IJIREEICE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

NCIET- 2020

National Conference on Innovations in Engineering & Technology
Govt., Polytechnic College, Palakkad, Kerala

Vol. 8, Special Issue 1, February 2020

A Survey on Malware Analysis

C. Reshma[1], Smithamol M B[2]

Student, Computer science, LBS college of engineering, kasargod, India[1]

HOD, Computer science, LBS college of engineering, Kasargod, India[2]

Abstract: Software that "deliberately fulfils the harmful intent of an attacker" is referred to as malicious software or malware. Malware is today one of the biggest security threats to the Internet. Malware refers to any binary or executable that is malicious. Viruses, worms, trojans, backdoors and adware are a few examples that fall under the umbrella of malware. Malware analysis is the process of analysing a malware sample/binary and extracting as much information as possible from it. The information we extract helps us understand the scope of the functionality of the malware, how the software was infected with the malware and how to defend against similar attacks in the future. Malware analysis experiments were carried out using the two techniques of malware analysis which are Static and Dynamic analysis. Static analysis is the process of analysing malware without executing or running it. The objective is to extract as much metadata from the malware as possible. Dynamic analysis is the process of executing malware and analysing its functionality and behavior. The objective is to investigate techniques that are used in order to effectively perform malware analysis and detection on enterprise systems to reduce the damage of malware attacks on the operation of organization's and to understand exactly how and what the malware does during the execution. The variants of malware families share typical behavioral patterns reflecting their origin and purpose. The behavioral patterns obtained either statically or dynamically can be exploited to detect and classify unknown malwares. The results showed that dynamic analysis is more effective than static analysis. Both the techniques are used for a comprehensive malware analysis and detection.

Keywords: Malware, Static analysis, dynamic analysis and Obfuscation.

I. INTRODUCTION

Malware analysis is the study or process of determining the functionality, origin and potential impact of a given malware sample such as a virus, worm, trojan horse, root kit, or backdoor. Malware or malicious software is any computer software intended to harm the host operating system or to steal sensitive data from users, organization's or companies. Malware may include software that gathers user information without permission. If an organisation discovers or suspects that some malware may have gotten into its systems, a response team may wish to perform malware analysis on any potential samples that are discovered during the investigation process to determine if they are malware and, if so, what impact that malware might have on the systems within the target organization's environment. Academic or industry malware researchers may perform malware analysis simply to understand how malware behaves and the latest techniques used in its construction. Vendors of software products and solutions may perform bulk malware analysis in order to determine potential new indicators of compromise, this information may then feed the security product or solution to help organizations better defend themselves against attack by malware. There are two fundamental approaches to malware analysis: static analysis and dynamic analysis. Static or Code Analysis is usually performed by dissecting the different resources of the binary file without executing it and studying each component. The binary file can also be disassembled (or reverse engineered) using a disassembler such as IDA. The machine code can sometimes be translated into assembly code which can be read and understood by humans: the malware analyst can then make sense of the assembly instructions and have an image of what the program is supposed to perform. Some modern malware is authored using evasive techniques to defeat this type of analysis, for example by embedding syntactic code errors that will confuse disassemblers but that will still function during actual execution. Dynamic or Behavioral analysis is performed by observing the behaviour of the malware while it is actually running on a host system. This form of analysis is often performed in a sandbox environment to prevent the malware from actually infecting production systems; many such sandboxes are virtual systems that can easily be rolled back to a clean state after the analysis is complete. The malware may also be debugged while running using a debugger such as GDB or WinDbg to watch the behaviour and effects on the host system of the malware step by step while its instructions are being processed. Modern malware can exhibit a wide variety of evasive techniques designed to defeat dynamic analysis including testing for virtual environments or active debuggers, delaying execution of malicious payloads, or requiring some form of interactive user input.

II. LITERATURE REVIEW

The obfuscation is a technique that makes programs harder to understand. For such a purpose, it converts a program to a new different version while making them functionally equal to each other. Originally, this technology aimed at protecting the intellectual property of software developers, but it has been broadly used by malware authors to elude detection. That is, in order to evade antivirus scanners, malwares evolve their body into new generations through the obfuscation technique. Clearly, it is important to analyses the obfuscation techniques to efficiently address malwares. There are many types of malware ENCRYPTED,

IJIREEICE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

NCIET- 2020

National Conference on Innovations in Engineering & Technology
Govt., Polytechnic College, Palakkad, Kerala
Vol. 8, Special Issue 1, February 2020

OLIGOMORPHIC, POLYMORPHIC AND METAMORPHIC MALWARE'S. The first approach to evade the signature-based antivirus scanners is to use encryption. In this approach, an encrypted malware is typically composed of the decryptor and the encrypted main body. The decryptor recovers the main body whenever the infected file is run. It uses a different key for making the encrypted part unique, but the decrypt remains constant which is used to detect by antivirus scanners. A polymorphic virus is a harmful, destructive or intrusive type of malware that can change or "morph," making it difficult to detect with anti-malware programs. A virus is said to be oligomorphic if it is capable of mutating it's decryptor only slightly. Interestingly, some products that were tested could not detect all instances of Memorial. A metamorphic virus is a type of malware that is capable of changing its code and signature patterns with each iteration. the obfuscation techniques commonly used in the polimorphic and metamorphic malware are Dead-Code Insertion, Register Reassignment, Subroutine Reordering, Instruction Substitution, Code Transposition, Code Integration. As a future trend, these obfuscation techniques will be more sophisticated and complex while being combined with one another. Problem of signature based detection system motivates the researchers to think about techniques to deal with new and unknown malware. Malware may perform malicious operations which are natural but sometimes they may appear in legitimate software's. Packed malware files and benign files are tested using different tools. Identified malware behavior in malicious samples also in benign files. Operations actions usually found in malware cannot be considered as an important element for malware detection because sometimes these are also performed in benign files. To determine the hidden behavior of a packed sample it is essential to execute it, there by dynamic analysis. Malware behavior analysis is the process to understand types and characteristics of malicious software. It is different from signature-based detection systems. Malware behavior analysis can also be done using dynamic analysis. combination of available tools and human expertise can be used in identifying malware behaviors. Malware analysis and detection is an essential technology that extracts the runtime behavior of malware and supplies signatures to detection systems and provides evidence for recovery, cleanup and forensics. To avoid the malware analysis and detection, malwares adopts measures such as: Shell Code, Polymorphism and Metamorphism which make the analysis and detection of malware very difficult. The target of Malware Detection is to judge the existence of malware in a file. It can be seen that, in SMD, it must first collect the malware sample before the detection. Behavior-based Malware Detection utilise the behaviour information of the malware during its execution as the detection basis. BMD will not be affected by shell code, Polymorphism and Metamorphism, and therefore can detect new malware. Based on malware behaviour extraction, the formal Malware Behaviour Feature (MBF) extraction method, detect newly appeared unknown malwares. Malware visualization is a field of knowledge that focuses on representing malware features in the form of visual cues. Visualization helps researchers to better understand malware graphically, highlighting certain interesting aspect of malware. Malware behavior image can be used in identifying malware variants. Malware behavior image could potentially introduce more ways for malware analysis, possibly through the use of image processing techniques. Single execution trace typically produces only part of the complete program behavior. Extended the analysis tool with the capability to explore multiple execution paths. The goal is to obtain a number of different execution paths, each with different behaviour also overview of the actions that an unknown sample can perform. The tool automatically provides the information under which circumstances a malicious action is triggered. Program processes interesting input (e.g., the local time, file checks, reads from the network). dynamically check for conditional branch instructions, snapshot is taken and again rewrite operation performed. For a significant fraction of malware samples in our evaluation set, the system is indeed exploring multiple paths. knowledge about a program's behavior is extended compared to a system that observes a single run. Mobile devices are one of the most needed - contains private information. Android is demanding software available there by attackers. Need to analyse the apps using some techniques to detect the malicious applications. Use of both static analysis and dynamic analysis and evolved the third technique named as hybrid analysis by mixing these. By comparing the pre-existing two techniques of malware detection internally, and also with the evolved technique and then compared the effectiveness of new evolved technique too. The HAAMD provides better and more accurate results as compared to individual static or dynamic analysis.

III. DISCUSSION

After analysing the literature survey some drawbacks are identified within these papers. Different malware analysis tools will make different classification of malware type. A malware poses different behaviour it is difficult to classify using analysis tools. When performing rewrite operations if memory locations are not updated related to argument value then the program executes impossible paths. Another drawback is the zero-day polymorphic malware. Sometimes benign files also contain similar operations as that of malicious files which allow anti-malware system to erroneously detect benign file as a malware

IV. FUTURE WORK

After analysing the literature survey, different tool will make different classification of malware type, hence human expertise is used to customise analysis result that generated by analysis tools, which can be extended further with better behaviour analysis approach, improve classification technique of malware and optimising malware detection. Emphasise on dynamic analysis is needed to discover hidden behaviour off packed sample. Packing information cannot be overlooked while describing malware, hence use of different machine learning & data mining algorithms may attempt to classify samples either as malware a benign with more accuracy.

IJIREEICE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

NCIET- 2020

National Conference on Innovations in Engineering & Technology

Govt., Polytechnic College, Palakkad, Kerala

Vol. 8, Special Issue 1, February 2020

V. CONCLUSION

Based on all literature survey's and the paper of malware analysis and detection in enterprise systems, it is evident that even though we have many tools to analyse malware behaviour statically and dynamically the intervention of human kind is required to analyse the behaviour and outputs from the analyse tools in order to determine whether next level of analyse is needed or not. Hence, it is time consuming. For this if implementing machine learning for malware analysis may become efficient, because day by day the malware is increasing rather than decreasing.

ACKNOWLEDGMENT

This work was supported by the department of computer science, Lbs college of engineering.

REFERENCES

[1] Andreas Moser, Christopher Kregel, & engin kirda, "*Exploring multiple execution paths for malware analysis",* IEEE Symposium on security & privacy, 2007
[2] Ilsun You, Kangbin Yim, "*Malware obfuscation techniques: A brief survey",* International conference on broadband, wireless computing, computer and applications, 2010.
[3] Mohamad Fadli Zolkipli, Aman janta, *"malware behaviour analysis: learning and understanding current malware threats",* Second international conference on network applications, protocols and services, 2010.
[4] Liu Wu, Ren Ping,Liu Ke, Duan Hai-xin, *" behavior based malware analysis and detection",* First international workshop on complexity & data mining, 2011.
[5] Syed Zainudeen Mohd Shaid, Mohd Aizaini Maarof, *"Malware behaviour image for malware variant identification",* International Symposium on Biometricand Security Technologies, 2014.
[6] Brightstarlang Wanswett, Hemanta Kumar Kalita, *"the threat of obfuscated zero day polymorphic malware: An analysis",* International conference on computational intelligence and communication networks, 2015.
[7] Mahima choudhary, Brij Kishore, *"HAAMD: hybrid analysis for android malware detection",* (ICCCI-2018)
[8] Mohammad Abu Qbeitah, Monther Aldwairi, *"Dynamic malware analysis of phishing emails",* (9th ICICS-2018)
[9] Om Prakash samantray, Satya narayan Tripathy, Susanta Kumar Das, *"A study to understand malware behaviour through malware analysis",* Proceeding of international conference on systems computation automation and networking, 2019.
[10] P.V.Shijo, A.Salim, *Integrated Static and Dynamic Analysis for Malware Detection,* the third ICoICT Conference,(2015), pp:804-811.
[11] Liu, Wu Ren, Ping Liu, Ke Duan, Haixin., *"Behavior-Based Malware Analysis and Detection",* Proceedings of Int. Workshop on Complexity data mining, 10.1109/IWCDM.2011.17, (2011), pp:39-42.
[12] Michael Bailey,Jon Oberheide,Jon AndersenZ, Morley Mao, Farnam Ja- hanian and Jose Nazario, *Automated classification analysis of internet malware,* In: Kruegel C., Lippmann R., Clark A. (eds) Recent Advances in Intrusion Detection. RAID 2007, Lecture Notes in Computer Science, vol 4637. Springer, Berlin, Heidelberg (2007), pp:178-197.
[13] Konrad Rieck, Thorsten Holz,Carsten Willems,Patrick Dusse and Pavel Laskov, *Learning and classification of malware behavior,* in DIMVA-2008, Springer Berlin Heidelberg, vol. 5137, (2008), pp:108-125.
[14] Nataraj, L., Karthikeyan, S., Jacob, G., And Manjunath,B. (2011). *Malware images: Visualization and automatic classification.* Proceedings of Visualization for Cyber Security (VizSec). 2011.
[15] Quist, D.A., and Liebrock, L.M. (2009). *Visualizing compiled executables for malware analysis.* 6th International Workshop on Visualization for Cyber Security, 2009 (VizSec 2009). pp. 27-32.
[16] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, and K. Rieck, *"Drebin: Effective and explainable detection of android malware in your pocket."* in Proceedings of the Network and Distributed System Security Symposium (NDSS), 2014.
[17] Ankita Kapratwar , *"Static and Dynamic Analysis for Android Malware Detection",* San Jose State University, May2016
[18] Yang-seo, choi, Ik-kyun Kim, jin-tae Oh et al. *PE File Header Analysis- based Packed PE File Detection Technique (PHAd).* International Symposium on Computer Science and its Applications 2008.
[19] Polymorphic-code. http://en.wikipedia.org/wiki/Polymorphic_code.
[20] Metamorphic-code. http://en.wikipedia.org/wiki/Metamorphic_code.
[21] https://en.wikipedia.org/wiki/Malware_analysis