International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 6 3951 - 3954

Detection of Malware By Using Support Vector Machine

Ms. Phalke Nutan Navnath
Department of Computer Engineering.
DGOI,FOE-swami-chincholi
Daund,pune,india
nutan.phalke@gmail.com

Prof. Amrit Priyadarshi.
Department of Computer Engineering.
DGOI,FOE-swami-chincholi
Daund,pune,india
amritpriyadarshi@gmail.com

***Abstract*—**Investigate the structure of the program by using bytes or text strings N-gram analysis is an approach . A basic issue with N-gram analysis is selection of feature ,the explosion of features that occurs when N is increased .The experimental details within this paper represent programs as operational code density histograms which are gained from dynamic analysis .A support vector machine is used for the creation of reference model ,also having two methods of feature reduction, first is area of intersect and subspace analysis using eigenvectors .then analysis show that the relationships between features are complex and simple statistics filtering approaches do not provide a viable approach. use eigenvector subspace analysis to produces a suitable filter.

Keywords- *Metamorphism malware, SVM, obfuscation, packers.*

*****

I. INTRODUCTION

The large growth of malware in every year which causes a serious security problem. so, malware detection is a mostly critical point in computer security also handle the suspected code for known security ,vulnerabilities which is become ineffective .to remove irrelevant feature which is created by malware writers, so that researcher has a need to apply different methods .In future research we can expand the detection methods by investigating N-gram size, An n-gram is a sub-sequence of n items from a given sequence. which is dramatically increasing the number of features. For this anticipated explosion of features we have chosen to investigate methods to remove irrelevant features. Principle Component Analysis (PCA) is one of the most popular method used to reducing features in subspace[7], This project aims to identify feature reduction in the original dataset space. For large datasets, the training process associated with learning machines can become large. Thus, the explosion of feature is occurs with N-grams for large values of N needs to be addressed[6]. Malware which is designed by attackers for disturbing whole computers. Malware variants will have various byte level representations while in principal under the same family of malware. The byte level content is different because little changes to the malware source code can result in significantly different compiled object code .In which programs are used as operational code (opcode) density histograms obtained through dynamic analysis. Dynamic analysis is the process of testing and evaluation of application or a program during running time. A SVM is used for classification of problems. It uses a technique called the kernel method to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs. We use static analysis for classification of malware. It identified a prefilter stage using values of files, that can reduce the feature set and therefore reduce the training effort .The result shows that the relationships between features are complex and simple statistics filtering approaches do not provide a Practical approach .it produce suitable filter.

II. LITERATURE SURVEY

A. .Detecting unknown A malicious code by applying classification

In [1] author uses opcode n-gram analysis of file for malware detection using static analysis which consist of different size i.e (N=1 to N=8).

B. opcode sequences as representation of executables for data-mining-based unknown malware detection,

In [2]author reduced the effort required for the training phase while labeling and also issue of unpacking malware.he evaluated various learning methods like KNN,Bayesian Network,SVM, etc.also shows that malware can be detected according to degree of occurrence of opcode.

C Towards an understanding of anti-virtualization and antidebugging behavior in modern malware,

In[3] he says different number of anti-analysis techniques used by the StrongOD plug-in is less than the number of analysis avoidance techniques available to malware. number of actions taken by StrongOD to prevent malware detecting the debugger and to defend against malware attacks on the debugger.like ill anti-attach handle,UnhandledExceptionFilter etc.

D Linear Programming: Foundations and Extensions

In [4] author says that their are number of investigation techniques are used for the classification of benign and alicious software so linear programming is one of the suitable

 ISSN: 2321-8169
3951 - 3954

technique for that to understand when a decision plane is applied how the area under each curve is interpreted .

E Callgraph properties of executables and generative mechanisms,

Bilar in [5] compared the statically generated the CFG of benign and malicious code.also their findings showed a difference in the basic block count for benign and malicious code.he concluded that malicious code has a lower basic block count, implying a simpler structure: Less interaction, fewer branches and less functionality provided.

III. IMPLEMENTATION DETAILS

The flowchart of project is shown in Fig.1. It works in following steps::

A. System Architectures

The use of SVM as tools for the detection of malware. Malware can be detected using SVMs through the opcodes chosen by the SVM as N-gram analysis is performed on that data. Also feature filter reduce feature also reduce computational overhead. training phase of SVM is a solution for n-gram analysis performed on large size of data. Experimental approach is that :

1)In test environment we are investigating the program then using different debug tool like ,which is monitoring that runtime opcode.

2)After completion ,then that data is parsed into opcode histograms.

3)Then after that the dataset is give to the SVM used for creation of a reference model.

4)Then use various filtering algorithms ,each filter processes the original dataset and compare with reference model which is produced by the SVM..

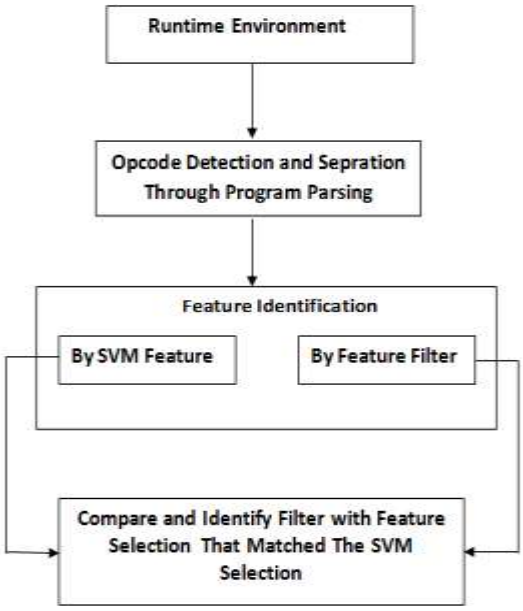

Figure1: System Architecture

The reference model is constructed by the SVM for to perform an exhaustive search by traversing through all the features, and searching for those opcodes that will have a positive impact on the classification of benign and malicious software. Feature filter algorithm are also find the features in dataset of the program and then compare with result of reference model if features are matched or same then it consider that there is a presences of malware in the program.

B. Dataset Creation

Operational code used in dataset creation i.e opcodes with machine language instruction that perform CPU operation on operand. such as arithmetic ,memory manipulation ,logical operation, program own control. we are directly considering 15 opcode such as[7]:-

Arithmetic operation:-add ,sub ,inc, adc (add with carry flag).

Memory manipulation:-lea(load effective address),mov, push, pop.

Logical operation:-XOR(Exclusive OR).

Program flow control:-call(jump to a function),ret(return from function),cmp(compare data),ja, je(jump if condition is met),rep(a prefix that repeat the specified operation).

opcode density histogram that obtained through dynamic environment by representing each executable file in dataset .also need to consider that only opcodes are consider ,operands associated with that opcodes are not consider. This separation of opcodes are done through the use of debugging tools,separating data into training i.e known data and test data through classification. Each training-set instance having assigned a target value i.e. benign or malicious. The goal of the SVM is that to construct a model that determine the target values of the test data. Ollydbg tool ran and correctly unpacked the malware,samples were restricted to programs that ollydbg correctly identified as packed or encrypted.

C. Support Vector Machine

To detect irrelevant opcode in program SVM technique is used. which is based on kernel method algorithm that algorithm depends on dot-product function .data placed into higher dimensional feature space through the use of kernel function, having two benefits:-

1)It has ability to generate a nonlinear decision plane.

2)Allows the user to apply a classification to the data has a non regular or unknown distribution.

Only unique opcodes are consider .Support Vector Machine create reference model for validation of filter experiment. Which traversing through all data set and searching only those opcode having good or positive impact on classification software.search is repeated for each opcode only average result of the opcodes are selected. Proposed Prefilter Approach

 ISSN: 2321-8169
3951 - 3954

To filter irrelevant opcode here we are using two approaches such as:

1) Area of Intersects:

Start with the investigating area of intersect between benign and malicious distribution using Linear Programming. considering only the characteristics of benign and malicious opcode.draw density curve of single opcode. vertically consider the number of program with that percentage of opcode and horizontally percentage of that measure the overlapping area of two density curves.it not always the case that if least area of intersect between benign and malicious it also has reduced rate of error and misclassification.SVM selecting different opcode:- ja ,adc ,sub ,inc,rep ,add,ret,push as a reference model it is not always possible to say that least area of intersect is better indicator of benign and malicious software. also their is need to consider closer inspection of the opcode distribution curve also need to identify the best indicator chosen by SVM over the other opcodes that have similar area of intersect and population ,so that use of Linear Programming(LP) used to understand how that the area under each curve is interpreted when decision plane is applied. having the component are: Constraints, Decision variable, Objective function

2) Subspace:

To determine importance of each opcode individually their usefulness and classification, so use eigenvector in subspace also principle component analysis(PCA) is transformation of covariance matrix,which is show in equation 1 below.

$$C_{ij} = \frac{1}{n-1}\sum_{m=1}^{m}\left(X_{jm} - \bar{X}_i\right)\left(X_{jm} - \bar{X}_j\right) - - - (1)$$

Where
C= Covariance matrix of PCA transformation;
X= dataset value;
$\bar{X}$ =dataset mean;
n and m = data length;

It compress the data by mapping data into subspace, reducing dimensionality through mapping that data into subspace. also create new set of the variable which are original data called principle components ,those principle component are order by their eigenvalue or usefulness .PCA are used to determine the number of principle components are correlated which are greater percent and most significant eigenvalues are selected to the eigenvector.PCA algorithm operate on covariance matrix of the training dataset ,which is calculated as shows in below equation-2 –

D. Dynamic Approach

Principle Component Analysis(PCA):

Step1. Take the whole dataset consisting of d-dimensional samples ignoring the class labels.

Step2. Compute the d-dimensional mean vector.

Step3. Compute the covariance matrix of the whole dataset.

Step4. Compute eigenvectors and corresponding eigenvalues .

Step5. Sort the eigenvectors by decreasing eigenvalues and choose k eigenvectors with the largest eigenvalues to form d x k dimensional matrix (where every column represents an eigenvector).

Step6. Use this d x k eigenvector matrix to transform the samples onto the new subspace

E. Mathematical Model

PCA algorithm operate on covariance matrix of the Training dataset which is as follows.

C=con(training data)
[V, λ]= eig (C)
d=diag
Significant value is calculated as

$$R_k = \sum_{k=1}^{8} V.d_k \quad \text{———(2)}$$

where
R =Sum of the matrix variance;
C =Covariance
V =Eigenvector
λ =EigenValue matrix;
d = EigenValue scalar;

IV. RESULT

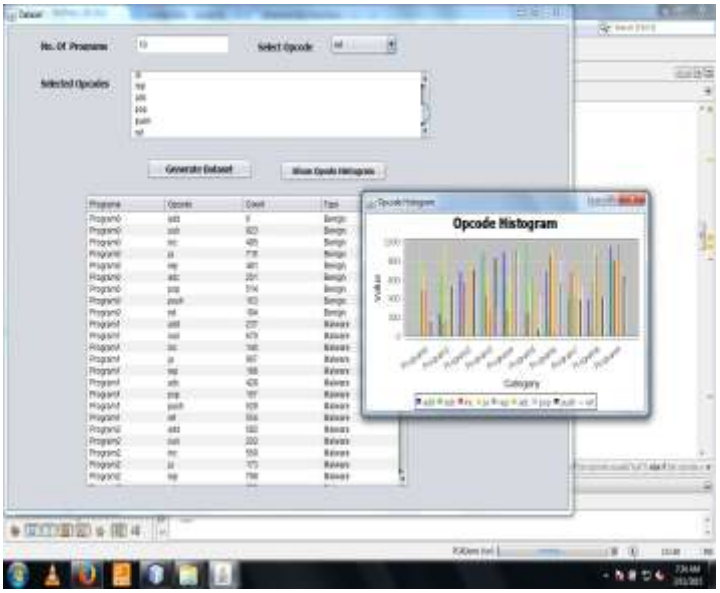

Figure 2: Create histogram of the opcode according to the percentage of the occurrence of opcode in the program

 ISSN: 2321-8169
3951 - 3954

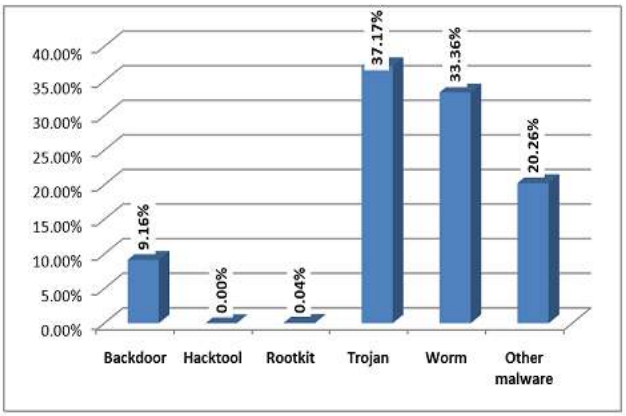

Figure3: Malware distribution in the test (WL) dataset, showing the percentage of different types of malware with respect to the total number of malware files

V. CONCLUSION

This project conclude that the use of Support Vector Machine as a means of identifying malware. It shows that presences of malware, that is packed/encrypted ,which can be detected using SVMs and by using the opcodes which is chosen by the SVM as a benchmark, identified a prefilter stage using eigenvectors that can reduce the feature set and therefore reduce the training effort.

1) *The identification of a high population opcode.*
2) *A subset of opcodes can be used to detect malware. However, the SVM analysis demonstrates that ja, adc and subare strong indicators of malware.*
3) *Using the, eigenvector prefilter, the dataset can safely remove irrelevant features.*

ACKNOWLEDGMENT

Our heartfelt thanks go to DGOI Faculty Of Engineering providing a strong platform to develop our skills and apabilities.I would like to thank to our guide respected teachers for their continuous support and incentive for us. Last but not least,I would like to thanks to all those who directly or indirectly help us in presenting the paper.

REFERENCES

[1] A. Shabtai, R. Moskovitch, C. Feher, S. Dolev, and Y. Elovici,"Detecting unknown malicious code by applying classification techniques on opcode patterns," Security Informatics, vol. 1, pp. 122, 2012.

[2] I. Santos, F. Brezo, X. Ugarte-Pedrero, and Y. P. G. Bringas Opcode sequences as representation executables for data-miningbased unknown malware detection Inform. Sci., 2011 [Online]. Available: http://dx.doi.org/10.1016/j.ins.2011.08.020

[3] X. Chen, Towards an understanding anti-virtualization and antidebugging behavior in modern malware ICDSN Proc., pp. 177186,2008.

[4] R. Vanderbei, Linear Programming: Foundations and Extensions Pub. New York, NY, USA: Springer, 2000, ISBN: 0792373421

[5] D. Bilar, Callgraph properties of executables and generative mechanisms, AI Commun., Special Issue on Network Anal in Natural Sci.and Eng., vol. 20, no. 4, pp. 231243, 2007.

[6] I. Santos, Y. K. Penya, J. Devesa, and P. G. Garcia, N-grams-based file signatures for malware detection, S3Lab, Deusto Technological Found., 2009 [Online]. Available: pgbg@technologico.deusto.es 7

[7] Philip O'Kane, Sakir Sezer, Kieran McLaughlin, and Eulgyu "SVM Training Phase Reduction Using Dataset Feature Filtering for Malware Detection" IEEE transactions on information forensics and security, vol. 8, no.3 march 2013