International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN(Online): 2347-5552, Volume-12, Issue-1, January 2024
https://doi.org/10.55524/ijircst.2024.12.1.7
Article ID IJIRD-1268, Pages 35-39
www.ijircst.org

Analysis of Cyber Security Threats Using Machine Learning Techniques

Ranjana B Nadagoudar

Assistant Professor, Department of Computer Science & Engineering, Visvesvaraya Technological University, Belagavi, India

Correspondence should be addressed to Ranjana B Nadagoudar; ranjanapriya8@gmail.com

Received 2 January 2024; Revised 15 January 2024; Accepted 26 January 2024

ABSTRACT- Nowadays malware detection is a problem that researchers have tried to solve for so many years by using enormous type of methods. The behaviors of two given malware variants remain similar, although their signatures could also be distinct. The proposed project mainly concentrates on classifying the malware families by considering the malware API sequence or API commands. This type of classification is helpful for the analyst as it helps them to get a better insight into the functioning of the malware.

KEYWORDS- Malware Detection, Malware Family Detection, KNN, SVM, API Calls Argument.

I. INTRODUCTION

A cyber or cyber security threat could also be a malicious act that seeks to wreck data, steal data, or disrupt digital life generally. Cyber-attacks include threats like computer viruses, data breaches, and Denial of Service (DoS) attacks. There are several sorts of computer security threats like Trojans, Virus, Adware, Malware, Root kit, hackers and far more.

Malware detection refers to the method of detecting the presence of malware on a number system or of distinguishing whether a selected program is malicious or benign. In order to guard a computer from infection or remove malware from a compromised computing system, it's essential to accurately detect malware. The proposed project is mainly concentrating on classifying the malware families by considering the malware API sequence or API commands. This type of classification helpful for the analyst as it helps them to get a better insight into the functioning of the malware. This is very helpful for analysts, because just by knowing the class/family of the malware they can have an idea about how to devise sanitation and detection techniques for that malware. Also by knowing the family to which a malware belong we have a general idea about its behavior. This helps in sharing of data between malware analysts.

There are two sorts of detection techniques that are normally employed by malware analysts, static and dynamic detection. Static detection is predicated on specific strings from the disassembled code without executing the binary file. This analysis can quickly capture the syntax but it's easily disturbed by code obfuscation and encryption technology. The second sort of detection is dynamic detection. It analyses the malware behavior like network activities, system calls, and file operations by executing the Malware. This system can detects newly created malware however, it requires more execution time.

II. RELATED WORK

Egele et al. [2] Automatic dynamic malware investigation procedures and techniques have been created; programmed dynamic examination delivers a report for each malware program, enumerating its run-time activities. The information created by these investigation devices clarifies the conduct of the malware program, empowering the convenient and applicable arrangement of countermeasures. Tsyganoket al. [3] the grouping blunder went from practically 9 percent to 22 percent. The arrangement blunder went from near 19 percent to 22 percent.Wang et al. [4] 2 to 3 API call successions have been created and used to portray eight dubious practices. The analysis included utilizing a Thomas Baye's algorithmic program to arrange whether the program was malevolent and achieved ninety 5 percent once 879 examples of 553 vindictive malware were instructed in 80th of the information.

Liu et al. [5] to scale back the overhead an ideal opportunity to build productivity by a serious half-hour, MapReduce reviewed. For recognizable proof of Trojans, malware, worms, and spyware, the trial result identifying with accuracy was forty-fifth (from five hundredth to 89%).

Ding Yuxin et al. [6] we utilize a powerful impurity examination strategy to stamp the framework call boundaries with spoil labels, at that point develop the administrator call guidance reliance diagram by following the proliferation of the pollutant information, constructed malware practices as reliance charts to discover the reliance connections between framework calls. They proposed a calculation to infer the conventional conduct chart, which is utilized to depict the social highlights of a malware family, in view of the reliance diagrams of malware tests.

Yousra Aafer et al. [7] an extreme examination was created to eliminate pertinent highlights from malware action got at the API level, and different classifiers were assessed

utilizing the made list of capabilities. Their discoveries show that by utilizing the KNN classifier, we are prepared to accomplish precision as high as 98 percent and a bogus positive rate as low as 3 percent.

AlirezaSouri et al. [8] the procedures overviewed don't appear to be adequate, while the natural component and progressed plan of malware are progressively advancing and along these lines ending up being more hard to identify. A logical and cautious review of interruption recognizable proof techniques for exploitation of information handling methodologies should be utilized. Likewise, in 2 key classes, it arranges malware recognition procedures alongside signature-based strategies and conduct based location. Impacts, we seem to conclude that with twenty ninth, j48 has 17 November, call tree has Bastille Day, NB has 9%, BF has five-hitter and furthermore the substitute methodologies have only 3 percent utilization of information preparing results, the SVM strategy has the most extent for malware discovery approach.

Deepak Koundel et al. [9]set up a way to deal with portray an application by exploitation information investigation as payment product or amiable application. We like to utilize different credits of an application for classifications of an application: I the authorizations utilized by an application, (ii) the consents empowered by battery use rating and (iii) the apparatus on the robot market not inheritable rating. To conclude the results, they utilized the Naive Bayes classifier to help the likelihood that is malware or not. These perceptions are communicated to the cloud any place a client peruses the Associate results being referred to as being malevolent or not to our worker.Dragos¸ Gavrilut et al. [10] An adaptable structure has been implicit which completely extraordinary AI calculations are utilized to effectively separate between malware documents and clean records, while during this paper we attempt to limit the thoughts behind our system by working principally with uneven course perceptions and also with course when effectively tried on medium-sized outcomes.

Chih-Ta Lin et al. [11] their strategy mixes the decision and furthermore the extraction of alternatives, which significantly diminishes the spatial property of training and characterization choices. Their procedure consolidates the decision and furthermore the extraction of choices, which essentially diminishes the spatial property of instructing and grouping choices. Helped malware practices got from a sandbox environment, pay in 5 stages: (a) removing data from conduct signs on the n-gram work territory; (b) developing a conduct log Experiments were done on a true informational collection of four, 288 examples from nine families, that the adequacy and furthermore the quality of our system were obvious.The [1] surveyed a method for detecting worms and other malware by using sequences of WinAPI calls and depending on fixed API call addresses.

While [2] developed automated dynamic malware analysis techniques and tools; automated dynamic analysis provides a report for each malware program, describing its run-time behavior. The information yielded by these analysis tools elucidates malware program behaviors, facilitating the timely and appropriate implementation of countermeasures.

[4] Developed and used two –to three API function call sequences to describe eight suspicious behaviors. The experiment involved using a Bayes algorithm to classify whether program was malicious and achieved 93.98% when 80% of the data were used to train in 914 samples with 453 malicious malwares.

III. METHODOLOGY

The study will be based on quantitative method whereby the accuracy of the proposed system will be measured using KNN and SVM. The proposed system is shown in below Figure 1.

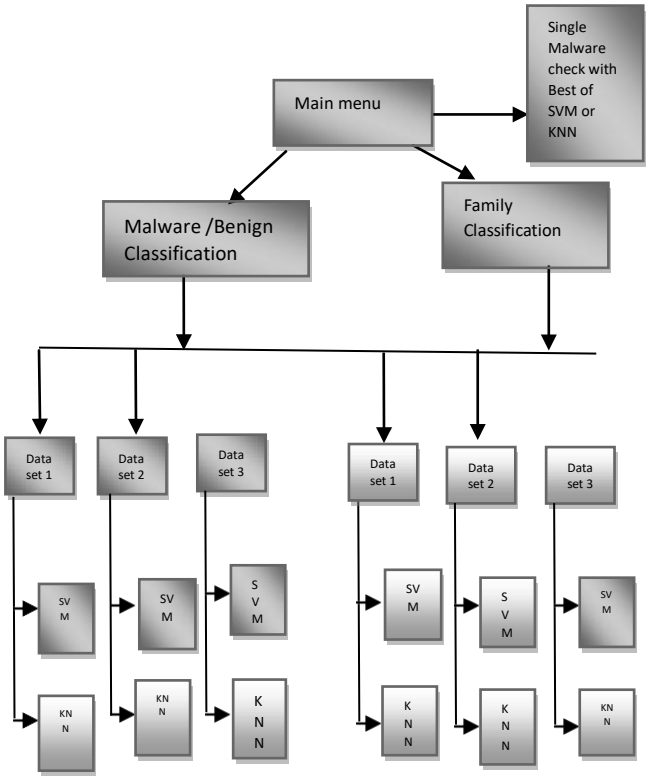

Figure 1: Proposed System for Malware Detection and Malware Family Classification

Accurate and sufficient number of features and cases in the dataset are very critical for accurate classification results. Hence, detecting malware must be automatic, efficient, effective and accurate. Malware can be detected and analyzed by either static or dynamic analysis using two techniques:

a) Code analysis without executing the software (signature based)
b) Behavioral analysis (anomaly based)

Researchers used a diversity of techniques for detecting malware despite how they handled the results. Figure 2 illustrates some of these techniques.

In the proposed system the malware datasets are collected from different well known websites which consists of malware API sequences. Along with the technology advancement, the malware authors have developed malicious code that hard and difficult to be analyzed and detected by researchers. For example, malware writers created malicious code with implement new technique mutation characteristic on that malware which causes an enormous growth in number of variation of malware.

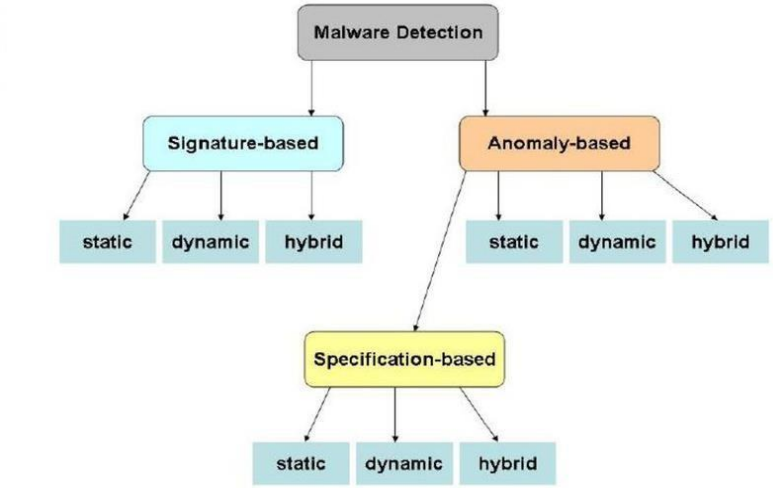

Figure 2: Malware Detection Techniques

A. K-nearest Neighbor

K-Nearest Neighbors (KNN) is one of the simplest, though, accurate machine learning algorithms. KNN may be a non-parametric algorithm, meaning that it doesn't make any assumptions about the info structure. In world problems, data rarely obeys the overall theoretical assumptions, making non-parametric algorithms an honest solution for such problems. KNN model representation is as simple as the dataset – there is no learning required, the entire training set is stored. KNN are often used for both classification and regression problems. In both problems, the prediction is predicated on the k training instances that are closest to the input instance. In the KNN classification problem, the output would be a category, to which the input instance belongs, predicted by the bulk vote of the k closest neighbors.

B. Support Vector Machines

Support Vector Machines (SVM) is another machine learning algorithm that's generally used for classification problems. The main idea relies on finding such a hyper plane, which would separate the classes in the best way. The term 'support vectors' refers to the points lying closest to the hyper plane, that might change the hyper plane position if removed. The distance between the support vector and the hyper plane is referred to as margin. Intuitively, we understand that the further from the hyper plane our classes lie, the more accurate predictions we can make. That is why, although multiple hyper planes are often found per problem, the goal of the SVM algorithm is to seek out such a hyper plane that might end in the utmost margins.

The proposed work consists of three main phases. They are

1)Malware/Benign Classification

In this phase based on the dataset attributes training and testing proportions have taken (ex: 80 samples of each for training and 40 samples of each for testing out of 120 malware cases) and it will classify which is the malware and benign using SVM and KNN.

2)Family Classification

There are so many malware classes listed above but for experiment purpose we are considering total 4 classes.Benign

- ***Dridex***

Dridex is malicious software (malware) that targets banking and financial access by leveraging macros in Microsoft Office to infect systems. Once a computer has been infected, Dridex attackers can steal banking credentials and other personal information on the system to realize access to the financial records of a user.

- ***Darkcomet***

DarkComet is a Remote Access Trojan (RAT) application that may run in the background and silently collect information about the system, connected users, and network activity. DarkComet may plan to steal stored credentials, usernames and passwords, and other personal and tip. This information could also be transmitted to a destination specified by the author.

- ***Cybergate***

CyberGate is one of many remote access tools (RATs) that allow users to control other connected computers remotely. Cyber criminals often use these programs for malicious purposes such as to steal personal, sensitive information and misuse it to generate revenue. People who have computers infected with programs like CyberGate should uninstall them immediately.

3) Single Malware Check

This step will extract the feature of the given malware file and it will find out best accuracy among SVM and KNN algorithm. Here we are checking the sample file Benign or Malware (of any family) in single test from given file, features are extracted and used to make prediction using KNN or SVM as Dataset1 given Best Accuracy. And it is used as Knowledge Base/Single Test.

IV. IMPLEMENTATION

During this step the research plan is designed and can be implemented in practice. The whole implementation process can be outlined in the following steps.

In this section we elaborate the complete framework for API extraction. Most of the malware within the dataset were compressed, packed and obfuscated. The freely available unpackers like UPX, ASPack, FSG and UPack are used in the automated system to unpack the executables before disassembly and analysis. For each category the extracted API's were further refined using DCFS measure. Fig 7.1 shows the system architecture of an automated process. The following are the steps followed by the automated system. Import table is employed by the loader at runtime to spot the addresses of the referred APIs in order that whenever an API is named, a jump to the API code is executed. Unpack the malware. Extraction of API Calls using IDA Pro and export into Mysql database using ida2sql python plug-in. Selection of relevant API Calls.

We have used the concepts of relevant API calls and Class-wise document frequency for choosing the relevant API calls. The aim is to spot a group of API calls that are common to the set of malware and similarly another set of API calls that are common to the benign executables.

In other words, the Class-wise document frequency is the number of executable programs in C that contain Ng. Fig. 2 is a flow chart describing the selection of relevant API calls using DCFS (Document Class wise Frequency feature selection).

Description of Dataset:

Table 1: Data sets Description

Dataset	Training	Testing
Dataset 1	80	40
Dataset 2	60	60
Dataset 3	20	100

There is totally 1858 API call that is used in malware. The API and sequence of API decides it is a malware or not. Sample malware files are collected form internet , we have allotted class code for them as

Table 2: Malware Families Classes List

Family Class	Family	Type Class	
1	benign	1	
2	dridex	2	These all are malwares, type class 2
3	darkcomet	2	
4	cybergate	2	

V. RESULT AND CONCLUSION

Total 4 classes of malware families have considered. They are Benign, Dridex, Darkcomet and Cyber gate. And there are 120 cases of each class, from that three dataset made for training and testing respectively. There are totally 1858 API call that are used in malware, the API and sequence of API decides it is a malware or not and sample malware files are collected form internet.

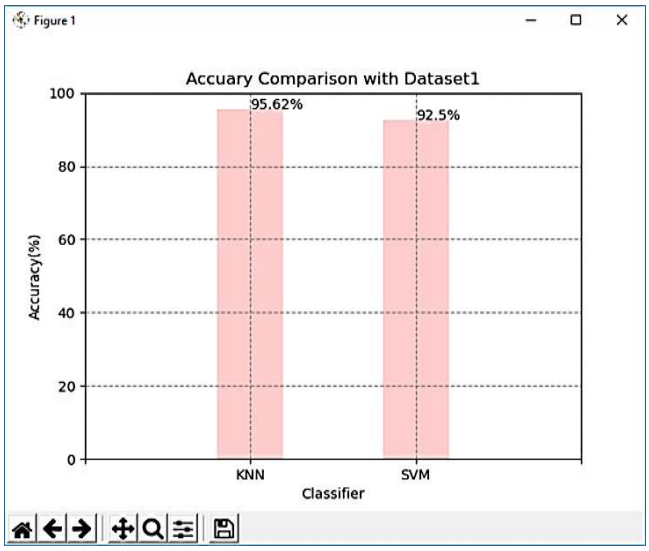

Figure 3: Accuracy Analysis of Dataset1

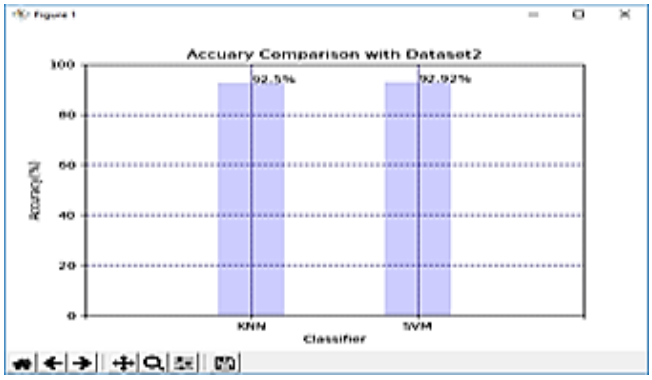

Figure 4: Accuracy Analysis of Dataset2

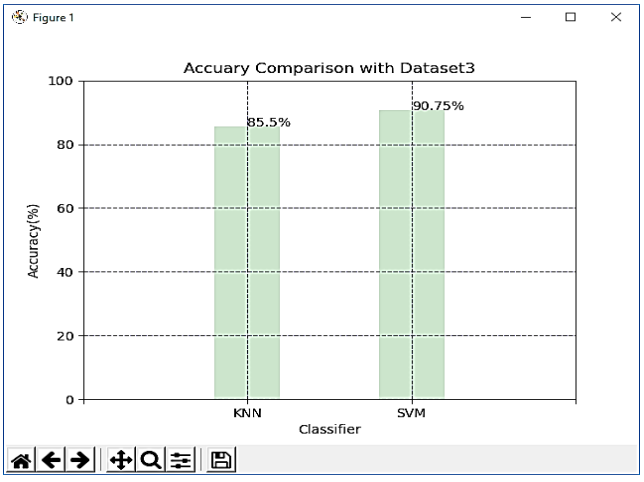

Figure 5: Accuracy Analysis of Dataset3

Proposed statistical analysis of Windows API calls in malware reflects the behavior of a piece of code. In this research project, the relevant APIs were extracted from each malware category and further refined using Document Class wise Frequency feature selection measure to classify the executable as malicious or benign. The entire static detection process was fully automated for classification system.

Table 3: Classification Comparison of KNN and SVM Algorithms

Dataset	Training	Testing	KNN Accuracy	SVM Accuracy
Dataset 1	80	40	98.12 %	96.25 %
Dataset 2	60	60	95.83 %	95.42 %
Dataset 3	20	100	94.25 %	90.00 %

Finally we concluded that, in malware detection and classification of malware family problems, different models gave different results. The lowest accuracy was achieved by SVM (85.5% and 90.75%). The highest accuracy was achieved with the KNN model and it was equal to 95.62% and 92.5%.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest

REFERENCES

[1] H. Sun, Y. Lin, and M. Wu, "Api monitoring system for defeating worms and exploits in ms-windows system," in Proceedings of the 11th Australasian Conference on Information Security and Privacy, 2006, pp. 159-170.

[2] M. Egele, T. S. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamicmalware-analysis techniques and tools," *ACM Computing Surveys*, Vol. 44, 2012, pp.6:1-6:42.

[3] K. Tsyganok, E. Tumoyan, M. Anikeev, and L. Babenko, "Classification of polymorphic and metamorphic malware samples based on their behavior," in Networks,2012, pp. 111-116.

[4] C. Wang, J. Pang, R. Zhao, W. Fu, and X. Liu, "Malware detection based on suspiciousbehavior identification, " in Proceedings of the 1st International Workshop on Education Technology and Computer Science, 2009, pp. 198-202.

[5] S. Liu, H. Huang, and Y. Chen, "A system call analysis method with mapreduce for malware detection," in Proceedings of the 17th IEEE International Conference on Parallel and Distributed Systems, 2011, pp. 631-637.

[6] Ding Yuxin, Xia Xiaoling, Chen Sheng, Li Ye, A malware detection method based on family behavior graph, *Computers & Security* (2017).

[7] YousraAafer, Wenliang Du, and Heng Yin, DroidAPIMiner: Mining API-Level Features for Robust Malware Detection in Android , Dept. of Electrical Engineering & Computer Science Syracuse University, New York, USA fyaafer, wedu, heying@syr.edu.

[8] Souri and Hosseini ,A state of the art survey of malware detection approaches using data mining techniques Hum. Cent.Comput. Inf. Sci. (2018) 8:3https://doi.org/10.1186/s13673-018-0125-x.

[9] Deepak Koundel, SurajIthape, Vishakha Khobaragade, Rajat Jain *B.E.* Computer Science JSPM's JSCOE Pune, India, Malware Classification using Naïve Bayes Classifier for Android OSThe International Journal Of Engineering And Science (IJES) Volume 3 Issue 4 Pages 59-63 2014 ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805.

[10] Dragos¸ Gavrilut¸ Mihai Cimpoes¸u1, Dan Anton1, Liviu Ciortuz, Faculty of Computer Science, University of Iasi, Romania, BitDefender Research Lab, Iasi, Romania, Malware detection using machine learning Conference Paper · November 2009, DOI: 10.1109/IMCSIT.2009.5352759 · Source: IEEE Xplore.

[11] Chih-ta lin, nai-jian wang, han xiao and Claudia eckert, Department of Electrical Engineering, National Taiwan University of Science and Technology Taipei, 106 Taiwan, Feature Selection and Extraction for Malware Classification, Journal of Information Science And Engineering 31, 965-992 (2015)