WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Online Cybersecurity for Malware Detection.

Sanjay S, Srinighil J[1], Vikram R[2], Pratheesh V R.[3]

[1,2,3]Dhirajlal Gandhi College of Technology, Salem.

A B S T R A C T

Malware code detection is a crucial task in the field of cybersecurity. Traditional methods for detecting malware involve signature-based detection and heuristics-based detection, which can be easily bypassed by sophisticated malware. In recent years, deep learning has emerged as a promising approach for malware code detection due to its ability to automatically learn complex features. Here we propose stacked bidirectional long short-term memory and generative pre-trained transformer based deep learning language models for detecting malicious code online without installing any antivirus software. The proposed algorithms, namely the bidirectional long short-term memory model and the generative pre-trained transformer to detect malicious code pieces by examining assembly instructions obtained from static analysis results of Portable Executable Files. BiLSTM model processes a sequence of input elements across time to learn and analyse the patterns. In contrast, the transformers-based GPT model enables modelling long dependencies between input sequence elements with parallel sequence processing, in which sequential data constituents can connect with others simultaneously. The ultimate objective of using deep learning for malware detection is to enhance cybersecurity by improving the accuracy, speed, and scalability of malware detection, and thereby reduce the risk of cyber attacks and data breaches.

Keywords: Malware, Cyber Security, BISLM, GPT.

1. Introduction

Malware (short for “malicious software”) is a file or code, typically delivered over a network, that infects, explores, steals or conducts virtually any behavior an attacker wants. And because malware comes in so many variants, there are numerous methods to infect computer systems. It has been designed to achieve the goals of attackers. These goals include disturbing system operations, gaining access to computing system and network resources, and gathering personal sensitive information without user’s permission.

Types of Malware

Here’s a list of the common types of malware and their malicious intent

Trojans

A Trojan (or Trojan Horse) disguises itself as legitimate software with the purpose of tricking you into executing malicious software on your computer. Trojans are commonly downloaded through email attachments, website downloads, and instant messages.

Spyware

Spyware invades your computer and attempts to steal your personal information such as credit card or banking information, web browsing data, and passwords to various accounts.

Adware

Adware is unwanted software that displays advertisements on your screen. Adware collects personal information from you to serve you with more personalized ads.

Rootkits

Rootkits enable unauthorized users to gain access to your computer without being detected.

Motivation and Background

The computer was attacked in just 8 hours of installation and in 21 days the computer was attacked 20 times and compromised 40 days after installation. The constant growth of Internet users and the provision of online services such as banking and shopping services, provide hacking criminals with a suitable environment to perform their cybercrimes, which leads to a rise in the expenses that are paid to protect the systems. The international damage cost that has been caused by cyber maliciousness takes the attention of the researchers due to its rapid growth. In 2021, this cost is predicted to be around 6 USD trillion according to Cybersecurity Ventures Official Annual Cybercrime Report. Malware is considered the biggest threat to cybersecurity and

falls under several types such as viruses, worms, trojan horses, rootkits, and ransomware since malware causes direct harm to the systems or steals their sensitive information.

Deep Learning

Deep learning is a special machine learning approach that facilitates the extraction of features of a high level of abstraction from low-level data. Deep learning has proven successful in computer vision, speech recognition, natural language processing and other tasks. It works best when you want the machine to infer high-level meaning from low-level data. For image recognition challenges, like ImageNet, deep learning-based approaches already surpass humans. It is natural that cybersecurity vendors tried to apply deep learning for recognizing malware from low-level data. A deep learning model can learn complex feature hierarchies and incorporate diverse steps of malware detection pipeline into one solid model that can be trained end-to-end.

Problem Statement

Malicious codes also gained the ability to spread rapidly in computer networks due to enhanced connectivity of end-user computers and servers, cloud systems, smartphones, and IoT devices. Model building for malware detection usually begins with feature extraction, as specified by either static or dynamic analyses, and sometimes hybrid analysis. The signature-based malware detection is straightforward and fast, yet it may be ineffective against sophisticated malware or overlook relations. The Machine Learning (ML) algorithms, in particular Deep Learning (DL) algorithms, have been deployed to eliminate the drawbacks of traditional, signature-based malware detection. In this project, we extract assembly codes using an open-source disassembler. This tool creates sequences as documents or sentences. Those data are then used for model development, given that the assembly code provides accurate information for obtaining critical coding patterns. we employ the disassembler output as input data to build a language model assisted with word embedding in a similar way to processing natural language. Then, by utilizing this language model, we aim to identify whether an executable file is malicious or benign.

Objective of the Project

The objective of the project is to design, development and an interactive visualization platform for hybrid analysis and diagnosis of malware and prevent from malware attack. To detect and prevent malicious code from functioning. To develop a model that predicts a label (malware or benign) of a given byte stream of a non-executable

2. Literature survey

2.1. A METHOD FOR AUTOMATIC ANDROID MALWARE DETECTION BASED ON STATIC ANALYSIS AND DEEP LEARNING

AUTHOR : MÜLHEM İBRAHIM; BAYAN ISSA

This rise of the usage of smartphones generally, and the Android system specifically, leads to a strong need to effectively secure Android, as the malware developers are targeting it with sophisticated and obfuscated malware applications. Consequently, a lot of studies were performed to propose a robust method to detect and classify android malicious software (malware). Some of them were effective, some were not; with accuracy below 90%, and some of them are being outdated; using datasets that became old containing applications for old versions of Android that are rarely used today

In this paper, static analysis is used and a functional API deep learning model is proposed, which takes as inputs the most useful observed features of android applications, and those are: the file size, Permissions, services, API function calls, broadcast receivers, Opcode sequences, and the fuzzy hash, which is used for similarity detection. They are automatically extracted using Bash script and Python3, which execute commands provided by the Androguard tool..

2.2. MALWARE DETECTION USING BYTE STREAMS OF DIFFERENT FILE FORMATS

AUTHOR: YOUNG-SEOB JEONG; SANG-MIN LEE

Web users are vulnerable to non-executable files such as Word files and Hangul Word Processor files because they usually open such files without paying attention. As new infected non-executables keep appearing, deep-learning models are drawing attention because they are known to be effective and have better generalization power. Especially, the deep-learning models have been used to learn arbitrary patterns from byte streams, and they exhibited successful performance on malware detection task

This paper aims at solving the malware detection task that is basically a binary classification; we want to develop a model that predicts a label (malware or benign) of a given byte stream of a non-executable. The author investigates two different non-executable formats (e.g., PDF and Hangul Word Processor (HWP)), and explain a motivation of using the two different formats for malware detection. The author demonstrates the benefit of it by experimental results of malware detection using our annotated datasets.

3. Existing System

Anti-virus companies mainly use signature-based detection techniques (it is a technique in which detection of malware is done based on features extracted from previously known malware) to capture malware, but using this technique only known malware can be detected. Zero-day malware (new and unseen malware) can't be detected using this approach. Moreover, malware writers practice evasion techniques like encryption and obfuscation to prevent them

from being detected at an early stage. After knowing the catastrophic effects of malware, it is necessary to protect systems from malware. The application of machine learning techniques to malware detection has been an active research area for about twenty years. Researchers have tried to apply various well-known techniques such as Neural Networks, Decision Trees, Support Vector Machines (SVM), ensemble methods and many other popular machine learning algorithms. Recent survey papers provide comprehensive information on malware detection techniques using machine learning algorithms.

Disadvantages

- Organizations are challenged to find, train, and retain malware analysis staff
- Malware analysis tools lack automation, integration, and accuracy
- Malware analysis can become a time-consuming and error-prone manual process across multiple disparate tools and disconnected workflows

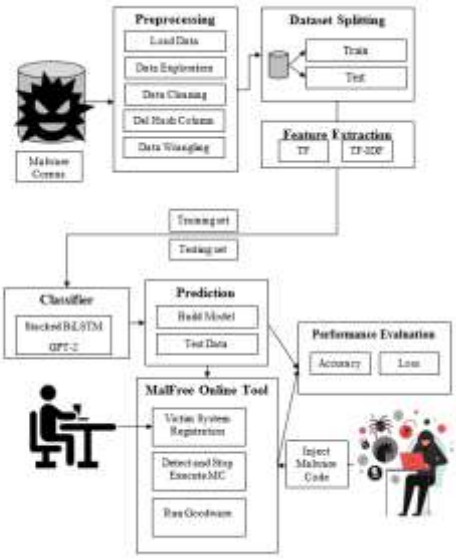
Recommended system

This project proposes malware detection approaches using natural language processing (NLP) techniques with DL algorithms. The proposed algorithms, namely proposes a stacked bidirectional long short-term memory (Stacked BiLSTM) and generative pre-trained transformer based (GPT-2) deep learning language models for detecting malicious code. Developed language models using assembly instructions extracted from .text sections of malicious and benign Portable Executable (PE) files. BiLSTM model processes a sequence of input elements across time to learn and analyze the patterns. In contrast, the transformers-based GPT-2 model enables modeling long dependencies between input sequence elements with parallel sequence processing in which sequential data constituents can connect with others simultaneously. Then use the perspective of NLP modeling by DL to extract similar characteristics, i.e., syntactic and semantic characteristics of assembly instructions. This models were designed to effectively learn and extract the features and characteristics of assembly language and classify the polarity of files.

Benefits

- Detects mismatched CPU types of modules
- MalFree doesn't interrupt the Goodware Portable Executable
- File(exe).

System Architecture



5. Objectives

- The objective of the project is to design, development and an interactive visualization platform for hybrid analysis and diagnosis of malware and prevent from malware attack.
- . To detect and prevent malicious code from functioning
- To develop a model that predicts a label (malware or benign) of a given byte stream of a non-executable

Acknowledgements

We would like to express our deep and sincere gratitude to professor Mr.R.Mahendran., for giving the opportunity and guidance throughout this research.

Conclusion

This project introduces MalFree, an interactive visualization platform for hybrid analysis and diagnosis of malware. This approach first represents the behavioral properties of the major malware classes (such as Trojan or backdoor), aiming to capture the common visual signatures of these malicious applications. MalFree implements a web-based prototype for demonstrating our approach to analyzing 60 malware samples from seven different classes. We focused on opcodes and operands, instead of opcodes only, to develop stacked bidirectional long short-term memory (BiLSTM) models and the decoder-based transformers generative pretrained transformers 2 (GPT-2) models. The resulting accuracy rate 95.4% shows that it is possible to classify malicious and benign assembly codes by GPT-2 with a custom pretrained model. By experimental results, we showed that using byte streams of different formats may contribute to performance improvements. This also allowed for faster detection of malware classes, permitting a quicker response in anti-malware cybersecurity applications. Overall, the application of this project can help identify malware types faster, prevent from malware attack and more accurately than contemporary approaches which can help save time when defending against malwares.

References

1. Caviglione, L.; Choras, M.; Corona, I.; Janicki, A.; Mazurczyk, W.; Pawlicki, M.; Wasielewska, K. Tight Arms Race: Overview of Current Malware Threats and Trends in Their Detection. IEEE Access 2021, 9, 5371–5396. [CrossRef]

2. Morgan, S. Cybercrime Damages $6 Trillion by 2021. 2017. Available online: https://cybersecurityventures.com/hackerpocalypsecybercrime- report-2016/ (accessed on 15 July 2021).

3. Cannarile, A.; Dentamaro, V.; Galantucci, S.; Iannacone, A.; Impedovo, D.; Pirlo, G. Comparing Deep Learning and Shallow Learning Techniques for API Calls Malware Prediction: A Study. Appl. Sci. 2022, 12, 1645. [CrossRef]

4. Villalba, L.J.G.; Orozco, A.L.S.; Vivar, A.L.; Vega, E.A.A.; Kim, T.-H. Ransomware Automatic Data Acquisition Tool. IEEE Access 2018, 6, 55043–55051. [CrossRef]

5. Urooj, U.; Al-Rimy, B.A.S.; Zainal, A.; Ghaleb, F.A.; Rassam, M.A. Ransomware Detection Using the Dynamic Analysis and Machine Learning: A Survey and Research Directions. Appl. Sci. 2022, 12, 172. [CrossRef]

6. Hansen, S.S.; Larsen, T.M.T.; Stevanovic, M.; Pedersen, J.M. An approach for detection and family classification of malware based on behavioral analysis. In Proceedings of the 2016 International Conference on Computing, Networking and Communications (ICNC), Kauai, HI, USA, 15–18 February 2016; pp. 1–5. [CrossRef]

7. Vignau, B.; Khoury, R.; Halle, S. 10 Years of IoT Malware: A Feature-Based Taxonomy. In Proceedings of the 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), Sofia, Bulgaria, 22–26 July 2019; pp.

458–465. [CrossRef]

8. Asam, M.; Hussain, S.J.; Mohatram, M.; Khan, S.H.; Jamal, T.; Zafar, A.; Khan, A.; Ali, M.U.; Zahoora, U. Detection of exceptional malware variants using deep boosted feature spaces and machine learning. Appl. Sci. 2021, 11, 10464. [CrossRef]

9. Sahay, S.K.; Sharma, A.; Rathore, H. Evolution of Malware and Its Detection Techniques. In Advances in Intelligent Systems and Computing; Springer: Singapore, 2020; Volume 933, pp. 139–150.

10. Kakisim, A.G.; Nar, M.; Sogukpinar, I. Metamorphic malware identification using engine-specific patterns based on co-opcode graphs. Comput. Stand. Interfaces 2019, 71, 103443. [CrossRef]