슬라이딩윈도우 기반 머신러닝을 활용한 웹쉘탐지 방안 연구

김기환O, 이동근*, 이 형*, 신용태(교신저자)**
한국전자통신연구원O
(주)에프원시큐리티*
숭실대학교 컴퓨터공학과**
e-mail: itconsult@hanmail.netO, leedg@f1security.co.kr*, yi@f1security.co.kr*, shin@ssu.ac.kr**

A Study on Sliding Window based Machine Learning for Web Shell Detection

Kihwan KimO, Lee DongGeun*, Hyoung Yi*, Yongtae Shin**
ETRIO
F1security Corp*
Dept. of Computer Science, Sungsil University**

● 요 약 ●

본 논문에서는 웹쉘을 탐지하기 위한 방법 중 하나로 슬라이딩윈도우 기반 머신러닝을 활용하는 방안을 제안하고자 한다. 웹 공격에 많이 활용되는 웹쉘의 탐지를 위하여 제안하는 슬라이딩윈도우 기반의 탐지 기법은 시간이 지남에 따라 발전해가는 웹쉘 탐지 우회 기술에 대응하여 보다 정확한 탐지를 제공하는 기술이며, 이를 기반으로 웹쉘의 다양한 변종 또한 탐지할 수 있다. 본제안의 경우 코드의 부분별 위험도를 측정 및 제공하여 보다 효과적으로 대응할 수 있을 것으로 전망된다.

키워드: 웹쉘, 웹쉘 탐지, 머신러닝,악성코드,WebShell, 슬라이딩윈도우

I. Introduction

웹쉘(Web Shell)이란 공격자가 악의를 가지고 제작한 프로그램 및 스크립트를 정의하는 명칭이며, 기본적인 동작 방법은 백도어와 유사하다. 웹쉘이 고안된 근본적인 이유는 웹쉘을 통해 이격되어 있는 웹서버를 효과적으로 모니터링하고, 문제가 발생했을 시 신속하게 수리하기 위함이나, 현재는 본래의 의도와 달리 원격 컴퓨터를 공격하는 용도로 악용되고 있다.

웹쉘은 주로 Server Side Script 언어 (ASP, PHP, JSP)로 제작되며, 공격자는 웹쉘 스크립트를 작성한 후 타깃으로 삼은 웹서버에 업로드를 하여 공격을 수행한다. 이러한 공격을 방지하기 위한 방안으로 여러 가지 방법론들이 제안되어 왔으며, 웹쉘의 패턴 및 해시를 기반으로 탐지하는 방법, 머신러닝을 활용해 탐지하는 방법 등이 제안되어 활용되고 있다.

본 제안에서는 이러한 웹쉘의 대한 보다 효과적인 대응을 위해 슬라이딩윈도우(sliding window)기법을 적용하여 스크립트를 탐지하는 방법을 제안하며, 이를 활용할 경우 기존에 대비해 향상된 성능과 정확도를 바탕으로 웹쉘에 대한 보다 효과적인 대응이 가능할 것으로 판단된다.

II. Preliminaries

1. Related works

1.1 웹쉘 탐지 우회 기법

웹쉘 탐지방법의 발달과 함께 이를 우회하기 위한 기법들도 발전하고 있으며, 현재는 한 줄 웹쉘(Single-line webshell, One-line webshell)과 같은 형태로 제작 및 배포되어 웹쉘 정규패턴을 이용한 탐지가 매우 제한적인 실정이다. 이는 한 줄 웹쉘의 특성상 변종의 제작이 매우 쉽고, 이러한 변종의 경우 기존에 활용하던 정적 탐지패턴의 정규식을 통해서는 탐지가 불가능하기 때문이다.

뿐만 아니라 다중 명령어로 구성된 웹쉘의 경우에도 공격자가 의도적, 지속적으로 내용을 변형할 경우 탐지가 어려워지며, 이에 따라 스크립트의 어느 부분이 웹쉘로서 악성행위를 수행하는지에 대한 보다 명확한 정의 및 분석이 요구된다.

1.2 머신러닝 기반 웹쉘 탐지 방안

인공지능 머신러닝의 학습 및 추론 기술은 데이터에 내제된 패턴, 규칙, 의미 등을 알고리즘 기반으로 스스로 학습하게 하여 새롭게 입력되는 데이터에 대한 결과를 예측 가능하도록 하는 기술이다.[1]

머신러닝 기반의 웹쉘탐지 방안으로는, 2018년 제안된 방안이 있다.[2] 해당 논문에서는 웹 스크립트에 포함된 글자의 정보량, 단어의 길이, 압축률, opcode의 순서 등을 활용하여 탐지를 수행한다고 명시하였으며, 이러한 방법은 [3], [4]에서 소개되었던 방법론과 더불어 웹쉘 탐지의 성능향상에 기여한다. 본고에서 제안하는 방법도 이와 같이 머신러닝 및 정보이론을 기반으로 하며, 슬라이딩윈도우 기법을 융합하여 전체 스크립트의 대한 위험도를 제시하여 스크립트의 위험도에 대한 보다 직관적인 기준을 제공한다.

III. The Proposed Scheme

슬라이딩윈도우기법은 전통적으로 데이터 통신 및 처리분야에서 사용되던 방법이며, 신호 및 데이터를 다룸에 있어 보다 섬세한 접근이 필요할 때 활용되는 방법이다. 예를 들면 그림 1과 같이 패킷 송수신 과정 중 오가는 패킷에 대한 신뢰성의 보장이 요구될 때, 혹은 그림 2와 같이 신호처리에서 부분적으로 나타나는 노이즈나 신호를 처리할 때 유용하다.

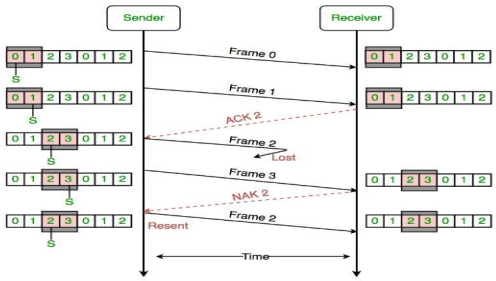

Fig. 1. 슬라이딩윈도우를 이용한 패킷 검증

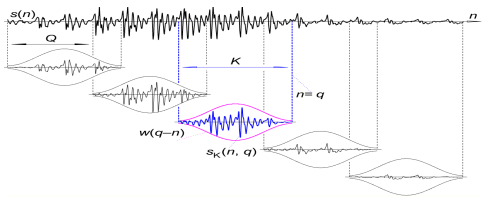

Fig. 2. 신호처리 분야에서의 슬라이딩윈도우

본 논문에서는 이러한 기법을 활용하여 스크립트에 부분적으로 존재하는 웹쉘을 탐지하기 위한 방법을 제안하고 있으며 이에 대한 접근은 그림 3과 같다.

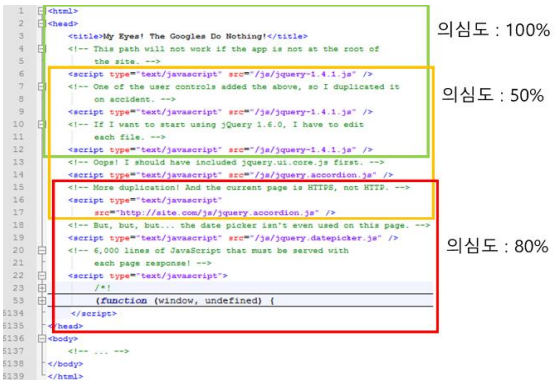

Fig. 3. 슬라이딩윈도우 기반 부분별 코드 위험도 측정

IV. Conclusions

웹쉘(Web Shell)은 자체가 악성코드가 아니기 때문에 백신, 스파이웨어 등에서 탐지가 쉽지않다. 본고에서는 보다 효과적인 웹쉘 탐지를 위한 방안으로 슬라이딩윈도우기반 머신러닝을 활용하는 접근을 제시하였다. 제시된 방법론을 활용할 경우 스크립트 전체에 대한 위험도가 아닌, 스크립트의 각 부분이 내포하고 있는 위험도에 대한 측정이 가능하며, 이로 인해 한 줄 웹쉘이나 탐지 우회를 위해 분산되어 있는 웹쉘 스크립트의 조합을 탐지하는데 있어 증강된 성능을 나타낼 수 있을 것으로 전망된다. 슬라이딩 윈도우 기법은 일부 제한된 경우에는 잘 동작하지만, 일부부문은 동작에 문제가 있을수 있으므로 다양한 머신러닝기법의 융합을 통하여 탐지 기법을 적용하는 연구를 지속적으로 해야 한다.

ACKNOWLEDGE

본 논문은 2018년 “산학연협력기술개발사업”의 지원을 받음

REFERENCES

[1] KihwanKim, et al. "Development of artificial intelligence application of web shell collection and analysis system“,KSii,2018.

[2] Cui, Handong, et al. "Webshell Detection Based on Random Forest-Gradient Boosting Decision Tree Algorithm." 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC). IEEE, 2018.

[3] Likarish, Peter, Eunjin Jung, and Insoon Jo. "Obfuscated malicious javascript detection using classification techniques." 2009 4th International Conference on Malicious and Unwanted Software (MALWARE). IEEE, 2009.

[4] Maiorca, Davide, et al. "Detection of Malicious Scripting Code Through Discriminant and Adversary-Aware API Analysis." ITASEC. 2017.