Static Way of Effective Feature Extraction and Malware Classification

Mahendra Deore[a], U.V. Kulkarni[b]

[a]Department of Computer Engineering, Cummins College of Engineering, Karve Road, Pune, Maharashtra, India

[b]Department of Computer Science and Engineering ,Sggs Institute of Engineering and Technology, Nanded, Maharashtra, India

Abstract

Today malware programmers use various mechanisms so that they can prevent the identification of unique properties of their program like encryption and obfuscation. The malware of malicious code hidden in different file format like Exe, PE, JPG which includes different malware like viruses, Trojans, spam, worms and likewise which causes damage to computer system. In this paper we are presenting the mechanism of identifying different malware through Portable Executable file (PE) by scanning this file and then breaking into different features of malware. This dataset is further classified using Naïve Byes and Support Vector Machine (SVM). Features used for classification are measurement of metadata, used symbols, sections and finally calculation of entropy. The main goal of this work is static approach on PE file and execution of classification algorithm to understand and determine the basic feature of malware infected files.

KEYWORDS – Malware Infection, Entropy, Naïve Byes algorithm, Metadata, Feature extraction.

I. INTRODUCTION

In last decade there is exponential growth in the internet usage. At the same time, the various security challenges are observed due to which internet becomes source of malicious activities. Malware is one of the major threat that harm user data. Software carried out suspicious actions, consist of information stealing, intelligence, are said to be malicious activities. By Definition of malware, it is mentioned as “malware is a computer code, software or we can say that the program especially designed to infect a user's computer and it’s harmful for computer”. According to research 2016, 6.5 million different hosts are attacked and around 4 million malware are observed in 2015 which carries unique characteristics [1] [2]. The Jupiter research (2016) predicts that there will huge loss of data and which will exponentially grow to $2.1 trillion worldwide by in 2020. There are numerous antivirus present in the market but unable to cope up with increasing diversity of malware. The inability of anti-virus software to provide protection against harmful attack leads to proceed in huge number of host are attacked.

In inclusion to that, the skill set required to develop a malware is decreasing day by day. Because of the maximum accessible of attacking software or we can say that the tools easily available on the internet. Latest survey indicates that numbers of attacks are being delivered by script-kiddies or are automated. Identification of malware for systems is most important aspect for single users and also for business, even because of attack can result in sufficient losses. In order to control massive loss of data, money and frequency of attack of malware requires accurate and time bounded detection method. Because of this reason, machine learning is majorly used. The existing work

focuses on malware feature extraction using machine learning and classification methodologies.

There are majorly two techniques followed for analysing malware from the exe file, 1. Static Technique 2. Dynamic Technique [3] [4]. Knowing that lot of feature or character of the file once we analyse the file like string contents, byte code , N-grams, different structural parts of PE header and all such metadata is extracted which further helps us to classify into malware families[5].

This paper proposes a unique approach to get various features from executable files with their family classification. By implementing the classification algorithm without unpacking or unzipping the sample exe file. In addition, there is no need for verification of newly packaged good ware because classification method of malware considers all samples of malware. we could able to check the infected file without actually unzipping or unpacking it and actually running it because execution of this malware infected file will be not suitable and from security point of view it will create problem and hence we could able to achieve the faster and speedy classification of data for further analysis .The accuracy of results will be measured by consider two parameters i.e. whether file contains malicious code or after classification, malware belong to which family [8]. The model will be evaluated for accuracy by using, WEKA Tool [6] [7].

II. LITERATURE REVIEW

E. I. Edem, describe a malware approach of classification that is mainly used to improve the precision and scalability. Using mix way of dynamic features and static features we can develop the hybrid approach. Many of the peoples giving attention to develop the classification of the malware which gets rise as per day. It has two major methods that are Static approach and dynamic approach. In static analysis technique the analysis is performed without executing a malware infected file. In the dynamic analysis we only monitor and observe actions made by the malware at the time of its execution. To find out and classification of malware, it makes comparison with another so that static approach is more useful. As a result, dynamic analysis has recently received remarkable attention as it is significantly less vulnerable to code obfuscating transformations. However, dynamic analysis has its own weaknesses [9].

Mansour Ahmadi developed a system Based on Microsoft malware classification challenge, in 2015 huge dataset 0.5 TB was released. This paradigm emphasizes the phases of extraction, and selection of a set of novel features for the effective malware representation. Features can be grouped according to malware behaviour, and characteristics and their fusion is performed according to a per-class weighting paradigm [10].

The system proposed by X. Hu. [11] is an approach to classify malware based on their static features. Before feature extraction, the data was pre-processed where they have reconstructed the PE headers. Elha di [12] proposed the **hybrid analysis** is used for the classification of malware. Firstly they've undergone dynamic analysis and then the static features are also examined. The file which is suspected of having malware is executed in the safe and controlled environment and then API calls are extracted using kernel hooking. If the file is packed then it is unpacked first. The static part includes construction of **call graphs** with the help of the API calls extracted and the resources of the OS used by these API calls. Damodaran, compares malware detection techniques based on static, dynamic, and hybrid analysis Specifically, to train Hidden Markov Models (HMMs) on both static and dynamic feature sets and compare the

resulting detection rates over a substantial number of malware families and also consider hybrid cases, where dynamic analysis is used in the training phase, with static techniques used in the detection phase, and vice versa [13]. Ashkan Sami [14] Proposed framework consists of three major parts. The first component is a PE analyzer that aims to analyze PE files and extract the API calls imported by PE file. Second component does feature generation and feature selection. Features included in this study, are discriminative and domain relevant features that are produced from the first component's output. Third part includes a classifier that classifies PE files based on features selected in the second part. They examined several classification methods such as Naive Bayes, J48, and Random Forest. Considering the highly imbalanced class distribution, and shows that Random Forest has the best performance.

III. Malware Methodology

3.1 Malware Classification

The classification of malware is a tough method. Code that permits unauthorized control of a system is clearly malicious. Malware comes in numerous forms and classes. These are typically classified per their propagation technique and their actions that are performed on the infected machine using the designed bug. The following list presents the common styles of malware.

Virus: A bug propagates from one program to a different or from one laptop to a different by inserting their code into alternative program.

Worm: its self-replicating programs that unfold from one laptop to a different by transmittal copy of itself via a network while not user authorization.

Trojan horse: Trojans mask themselves by showing to be one thing legitimate. Trojans usually destroy knowledge or commit to extract steer as well as monetary knowledge & passwords.

Spyware: Spyware is any code put in on system while not user's data. It's a collective term for code that monitors and gathers personal data concerning the user and sends that data back to the assaulter thus the assaulter will use the stolen data in some ill-famed means. It typically enters a system once free or trial code is downloaded and put in on the system while not the user's data, changes the setting of your browser or adds abominable browser toolbars.

Scareware: Scareware is a malware masquerading as free or trial anti-virus code or another free on-line scam. It will be put in by the user once downloading phoney security code, gap attachments or by visiting a malicious web site.

Adware: Adware is advertising supported code that mechanically plays, displays, or downloads advertisements to a laptop once malicious code is put in or application is employed. This piece of code is usually set into free downloaded code. The foremost common supply of adware programs square measure free games, peer- to-peer shoppers like KaZaa, BearShare etc.

Botnet: A botnet is remotely controlled autonomous code. It is sometimes a zombie program (worms, Trojans) underneath common management for any network infrastructure.

3.2 Malware detection methodology

The malware analysis has two main methodologies static and Dynamic. In static we have no need of execution of a program into that two main methods are available SAFE and SAVE for the detection of malware, mostly the analysis is based on the PE header of infected file that is directly based on the byte code or disassemble the given

PE file to separate the data and code or other relevant detailed information available into the PE header. The main issue during the analysis is with the packing and obfuscation. Malware analysis is critical to develop effective malware detection technique. It's the method of analyzing the aim and practicality of a malware, thus the goal of malware analysis is to know how a selected piece of malware works so defence is designed to defend the organization's network.

3.2.1 Static Analysis

It is additionally known as code analysis. It's the method of analyzing the program by examining it i.e. software system code of malware is discovered to achieve the information of however malware's functions work. During this technique reverse engineering is performed by mistreatment destruct tool, decompile tool, debugger, source code analyser tools like IDA pro and Ollydbg so as to know structure of malware [15]. Before program is dead, static info is found in the feasible as well as header information and also the sequence of bytes is employed to see whether or not it's malicious. Activity technique is one in all the techniques of static analysis. With static analysis feasible file is destruct the .mistreatments disassemble tools like XXD, Hexdump, NetWide command, to urge the programming language program file. From this file the opcode is extracted as a feature to statically analyze the appliance behaviour to observe the malware. As the name suggests it is performed without execution of file [3] [9]. We say that the source code of the malware is perusal into the static analysis and is trying to reduce the behavioral stuff of the file. Static analysis has included various techniques [16] [17].

1. **File Format Inspection:** By using metadata it provide useful information like header of Windows PE (Portable Executable) and gives more information during compilation, imported and exported functions etc.
2. **String Extraction:** with the help of String we can observe the system output and another information about the PE [18] [19].
3. **Fingerprinting:** This method produce cryptographic and hash computation which is used to find artifacts such as filename, username and also strings of registry.
4. **AV scanning:** Inspected file is contains a well-known malware, is go through scanned by all anti-virus scanners that are available to detect it. So this might seem similar to scanners, means AV vendors or sandboxes doing their detection to confirm the results.
5. **Disassembly:** Is used to reverse the machine code into an assembly language which separates data and code. It is considering as most common and consistent method during static analysis [6].

Certain type of tools is used for static analysis. Using static analysis we can determine all type of behavioural scenes. Researcher is to research the code to find all of the ways of malware execution which is have no limit for the present situation, So that file is not executed by it and also it is not able to result in bad consequences for the system [20]. Static analysis is the mechanism that takes more time to analyze. Therefore mostly we don't prefer static mechanism worldwide for dynamic environment like anti-virus system, but mostly it is used for research purposes. This signature can also be file fingerprints for ex. SHA1 hashes or MD5, file metadata and static strings. In this case of detection that is to be follows, when the file reaches at the system then the antivirus software is possibly analyzed statically. If there is matched found of any of the signatures then there will be an alert triggered which is declaring that the file is suspicious. Mostly this type of the analysis is considered good and enough so because of this well-known malware is able to detect based on the hash values [3].The another

way to use virtual environment like sandboxes to run the files and check for states. This method is more time consuming as well as it is much safer as the file is checked before its execution. Advantage of this method is to identify not only malware which are already belongs to malware families but it also check for zero day attacks and polymorphic virus.

3.2.2 Dynamic analysis

In the Dynamic approach we are execute a file in the virtual machine (VM) [20].It is additionally known as activity analysis. Analysis of infected file throughout its execution is understood as dynamic analysis [6]. Infected files are analysed in simulated setting sort of a virtual machine, simulator, emulator, sandbox etc [7]. Subsequently malware researchers use SysAnalyzer, method explorer, ProcMon, RegShot, and different tools to spot the general behaviour of file [4]. In dynamic analysis the file is detected when execution it. In real setting, throughout execution of file its system interaction, its behaviour and result on the machine area unit monitored. The advantage of dynamic analysis is that it accurately analyses the called well as unknown, new malware. It's simple to observe unknown malware conjointly it will analyse the obfuscated, polymorphic malware by perceptive their behaviour however this analysis technique is longer intense. It needs the maximum amount time on prepare the setting for malware analysis like virtual machine setting or sandboxes.

3.2.3 Hybrid Analysis

This technique is projected to overcome the constraints of static and dynamic analysis techniques. It foremost analyses the signature specification of any malware code & then combines it with the opposite activity parameters for improvement of complete malware analysis. This approach hybrid analysis overcomes the limitations of each static and dynamic analysis.

3.3 Malware analysis techniques

The malware classification method discovered by the Anderson [21] and it is used with the help of different sources like dynamic call sequences control flow graphs, partition of executable and the file signatures. The training set of the malware and benign executable is used to know the weight of model. At the time of testing, the weights of these inputs are finally used for classify into the malware and benign.

3.3.1 Signature Based detection

It is also known as Misuse detection. It maintains the information of signature and detects malware by examination pattern against the information. General flow of signature-based malware detection and analysis is explained thoroughly in [17]. Most of the antivirus tools square measure supported the signature- primarily based detection techniques. These signatures square measure created by examining the disassembled code of malware binary. Disassembled code is analysed and options are extracted. These options are employed in constructing the signature of explicit malware family. A library of renowned code signatures is updated and fresh perpetually by the antivirus software package merchandiser therefore this method will notice the renowned instances of malware accurately. the main benefits of this method is that it will observe identified instances of malware accurately, less quantity of resources are needed to find the malware and it chiefly concentrate on signature of attack. The foremost disadvantage is that it can't notice the new, unknown instances of malware as no signature is on the market for such style of malware Signature-based

detection works by checking out explicit sequences of bytes inside associate object so as to spot exceptionally a specific version of a pestilence. Also known as string scanning, it's the only variety of scanning, constructed upon databases that have virus signatures. When a new virus emerges, its binary form will be specifically and uniquely analysed by a pestilence researcher and its sequences of bytes are going to be supplemental to the virus information [22]. A virus is identified by its sequences of bytes and what's referred to as a pestilence signature. Additionally, a hash price is another style of signatures. An oversized quantity of information is converted into one value by a function or a procedure called a hash function [20].

3.3.2Heuristic Based technique

It is additionally known as behaviour or anomaly-based detection. The second kind of on-access scanning is heuristic-based detection that was developed to beat the restrictions of signature-based detection. The most purpose is to investigate the behaviour of well-known or unknown malwares. Behavioural parameter includes numerous factors like supply or destination address of malware, sorts of attachments, and different numerable applied math options. it always happens in 2 part: training part and detection phase. Throughout training part the behaviour of system is discovered within the absence of attack and machine learning technique is employed to form a profile of such traditional behaviour. In detection part this profile is compared against the present behaviour and variations are flagged as potential attacks [22].

3.3.3 Specification Based technique

It is derivative of behavior-based detection that tries to beat the standard high warning rate related to it. Specification based mostly detection depends on program specifications that describe the intended behavior of security essential programs. It involves monitor program executions and detective work deviation of their behavior from the specification, instead of detective work the prevalence of specific attack patterns. This system is comparable to anomaly detection however the distinction is that rather than hoping on machine learning techniques, it will be supported manually developed specifications that capture legitimate system behavior [22].

From last decade data processing has been the most focus of the many malware scientist for detective work the new, unknown malwares; they need further information mining as a fourth planned malware detection technique. Data processing helps in analyzing the information, with automatic statistical analysis techniques, by distinguishing significant patterns or correlations. The results from this analysis may be summarized into helpful data and might be used for prediction. Machine learning algorithms area unit used for detective work patterns or relations in information, that are more accustomed develop a classifier [23]. The common technique of applying the information mining technique for malware detection is to begin with generating a feature sets. Classification of Malware mainly based on the signature-based method.

IV. PROPOSED SYSTEM

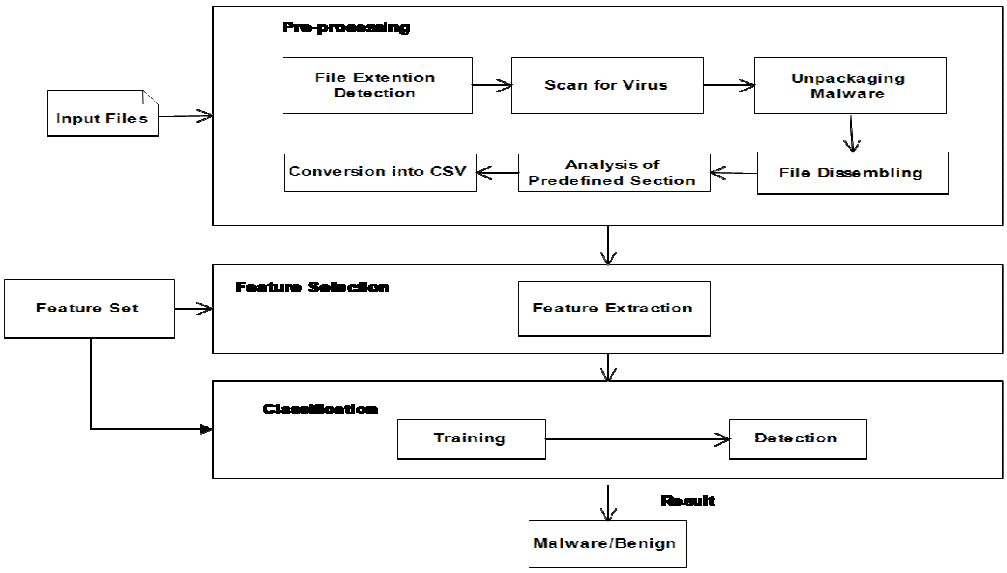

Figure1. System Architecture

The present work is based on Portable executable file generally known as PE file. The figure 1 presents, the process of scanning which carried out through Virus Total tool which analyse the given executable file is malware infected or not. The malware classification will be done on infected files only. In the next phase, we need to disassemble the infected file to separate out the code and data of infected file. After separation of code and data from infected file, the features are extracted by performing analytical operation on all related files. The total extracted features are collected and stored in the xml file and converted into csv file for the final result. The endmost or final result is unveil using WEKA Libraries in Java Technology by machine learning classification algorithms.

4.1 Algorithm for Family Classification-

En-Entropy, Col-column, Md5-MD5Hash
Read XML
    Get En from col overview
        If child node==entropy
            Get entropy. Value in En
        End if
Read Ecsv
    For each value of En in col entropy
        If entropy. Value==En
            Get md5
        End if
    End for
Read Fcsv
    For each value of md5 in col MD5hash
        If MD5hash.value==md5
            Get Virus Family
        End if

End for

We develop above algorithm for the classification of executable file into the appropriate family of malware, for that we read the XML file to get the entropy; the XML file contains parent child nodes. After that we need to read the train entropy dataset that contains the entropy with the MD5Hash, into that we need to match our entropy value with corresponding entropy and get the MD5Hash value from that dataset. After that for the classification into appropriate family we need to read another train dataset that contains the family related with the MD5hash value, into that we need to match our MD5Hash value. With the help of that hash value we can extract the related family of
Corresponding hash value.

4.2 Classification Explanation:
We get the efficiency of the approach with the help of various classification algorithms of the Machine Learning. They are, Support Vector Machine [19], Naive Bayes and Random Forest etc. After the step of feature extraction we passed it towards the WEKA libraries [24] for the classification. It is showing that all the classifiers were evaluated by using 10 fold cross validation. This is considering as a well-accepted and also standard method for the classification. WEKA shows the accuracy of each algorithm after that based on Entropy in the malware files which is very useful method for static detection of malware, on basis of this we can identify and classify the malware family. We compare MD5 value from related entropy through which exact malware family detection is done. Figure 2 depicts the process of Malware classification through Entropy and MD5 value.

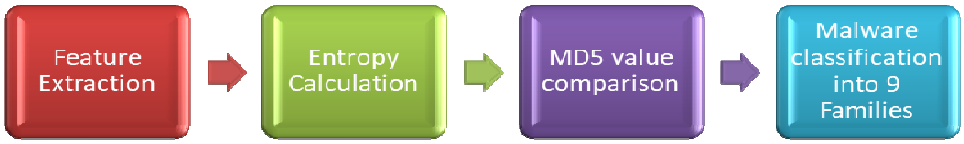

Figure1. Malware Family Classification Process

First of all we can extract the features from the PE file which is malware infected file, that file we get from the available dataset over the internet. After that we followed the procedure shown in the figure4.The analyzed file has been classified into the Family is TROJ_SPNR.07EM11 which is belongs to one of the classes of malware families.

V. RESULTS AND DISCUSSION

5.1. Data feature importance:
The analyzed data collected from the TekDefence.com. It is a freeware data source with password protected. Primarily the analysis is tested on the bx89.exe PE file. Many of disassembled code is containing the data define instruction such as db and dd.

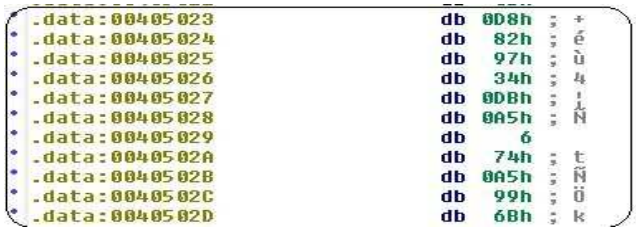

Figure 2: Assembly view

Figure 3 shows the sample which has no API call that describes only few assembly instructions. Above discussion is only for the malware infected files. We have 2723samples available for the classification in to the appropriate families.

5.2. Results and Graphs

As discussed earlier, we had used three classification algorithm of machine learning for malware classification families like Random Forest, Naïve Byes and Support vector machine. These are most commonly used algorithms therefore we had taken for classification. From figure 4 graph it can be analyzed that we had achieved highest classification accuracy through Support Vector Machine (SVM) which is 93.33%. After that we had got the accuracy from Naïve Byes (86.66%) and then lastly from Random forest of around 66.66%. This has been achieved by keeping 10 folds of datasets. In cross fold validation we divide the original dataset into 10 different sets of data, and is set act as train and test dataset for cross validation.

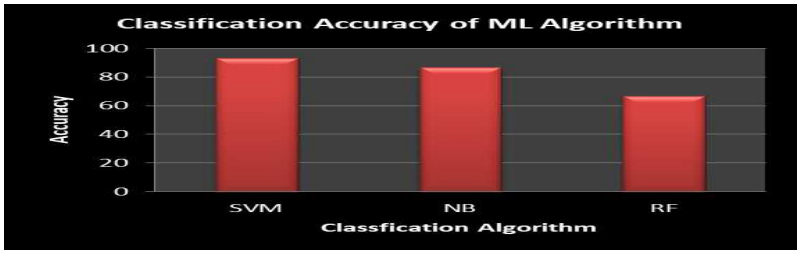

Figure 3: Classification Accuracy

With the help of HXD tool, we extract the embedded data into the Executable file from the Figure 5 and figure 6 we can conclude that, there is major difference between the benign file and malware file. We can see that with the help of the statistic report of the benign and malware file.

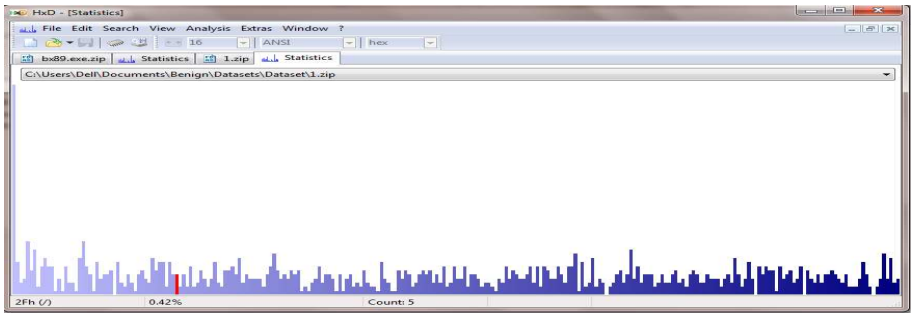

Figure 4: Benign File Statistic Report

The above figure graph is showing for benign file related to statistical report which is comparatively lower than the malware file. Benign file has reduced noise as compared to the malware file. Therefore the waves seen in the diagram are pretty much low down in benign file.

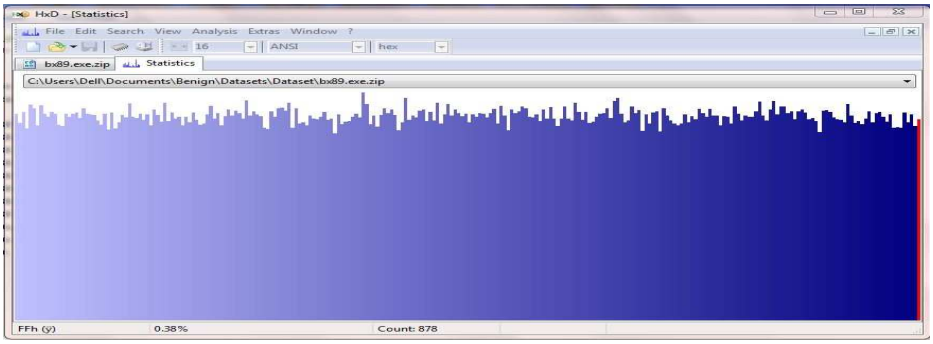

Figure 5: Malware File Statistic Report

The above figure graph is showing for malware file which is comparatively higher than benign file. From above figures it is conclude that the malware consume file is shown the higher graphs compare to the normal files that does not have any noise with the help of HXD tools we can analyze the noise in the actual PE file based on that we can conclude that file is malware infected or not.

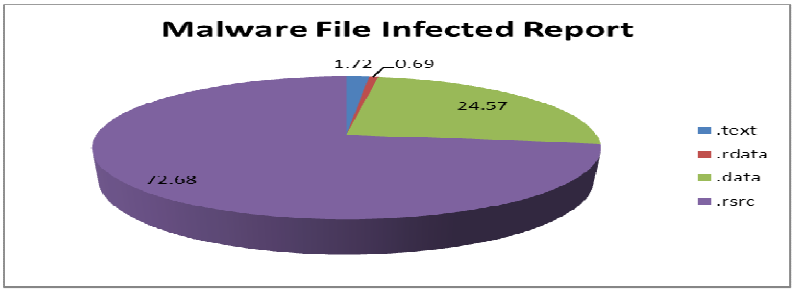

Figure 6: Section-Wise Report of malware File

The pie chart shows the Section-wise report in figure 7 of the malware infected file. The sections in pie charts contain .data, .text, .rdata, .rsrc. We can see that the maximum malware is found in the .rsrc section which is 72.68% followed by .data section which is comparatively lower than .rsrc section which is 24.57%, the minimum malware is found in the .rdata and .text section which is 0.69% and 1.72% respectively.

VI. CONCLUSION & FUTURE WORK

The proposed system describes the mechanism of identifying different malware through Portable Executable file (PE), by scanning this file and then breaking into different features of malware. Huge growth in unknown malware arising from multitudinous automated obfuscations, there's a requirement to ascertain malware detection strategies that are strong and economical. System has checked for various classification algorithms, got different results from different algorithms. The lowest accuracy was achieved by Random Forest (66.66%), followed by Naïve Byes

(86.66%) and Support Vector Machines (93.33%). The highest accuracy was achieved with SVM algorithm for multi-class classification.In future the modified SVM or Naïve Byes classification algorithm can be implemented so that it can increase the overall classification efficiency as our own contribution. Currently we had considered the data around 5 GB of Size, but we wish to process big data of around 500 GB and through this big data processing we will be minimizing the time constraint of data processing and improve classification accuracy of malware families.

VIII. REFERENCES

[1] Kaspersky Lab, WWW document. Available at:https://securelist.com/analysis/kaspersky-security bulletin/73038/kaspersky-security-bulletin-2015-overall-statistics-for-2015/, 2016.

[2] Kaspersky Labs, http://usa.kaspersky.com/internet-securitycenter/internet-safety/what-is-malware-and-how-to-protect-against- WJZS9xt942x, 2017.

[3] Matthew G. Schultz, Eleazar Eskin, Erez Zadok, and Salvatore J. Stolfo, “Data Mining Methods for Detection of New Malicious Executables”, in Proceedings of the Symposium on Security and Privacy, pp. 38-49, 2001.

[4] Pham Van Hung, “An approach to fast malware classification with machine learning technique”, Keio University,5322 Endo Fujisawa Kanagawa 252-0882JAPAN, 2011.

[5] Z. Fuyong and Z. Tiezhu, "Malware Detection and Classification Based on N-Grams Attribute Similarity", 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, pp. 793-796, 2017.

[6] Alazab, M, Layton, R, Venkatraman, S & Watters, “Malware Detection Based on Structural and Behavioural Features of API calls", in the 1st International Cyber Resilience Conference, Perth Western Australia, pp. 1-10,2010.

[7] Baskaran, Balaji, and AncaRalescu. “ A Study of Android Malware Detection Techniques and Machine Learning” , Proceedings of the 27th Modern Artificial Intelligence and Cognitive Science Conference, 15-23,2016.

[8] Vinod P. V.Laxmi,M.S.Gaur, “Survey on Malware Detection Methods”, 3rd Hacker, Workshop on Computer and Internet Security, Department of Computer Science and Engineering, Prabhu Goel Research Centre for Computer & Internet Security, IT, Kanpur, pp-74-79, March,2009.

[9] E. I. Edem, C. Benzaïd, A. Al-Nemrat and P. Watters, "Analysis of Malware Behaviour: Using Data Mining Clustering Techniques to Support Forensics Investigation", 2014 Fifth Cybercrime and Trustworthy Computing Conference, Auckland, pp. 54-632014.

[10] Mansour Ahmadi, Dmitry Ulyanov, Stanislav Semenov, Mikhail Trofimov, Giorgio Giacinto,"Novel Feature Extraction, Selection and Fusion for Effective Malware Family Classification", CODASPY ’16, March 09-11, 2016, New Orleans, LA, USA,.2016.

[11] X. Hu et al.,”Scalable malware classification with multifaceted content features and threat intelligence”, IBM Journal of Research and Development, vol. 60, issue 4, 2016.

[12] Ammar Ahmed E. Elhadi et al., “Malware Detection Based on Hybrid Signature Behaviour Application Programming Interface Call Graph”, American Journal of Applied Sciences, vol. 9, pp. 283-288 , 2016.

[13] Damodaran, A., Troia, F.D., Visaggio, ” A comparison of static, dynamic, and hybrid analysis for malware detection “,J Comput Virol Hack Tech,pp 13: 1, 2017

[14] Ashkan Sami, Babak Yadegari, Hossein Rahimi, Naser Peiravian, Sattar Hashemi, and Ali Hamze ,”Malware detection based on mining API calls”, In Proceedings of the 2010 ACM Symposium on Applied Computing (SAC '10). ACM, New York, NY, USA, pp 1020-1025,2010

[15] Pham Van Hung, “An approach to fast malware classification with machine learning technique”, Keio University,5322 Endo Fujisawa Kanagawa , 252-0882JAPAN,2011.

[16] R.K. Shahzad, S.I. Haider, and N. Lavesson, “Detection of spyware by mining executable files”, in Proceedings of the 5th International Conference on Availability, Reliability, and Security. IEEE Computer Society, pp. 295-302,2010

[17] Ronghua Tian, “An Integrated Malware Detection and Classification System”, Changchun University of Science and Technology, Thesis, August, 2011.

[18] Sunita Beniwal, Jitender Arora, “Classification and Feature Selection Techniques in Data Mining”, International Journal of Engineering Research & Technology (IJERT),Vol. 1 Issue 6, ISSN: 2278-0181, August – 2012.

[19] B. Sanjaa and E. Chuluun, "Malware detection using linear SVM", Ifost, Ulaanbaatar, pp. 136-138,2013.

[20] Robin Sharp, “An Introduction to Malware”, Springer 2012. Retrieved on April, 10, 2013.

[21] Anderson, B., Quist, D., Neil,” Graph-based malware detection using dynamic analysis “,Computer Virol , pp 7: 247, 2011.

[22] Robiah Y, Siti Rahayu S., Mohd Zaki M, Shahrin S., Faizal M. A., Marliza R., ”A New Generic Taxonomy on Hybrid Malware Detection Technique”, (IJCSIS) International Journal of Computer Science and Information Security, Vol. 5, No. 1, 2009.

[23] Raja Khurram Shahzad, Niklas Lavesson, Henric Johnson, “Accurate Adware Detection using Opcode Sequence Extraction”, in Proc. of the 6th International

Conference on Availability, Reliability and Security (ARES11),Prague, Czech Republic. IEEE, pp. 189-195,2011.

[24] V. Mehra, V. Jain and D. Uppal, "DaCoMM: Detection and Classification of Metamorphic Malware", Fifth International Conference on Communication Systems and Network Technologies, Gwalior, pp. 668-673, 2015.