# Behavioral detection of malicious activity in an EDR context

Ianău Andrei-Ioan

\* University of Medicine, Pharmacology, Sciences and Technology “George Emil Palade”

Instructor: Sándor Miklós Szilágyi

***Abstract*-** The proliferation of malware has made it necessary to improve automatic detection and classification methods. Machine learning, particularly the use of neural networks and deep learning techniques, has proven to be effective in identifying patterns in large datasets. These techniques, which have advanced significantly in recent years, have demonstrated superior classification accuracy in areas such as computer vision and natural language processing. The increase in the number and variety of malware samples necessitates the use of machine learning techniques in order to accurately detect and classify them. Moreover, the increasing capabilities of malicious actors to hide and obfuscate the malware comes as a hindrance in detecting them. Even so, the behavior of the malware remains the same so the dynamic analysis proves useful in identifying and stopping 0 day exploits and advanced persistent threats. The behavior of a program is intrinsically the same across multiple layers of intent if the scope remains the same.

***Index Terms***- behavior, machine language processing, malware, natural language processing.

Introduction

Similar to the NLP process Teufl et al. call the process Machine Language Processing (MLP). In [1] the authors use the e-Participation analysis architecture, extract the various NLP techniques and adopt them for the malware analysis process. In [2] they transfer the performance improvements achieved in the area of neural networks to model the execution sequences of disassembled malicious binaries. Pre-trained neural Language Models (PTLM), such as CodeBERT, are recently used in software engineering as models pre-trained on large source code corpora as described in [3].

The authors in [4] present hybrid-Flacon, a hybrid pattern Android malware detection and categorization framework. In the [5] the researcher explores an ensemble mechanism, which presents how the combination of byte-code and native-code analysis of Android applications can be efficiently used to cope with the advanced sophistication of Android malware. From [6] we get to see how they present a simple gradient-descent based algorithm for finding adversarial samples, which performs well in comparison to existing algorithms.

Huang et al. design Gossip in [7], a novel approach to automatically detect malicious domains based on the analysis of discussions in technical mailing lists (particularly on security-related topics) by using natural language processing and machine learning techniques. In order to reduce the manpower of feature engineering prior to the condition of not to extract pre-selected features In the paper [7] they develop a coloR-inspired convolutional neuRal networks (CNN)-based AndroiD malware Detection (R2-D2) system.

In the paper [8] Sewak et al. investigate and compared one of the Deep Learning Architecture called Deep Neural Network (DNN) with the classical Random Forest (RF) machine learning algorithm for the malware classification. Similar to natural language processing the authors from the paper [9] propose a novel and efficient approach to perform static malware analysis, which can automatically learn the opcode sequence patterns of malware. Other influential work includes findings in the paper [2]. To this end the researchers from the paper [10] propose a joint learning approach to generating instruction embeddings that capture not only the semantics of instructions within an architecture, but also their semantic relationships across architectures. ([**Uhlig et. al., 2022**](https://www.paperdigest.org/paper/?paper_id=arxiv-2208.11367)) propose deep learning approximate matching (DLAM), which achieves much higher accuracy in detecting anomalies in fuzzy hashes than conventional approaches.

([**Zhang et. al., 2019**](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1016_j.future.2018.12.028)) propose a cloud-based malware detection system called SaaS by leveraging and marrying multiple approaches from diverse domains such as natural language processing (n-gram), image processing (GLCM), cryptography (fuzzy hash), machine learning (random forest) and complex networks. (**[Sahabandu et. al., 2022](https://www.paperdigest.org/paper/?paper_id=arxiv-2204.06624)**) propose a binary code feature extraction model to improve the accuracy and scalability of ML-based ISA identification methods. ([Nagano et. al., 2017](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1145_3022227.3022306)) propose a new method which automatically detects new malware subspecies by static analysis of execution files and machine learning.

All of the mentioned works use a static analysis approach in detecting the malicious samples. But there are also papers that take into consideration the dynamic analysis methodology to asses if a sample is malicious. ([Wang et. al., 2018](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1109_tifs.2017.2771228)) propose an effective and automatic malware detection method using the text semantics of network traffic. The approach assessed is a novel dynamic malware analysis method, which may generalize better than static analysis to newer variants ([Kim, 2018](https://www.paperdigest.org/paper/?paper_id=arxiv-1802.05412)). To leverage the application of deep learning architectures towards cyber security (**[Vinayakumar et. al., 2019](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1007_978-3-030-16837-7_7)**) consider intrusion detection, traffic analysis and Android malware detection. An anomaly detection method is implemented using Natural Language Processing (NLP) based on Bags of System Calls (BoSC) for learning the behavior of applications on Windows virtual machines running on Xen hypervisor (**[Peddoju et. al., 2020](https://www.paperdigest.org/paper/?paper_id=doi.org_10.14569_ijacsa.2020.0110559)**). ([**Chen et. al., 2022**](https://www.paperdigest.org/paper/?paper_id=arxiv-2208.14221)) present AVMiner, an expansible malware tagging system that can mine the most vital tokens from AV labels. (**[Maniriho et. al., 2022](https://www.paperdigest.org/paper/?paper_id=arxiv-2209.03547)**) present MalBehavD-V1, a new behavioural dataset of Windows Application Programming Interface (API) calls extracted from benign and malware executable files using the dynamic analysis approach. MDTA is the best suitable and manageable approach for analyzing behavioral reports using a machine learning algorithm for providing security measures to identify malware without the intervention of the investigator (**[Vanjire et. al., 2021](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1007_978-981-16-1866-6_10)**). ([**Poudyal et. al., 2021**](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1109_access.2021.3109260)) propose a deep inspection approach for multi-level profiling of crypto-ransomware, which captures the distinct features at Dynamic link library, function call, and assembly levels. (**[Parildi et. al., 2021](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1007_s00521-021-05861-7)**) present an alternative method for malware detection, which makes use of assembly opcode sequences obtained during runtime. Other influential work includes (**[Kolosnjaji et. al., 2016](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1007_978-3-319-50127-7_11)**). ([Qin et. al., 2020](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1109_icbaie49996.2020.00041)) propose a Dynamic Ransomware Detector based on the improved TextCNN(DRDT). ([Zago et. al., 2020](https://www.paperdigest.org/paper/?paper_id=pubmed-32215308)) selected 50 among the most notorious malware variants to be as exhaustive as possible.

Discussion

In [2] the authors look into

Identify the constructs of a Journal – Essentially a journal consists of five major sections. The number of pages may vary depending upon the topic of research work but generally comprises up to 5 to 7 pages. These are:

1. Abstract
2. Introduction
3. Research Elaborations
4. Results or Finding
5. Conclusions

**In Introduction you can mention the introduction about your research.**

1. IDENTIFY, RESEARCH AND COLLECT IDEA

It's the foremost preliminary step for proceeding with any research work

1. WRITE DOWN YOUR STUDIES AND FINDINGS

Now it is the time to articulate the research work with ideas gathered in above steps by adopting any of below suitable approaches:

## A. Bits and Pieces together

In this approach combine all your researched information in form of a journal or research paper. In this researcher can take the reference of already accomplished work as a starting building block of its paper.

Jump Start

This approach works the best in guidance of fellow researchers. In this the authors continuously receives or asks inputs from their fellows. It enriches the information pool of your paper with expert comments or up gradations. And the researcher feels confident about their work and takes a jump to start the paper writing.

## B. Use of Simulation software

There are numbers of software available which can mimic the process involved in your research work and can produce the possible result. One of such type of software is Matlab. You can readily find Mfiles related to your research work on internet or in some cases these can require few modifications. Once these Mfiles are uploaded in software, you can get the simulated results of your paper and it easies the process of paper writing.

As by adopting the above practices all major constructs of a research paper can be written and together compiled to form a complete research ready for Peer review.

1. GET PEER REVIEWED

Here comes the most crucial step for your research publica

1. IMPROVEMENT AS PER REVIEWER COMMENTS

Analyze and understand all the provided review comments thoroughly. Now make the required amendments in your paper. If you are not confident about any review comment, then don't forget to get clarity about that comment. And in some cases there could be chances where your paper receives number of critical remarks. In that cases don't get disheartened and try to improvise the maximum.

**After submission IJSRP will send you reviewer comment within 10-15 days of submission and you can send us the updated paper within a week for publishing.**

This completes the entire process required for widespread of research work on open front. Generally all International Journals are governed by an Intellectual body and they select the most suitable paper for publishing after a thorough analysis of submitted paper. Selected paper get published (online and printed) in their periodicals and get indexed by number of sources.

**After the successful review and payment, IJSRP will publish your paper for the current edition. You can find the payment details at:** [**http://ijsrp.org/online-publication-charge.html**](http://ijsrp.org/online-publication-charge.html)**.**

1. CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.