있다. 본고에서는 Shingle을 활용하여 모든 ASCII문자열을 "단어"로 활용할 수 있는 전집(corpus)의 구성기법을 제안한다. 제안 기법은 모든 ASCII문자열을 고려하여 네트워크 문서의 단어를 구성하는 방법으로 Hashing기법을 활용하여 수행 복잡도 문제를 해결한다.

(그림 4)는 제안하는 네트워크 플로우(문서)의 단어 구성 기법을 도식한다. 길이 L인 shingle은 L개의 ASCII코드의 연속구성이다. 예를 들어 문서 "FF FE 48 45 4C 4C 4F EE A0 A2 A3 AF"의 4-shingle은 "FF FE 48 45", "FE 48 45 4C", ..., "A0 A2 A3 AF"로 총 9개가 된다.

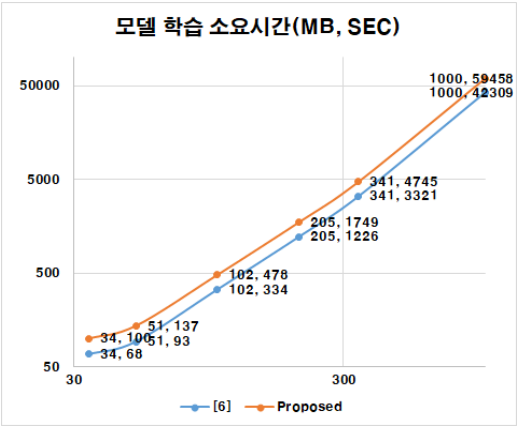


(그림 4) Shingle(i, j)의 해싱 방법
(Figure 4) Hashing of shingle(i, j)

4.1.1절에서 설명한바와 같이 전집을 구성하고, 전집의 각 문서 i에 대하여 길이 L인 모든 shingle에 Hash값을 계산(Rabin fingerprint)한다. 전집의 모든 shingle에 대해서 해시테이블을 구성한다. 해시테이블에서 χ회 이상 충돌(collision)이 발생한 모든 shingle을 전집 D에서의 단어로 사용한다. 예컨대, $\chi=2$일 때 두 개의 문서 "FF FE 48 45 4C 4C 4F EE A0 A2 A3 AF", "FF FE 48 45 4C"로 구성된 전집을 고려하면 "FF FE 48 45", "FE 48 45 4C"의 2개의 shingle이 각 2회 충돌이 발생하며 두 문서의 단어구성은 동일하게 될 것이다. 제안 방법은 네트워크 트래픽에서 자주 발견되는 이진문자열의 집합을 탐지규칙 시그니처의 후보군으로 고려할 수 있도록 한다. 그러나 중요한 shingle이 단 한번만 사용되는 경우* 충돌이 발생하지 않아 시그니처로서 활용할 수 없을 수도 있다.

성능 측면에서의 논의: 탐지규칙 시그니처의 추출에 있어 정확도 측면에서 가장 우수한 방법은 모든 Shingle을 단어로 활용하는 것이다. 가장 Naive한 이 방법은 LDA알고리즘을 적용함에 있어 수행시간 측면에서의 심각한 성능 문제를 야기한다. 네트워크 플로우의 단어 구성 문제는 결국 수행시간 vs 시그니처 정/오탐의 트레이드오프(trade-off) 문제로 귀결되며 [6]에서 제시되었던 이

* 확률적으로 매우 드문 일이다.



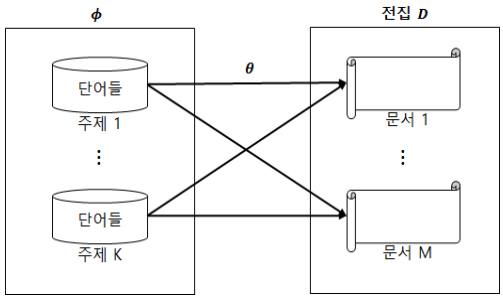
(그림 5) LDA모델 학습 소요시간
(Figure 5) LDA model training time

진 실행코드를 탐지규칙 시그니처로 활용할 수 없었던 문제를 해결하였다. 본 연구팀은 실험적으로 LDA알고리즘의 수행시간이 전집의 크기에 기하급수적으로 비례함을 실험적으로 규명하였으며 [6], χ값을 선택함에 있어 전체 시그니처 추출(LDA알고리즘에 거의 의존적임)시간을 고려한다. (그림 5)는 LDA모델 학습에 필요한 시간을 로그스케일로 도식한 것이다. 학습 데이터의 크기에 따라서 기하급수적으로 수행시간이 증가한다는 점은 [6]의 결과와 동일하다. 그러나 본고에서 제안하는 기법은 1.2배 이상 많은 비트패턴을 최종 시그니처 후보군으로 고려하므로 동일한 데이터셋을 사용하여 LDA모델을 학습하는데 [6]에 비해 1.5배의 시간이 소요된다.

4.2 탐지규칙 시그니처 추출

3장에서 논의한 바와 같이 LDA에서 전집 D의 문서 i는 인자 α,β를 갖는 2개의 디리클레분포에 의해 K개의 주제들의 집합에서 생성된다고 가정한다. 문서 i가 주제 $k=\{1,2,...,K\}$에 관련될 확률은 $\theta_{i,k}$이며 이는 인자 α를 갖는 디리클레분포 $\theta(=Dir(\alpha))$를 따른다. 또한 모든 단어 $w\in V$는 인자 β를 갖는 디리클레분포 $\phi(=Dir(\beta))$에 따르는 확률 $\phi_{k,w}$에 따라 주제 k와 연관된다. 확률분포 θ,ϕ를 (그림 6)에 도식하였다.

수식 (1)에 기술된 바와 같이 LDA알고리즘은 전집 D와 인자값(α,β)과 주제의 개수(K)로부터 관찰 불가능한 확률분포 (θ,ϕ)를 추론(estimation)한다. LDA의 최종적인 결과물로서 각각의 주제와 관련된 콘텐츠 문자열



(그림 6) LDA에 기반한 전집의 디리클레확률 연관관계
(Figure 6) Dirichlet allocation of corpus D based on LDA

(시그니처)인 단어와 시그니처의 관계의 크기는 $\phi_{k,w}$로 주어진다. 제안 기법에서는 탐지의 대상이 되는 악성트래픽과 관련된 주제에 관련된 단어의 ϕ값이 임계치 이상인 단어를 해당 악성트래픽을 탐지할 수 있는 시그니처로 활용한다.

$$\text{시그니처}=\{w|\phi_{k=\text{악성},w}>\text{임계값}\} \quad (2)$$

5. 성능 평가

5.1 실험 환경

5.1.1 데이터셋

본고에서는 제안 기법의 성능 검증을 위해 contagio [21]에 공개된 80여종의 네트워크 위협 데이터셋과 MACCDC(Mid-Atlantic Collegiate Cyber Defense Competition) [22] 및 CAIDA(Center for Applied Internet Data Analysis) [23]의 협조를 통한 수집한 정상 데이터 1T바이트를 혼합하여 사용하였다. contagio에서 제공하는 네트워크 위협 데이터는 ①트로이 목마, ②오버 플로우 공격, ③서비스 거부 공격, ④SQL삽입 공격, ⑤쉘 코드, ⑥임의 코드 실행, ⑦기타 등 7종으로 분류하였다. 실험에 사용된 네트워크 위협중 상당수는 CVE(Common Vulnerabilities and Exposures) 데이터베이스에 등록되어 있다.

5.1.2 매개 변수 세팅 및 알고리즘 구현

디리클레분포의 인자값(α,β): Heinrich의 연구[24]는 Gibbs샘플링 기법을 사용할 때 LDA알고리즘에 사용되는 디리클레분포의 최적 인자값을 도출하였다. 우리는 [24]에서 도출한 최적 인자값$(\alpha=50/K,\beta=0.01)$을 활

용 하였다.

토픽의 개수(K): LDA알고리즘은 전집에 함축된 주제의 개수를 매개 변수로 받는다. 본고에서는 망에 통상적으로 흐르는 응용 프로토콜의 개수에 대한 연구[25]을 참고하여 $K=30$으로 설정하였다.

shingle 길이(L): shingle의 길이를 작은 값으로 설정하면 전집에 포함되는 단어가 많아져 LDA알고리즘을 통해 추출한 시그니처의 정밀도가 증가한다. 본고에서는 L값을 4바이트로 한정하였다. 왜냐하면 4바이트보다 작은 shingle의 길이를 사용하면 전집 구성과정에서 시그니처로서 활용도가 떨어지는 shingle을 다수 포함한다. 이는 LDA알고리즘 수행에 있어 과도한 연산을 요구한다.

5.1.3 LDA알고리즘의 구현

우리는 네트워크 패킷 jpcap(pcap library in JAVA)[26]을 활용하여 전처리 과정을 JAVA코드로 구현하였으며, LDA 및 Gibbs샘플링 기법을 구현한 JGibbLDA[27]를 활용하여 실험을 진행하였다.

5.2 실험 결과

5.2.1 추출한 탐지규칙 시그니처 개수

표 1에 공격 유형별로 추출한 탐지규칙 시그니처 평균 개수를 나타냈다. 80여종의 실험에 사용했던 모든 네트워크 위협에 대하여 제안 기법이 탐지규칙 시그니처를 추출한다. [6]에서는 큰 L값(32Byte)에 대하여 특정 네트워크 위협들은 시그니처를 추출하지 못하는 경우도 있다. 그러나 본고에서는 실험에 활용된 모든 공격트레이스에 대하여 탐지규칙 시그니처를 추출한다. 우리는 [6]에서의 실험결과를 통해 300M정도의 전집크기를 사용하면 대략 1시간 이내에서 시그니처 추출결과를 얻을 수 있음을 고려하여 χ값을 설정하였다. 즉, 네트워크 캡처 파일의 용량이 크다면 보수적으로 χ값을 설정하였으며, 용량이 작다면 진보적인 χ값(=2)을 설정하여 실험을 수행하였다.

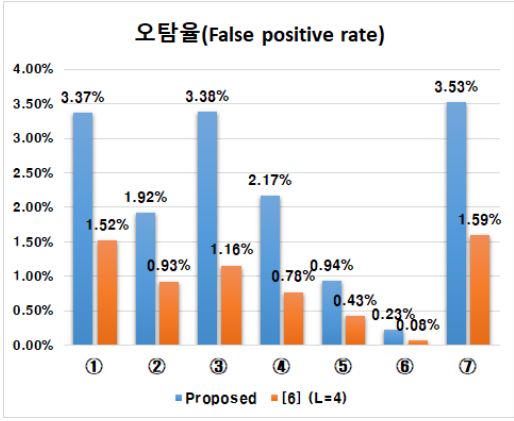
(표 1) 추출한 탐지규칙 시그니처 평균개수 (공격 유형별)
(Table 1) The average number of the extracted signatures (categorized into attack types)

L	①	②	③	④	⑤	⑥	⑦
4B	613	17	7	27	8	7	8

5.2.2 추출한 탐지규칙 시그니처의 정/오탐율

이절에서는 LARGen[6]에서 사용했던 출력가능한 ASCII문자열을 이용한 전집구성 방법과 제안방법의 탐지규칙 시그니처의 정탐율, 오탐율을 비교한다.

[6]과 제안방법 모두 추출한 탐지규칙 시그니처의 정탐율은 100%이다. 이는 네트워크 패킷 페이로드의 부분 문자열을 시그니처로 사용하므로 자명한 결과라 할 수 있다.



(그림 7) 추출한 탐지규칙 시그니처 오탐율 최댓값
(Figure 7) The maximum false positive rates of the extracted signatures

탐지규칙 시그니처의 추출에 있어서 가장 중요한 성능척도는 추출한 탐지규칙 시그니처의 오탐율이다. (그림 7)은 LARGen[6]과 제안방법을 이용해 추출한 탐지규칙 시그니처의 오탐율(false positive rate)을 공격 유형별로 그래프로 나타낸 것이다. 5.1.2.에서 기술한 바 제안 기법은 LARGen보다 적은 수의 시그니처를 추출한다. 그럼에도 불구하고 제안 기법에 의해 추출된 탐지규칙 시그니처는 0.23%(⑥임의 코드 실행) ~ 최대 3.53%(⑦기타)의 오탐율을 보인다. 이는 LARGen의 최대 오탐율인 1.6%보다는 다소 높은 값이다. 그러나 제안방법은 모든 ASCII문자열을 네트워크 위협 시그니처로 추출할 수 있다. 일반적으로 기계학습을 활용한 스킴들의 성능을 고려하면 5%이내의 오탐율은 우수한 성능임에 틀림없다 [25].

6. 결 론

본고에서는 네트워크 위협 트래픽을 분석하고 침입탐지시스템의 탐지규칙에 사용할 수 있는 시그니처를 추출

하는 기법을 제안하였다. 제안 기법은 LDA알고리즘을 이용하여 네트워크 위협 트래픽을 탐지할 수 있는 탐지규칙 시그니처를 추출한다. 본고에서는 제안 기법의 성능을 평가하기 위해 다양한 실험을 수행하였다. 실험결과에 따르면 제안 기법을 통해 추출한 네트워크 위협 탐지규칙 시그니처는 최대 3.53%의 오탐율을 보였다. 제안하는 기법은 네트워크 패킷 페이로드의 형태와 관계없이 모든 ASCII문자열에 대하여 네트워크 위협 탐지규칙 시그니처를 추출할 수 있다. 따라서 기존에 성능문제로 해결하지 못하였던 이진 실행코드형태로 전파되는 악성코드에 대한 탐지규칙 시그니처를 추출할 수 있다. 이는 4장에서 제시한 네트워크 플로우를 통한 단어 구성방법에 대한 연구에 기반을 둔다. 우리는 현재 딥러닝(deep learning), 특히 RNN(Recurrent Neural Networks)기법을 탐지규칙 시그니처 추출 분야에 적용하기 위한 연구를 진행 중이다.

참고문헌(Reference)

[1] A. Wool, “A quantitative study of firewall configuration errors”, Computer, vol. 37, no. 6, pp. 62 – 67, 2004. https://doi.org/10.1109/mc.2004.2

[2] Y. Qi, B. Yang, B. Xu, and J. Li, “Towards system-level optimization for high performance unified threat management,” in proc. of IEEE ICNS 2007. https://doi.org/10.1109/icns.2007.126

[3] T. Krueger, C. Gehl, K. Rieck, and P. Laskov, “Tokdoc: A selfhealing web application firewall,” in Proceedings of the 2010 ACM Symposium on Applied Computing. ACM, 2010, pp. 1846 – 1853. https://doi.org/10.1145/1774088.1774480

[4] X. Zhang, C. Li, and W. Zheng, “Intrusion prevention system design,” in Computer and Information Technology, International Conference on. IEEE Computer Society, 2004, pp. 386 – 390. https://doi.org/10.1109/cit.2004.1357226

[5] SNORT, https://www.snort.org/.

[6] S. Lee et al. “LARGen: Automatic Signature Generation for Malwares Using Latent Dirichlet Allocation,” IEEE Transactions on Dependable and Secure Computing (2016). https://doi.org/10.1109/tdsc.2016.2609907

[7] Hofmann, Thomas. "Probabilistic latent semantic analysis." Proceedings of the 15th conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 1999. http://www.iro.umontreal.ca/~nie/IFT6255/Hofmann-UAI99.pdf

[8] Blei, David M. "Probabilistic topic models." Communications of the ACM 55.4 (2012): 77-84. https://doi.org/10.1145/2133806.2133826

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet Allocation,” the Journal of machine Learning research, vol. 3, pp. 993 – 1022, 2003. https://endymecy.gitbooks.io/spark-ml-source-analysis/content/%E8%81%9A%E7%B1%BB/LDA/docs/Latent%20Dirichlet%20Allocation.pdf

[10] J. Newsome, B. Karp, and D. Song, “Polygraph: Automatically generating signatures for polymorphic worms,” in Security and Privacy, 2005 IEEE Symposium on. IEEE, 2005, pp. 226 – 241. https://doi.org/10.1109/sp.2005.15

[11] R. Perdisci, D. Dagon, W. Lee, P. Fogla, and M. Sharif, “Misleading worm signature generators using deliberate noise injection,”in Security and Privacy, 2006 IEEE Symposium on. IEEE, 2006. https://doi.org/10.1109/sp.2006.26

[12] M. M. Mohammed, H. A. Chan, and N. Ventura, “Honeycyber: Automated signature generation for zero-day polymorphic worms,” in Military Communications Conference, 2008. MILCOM 2008. IEEE. IEEE, 2008, pp. 1 – 6. https://doi.org/10.1109/milcom.2008.4753178

[13] Jolliffe, Ian. Principal component analysis. John Wiley & Sons, Ltd, 2002. http://dx.doi.org/10.1007/b98835

[14] G. Tahan, C. Glezer, Y. Elovici, and L. Rokach, “Auto-sign: an automatic signature generator for high-speed malware filtering devices,” Journal in computer virology, vol. 6, no. 2, pp. 91 – 103, 2010. https://doi.org/10.1007/s11416-009-0119-3

[15] A. Tongaonkar, R. Keralapura, and A. Nucci, “Santaclass: A self adaptive network traffic classification system,” in IFIP Networking Conference, 2013. IEEE, 2013, pp. 1 – 9. http://ieeexplore.ieee.org/document/6663505/

[16] Z. Zhang, Z. Zhang, P. P. Lee, Y. Liu, and G. Xie, “Proword: An unsupervised approach to protocol

feature word extraction," in INFOCOM, 2014 Proceedings IEEE. IEEE, 2014, pp. 1393 – 1401. https://doi.org/10.1109/infocom.2014.6848073

[17] H. J. Wang, C. Guo, D. R. Simon, and A. Zugenmaier, "Shield: Vulnerability-driven network filters for preventing known vulnerability exploits," ACM SIGCOMM 2004. https://doi.org/10.1145/1015467.1015489

[18] Z. Li, G. Xia, H. Gao, Y. Tang, Y. Chen, B. Liu, J. Jiang, and Y. Lv, "Netshield: massive semantics based vulnerability signature matching for high-speed networks," ACM SIGCOMM 2010. https://doi.org/10.1145/1851182.1851216

[19] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Proceedings of the National academy of Sciences, vol. 101, no. suppl 1, pp. 5228 – 5235, 2004. http://psiexp.ss.uci.edu/research/papers/sciencetopics.pdf

[20] Sood, Aditya K., Richard J. Enbody, and Rohit Bansal. "Dissecting SpyEye – Understanding the design of third generation botnets." Computer Networks 57.2 (2013): 436-450. https://doi.org/10.1016/j.comnet.2012.06.021

[21] M. Parkour, "blog sobre comparticion de malware, recurso en l´ınea disponible," 2014. http://contagiodump.blogspot.com/

[22] Netresec, "Capture files from Mid-Atlantic CCDC," http://www.netresec.com/?page=MACCDC, 2014.

[23] CAIDA. http://www.caida.org/home/

[24] G. Heinrich, "Parameter estimation for text analysis," in Technical Report. Fraunhofer IGD, Darmstadt, Germany, 2009. http://www.arbylon.net/publications/text-est2.pdf

[25] Kim, Hyunchul, et al. "Internet traffic classification demystified: myths, caveats, and the best practices." Proceedings of the 2008 ACM CoNEXT conference. ACM, 2008.https://doi.org/10.1145/1544012.1544023

[26] jpcap. http://jpcap.sourceforge.net/

[27] A Java Implementation of Latent Dirichlet Allocation (LDA) using Gibbs Sampling for Parameter Estimation and Inference. http://jgibblda.sourceforge.net/

◐ 저 자 소 개 ◑

이 성 일(Sungil Lee)
1998년 순천향대학교 컴퓨터공학과(공학사)
2005년 순천향대학교 정보보호대학원 정보보호학과(공학석사)
2008년~현재 한국전자통신연구원 부설연구소 선임연구원
관심분야 : 정보보호, 포렌식, 인공지능
E-mail : silee@nsr.re.kr

이 수 철(Suchul Lee)
2008년 서울대학교 컴퓨터공학부(공학사)
2014년 서울대학교 대학원 컴퓨터공학부(공학박사)
2016년~현재 한국교통대학교 철도대학 철도경영·물류·컴퓨터학부(컴퓨터정보공학전공) 조교수
관심분야 : 정보통신 및 보안
E-mail : sclee@ut.ac.kr

이 준 락(Jun-Rak Lee)
1984년 인하대학교 수학과(이학사)
1986년 인하대학교 대학원 수학과(이학석사)
1991년 인하대학교 대학원 수학과(이학박사)
1995년~현재 강원대학교 인문사회과학대학 교양학부 교수
관심분야 :해석학, 데이터베이스, 정보통신 및 보안
E-mail : jrlee@kangwon.ac.kr

염 흥 열(Heung-Youl Youm)
1981년 한양대학교 전자공학과(공학사)
1983년 한양대학교 대학원 전자공학과(공학석사)
1990년 한양대학교 대학원 전자공학과(공학박사)
1990년~현재 순천향대학교 공과대학 정보보호학과 교수
관심분야 : 인터넷보안, USN보안, IPTV보안, 홈네트워크 보안, 암호 프로토콜
E-mail : hyyoum@sch.ac.kr