Preprints.org

Article

Not peer-reviewed version

PDF Malware Detection using Machine learning

Awss AlMahadeen * and mouhammd alkasassbeh

Posted Date: 30 January 2023

doi: 10.20944/preprints202301.0557.v1

Keywords: PDF; Malware; Machine Learning; Python; Random Forest

Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

PDF Malware Detection Using Machine Learning

Awss AlMahadeen * and Mouhammd Alkasassbeh

Princess Sumaya University for Technology Amman Jordan; m.alkasassbeh@psut.edu.jo
* Correspondence: aws202028022@std.psut.edu.jo

Abstract: Portable Document Format (PDF) is one of the most widely used files types worldwide in data exchange, this has encourage hackers to utilize such files to spread any malicious content through PDF, utilizing different methods and techniques to accomplish that, on the other hand, security researches kept trying to improve detection methods to cope up to the rapidly increasing number of malwares daily, one of the commonly used detection technique nowadays is by utilizing artificial intelligence and Machine learning classificat; thision to help detecting PDF Malwares, in this paper, we utilize machine learning classifier Random Forest on a newly released PDF Malware dataset CIC-Evasive-PDFMal2022 to achieve the main goal of detecting malicious PDF documents, results showing a detection accuracy of around 99.5%.

Keywords: PDF; malware; machine learning; python; random forest

I. Introduction

A malware, stands for Malicious software, is a piece of code that cause harm for the infected computer or network, malware can aim to steal secret data, or cause any kind of damage by effecting the performance of PC or even try to gain unauthorized access to the target machine, Malware have improved over the past years to become more and more sophisticated including different types of malicious software including computer viruses, worms, trojans, ransomwares, keyloggers, rootkits and spyware.

For instance, the first created computer virus called "Brain A." appeared in 1968 [1] and was developed by two brothers in Pakistan infecting the boot sector of floppy disk by then without causing any harm for the infected PC, nevertheless, it pointed out that the computer system is vulnerable.

According to Internet Crime Report [2] In 2021, the Internet Crime Complaint Center received around 3,729 new complaints identified as ransomware expected losses of more than $49.2 million, The raising challenge of information security, raised the importance to develop different mechanisms in order to detect Malwares efficiently, malware detection techniques (Figure 1) has improved over time [3], from signature based detection, which typically inspect the metadata and payload of the file and tries to match those to a pre-defined set of malicious signatures/patterns, such method has some limitation as hackers were able to evade detection by obfuscating the malicious content to avoid pattern matching, another limitation is that the signature database has to be updated frequently to keep up with the new malwares firing up on a daily basis, another technique is to use behavior-based detection, where the code has to be run in an isolated environment (i.e sandbox) [4] and monitor the behavior of the program/application, including initiated process, API calls, CPU and memory impact in order to confirm if the program is clean or malicious, however, behavior-based analysis fails to detect malwares, as hackers started to use some evasion techniques, for example, some malwares uses time-delay method, to delay the execution of the malicious content right after it has been classified as clean, other malwares has some advanced features that can detect the existence of a sandbox environment, and thus stays idle until it has been released to the production environment.

© 2023 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

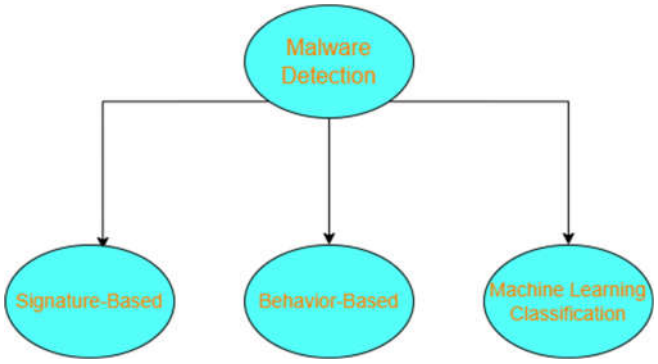

Figure 1. Malware Detection Approaches.

Limitations of the previous detection methods, emerged the use of machine learning classification to address the raising concern of malware detection, as ML will determine whether a file is malicious or benign after being trained on specific dataset extracting the features of the data and making a decision afterwards, there are different machine learning algorithms used in the world of data science, in this paper, we used Random Forest classifier as its known to be one of the most accurate ML algorithms.

PDFs has a different type of structure than regular text files, for instance, PDF contains different layers and data types, in addition, it has specific file structure [Figure 2 PDF File Structure], according to [5] a PDF file contains different layers:

Metadata which includes details about the PDF file such as author field, creation date, keyword and copy rights [6]

Objects Which can be defined as the basic component of PDF file, where each object can contain different types of data, following are the 8 types of objects that build PDF documents:

- Boolean Values
- String: Limited length sequence of bytes
- Names: unique character sequence
- Arrays: 1-Dimensional array containing ordered PDF objects
- Null: if the "null" object is present in the PDF document, that means that the reference object doesn't exist in the PDF file
- Integers and real numbers
- Dictionaries: set of key pair that links each object name to its content
- Streams: Unlimited sequence of bytes, where reading process for such object can be incremental

As explained earlier PDFs has its own structure which makes it support different types of data, following are the four parts of PDF file which is brief in (Figure 2):

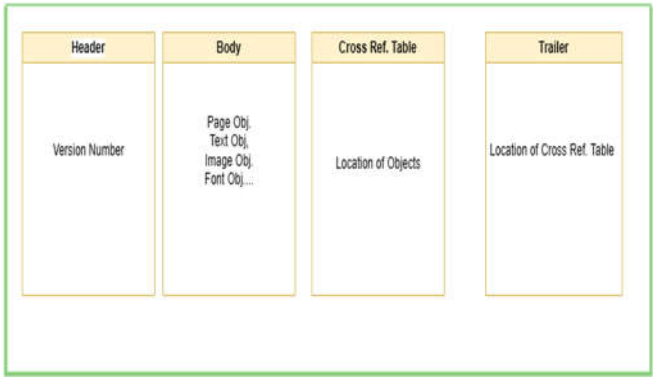

Figure 2. PDF File structure.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

- Header: the first line of PDF file containing the version of the PDF under the format of "%PDF-a.b" where a.b represent the version
- Body: which contains the main content of the PDF document (i.e PDF objects)
- Cross-reference table: contains the location/address of each object of the body part.
- Trailer: contains the offset location of cross reference table, by which a PDF viewer application can start reading the file from the trailer file to proceed to Cross-ref table then the main PDF file object

II. Literature Review

Many researchers have focused on finding the optimal Malware detection method for PDF malware, for instance authors of [7] have presented a comprehensive survey of Malware detection approaches between signature-based, behavior-based and Machine learning based, listing the pros and cons of each approach offering the different Malware obfuscation mechanisms, concluding that signature-based is the most efficient in detecting well-known malicious codes, while behavior-based is more effective in detecting more complex Malwares and zero-day Malwares. Authors of [8] focused on the most recent PDF-Malware detection techniques, including the PDF feature extraction and analysis and surveying the variety of detection approaches including statistical analysis, which may focus on byte0level comparison between Malicious and benign PDFs to detect Malware for example, another methods of detection could be signature matching and ML classification, Marwan Albahar In [9] applied support vector machine (SVM) and convolutional neural network (CNN) machine learning classifiers on two PDF Malware datasets retrieved from Virus Total [10] consist of 10,603 malicious files (collected Dec./2017) and Contagio [11] dataset consist of approximately 20,000 malicious and benign files (collected Nov./2017) , the two models achieved around 100% accuracy for both classifiers.

Other past work focused on feature extraction methods, for instance authors in [12] stressed on enhancing PDF Malware classification by identifying the most valuable features in PDF file using two PDF analysis tools PeePDF and PDFiD then evaluated those features via a wrapper function decreasing the feature set by 60% resulting in top 14 features extracted from PDF documents, and evaluating the classification on a dataset retrieved from VirusTotal using multiple ML classifiers Random-Forest, SVM,DNN and C50 Decision Tree, achieving a max accuracy of 96.8%

Couple of researches focused on the same dataset CIC-Evasive-PDFMal2022 [13], authors of [14] presented a new detection system utilizing "Adaboost" decision tree algorithm, splitting the data into training and testing dataset with 80:20 achieving accuracy of around 98.8% and a low PDF Malware detection overhead in respect to timing of couple of micro seconds, meanwhile in [15] researches used two datasets of Contagio and CIC-Evasive-PDFMal2022 on two different machine learning classifiers of Long Short Term Memory S-LSTM and IWO-S-LSTM achieving an accuracy of 97.06% and 98.20% on our concerned dataset of CIC-Evasive-PDFMal2022.

Authors of [16] proposed a framework to detect PDF Malware by extracting 28 features from PDF files and evaluating the proposed framework on couple of machine classifiers including RF, MLP, SVM and Adaboost, testing the classification on two datasets "Contagio" and "Evasive-PDFMal2022" achieving a max accuracy of 98.44% on the second dataset using Random-Forest ML classifier.

III. Methodology

In this paper we evaluated CIC Evasive-PDFMal2022 on Random Forest Machine learning classification via Python code built from scratch and complied on Jupyter Anaconda 3 compiler involving couple of steps, following a summary of the steps (Figure 3 Flow Diagram):

1- Import collected dataset into python code.
2- Splitting the dataset into features and target arrays dataset.
3- Dataset preprocessing and cleaning.
4- Splitting dataset into Training and testing datasets.
5- Using Random Forest classification on the Training and Testing dataset.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

6- Producing prediction results.

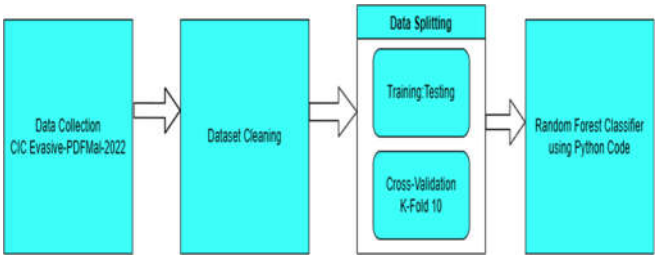

Figure 3. Flow Diagram.

A. ***Dataset****:*

As discussed earlier the dataset using in this paper is the CIC Evasive-PDFMal2022 retrieved by the Canadian Institute for Cybersecurity with the of 11,173 Malicious files from Contagio and 20,000 Malicious file from VirusTotal and 9,109 benign file [11] performing feature extraction of 32 features, employing K-means machine learning after that groups the data into 2 clusters, Samples that fall into the wrong maliciously labeled cluster are considered an evasive set of malicious records. The intuition is that the characteristics of these samples are not very similar to the rest of the class, so they are not clustered. Most of the samples with the same label.

The next step is removing all the duplicate records of the two datasets then combining all those datasets resulting in 10,025 PDF records of which 4,468 benign files and 5,557 Malicious files, therefore this dataset is a balanced dataset, Figure 4 summarizes the steps to generate the new dataset.

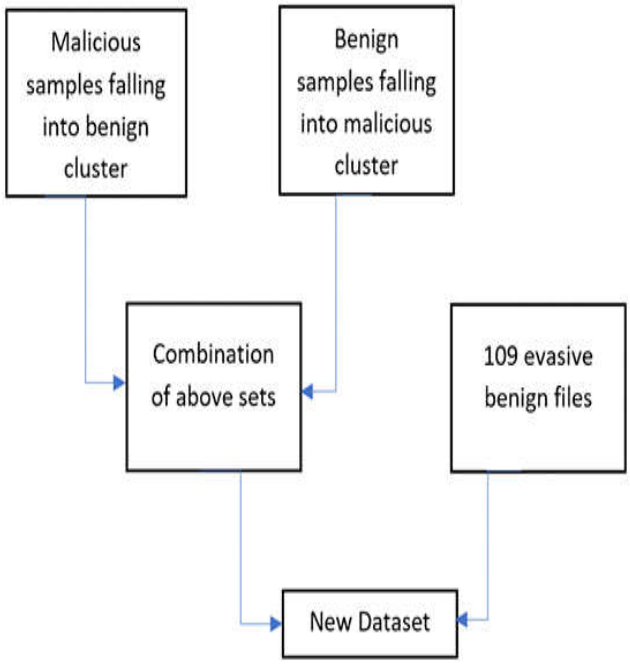

Figure 4. Dataset generation process.

The dataset consists of around 32 features extracted from the collected Malicious and benign PDF files to be used for classification later [11] ,features include general PDF features such as PDF size, encryption, Metadata size, text, header, avg. size of the media, and PDF structural features, such as, number of specific keywords as “stream”, “/JS”, “/Action” and “/AA”, following is the full list of the proposed features in this dataset [13]:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

General features

- PDF size
- title characters
- encryption
- metadata size
- page number
- header
- image number
- text
- object number
- font objects
- number of embedded files
- average size of all the embedded media

Structural features

- No. of keywords "streams"
- No. of keywords "endstreams"
- Average stream size
- No. of Xref entries
- No. of name obfuscations
- Total number of filters used
- No. of objects with nested filters
- No. of stream objects (ObjStm)
- No. of keywords "/JS", No. of keywords "/JavaScript"
- No. of keywords "/URI", No. of keywords "/Action"
- No. of keywords "/AA", No. of keywords "/OpenAction"
- No. of keywords "/launch", No. of keywords "/submitForm"
- No. of keywords "/Acroform", No. of keywords "/XFA"
- No. of keywords "/JBig2Decode", No. of keywords "/Colors"
- No. of keywords "/Richmedia", No. of keywords "/Trailer"
- No. of keywords "/Xref", No. of keywords "/Startxref"

B. Random Forest:

We choose to use Random Forest Classifier among other machine learning classifiers, as found it produced the most accurate results in comparison to all the old related works, in addition, it's a well-known ML classification algorithm which belongs to supervised learning, which basically as the name propose contains a number of decision trees on different subsets of the provided dataset and then calculates the average of prediction accuracy lastly producing a more accurate results.

Other features that lead to choosing Random Forest classifier is that it takes less training time compared to other ML classifiers, it also maintains accuracy whenever a large part of data is missing [17].

C. Evaluation metrics

To demonstrate the classifier performance over the targeted dataset, we used couple of machine learning metrics that helps providing more insights into the Malware/Benign classification, following are the used metrics:

- **Accuracy:**

Represents the ratio of correct predictions over the total made predictions. And it may be expressed mathematically as follows [18]:

$$\text{Accuracy} = (TP+TN)/(TP+TN+FP+FN)$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

- **Recall:**

Represents the ratio of how many of the actual labels were predicted. And it can be represented by the following equation [18]:

$$\text{Recall} = (TP)/(TP+FN)$$

- **Precision:**

Represents the ratio of how much of the predicted is correct. And it can be represented by the following equation [18]:

$$\text{Precision} = (TP)/(TP+FP)$$

D. ***Data preprocessing and cleaning:***

To get the best and most accurate results some data preprocessing and cleaning steps were performed, following are the details of those steps.

Data Cleaning

The raw data records available in the dataset needed some cleaning steps before moving on to processing, cleaning steps included:

1) converted last column to 1 for 'Malicious' and 0 for 'Benign'.
2) Removed some invalid records containing irrelevant data in JavaScript and Object fields.
3) Removed Records containing "Nan" for all the features (representing 'Null' value).

Data Splitting:

For the data splitting into Training/Testing datasets we utilized 2 approaches in order to come with the best results later on classification phase:

a) Data splitting into Training/Testing of ratio of 80%:20%
b) Using K-Fold Cross validation with 10 Folds

As known K-Fold cross validation, as it implies it will cross validate the whole given data by splitting the data into Training: Testing Folds sequentially, where the data is split into k-Folds (in our experiment 10 folds) then we use the first one for testing and other 9 folds for training and note the score, then use the 2nd one for testing and others for training and note the score, …, until the 10th fold after that the average the score.

After performing the data splitting we input the new produced data into classification process with Random Forest Classifier imported into our Python code which will use the training data to predict the Malicious and clean files of the Test data and then outputs the prediction metrics of Precision, Recall, F-score, Confusion Matrix and Accuracy of the prediction results.

In the following section, we will demonstrate the results of the experiment performed on the Evasive-PDFMal2022 dataset.

IV. Results

As discussed earlier we performed PDF Malware detection using Random Forest Classifier on CIC Evasive-PDFMal2022 dataset by building a Python code to perform classification, testing to split the dataset in 2 forms.

a) Training and Testing data of 80:20 ratio and
b) using Cross validation with 10 K-fold,

Results shows an accuracy of 99.5% when using Training: Testing split and around 98.9% when using Cross-Validation of 10 Folds (results of both experiments are detailed in Tables 1 and 2)

In addition, we generated the confusion matrix to show the number of samples classified as True-Positive, False-Positive, True-Negative and False-Negative classified samples, Figure 5 contains visual representation of CM.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

Figure 5. Confusion Matrix.

Table 1. Results with Data splitting 80:20.

Metric	Score
Accuracy	99.5%
Precision	99.35%
Recall	99.72%
F1-Score	99.5%

Table 2. Evaluation Result with K-Fold.

Metric	Score
Accuracy	98.9%
Precision	99.35%
Recall	99.72%
F1-Score	99.5%

In relation to other related work done on the same Dataset the following table (Table 3) containing a comparison of the classification accuracy.

Table 3. Comparison of Accuracy to other work.

Reference	Accuracy
[14]	98.8%
[15]	98.2%
[16]	98.44%
[17]	99.5%
Our Work	99.5% and 98.9%

As observed, our work outstands previous related work on the same dataset by achieving accurate classification towards Malicious and benign PDF files.

V. Conclusions

As observed throughout the research, PDF is one of the most used file formats for data exchange all over the world, this resulted in an increased attack surface which attracted the attention of security hackers to explore new methods for the sake of compromising victims systems for different targets, including causing damage, stealing confidential information or gaining unauthorized access, such new techniques, raised the concern to come up with counter defensive techniques by security researchers, going for static analysis to dynamic analysis up to using Machine Learning algorithms to detect malicious and benign softwares and files accurately.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2023 doi:10.20944/preprints202301.0557.v1

Many previous work focused on finding the optimal and most accurate detection via different ML classification algorithms, focusing on various datasets, in our research, we focused on one of the most recent datasets, one that focus on the PDF malwares that tends to use evasive properties to try to get pass through different detection tools, and tried to find the most successful ML classifier to detect any malicious PDF file comparing to other related work on the same datasets, as conclusion we found that Random-Forest classification achieved the most accurate detection results.

VI. References

1. N. Milosevic, "History of malware," 02 2013.
2. Internet Crime Report, 2021, https://www.ic3.gov/
3. Mouhammd Alkasassbeh, Mohammad A. Abbadi, Ahmed M. Al-Bustanji. " LightGBM Algorithm for Malware Detection." *Applied Sciences* volume 1230 (2022)
4. 14. Or-Meir, O.; Nissim, N.; Elovici, Y.; Rokach, L. Dynamic malware analysis in the modern era—A state of the art survey. ACM Comput. Surv. 2019, 52, 1–48. [CrossRef] 15. Albulayhi, K.; Abu Al-Haija, Q.; Alsuhibany, S.A.; Jillepalli, A.A.; Ashrafuzzaman, M.; Sheldon, F.T. IoT Intrusion Detection Using Machine Learning with a Novel High Performing Feature Selection Method. Appl. Sci. 2022, 12, 5015.
5. Document management – portable document format – part 1: Pdf 1.7. Standard, International Organization for Standardization, Geneva, CH, Mar. 2008.
6. [6]PDF properties and metadata, Adobe Acrobat Accessed 6,Dec 2022
7. Aslan, Ömer & Samet, Refik. (2020). A Comprehensive Review on Malware Detection Approaches. IEEE Access. 8. 1-1. 10.1109/ACCESS.2019.2963724.
8. Elingiusti, Michele & Aniello, Leonardo & Querzoni, Leonardo. (2018). PDF-Malware Detection: A Survey and Taxonomy of Current Techniques. 10.1007/978-3-319-73951-9_9.
9. Albahar, Marwan & Thanoon, Mohammed & Alzilai, Monaj & Alrehily, Alla & Alfaar, Munirah & Al-Ghamdi, Maimoona & Alassaf, Norah. (2021). Toward Robust Classifiers for PDF Malware Detection. Computers, Materials and Continua. 69. 2181-2202. 10.32604/cmc.2021.018260.
10. VirusTotal https://virustotal.com/.
11. Contagio Malware Dump, "External data source," [Online]. Available: http://contagiodump.blogspot.com.au .
12. Falah, Ahmed & Pan, Lei & Huda, Shamsul & Pokhrel, Shiva & Anwar, Adnan. (2021). Improving malicious PDF classifier with feature engineering: A data-driven approach. Future Generation Computer Systems. 115. 314-326. 10.1016/j.future.2020.09.015.
13. CIC-Evasive-PDFMal2022 Dataset CIC-Evasive-PDFMal2022 | Datasets | Canadian Institute for Cybersecurity | UNB
14. Abu Al-Haija, Q.; Odeh, A.; Qattous, H. PDF Malware Detection Based on Optimizable Decision Trees. Electronics 2022, 11, 3142.
15. Chandran, P. & Hema, Rajini & Jeyakarthic, M.. (2022). Invasive weed optimization with stacked long short term memory for PDF malware detection and classification. International journal of health sciences. 4187-4204. 10.53730/ijhs.v6nS5.9540.
16. Issakhani, Maryam & Victor, Princy & Tekeoglu, Ali & Lashkari, Arash. (2022). PDF Malware Detection based on Stacking Learning. 562-570. 10.5220/0010908400003120.
17. S. Y. Yerima, A. Bashar and G. Latif, "Malicious PDF detection Based on Machine Learning with Enhanced Feature Set," 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN), Al-Khobar, Saudi Arabia, 2022, pp. 486-491, doi: 10.1109/CICN56167.2022.10008374.
18. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
19. Han, J., Kamber, M., & Pei, J. (2012). Data mining: concepts and techniques. Elsevier.
20. Internet Security Report, 2021, WatchGuard's Internet Security Report – Q4 2021
21. Pedro Ernesto Aquino, How Has Ransomware Changed Cyber Insurance,2022
22. Alzubi, Jafar & Nayyar, Anand & Kumar, Akshi. (2018). Machine Learning from Theory to Algorithms: An Overview. Journal of Physics: Conference Series. IEEE 1142. 012012. 10.1088/1742-6596/1142/1/012012.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.