An Analysis to Detect Malware using Machine Learning

Amit Sahu[1], Prachi Parwar[2], Deepak Agrawal[3]

[1]*Research Scholar, Takshshila Institute of Engineering and Technology, MadhyaPradesh India*
[2]*Professor, Takshshila Institute of Engineering and Technology, MadhyaPradesh India*
[3]*Hod, Takshshila Institute of Engineering and Technology, MadhyaPradesh India*

Abstract- *In recent times, malware has progressively turn out to be a dangerous threat to computer systems and mobile devices while traditional software inventions like antivirus and malware have not been so effective in protecting from the ever-evolving and advanced malware programs. In this paper, we proposed a new approach of malware detection using machine learning concepts.*

Antivirus companies are receiving thousands of malwares on the daily basis, so detection of malwares is complex and time consuming task. There are many malwares detection techniques like signature based detection, behavior based detection and machine learning based techniques, etc. The detection system work on signature analysis doesn't succeed to detect new unknown malware. In behavior analysis system, if the antivirus program detect try to modify or alter a file or communication over network then it will produce alarm signal, but there is still a chance of false positive rate.

N-gram approach in opcode and machine learning approach is best for classification of the malware.

Keywords- Malware Detection, Opcode, N-gram, Machine Learning, Malware Analysis, Data Mining, Random Forest.

I. INTRODUCTION

A Malicious software (malware) is computer program developed to damage your computer, network server and client. It can create several hazards like failure of system and loss or leakage of confidential information

The high distribution rate of computer malwares and traditional signature based antivirus unable to distinguish polymorphic, earlier unobserved malicious softwares. Rapid malware transmission is the serious problem for the world, thus becoming a serious threat. The traditional heuristic examination of static malware study is not effective and efficient compared against the great distributing rate of malware.

One planned method (solution) is by using automatic dynamic (behaviour) malware study along with machine learning (classification) techniques to attain efficiency in detecting malware.

II. MALWARE

Malware is malicious software which compromises the device. It is bad, unauthorised which breaks all the permissions given by the apps. There are many types of malware such as Worms, spam, ransom ware, spyware, rootkit, adware etc. The users who install these apps are in trouble later by installing. It is programmed by the attackers to steal all the credential information, e-mails, financial transactions, networking etc. So we need a mobile security application to protect against these malware attacks and to destroy it [37].

a) Trojans: It is a type of malware which appears to be as legitimate software. It steals the confidential data from the users without their knowledge. By this easy technique it gets access to the browsing [3] history, personal information, IMEI numbers, contacts, messages etc. For ex: A fake Facebook login will be given the user tries to sign in by his email id and password the hackers watches all this information and gains all the necessary information from the user this won't be known to the user.

b) Backdoor: It is a malicious code which exploits the device privileges and hides itself from the antivirus. It runs in the background so that it is not known to the victim.
Ex: Finspy, it is installed in the system and it automatically executes the files. It compromises the overall device security [46].

c) Worms: It creates a copy of it and sends it throughout the network.
For Ex: Cabir worms spread using the Bluetooth feature, by sending and receiving the files it spreads through the devices.

d) Spywares: This is malicious software which behaves differently.
Ex: Nickspy and GPSspy it sends the users information such as texts messages, contacts etc to the attacker who installed the software on victims device.

e) Keyloggers: It notifies the keystrokes and has a history of information. It is not known to the users whether it is installed or not. Whatever is typed in the keyboard is recorded; even the antivirus software's cannot find this malware.

III. EVALUATION OF MALWARE

In this era, several malware attacks occur that are important security threats for business organizations and daily users. Fig. 1 presents real state of malware attacks over the last 10 years. The whole number of malware has significantly increased. For example, Symantec informed that more than 357 million fresh variants of malware were detected in 2016. One of the core reasons for this bulk of malware samples is the wide use of obfuscation methods by malware creators, malicious files from the same malware family (i.e. similar code and common origin) are continually altered. In Intrusive Software Detection Based on Binary Analysis and Machine Learning order to cope with the rapid evolution of malware, it is essential to develop robust malware classification techniques that are accepting new modifications of malware files that are placed in the same family.

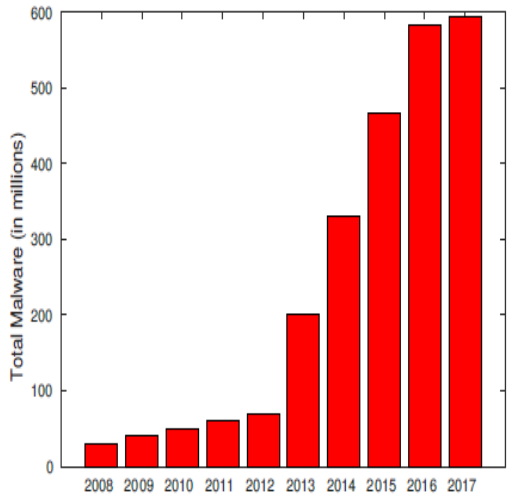

Fig -1: Last 10 years malware statistics.

Industrial control systems have not been safe either.Some industrial control systems control the electrical grid and their disruption could lead to loss of lives in hospitals due to lack of electricity. As cybercrime because more profit oriented, many enterprises have been the target of various malicious attacks that lead to heavy paydays as seen in the most recent swift attack on the Bangladesh bank where the attacker walked away with $81 million. Advanced Persistent targeted attacks that are specially designed for a system are now the bane of many company information security personnel. Still, with the Internet of Things evolution, it is only expected that many more systems will be compromised using malware as now, there will be many more viable entry points to any network. Figure 1 shows the main evolution milestones of the malware world.

IV. MALWARE ANALYSIS TECHNIQUES

Malware analysis involves dissecting a file sample in order to understand its behaviour and therefore its intent when introduced into a system. The expected result at the end of the analysis is to be able to devise ways in which to stop it from infecting systems and methods of removing it from systems it has already infected. [23]. Mitigation efforts could be range from end user training in order to avoid infection to development of detection signatures. There exist 2 main categories of malware analysis, static and dynamic.

Static malware analysis

Static malware analysis refers to analysis of malware without executing the source code. This study depends on features integral in the malware program. As mentioned earlier, detection rates for static detection are relatively lower than dynamic due to proliferation of malware obfuscation [24]. Additional drawback is that static malware analysis needs access to a signature database to evaluate malware.

With static malware detection efforts, malware analysts, evaluates various data sections of malware to extract malware signs. Some of these sections could be file headers and class names. Whilst static malware analysis may not be fruitful in complex malware cases, it still remains an important initial malware analysis step as information extracted during this stage can assist a malware analyst deduce the next step or even conclude with certainty that a file is malicious based on initial knowledge.

Dynamic malware analysis

Due to obfuscation techniques mentioned earlier, dynamic malware analysis is often necessary to determine malware effects and mitigation. With dynamic malware analysis, malware is executed and its activities monitored [23]. Dynamic malware analysis is time consuming and often requires specialised tools to accomplish. Increasing malware complexity, hence requiring dynamic analysis, is a tactic meant to waste malware analyst resources, thereby increasing chances for the malware to achieve its goals. Some of the tools used in Dynamic analysis include REMnux and OllyDbg. Dynamic malware detection involves executing a suspected malware sample and observing its behaviour as it is running. Dynamic malware evaluation usually occurs after static malware analysis, in order to analyse malware samples which have been

obfuscated making it almost difficult for static analysis. Dynamic analysis requires specialised tools and is fraught with many risks ranging from accidently attacking one's own network to accidently attacking another workstation on the internet by accident [26]. Ideally, a separate network with a separate internet connection should be configured to isolate network traffic from the main network. Recently, there has been a growth in the development of automated dynamic analysis tools that provide online sandboxing analysis and/or already configured systems that can be used for sample analysis. An example of such a system is cuckoo which has the malware online analysis tool and the cuckoo sandbox product available to malware analysts. Although this provides a way of making dynamic analysis faster and more efficient, complex malware samples are known to thwart these systems by detecting the dynamic analysis environment and then executing in a dubious way to mislead the analyst. The execution of samples in these environments also requires time.

V. WORK TO BE DONE

Amr I. Elkhawas, Nashwa Abdelbaki,[2018],In this paper we introduced our novel approach in using trigrams and PE file attributes as features for malware detection. We took a text mining approach to make our detection method more robust to polymorphism and metamorphism. We used opcodes trigram sequences as the main feature for our machine learning algorithm. We used Support Vector Machine (SVM) as our classifying algorithm which is a discriminative classifier model that gives a definite decision whether the predicted outcome belongs to the learned class or not.

Limitations can be eradicated by removing trigram, sequences of three-gram consecutive opcodes are passed in the code of malware which is time consuming. Bi-grams are most preferred usage in n-gram opcodes. PE header attributes have limited information in database and they cannot predict any new malware family. Feature extraction was done with limited steps.

Muhammad Murtaz, Hassan Azwar, Syed Baqir Ali, Dr. Saad Rehman,[2018] ,This study condenses the progression of malware detection techniques supported machine learning algorithms centred on the Android Operating systems. The model uses grouping strategies including stream based, bundle based and time-based highlights to describe malware families. During this analysis, a brand-new detection and characterization system for investigation significant deviations within the network behaviour of a smart-phone application is planned. The most goal of the planned system is to guard mobile device users and cellular infrastructure corporations from malicious applications simply nine traffic feature measurements.

Limitations are used of limited datasets. This study condenses the progression of malware detection techniques supported machine learning algorithms centred on the Android Operating systems.

VI. PROPOSED WORK

Amr I. Elkhawas, Nashwa Abdelbaki,[2018],In this paper they introduced Trigram approach and SVM(Support Vector Machine). The following text shows the pseudocode for the disassembly of the PE file.

```
read input_file
open output_file
get entry point of input_file
get raw size of input_file
get image base of input_file
set offset as entry point
while offset < (offset + raw size):
i = get instruction offset
x = get instruction string at (i + image base)
```

The following stage is the data mining phase. Disassembled file is breakdown into opcode trigram sequences as a list of tuples. We use another python script to automate this task. The text below represents the pseudocode used for trigram data mining.

```
define function  find_ngrams(input_list, n):
return input_list[i:] for i in range(n)
open input_file:
for line in input_file:
split words in line by 'space'
append only the first word in line to list
##Uses only first 1000 instructions
call find n_grams(list[:1000],3)
write to output_file
```

VII. RESULT

Amr I. Elkhawas, Nashwa Abdelbaki [2018] , In this paper they found the accuracy of this model is 89.47 %. The 95% Confidence Interval value means that 95% chance for the mean of this sample lies between the range of (0.8347 and 0.9386). The size of the confidence interval is inversely proportional to the sample size.

In this paper they found out following results

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  20   3
         1  13 116

               Accuracy : 0.8947
                 95% CI : (0.8347, 0.9386)
    No Information Rate : 0.7829
    P-Value [Acc > NIR] : 0.0002444

                  Kappa : 0.6523
 Mcnemar's Test P-Value : 0.0244489

            Sensitivity : 0.9748
            Specificity : 0.6061
         Pos Pred Value : 0.8992
         Neg Pred Value : 0.8696
             Prevalence : 0.7829
         Detection Rate : 0.7632
   Detection Prevalence : 0.8487
      Balanced Accuracy : 0.7904

       'Positive' Class : 1
```

VIII. CONCLUSION

Amr I. Elkhawas, Nashwa Abdelbaki [2018] says that the similar dataset was used afterwards dropping opcode trigram sequence as a feature with the identical SVM configuration with the detection rate for malicious files of 88%. Our result also suggest that our dataset size is properly adequate to stay functioning in this track.

REFERENCES

[1] Firdausi, Ivan, et al. "Analysis of machine learning techniques used in behavior-based malware detection." Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference on. IEEE, 2010.

[2] Lu, Yi-Bin, Shu-Chang Din, Chao-Fu Zheng, and BaiJian Gao. "Using multi-feature and classifier ensembles to improve malware detection." Journal of CCIT 39, no. 2 (2010).

[3] Santos, Igor, et al. "Opcodes-sequence-based semi supervised unknown malware detection." Computational Intelligence in Security for Information Systems. Springer Berlin Heidelberg, 2011. 50-57.

[4] Gavriluţ, D. and Ciortuz, L., 2011, September. Dealing with Class Noise in Large Training Datasets for Malware Detection. In Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2011 13th International Symposium on (pp. 401-407). IEEE.

[5] Zhao Z. A virus detection scheme based on features of Control Flow Graph. In Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on 2011 Aug 8 (pp. 943-947). IEEE.

[6] Gavrilut, D., Benchea, R., & Vatamanu, C, Optimized zero false positives perceptron training for malware detection. In Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2012 14th International Symposium on (pp. 247-253). IEEE.

[7] Raman, K., 2012. Selecting features to classify malware. InfoSec Southwest, 2012.

[8] Shahzad, F., Shahzad, M. and Farooq, M., 2013. In execution dynamic malware analysis and detection by mining information in process control blocks of Linux OS. Information Sciences, 231, pp.45-63.

[9] Santos, I., Brezo, F., Ugarte-Pedrero, X., & Bringas, P. G. (2013). Opcodes sequences as representation of executables for data-mining-based unknown malware detection. Information Sciences, 231, 64-82.

[10] Jiang, Q., Liu, N. and Zhang, W., 2013, December. A Feature Representation Method of Social Graph for Malware Detection. In Intelligent Systems (GCIS), 2013 Fourth Global Congress on (pp. 139-143). IEEE.

[11] Khammas, Ban Mohammed, et al. "FEATURE SELECTION AND MACHINE LEARNING CLASSIFICATION FOR MALWARE DETECTION." Jurnal Technology 77.1 (2015).

[12] Ranveer, S., & Hiray, S. SVM Based Effective Malware Detection System. In: 2015 International Journal of Computer Science and Information Technologies, Vol. 6 (4) , 2015.

[13] Mauricio Macías, et. al. Proposed "SGSI Support Through Malware´s Classification Using a Pattern Analysis" 9781509011476/16/$31.00©2016IEEE.

[14] Jhu-Sin Luo, Dan Chia-Tien Lo," Binary Malware image Classification using Machine Learning with Local Binary Pattern", 978-1-5386-2715-0/17/$31.00 ©2017 IEEE.

[15] Ken F. Yu, et. al. Proposed "Machine Learning in Malware Traffic Classifications", 978-1-5386-0595-0/17/$31.00 ©2017 IEEE.

[16] Amr I. Elkhawas, Nashwa Abdelbaki," Malware Detection using Opcode Trigram Sequence with SVM" Information Security- School of Communication and Information Technology Nile University Giza, Cairo, IEEE-2018.

[17] Muhammad Murtaz, Hassan Azwar, Syed Baqir Ali, Dr. Saad Rehman," A framework for Android Malware detection and classification" NUST, College of Electrical and Mechanical Engineering Islamabad, Pakistan, IEEE-2018.

[18] Udayakumar N, Vatsal J. Saglani, Aayush V. Gupta, Subbulakshmi T, "Malware Classification Using Machine Learning Algorithms"Proceedings of the 2nd International Conference on Trends in Electronics and Informatics (ICOEI 2018) IEEE Xplore ISBN:978-1-5386-3570-4.