Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.31, NO.2, Apr. 2021 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2021.31.2.211

Hadoop 환경의 기계 학습을 이용한 HTTP 공격 탐지 모델

김 형 기,[1†] 김 문 현[2‡]
[1,2]성균관대학교 정보통신대학원 (대학원생, 교수)

HTTP Intrusion Detection Model Using Machine Learning in Hadoop Environment

Hyung-gi Kim,[1†] Moon-hyun Kim[2‡]
[1,2]Graduate School of Information & Communications SungKyunKwan University
(Graduate student, Professor)

요 약

모바일 시대가 된 이후 인터넷 트래픽 사용량은 기하급수적으로 증가 하고 있으며 이를 통한 각종 침해 사고와 비정상 트래픽이 급증하고 있다. 최근의 침해 사고는 보다 다양화 되고 지능적이며 복합적인 형태로 발생되고 있으며 이를 탐지하기 위해서는 기존의 방법들 이외에 다양한 방법이 요구 되고 있다. 이에 본 연구에서는 HTTP 서비스 트래픽을 빅 데이터로 수집하여 Web 서비스 침해 시도를 탐지하기 위한 방안으로 기계학습 기반의 지도 학습인 SVM과 의사결정 트리를 이용해 구현하여 검증해 보고 지도 학습의 한계를 보완하기 위해 패턴을 사용하지 않는 비지도 학습 기반의 Word to Vector 방식을 적용 할 수 있는 방안에 대해서 연구하고자 한다.

ABSTRACT

Since the mobile era, Internet traffic usage has been increasing exponentially, and through this, various intrusion accidents and abnormal traffic are increasing rapidly. Recently, infringement accidents are occurring in a more diversified, intelligent and complex form, and various methods other than the existing methods are required to detect them. Therefore, in this study, a method to detect web service intrusion attempts by collecting HTTP service traffic as big data is implemented and verified using SVM and decision tree, which is a machine learning-based supervised learning without using patterns. In order to compensate for the limitations, I would like to study how to apply the Word to Vector method based on unsupervised learning.

Keywords: Machine Learning, Word to Vector, SVM, Hadoop

I. 서 론

최근의 인터넷상 공격은 기존의 단순한 서버를 공격하는 것에 그치지 않고 APT(Advanced Persistent Threat) 공격과 다양한 네트워크상의 위협들이 증가하고 있다. 추가로 일반 사용자들을 대상으로 하는 악성 코드 공격 또한 증가하고 있으며 이로 인해 인터넷 환경은 점차 위험해 지고 있다[1,2].

이런 고도화된 공격을 피해자에게 전달하기 위한 다양한 방법이 사용되고 있고 그중 Email을 이용한 HTTP 주소 기반의 전달 방법이 많이 사용되고 있

Received(12. 28. 2020), Modified(02. 15. 2021), Accepted(03. 08. 2021)
† 주저자, sokoban@kakao.com
‡ 교신저자, mhkim@skku.edu(Corresponding author)

으며 일반적인 웹페이지에 삽입된 악성 URL 역시 위험한 요소 중 하나이다. 이러한 위험들은 지금까지 다양한 보안장비로 탐지, 분석, 대응하기 위한 노력이 이루어졌고 최근에는 IDS(Intrusion Detection System), IPS(Intrusion Prevention System), Firewall 등의 보안장비의 로그와 기계학습을 이용하여 공격을 탐지하기 위한 연구가 지속하고 있다.

하지만 점차 고도화되고 정상에 가까운 공격들, 특히 논리적인 허점을 이용한 공격들에 대해서는 기존의 패턴 기반 장비에서는 그 한계가 명확하게 드러나고 있다. 물론 이러한 한계를 극복하기 위한 다양한 방법들이 연구, 소개되고 있으며 기계 학습을 사용하는 방법은 이미 많은 연구에서 발표되었고 관련된 장비나 연구 자료를 구할 수 있다[7,9,17,18].

다만 대다수의 연구가 악성 코드의 특성을 분석하여 탐지하는 데에 주로 초점이 맞추어져 있거나 HTTP URL 상에서 이미 확보된 악성 URL 정보에 의존하거나 악의적인 도메인을 탐지하는 데에 중점을 두어 HTTP 트래픽에 대한 보안 탐지 부분에서는 False Negative와 False Positive의 문제가 많이 발생하고 있으며 이러한 점을 보완하여 End Point나 추가적인 정보를 수집하여 탐지에 적용하는 방안이 소개되고 있다[15, 16].

또한 웹과 모바일 서비스에서 HTTP를 이용하는 경우라면 다양한 취약점에 노출되며 취약점마다 다양한 공격 지점이 존재하므로 이를 모두 인지하는 것은 어려우며 공격자가 테스트를 통해서 우회도 가능할 수 있다. 또한 Zero-Day 공격의 경우 어떠한 보안장비에서도 정확하게 탐지할 수 있는 것을 보장하기는 힘들다.

이러한 단점과 기존의 연구를 보완하기 위해 본 연구에서는 서비스상에서 발생하는 모든 HTTP 트래픽을 대상으로 패턴이나 Feature를 추출하지 않고 공격 문을 NLP(Natural Language Processing)를 이용하여 분석/탐지하는 것을 목표로 한다. 우선 HTTP Packet 전체에 대한 각 요소를 Feature로 뽑아 이를 기계 학습으로 분석하여 기존의 연구들의 한계를 증명한 다음, 공격 구문 자체만을 대상으로 재처리를 수행하고 이를 NLP를 이용한 분석 이후 전체 구문을 벡터화하고 이를 지도학습을 통해서 공격 URL을 탐지 하는 것에 대해서 연구한다.[2,3].

II. 관련연구

본 연구에서는 HTTP Packet의 L7 영역의 데이터를 수집하여 이를 Hadoop을 통해 수집/정제하여 기계학습에 적용하는 방법을 사용한다. 먼저 트래픽 상에서 Packet을 수집하는 구간과 이를 저장/분석하는 Hadoop의 구간, 세 번째로 이러한 데이터를 기반으로 하는 기계학습 구간, 총 3가지의 중요 구간으로 나눌 수 있다.

2.1 트래픽 상의 HTTP Packet의 수집

초기 인터넷 서비스의 형태는 주로 HTTP 프로토콜을 웹상의 콘텐츠 제공에만 사용하였지만, 동적 웹 콘텐츠의 증가와 모바일 서비스가 증가함에 따라 HTTP 프로토콜을 이용한 REST API가 더욱 활성화 되고 있어 기존 인터페이스를 통한 통신을 제외한 대부분의 통신이 HTTP를 추구하고 있다.

HTTP가 서비스 되는 대부분의 환경은 라우터나 스위치를 통해서 서비스되고 있으므로 HTTP Packet을 수집하기 위해서 스위치에서 제공하는 Packet Mirror 기능을 이용하여 Packet을 수집한다.

추가로 HTTPS의 경우 암호화된 트래픽으로 트래픽 복제를 통해 수집할 경우 암호화된 문장만 볼 수 있으므로 분석을 위해 평문을 수집하기 위한 SSL을 복호화 하여 수집하여야 한다[14].

2.2 트래픽상의 비정상 HTTP Packet

비정상 HTTP 트래픽은 공격의 관점에서 분석가들에게 다양한 의미로 해석될 수 있으며 수집되는 트래픽의 종류에 따라 각기 구분될 수 있다.

Table 1.에는 일반적인 웹 사이트의 사용자 리스트를 출력하는 화면에서 admin 사용자를 조회하는 정상 웹 요청과 여기에 SQL Injection 공격을 이용하여 DB 내의 사용자의 정보를 가져오는 공격 문으로 구성된 두 가지의 로그가 있다[11].

실제 HTTP 요청문 내의 특정 변수에 공격을 위한 공격 구문이 삽입되어 있고 연구를 위해 구축한 환경에서는 모든 사용자의 ID/Password 정보가 반환되었다.

이러면 쉽게 서명 기반의 IDS 장비로도 공격을 탐지할 수 있을 것이다. 하지만 특정 웹

Table 1. Normal And Attack Web Log

Type	Log
Normal Web Log	http://192.168.0.137/admin/list/?id=admin&Submit=Submit&title=%EC%9E%AC%EB%AC%B4%20Admin%20!
Attack Web Log	http://192.168.0.137/admin/list/?id=%27+and+1%3D0+union+select+null%2C+concat%28user%2C%27%3A%27%2Cpassword%29+from+users+%23&Submit=Submit&title=%EC%9E%AC%EB%AC%B4%20Admin%20!

Application에 존재하는 다양한 취약점과 알려지지 않은 Zero-Day 취약점은 서명 기반으로는 탐지할 수 없다. 더욱이 파일 업로드 공격이 성공되어 웹 서버에 /admin/list/websheell.php 라는 WebShell이 동작되고 있고 여기에 공격자가 접근하더라도 패턴이 없다면 탐지하는 것이 불가능하다.

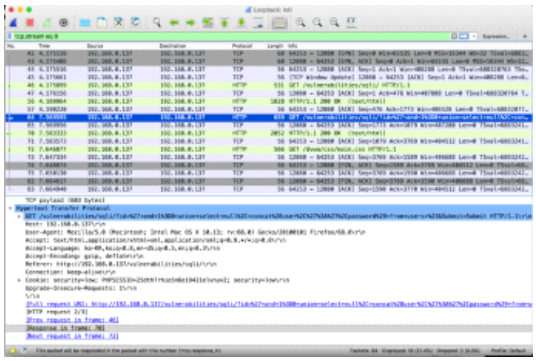

Fig. 1. SQL Injection Attack Packet

2.3 Hadoop

많은 양의 Packet 데이터를 저장하고 효과적으로 분석하기 위해서는 기존의 RDBMS 보다 최근에 제공되는 분산 기반의 데이터 분석 솔루션을 사용하는 것이 효과적이다. 그중에서도 Hadoop은 다양한 연관 솔루션들인 Hive, HBase를 제공하며 더욱이 Spark나 Kafka와 같은 다양한 솔루션들과도 쉽게 연동되기 때문에 데이터 분석을 위한 환경 구성이 쉬우며 Hive를 사용할 경우 구분자로 구분된 TXT 파일을 업로드 하는 것만 으로도 SQL 질의를 사용할 수 있기 때문에 사용하고자 한다.

2.4 기계 학습

2.4.1 지도 학습

지도 학습은 산출될 수 있는 결과 값인 Class가 있고 이에 개별적인 데이터들의 결과 값인 Label이 정해진 Training Set 데이터를 이용하여 입력된 데이터들에 대한 결과 값을 산출하는 데 사용되는 기계 학습이다. 입력되는 데이터들은 학습을 위한 Training set 말고도 검증을 위한 Validation Set, 그리고 테스트를 위한 Test Set 등 3가지로 구성된다.

지도 학습의 일반적인 출력 값은 선형(regression) 값처럼 연속 값일 수도 있으며 또는 입력된 데이터에 대한 분류 값 즉 classification 값일 수도 있다.

2.4.2 비지도 학습

지도 학습이 알려진 결과에 대해 분류를 한다면 비지도 학습은 이러한 결과에 대한 결과 값을 알지 못하는, 즉 출력 값에 대한 정보 없이 학습해야 하는 모든 종류의 기계 학습을 지칭한다. 즉 비지도 학습은 입력되는 데이터만으로 얻고자 하는 정보를 추출할 수 있어야 한다.

비지도 학습은 크게 비지도 변환과 군집(Clustering)으로 나누어질 수 있다. 군집은 입력되는 대상에게 적용된 항목들을 그룹화 하는 것을 말하며 비지도 변환은 입력된 데이터를 사람이나 기계 학습에 적용할 수 있도록 반드시 필요한 특징만을 추출할 수 있도록 하는 차원 축소 등이 있다[8].

III. Packet 수집과 Hadoop을 이용한 공격 탐지 시스템의 설계 및 구현

일반적인 인터넷 서비스 기업이나 사무실 환경이라고 할지라도 인터넷 트래픽은 1Gbps ~ 10Gbps 수준으로 생각된다. 하지만 본 연구에서는 최소 100Gbps 이상의 HTTP 트래픽에서 적용이 가능한 모델을 찾는 것을 추가 목표로 하였다.

100Gbps를 TCP 오버헤드를 제외한 데이터 트래픽이라면 초당 12.5Gbyte이다. HTTP 요청을 평균 800byte, 응답을 MTU 최대 크기인 1500byte인 대칭 형태의 HTTP 서비스라면 초당

약 583만 pps가 된다. 이 정도의 트래픽을 저장하여 실시간으로 기계 학습으로 분석하는 환경을 구축하는 것도 한 개의 연구 과제가 될 수 있으나 본 연구에서는 실시간은 제외하고 대단위 데이터를 저비용으로 저장하여 분석이 가능한 Hadoop을 이용하고자 한다.

다만 실제 100Gbps 망에서의 수집/분석은 수행하지 못하고 샘플링의 개념으로 대규모 서비스를 가정한 2Gbps 수준의 트래픽을 대상으로 실험은 진행하였다.

3.1 Packet 수집 시스템의 구현

실제 인터넷상에서 사용자들이 서비스를 사용하면서 발생하는 데이터 Packet을 수집하기 위해 서버의 상단 스위치에서 트래픽 복제 방식으로 트래픽을 수집 하였다. 단 모든 트래픽을 수집하기에는 수집 시스템의 한계가 있으므로 HTTP 응답의 Payload를 제외한 HTTP에서 분석, 활용 가능한 부분을 수집하여 본 연구에 사용하였다. 일반적인 소규모 서비스 상에서는 모든 Packet을 Pcap 방식으로 저장하여 이를 분석에 사용할 수 있으나 대용량 서비스 환경에서 이를 모두 저장/분석하는 것은 수집 및 저장 과정에서 많은 부하가 발생한다. 그러므로 패킷을 수집하는 과정에서 Pcap 라이브러리를 활용하여 응답 Packet의 Payload 데이터를 제외한 HTTP Header, Cookie, URI, Path, Query 등을 추출 필드 구분자를 두고 요청과 응답을 각각 한 개의 문자열로 TXT파일 형식으로 Hadoop에 저장하여 사용한다.

3.2 데이터 분석 시스템

3.1에서 설명한 수집 시스템과 함께 1차 통계 작업 및 데이터의 Feature 추출을 위한 작업을 Hadoop을 중심으로 분석 시스템을 구축하였으며 전체적인 작업의 흐름은 Fig 2.와 같다.

HTTP 데이터들은 HDFS 상의 데이터 노드에 저장되고 저장된 데이터들은 분석을 위해 MapReduce를 사용한다. 다만 이때 직접적인 MapReduce 작업을 프로그램으로 개발하기보다는 Apache Hive에서 제공하는 SQL 언어를 이용하여 Feature의 분석 및 통계 데이터의 추출 등을 수행하였다. 필드 구분자로 저장된 각각의 필드는 Hive

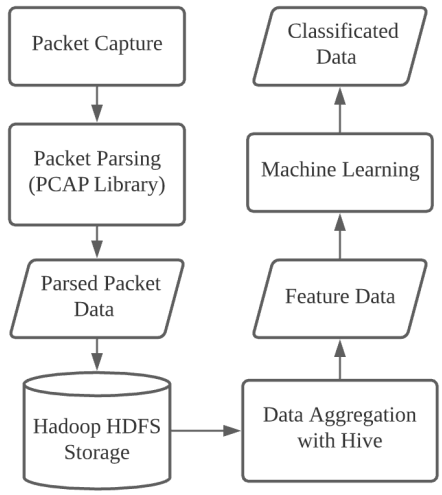

Fig. 2. Data Collection/Analysis Flow Diagram

를 이용하여 필드를 분석하고 Feature를 추출하는 용도로 사용하였다.

Hadoop은 대용량 데이터에서 통계 추출과 이를 통한 Feature의 추출 역할을 수행하며 상세한 기계 학습은 통계와 분석에 최적화된 scikit-learn과 이용하고자 한다.

3.3 기계 학습의 적용

기계 학습은 크게 지도 학습과 비지도 학습으로 나누어질 수 있으며 본 연구에서는 지도 학습을 적용해 보고 이에 대한 개선으로 word2vec과 지도학습 두 가지를 적용한 공격 탐지 방안을 연구해 보고자 한다.

3.3.1 Scikit-Learn

Scikit-Learn은 Python 기반의 모듈로서 Tensor Flow와 함께 가장 인기 있는 오픈 소스 기계 학습 소프트웨어이다.

지도 학습, 비지도 학습을 중심으로 이를 지원하기 위한 Feature 및 모델 그리고 데이터 변환 등에 대한 다양한 알고리즘과 기능을 제공하고 있다.

Scikit-Learn은 손쉽게 모델을 만들고 이를 구성하여 적용해 볼 수 있는 장점이 있지만, 서비스 영역에서 대용량 데이터를 처리해야 할 경우 GPU를 지원하지 않으므로 대용량 데이터의 처리가 힘든 단점이 있다.

IV. 실험 및 결과

실험의 목표는 HTTP 서비스에 구축된 네트워크 기반 IDS에서 서명 없이는 탐지하지 못하는 Zero-Day와 같은 최신 공격과 서명을 우회하기 위해 변조된 공격을 서명 없이 탐지하는 것을 목표로 한다.

기존의 서명 기반 방법뿐만 아닌 통계적 분석 방법과 SVM(Support Vector Machine), Random Forest 등의 기계 학습 알고리즘을 이용한 방법 등을 혼합하여 사용하고자 한다.

실험 평가 방법으로는 분석에 사용된 데이터 내에 정상과 비정상을 분류하였고 이를 실질적으로 기계 학습에서 정확하게 분류 해낼 수 있는지에 대해서 평가하였다.

마지막으로 실제 데이터 상에서 수동으로 분류한 공격 이외의 비정상적인 요청에 대해서도 효과적으로 탐지해 낼 수 있는지를 평가하게 되고 전체 공격의 80% 수준을 90%의 확률로서 탐지해 내는 것을 목표로 하고 있다. 하지만 공격에 대한 정확한 기준이 명확하지 않은 경우를 고려하여 오차는 +-5% 수준을 허용하였다.

4.1 기계 학습을 통한 탐지

4.1.1 Feature 추출

Hadoop에 저장된 데이터들 중에서 공격자와 정상을 파악하기 위해서 서버로 인입되는 HTTP 요청에 대해 Table 2와 같은 Feature를 추출하여 이를 이용하여 기계 학습을 적용해 보고자 한다.

Table 2. Feature for HTTP Packet

Feature	Description
Unique Reference	Unique Reference URL Count
Unique Host	Unique Destination Host Count
Unique Dest IP	Unique Destination IP Address Count
Unique User-Agent	Unique User-Agent Connection Count per IP
Unique Path	Unique Connection Path Count
Unique Query	Unique Connection query Count per IP
Total Connection Count	Total Connection Count to Server

IP	Referer	Host	Dest IP	User Agent	Path	Query	Count	Label
117.x.x.x	1	6	1	1	552	36	1002	Normal
61.x.x.x	2	2	9	1	1	2	945	Attack
212.x.x.x	92	4	14	1	49	54	207	Normal
114.x.x.x	19	7	23	8	6	27	308	Normal
117.x.x.x	1	1	5	1	1	139	206	Attack
222.x.x.x	90	2	10	1	53	58	205	Normal
115.x.x.x	1	1	12	1	1	205	205	Attack
115.x.x.x	1	1	12	1	1	203	203	Attack
115.x.x.x	1	1	12	1	1	203	203	Attack
66.x.x.x	40	1	20	2	149	175	203	Normal
114.x.x.x	1	1	12	1	1	201	201	Attack
221.x.x.x	85	6	14	1	57	32	201	Attack
66.x.x.x	38	1	20	2	150	174	200	Normal
182.x.x.x	2	2	5	1	5	48	54	Attack
1.x.x.x	16	2	4	1	3	21	54	Normal
66.x.x.x	43	6	8	2	20	37	54	Attack

Fig. 3. Feature Data

실험의 목적 중 한 가지는 일반적인 Feature 추출 시에 데이터를 분석하여 추가적인 전처리를 통해 추출되는 반면 데이터 자체의 통계만을 이용하여 공격자들을 탐지하는 부분에 대해서 살펴보기 위해 1차적으로 위의 Feature들을 사용하였다.

각 Feature를 추출하기 위한 Key는 Source IP를 기준으로 하여 각각의 Feature를 산출하였으며 Label은 공격과 정상 두 가지를 분류하여 기계 학습에 적용하였다.

실제 추출된 데이터는 Fig 3.과 같으며 Labeling을 위해 임계치에 기반을 둔 통계나 비정상적인 통신 Packet을 발생시키는 IP들만을 추출하는 것은 실제 공격자를 찾는 기준이 될 수 없어 네트워크 상의 공격을 탐지하는데 사용되는 IDS의 탐지 데이터에 기반을 두어 공격자를 분류하고 그 이외의 데이터를 정상적인 접근으로 분류하였다.

4.1.2 지도 학습

Fig 3.의 데이터를 지도 학습을 통해 학습하기 위한 알고리즘으로는 의사결정 트리(Decision Tree)를 사용하여 지도 학습을 진행하였다.

먼저 의사결정 트리를 이용하여 학습을 진행해본 결과 전체 입력 Feature 중 Unique Path가 가장 높은 중요도로 나타났으며 다음으로 Unique

Table 3. Attack Type for Connection

Label	Description
Web Hacking	Web hacking attempt conforming to WOWASP TOP10
Brute Force	Mass access for account takeover
Web Crawling	Mass access to collect large amounts of information
information gathering	Collecting information on services for attacks

Referer와 Total Connection Count였다. 이러한 결과를 미루어 보았을 때 아래 Table 3.의 공격 유형을 해당 결과에 대입하여 실제 데이터에 적용해 보았다.

Unique Path가 5이하 이고 Unique Referer가 6 이하인 경우 주로 Brute Force 공격과 DDoS 공격에 대한 탐지가 가능하였고 Total Count가 212회 이하이고 Unique Referer가 24회 이하인 경우 Web Crawling이나 정보 획득 공격임을 탐지할 수 있었다. 추가로 SVM 알고리즘을 적용해 보았을 때 알고리즘상 정확도 87%로 나왔으나 이를 실제 데이터에 적용한 결과를 수동으로 Web Hacking을 검사하였을 때 Web Hacking의 탐지율은 약 30% 수준이였다.

의사결정 트라와 SVM 두 가지 알고리즘에서 모두 단순 임계치 성격의 공격 유형을 찾을 수는 있었지만 Web Hacking에 대한 탐지율은 30% ~ 40% 수준 이하였다.

4.1.3 지도 학습의 결과

지도학습의 적용 결과 Fig 3.의 Feature와 Label로는 다양한 서비스에 들어오는 전체 접속 데이터 중 HTTP 공격에 대한 탐지가 힘들다는 것을 알 수 있었다.

대용량의 데이터라면 통계 기반의 Feature를 통해 공격자에 대해 어느 정도 수준의 분류가 가능할 뿐 목표인 HTTP 내의 공격을 수행한 공격자에 대해 분류하는 기준이 되지 못하며 더욱이 일반적인 공격자의 특성이 정상 IP와 유사하게 나타나는 경우를 보이고 있으므로 대부분은 Over Fitting이 발생되었다.

이를 보완하기 위해서 각 Feature를 통계 데이터로 산출하기 이전에 각각의 항목에 대해서 정상과 비정상을 기입하여 스코어링을 하는 경우에는 HTTP의 공격자에 대한 분류가 가능하나 Table 4.에서 표기한 것과 같이 각 항목들의 비정상적의 데이터가 서비스별로 구분되어 축척 되어야 하며 Threat Intelligence를 통해서 위협 IP DB를 별도로 구축하는 것이 필요하므로 이 부분은 제외하고자 한다. 현재에도 많은 논문에서 이를 다루고 있는 것으로 파악된다[15].

Table 4. Feature Classification

Feature	Description
User-Agent	Known Malicious UA : Bad Etc UA : Normal
Country	Same Country : Normal Other : Suspicious
Port	Service Port(80,443): Normal Etc : Suspicious
Path	Known Path : Normal Unknown Path : Suspicious
Reputation DB	Known Malicious IP : Bad Etc IP : Normal

4.2 Word2Vector와 지도학습

4.1의 기본적인 통계 기반의 Feature 방식으로는 HTTP 상의 공격을 탐지하는 데에 그 한계가 있다는 것이 증명되었다.

이번 실험에서는 Word to vector와 지도 학습의 Random Forest, SVM 을 함께 적용하여 웹 서비스의 다른 변수에 영향을 받지 않고 오직 공격 구문만을 이용하여 탐지하는 것이 가능한지를 알아보고자 하며 이를 위해 웹 요청 문만을 대상으로 하여 진행하였다.[9].

다만 Web Hacking 내의 공격 유형을 분류해 내는 것이 아닌 정상과 공격 두 가지를 분류해 본다.

4.2.1 공격 데이터의 Labeling

벡터화의 적용 대상은 GET 방식의 URL Query 문 자체를 그 대상으로 하였으며 정확한 모델을 생성하기 위해 IDS에 탐지된 공격 문과 웹 요청의 정상 Query를 정확하게 수동으로 구분하여 등록하였다. 이를 위해서 다량의 서비스 데이터를 사용하기보다는 약 3가지 수준의 접근 도메인에서 수집

된 데이터를 이용하여 학습을 진행하였다.

수집된 URL은 Table 1.의 형태로 공격 데이터와 비공격 데이터의 Labeling을 위해서 현재 존재하고 있는 XSS, SQL Injection, Directory Traversal 등의 사전에 탐지된 공격이 수행된 웹 로그를 육안으로 재 검토하여 약 4,200개를 공격으로 분류하여 등록하였고 웹 서버의 정상적인 웹 요청 약 4,200건을 정상으로 분류하여 등록하였다. 공격문에는 다양한 공격을 탐지할 수 있게 하기 위해서 다양한 공격 도구에서 사용하는 공격 패턴도 일부 입력하였다[13].

4.2.2 웹 로그의 재처리

일반적인 웹 로그의 포맷은 다양한 변수들로 이루어져 있으며 데이터 전송을 위해 URL Encoding을 사용한다. 이러한 웹 요청을 Decoding 하면 다양한 특수문자 등이 포함되어 있다. 더욱이 여기에 공격 구문이 추가될 경우 기계가 정상과 비정상을 구별하기 더욱 힘들어지게 된다.

Table 1.에 정상 웹 요청 구문에 MySQL 서버를 공격하기 위한 공격 구문을 URL Decoding 해보면 ″′ and 1=0 union select null, concat(user,′:′,password) from users #″와

```
spstr = {"!" : "spExclamation"
,"#" : " spHashtag "
,"," : " spComma "
,"$" : " spDollar "
,"%" : " spPercent "
,"'" : " spApostrophe "
,"(" : " spLeftparenthesis "
,")" : " spRightparenthesis "
,"*" : " spAsterisk "
,"+" : " spPlussign "
,"-" : " spHyphenminus "
,"." : " spFullstop "
,":" : " spColon "
,"_" : " spLowline "
,";" : " spSemicolon "
,"<" : " spLessthan "
,"=" : " spEqual "
,">" : " spGreaterthan "
,"@" : " spAt "
,"`" : " spGraveaccent "
,"\\" : " spBackslash "
}
```

Fig. 4. Special Character Replacement

같이 MySQL의 정보를 탈취하기 위한 공격 코드임을 알 수 있다.

실제 공격 구문에는 다양한 특수 문자가 사용되고 정상적인 웹 요청에도 사용자의 입력에 의한 정상적인 특수문자의 입력 등이 존재할 수 있다. 이러한 값들을 모두 Vector로 변환하여 모델을 생성하여야 했다.

Word to vector를 적용하기 위해 NLP를 이용하여 단어를 토큰으로 추출할 경우 특수문자는 모두 제거될 것이나 웹 공격의 많은 부분이 특수문자로 이루어져 있는 만큼 특수문자가 제거된다면 정확한 분류가 힘들어지게 되며 특수문자에 대한 정확한 측정이 가장 중요한 요소인 만큼 이를 고려하기 위해 특수문자를 모두 기계에서 인식할 수 있게 Fig 4.와 같이 Query문 상의 특수문자를 모두 변환한 다음 별도로 저장하여 사용 하였다.

4.2.3 Word to vector의 학습

기계 학습을 위한 사전 데이터 준비와 지도 학습을 위한 Labeling이 완료되었으므로 실질적인 Word to Vector 모델을 생성한다.

Word to vector을 사용하여 모델을 구성하는 경우 단어들을 백터화하여 서로 간의 관계를 산출하며 이때 Word to Vector 분석 라이브러리 상에서 다양한 변수를 설정해야 하며 그중에서 몇 가지 중요한 변수에 대해서 설정을 하였다.

대표적으로 Table 5.의 Dimension Size는 벡터화 시의 차원을 과도하게 설정할 경우 학습 시간이 오래 걸리거나 제대로 된 백터화가 이루어지지 않으므로 word2vec 라이브러리에서 권고하는 100개~300개 사이를 테스트하여 200개 차원을 설정하였다. 그리고 결과에 크게 영향을 미치는 것 중에 하나로 Minimum Count가 있으며 금번의 요청문의

Table 5. Word to Vector Parameters [14]

Parameter	Description
Dimension size	Dimensionality of the word vectors
windows size	Maximum distance between the current and predicted word within a sentence
Minimum count	Ignores all words with total frequency lower than this

벡터화에서 결과에 가장 큰 영향을 미쳤다. 공격 문은 일반적이지 않은 문자열이 주로 사용되므로 만약 해당 수치를 크게 잡을 경우 공격 문들이 제대로 벡터화되지 않는 문제가 발생하였다. 수치를 20-30 수준으로 설정하였을 경우 공격 문이 대부분 제외되었고 수치를 5-7사이로 설정하여 사용하였다. 유일한 공격 문이 있는 것을 고려하여 1-3 사이로 설정한 경우 오히려 오탐율이 증가하였다.

측정된 수치를 기반으로 Labeling 된 데이터에서 497,472개의 단어와 8,494개의 문장을 추출하였고 대해서 학습을 완료한 이후 모델을 생성하였다.

4.2.4 Random Forest VS SVM 적용

생성된 모델을 기반으로 하여 실제 분류를 위해서 Random Forest와 SVM을 사용하여 공격 구문을 분류해 보았다.

Random Forest의 경우 입력된 총 8,757개의 웹 요청 중에서 3.2%인 288건을 공격으로 분류했고 이중 실제 공격이 271건으로 약 94%의 정확도를 보여 주었다. 남은 웹 로그를 분석해본 결과 14건 정도가 탐지되지 않아 미탐율은 약 0.1%였다. 이에 대비 SVM의 경우 8,757개 중 57건을 공격으로 분류하였고 이 중에서 실제 공격은 26건으로 분류 개수 대비 약 45%의 정확도와 전체 공격 대비 9% 수준의 탐지율을 보여 주었다. 동일한 벡터 데이터와 웹 URL Query를 사용하였지만, 머신러닝 알고리즘의 특성으로 인해 Random Forest가 더 뛰어난 결과를 나타내 주었으며 Random Forest의 경우 정상 요청문과 공격 구문을 더 학습시킬수록 실제 공격 탐지율을 더 향상할 수도 있다. 실제 탐지된 웹 로그는 Fig 5.와 같으며 탐지된 문자열에 대한 검토 결과 존재하지 않는 페이지에 대한 접근 시도 즉 스캐너를 이용한 정보 수집과 Command Injection 공격을 수행한 것이 탐지되었다.

```
/cgi-bin/awstats.pl?configdir=|echo;echo+START;uname%20-a%3bid;
/cgi-bin/awstats.pl?pluginmode=:system(\"echo+START;uname%20-a%
/cgi-bin/awstats.pl?update=1&logfile=|echo;echo+START;uname%20-
/w1ltw1lt.at.ISC.SANS.DFind:)
/x0x0x0x0x0x0x0x0/ThisFileMustNotExist
http://88.191.24.10:80/mysql/dbadmin/
http://88.191.24.10:80/mysql/mysqlmanager/
http://88.191.24.10:80/mysql/sqlmanager/
http://sevy.eu.org/azenv.php
http://88.191.24.10:80/phpmyadmin/
/w00tw00t.at.ISC.SANS.test0:)
http://proxyjudge1.proxyfire.net/fastenv
/w00tw00t.at.ISC.SANS.DFind:)
```

Fig. 5. Detected Attack Log

V. 결 론

도출된 최종 모델은 Word to Vector를 이용하여 공격 구문을 기계가 인식하도록 하고 이를 지도학습을 이용한 분류를 통해 웹 요청에서 공격을 탐지하는 것을 제시한다.

웹 요청에서 발견할 수 있는 다양한 Feature를 이용하여 비정상적인 웹 요청을 탐지하고자 하였으나 패턴화를 통한 지도학습의 경우 비정상을 모두 분류하기 힘들었다. 이에 일반적인 웹 공격의 경우 동일한 문자열 패턴을 띄는 경우가 많은 점을 이용하여 기존에 존재하는 다양한 공격 구문을 공격으로 정하고 정상 구문을 정상으로 Labeling 하여 각 문자열을 백터화 하여 인식 시켜 학습하였다. 그 결과 패턴에 등록된 공격 구문의 경우 정확하게 인지할 수 있었고 추가적으로 공격 문과 유사한 형태도 탐지할 수 있었다.

다만 일부 특수 문자가 많이 사용된 웹 요청의 경우 정상 문장임에도 공격 문으로 인식되는 경우가 있었으며 이러한 경우는 모두 Labeling을 통해 정상 요청으로 분류해 주지 않으면 오탐이 많아지는 단점이 존재하였다.

해당 모델은 추가 목표였던 대단위 트래픽 하에서도 지속적인 정상 요청에 대한 예외 학습을 통해 적용이 가능하다고 판단되며 패턴에 의존하지 않고도 높은 확률로 웹상의 공격을 탐지할 수 있어진다. 이를 통해 기존의 보안 장비에서 제공할 수 없는 Zero-Day 공격 등을 탐지할 수 있다고 생각하며 HTTP 응답에 이를 적용하여 학습한다면 정상/비정상을 Labeling 하는 것으로 정보의 유출 탐지와 스팸, 악성 메일의 탐지나 다양한 비즈니스 구성 요소상에서 발생할 수 있는 위협을 사전에 탐지할 수 있는 시스템에 적용할 수 있다고 생각된다.

References

[1] Dragos Gravrilut, Mihai Cimpoesu, Dan Anton, Liviu Ciortuz, "Malware Detection Using Perceptrons and Support Vector Machinesg", 2009 Computation World: Future Computing, Service Computation, Congnitive, Adaptive, Content, Patterns, pp. 283-288, Dec. 2009

[2] Chan Kyou Hwang, Jong Kyu Seong, Min Hyung Lee, Jae Hyung Yoo, "Design of Implementation on real-time Anomaly Traffic Lookup & Analysis System", The Committe on Korean Network Operations and Management Review Vol.10, No. 1, pp. 43-55, Aug. 2007
[3] Wonchul Kang, Yeonhee Lee, Youngseok Lee "A Hadoop-based Traffic Analysis System Architecture for Multiple Users", Proceedings of the Korean Information Science Society Conference Vol. 38, No. 1, pp. 252-255, Jun, 2011
[4] Taeshik Shon, Jongsub Moon, "A hybrid machine learning approach to network anomaly detection", Information Sciences Journals Vol. 177, Issue. 18, pp. 3799-3821, Sep. 2007
[5] Jesus Mena, Machine Learning Forensics for Law Enforcement, Security and Intelligence, CRC Press, Aug. 2011
[6] Steven Bird, Ewan Klein, Edward Loper, Natural Language Processing with Python, O`Reilly Media Inc, Jun. 2009
[7] Hae-Duck J. Jeong, Myeong-Un Ryu, Min-Jun Ji, You-Been Cho, Sang-Kug Ye, Jong-Suk R. Lee "DDoS Attack Analysis Using the Improved ATMSim" , Journal of Internet Computing and Services Vol. 17, No. 2, pp. 19-28, Apr. 2016
[8] Andreas Muller, Sarah Guido, Introduction to Machine Learning with Python, O`Reilly Media Inc, May. 2016
[9] Chinuk Lee, Kook Hyun Yoo, Byeong Min Mun, Suk Joo Bae "Informal Quality Data Analysis via Sentimental analysis and Word2vec method", Journal of the Korean society for quality management Vol. 45, No. 1, pp. 117-128, Mar. 2017
[10] Hadi Pouransari , Saman Ghili "Deep learning for sentiment analysis of movie reviews", Computational and Mathematical Engineering, Stanford University, Aug. 2019
[11] Gajanan Shinje, Sanjhya S. Waghere, "Analysis of SQL Injection Using DVWA Tool", Annals of Computer Science and Information Systems Vol. 10, pp. 107-110, Mar. 2017
[12] Gensim Word2vec embeddings, "word2vec", https://radimrehurek.com/gensim/models/word2vec.html, Jul. 2019
[13] Github Web Attack Payloads, "Attack payload", https://github.com/foospidy/payloads, May. 2020
[14] Yousef Bakhdlaghi, "Snort and SSL/TLS Inspection", SANS Institute, 11200 Rockville Pike, Suite 200, North Bethesda, MD 20852 , 23 pages, Apr. 2017
[15] Dharmaraj R. Patil, Jayantrao B. Patil, "Feature-based Malicious URL and Attack Type Detection Using Multi-class Classification", The ISC International Journal of Information Security Vol. 10, No. 2, pp. 141-162, Jul. 2018
[16] WENCHUAN YANG, WEN ZUO, BAOJIANG CUI, "Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network", IEEE Access Vol. 7, pp. 29891-29900, Mar. 2019
[17] Mohammad Almseidin, Maen Alzubi, Szilveszter Kovacs, Mouhammd Alkasassbeh, "Evaluation of Machine Learning Algorithms for Intrusion Detection System", IEEE 15th International Symposium on Intelligent Systems and Informatics

(SISY), pp. 277-282, Oct. 2017
[18] Yao Pan, Fangzhou Sun, Zhongwei Teng, Jules White, Douglas C. Schmidt, Jacob Staples, Lee Krause "Detecting web attacks with end-to-end deep learning" Journal of Internet Services and Applications Vol. 10, Article No. 16, Aug. 2019

〈저자소개〉

김 형 기 (Hyung-gi Kim) 정회원
2003년 8월: 부산 동의대학교 컴퓨터 공학과 졸업
2016년 8월: 성균관대학교 정보통신대학원 정보보호학과 석사 수료
2008년 4월~2011년 2월: NHN Japan (Line) 보안부서 근무
2011년 2월~2012년 12월: Naver Cloud Platform 보안부서 근무
2012년 12월~현재: KAKAO Corp. 보안부서 근무
〈관심분야〉 정보보호, 침입탐지, 어뷰징 탐지, 기계학습, 데이터 처리

김 문 현 (Moon-hyun Kim) 정회원
1978년 2월: 서울대학교 전자공학과 졸업
1980년 2월: KAIST 전기 및 전자공학 석사
1988년 2월: University of Southern California 컴퓨터공학 박사
1988년 3월~현재: 성균관대학교 소프트웨어대학 교수
성균관대학교 정보통신대학원 교수
〈관심분야〉 인공지능, 기계학습, 정보보호