International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 5, pp 32-40, May-2023

https://doi.org/10.47001/IRJIET/2023.705005

A Survey on Malware Attacks Analysis and Detected

[1]Naz faith M Jameel, [2]Muna M. T. Jawhar

[1,2]Software Department, College of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq

***Abstract* - Malware is one of the biggest problems modern internet users face. Private data and pricey computing resources are seriously threatened by the rise in malware attacks. Anti-malware businesses rely on signatures, which do in fact involve regular expressions and strings, to find malware and its related families. Recent malware attacks in recent years have demonstrated that signature-based techniques are error-prone and easily avoided by sophisticated malware programs. This essay provides an introductory overview of malware and analysis techniques used, as well as detection techniques used by researchers.**

Keywords: Malware, dynamic analysis, static analysis.

I. Introduction

One of the biggest risks in the world today is malware. World due to the swift improvements in communication devices and the Internet. Malicious software or malware is the result of several viruses that have been developed to carry out malicious activity, to steal information, to scan, or otherwise. The malware mainly aims to cause massive destruction of information and resources, or gain unauthorized access to the network. Moreover, it is described as "a class of computer program designed to damage a legitimate user's machine in a variety of ways." [1] These threats are widespread programs and viruses as well as in terms of both number and quality.

Researchers in recent years have become more interested in safeguarding delicate and important data from the increasingly electronic and cyber attacks.

The researchers suggested a variety of solutions to defend malware attacks. Most solutions generally use two main ways to detect harmful programs: The signature approach and anomalous approach or behavior -based approach. A third type is the hybrid between the previous two types. Signature methods are very effective in discovering well -known malware [2]. The detection system for the signing -based malware analysis or the file fingerprint and compares after put it in a database. for the well-known files' signatures. The signature is a series of data in a file, and most harmful programs can change their signatures or create new inconsistencies each time to prevent detection, so the operations of detecting the sale fail at some time in the event of incompatibility in the database. The second type of detection is adopted anomalous behavior of harmful programs to detect them. Although the methods of detecting anomalies are better than the signing methods, they have a high false warning rate. [3]

The problem of discovering harmful programs is that it is a task classification, either working on a set of data containing executive files classified into harmful or benign programs. In this case, we need algorithms that deal with large data and have the ability to classify data into two categories, either harmful or benign programs. [4]

Recently, many researchers used automatic learning techniques to deal with big data and difficult tasks including classification and detection of malware. It is usually based on the analysis of binary files and is divided into two categories: static or dynamic analysis. [5]

In this research paper, harmful programs were studied and their types were identified. In the next section, identify the basic detection methods and then compared to research published in the field of detection of malware, methods used and the results obtained.

II. Malware Types \ Types of malware [6]

Malware is computer software that has been installed on a user's computer without their consent. It can sabotage a computing device, steal data, or gain unauthorized access. The list that follows shows popular malware subcategories:

- Virus: A program that replicates by inserting its code into other programs, to benefit itself. An infection can move from one machine to another as well as from one application to another.
- Worms: harmful software that repeat themselves on the destroy the data and files on the computer. Worms may likewise encrypt data or send spam email messages. Worms transport themselves in their own containers, in contrast to viruses.
- Trojan horse: Disguised as authentic software, Trojan horses do unauthorized and undesirable acts. Trojans give assailants access to a functioning Using a computer, retrieve user information such as password, confidential bank information, etc.
- Spyware:\Spyware is computer software that continuously monitors user behavior. It serves to collect data pertaining to users such as regularly visited web

ISSN (online): 2581-3048
Volume 7, Issue 5, pp 32-40, May-2023
https://doi.org/10.47001/IRJIET/2023.705005

pages without their knowledge, and payment card details, then transmits these information returned to the assailants.

- Rootkit: A group of harmful software known as a rootkit programs created to enter a computer network and provide various malware kinds access to the system.
- Ransomware: is an undesirable application that gives a hacker access to the victim's computer and locks it, preventing them from accessing important data. Ransomware encrypts crucial information about the compromised device or network and then demands the cost of lifting the restrictions.
- Adware: Ad-A form of malware is supported software that constantly brings commercials to your computer. Adware is usually packaged with no cost-download apps and software such as free-to-play games.
- Botnet: A malicious program that remotely managing a number of devices such as computers, smart phones and IoT devices, infested with malware and managed by a cybercriminal. Botnets are commonly utilized for denial-of-service attacks or spam operations. Users frequently not aware that their infrastructure is infected with malware in a botnet.

III. Detection Methods

Malware detection techniques are divided into numerous categories based on their perspectives. The two primary techniques for malware detection signature-based and heuristic are covered in this section. Figure 1 displays the key ways to detect malware.

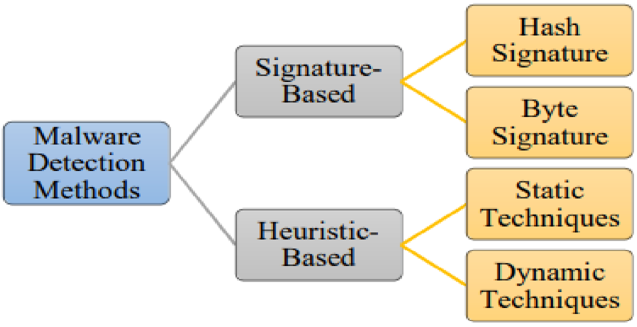

Figure 1: Methods of malware detection

1) Signature-based disclosure

The nearly all antivirus programs available use the signature-based strategy. This technique uses the malware file that was seized to extract a distinctive signature, which is then used to locate malware with a similar signature. [7]

Most antivirus programs use this type. This type relies on the malware's own a signature to recognize it. A signature is a group of characters that may be used to recognize certain viruses is a file hash or a hash of bytes. [8] Consequently, this technique has a low false positive (FP) rate. But in this type we cannot detect new species or species that change their signature. The method based on signatures relies about the use of static parsing to take out special Bytes in succession called tags.

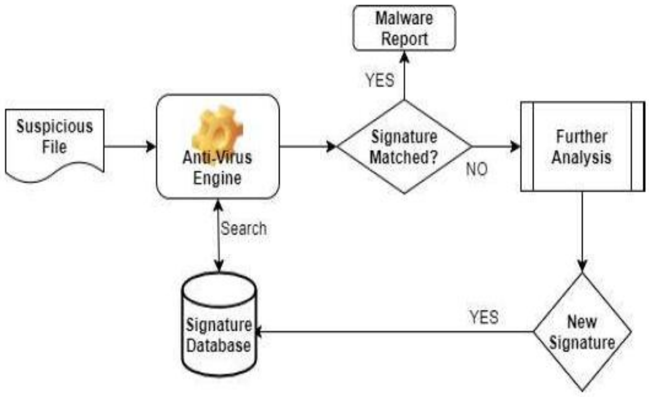

Figure 2: Signature-based general flow

2) Heuristic-Based Detection

Heuristic detection is additionally known as anomalous detection or behavioral detection. The actions taken this detection identify the virus as examined in the training phase while it is running. Then, according to a pattern discovered while training testing, the file is categorized as harmful or benign during the testing phase. This type is able to recognize all unfamiliar malware. However, the main action -based disadvantages are the large false positive (FP) rate.

Detection is generally based file playback behaviors, and these technologies consist of Naive Bayes, Support Vector Machine, Decision Tree, Random Forest, as well as machine learning and deep learning techniques. [9]

IV. Malware Analysis Techniques

The goal of malware analysis to research harmful files to gain a greater comprehension of various malware characteristics, including behavior, development over time, and target specificity.

The methods employed in malware analysis are primarily broken down into three categories: static, dynamic, and mixed analysis. Additionally, memory-based analysis is yet another excellent tool technology in analyzing malware. [10]

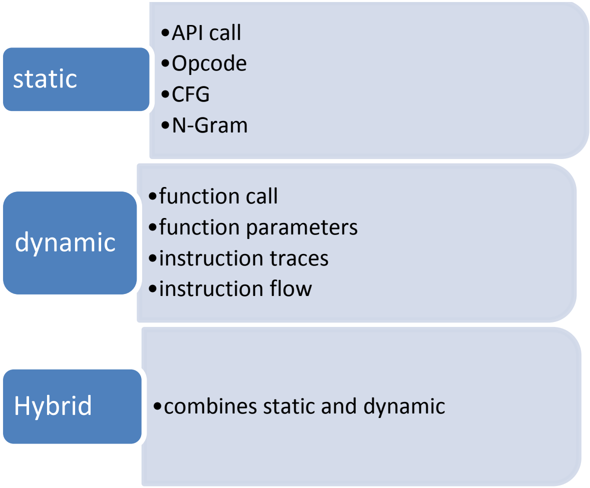

Figure 1: Malware analysis techniques and features

1) Static Analysis

This technology relates to parsing without launching them, portable files with execution (PE files). Malware frequent AWly employs package binary software, the ASP Pack Shell and UPX, to prevent being parsed [11]. The PE file must first be decompressed. It can be parsed to disassemble a IDA Pro and Olley Dbg are two disassembler tools that can be used with Windows executable files displays assembly guidelines, provides malware knowledge and pattern extraction to identify the attacker..

Static analysis allows for the extraction of the detection pattern such as calls to the Windows API, string signatures, and control flow diagram (CFG), opcode frequency. (opcodes) and n-grams byte sequence.[12].

We will go over the main aspects of static analysis in the sections that follow.

Calls from the Windows API (Application Programming Interface) are almost universally used by programs to interact with the operating system. For instance, "OpenFileW" is the Windows.API.in "Kernel32.dll" It either opens an existing file or generates a new one. As a result, API calls are a vital indicator in the identification of malware since they expose the activity of applications. For instancen, Windows API requests. "WriteProcessMemory", "LoadLibrary" and "CreateRemoteThread" are suspect a characteristic of malware to inject a a process with a DLL, while they Rarely do they come together in a genuine group.

Chains are a reliable indication of a malicious presence Strings disclose the attacker's intents and objectives because they frequently include important information on semantics [13]. For instance, the string that follows. When a malicious file is located external to the standard PE.0header, it is a common feature of compilers and contractors, it will display the message" DOS mode cannot be used to run this software.

Control Flow Graph (CFG): A directed graph called a CFG shows the control flow of a program, in which codes in blocks shown using nodes and edges govern direction of the flow. CFG may be used in malware detection to record a PE file's operations and extract the structure of the program.[14]

Opcodes are the first piece of machine code (also called machine language) instructions that specify the procedure to be performed using the CPU. A complete teaching machine language consists of opcode and possibly a single or more operands (eg, "moveax 7", "add eaxecx" and "sub ebx 1").

An opcode may be utilized as an advantage in detecting malware from measuring the regularity of opcodes or comparing opcode series.

N-grams.are all consecutive overlapping strings of length N [15]. For instance, the term "MALWARE" is a seven character string that may be broken down into three parts grammes such as: "MAL", "ALW", "LWA", "WAR" and "ARE". NGrams has been executed with many detecting capabilities like as opcodes and API requests. Furthermore to the features mentioned previously, additional characteristics including file size and function length have been employed in static analysis Network features such as Static analysis also includes features for TCP/UDP ports, IP addresses at a destination, and HTTP requests.[16].

The most important researches on virus signature evasion methods was completed by Vigna and Kirat [17]. They were successful in stealing techniques. From 2,810 malware samples are categorized into 78 elusive signature approaches that are all comparable.

Al-Hashemi and Hamza introduced a fresh method that creates a digital image of the special opcode by removing it from the executable file. Using the Local Binary Pattern next (LBP), which is a visual feature extraction one of the most common extraction of texture techniques for processing images. Lastly, malware detection relies on machine learning techniques The suggested detecting method obtained an accuracy rate of 91.9% [18].

Cheng et al. [19] analyze the sequence of native APIs using the WinDbg tool and apply the Support Vector Machine to find malware that uses shell code. They were successful in achieving an accuracy rate of 94.37% while using a fairly limited training set. The false negative rate, however, was 44.44%.

ISSN (online): 2581-3048
Volume 7, Issue 5, pp 32-40, May-2023
https://doi.org/10.47001/IRJIET/2023.705005

Table 1: Shows the results of surveyed papers that applied static analysis in their malware detection approaches

Research	Method	accuracy rate	false negative
Al-Hashemi and Hamza	LBP	91.9%	
Cheng et al	SVM	94.37%.	44.44%.

2) Dynamic Analysis

Another name for it is behavior analysis. In this analysis, in this type, In a supervised setting, suspicious file execution and monitoring are performed such as a VM or an emulator.

In contrast to static analysis, dynamic analysis is more efficient since it does not need disassembling the infected file for analysis. In addition, the dynamic analysis is capable of identifying both known and undiscovered malware. New malware cannot also avoid dynamic detection though dynamic parsing is time consuming and resource-intensive [20].

Dynamic analysis uses malware that is executed in a controlled setting to see malicious files in action without getting harmed. Different control environments come in different forms such as emulators, debuggers, emulators, and simulated machines.

A malicious application can be executed in a controlled environment called an emulator. A full simulation the system is in charge of the hard drive, CPU, and resources.

Another sort of a managed environment is a debugger, a program that monitors and verifies that other binary programs are running properly. Debugging tools like Win Dbg,. Olly Dbg, and GDB can be used to track the way questionable binaries execute at the instruction level.

Another environment is simulation, which is software that simulates a procedure that the user may see without actually completing that process. Emulators like as CWS and box, Norman sandbox, and Detours permitting malware to be executed in a supervised virtual setting and observe its actions a process's function calls to any DLL be intercepted, detours are employed (DLL injection), while CWS and box Implements API hooking to record Windows API calls made by malicious software as opposed to that, Norman sandboxing mimics Windows and LAN connection, and Internet connection on the virtual machine. [21]

The most prevalent controlled setting is the virtual machine (VM). A VM is a computer program that is able to run apps and an operating system. These programs are kept separate from the host computer. Therefore, opening a file or program the host computer cannot be harmed from inside a virtual machine. A virtual machine is created, executed, and managed by a program called Virtual Machine Monitor (VMM). Also, it's in charge of assigning hardware for the virtual machine.[22]

Moreover, the majority of virtual machines and debuggers produce related drivers and files specific tool, and Viruses may search for these tools to see if virtual machines or debuggers are present [23]. Most dynamic techniques have API calls were utilized to represent malware behavior.

In [24] API calls that are linked and have the same semantic goals are mapped into sequences. The decision tree was used to obtain its maximum accuracy of 97.19%.

Ki et al. [25] Apply the Longest Common Sequence (LCS) algorithm to compare related sequences. The strategy produced accurate results of 99.8% and zero (0) false positives. Moreover, Fan et al. [19] Use API hooking to track which APIs malware is trying to occult. This technology observes both normal Native APIs and APIs. Such as low-level, undocumented APIs. Utilizing decision tree and Naive Bayesian techniques, the detection rate was 95%.

Table 2: The results papers that applied dynamic analysis in their malware detection approaches

Researcher	Methods	Accurse	False positives
[33]	Decision tree	97.19%	
Fan et al. [35]	decision tree and Naive Bayesian algorithms	95%	
Ki et al.	LCS	99.8%	0

3) Hybrid analysis

Hybrid analysis uses static and dynamic analysis to compile information on malware with the use of hybrid analysis. Gain for security researchers' benefits of static and dynamic analytics. Hence, the capacity for correctly recognizing harmful programs increases [26]. Both analyzes have distinct benefits and restrictions Static analysis is quick and inexpensive and more secure vs dynamic analysis Nonetheless, malware avoids this by using concealment techniques. Yet, dynamic analysis is trustworthy and can defeat obscurity strategies. Moreover, it is capable identify unknown malware versions and malware groups. Though, it is time consuming and resources [27].

V. Machine learning in Malware detection

In this section, we review some of the researchers' work in this field and the techniques that were used with the results they obtained. The research in [28] used CNN-LSTM To address serious shortcomings in malware detection the suggested CNN-LSTM method is employed for the early

ISSN (online): 2581-3048
Volume 7, Issue 5, pp 32-40, May-2023
https://doi.org/10.47001/IRJIET/2023.705005

identification of malware accuracy is 99%.The accuracy of the other classifiers is 98% for DT and 95% for SVM. The LSTM model's accuracy is 99%, its recall accuracy is 99%, and its F1 score is 1.

In the study [29], CNN was utilized to identify malicious PDF files by analyzing them at the byte level. The harmful samples come from Virus Total. The network demonstrated its ability to properly differentiate between different virus families Model accuracy CNN 99.83%.

In research [30] used decision trees with the Ada Boost algorithm. The suggested the system was tested and trained using a fresh, comprehensive dataset for PDF documents called Evasive-PDFMal2022 accuracy 98.84%.

Research [31] used Faster R-CNN (F-RCNN) On the Kaggle website, the benchmark database from the Microsoft Malware Classification challenge has been utilized to evaluate system performance an overall categorization of average accuracy of 98.12%.

In [32] the researcher used DT, CNN, and SVM algorithms completely reliant on information from the Canadian Institute for Cyber Security. The results show that, DT (99%), CNN (98.76%), and SVM (96.41%).

In this research [32], we investigated how ML and DL techniques appropriate the scope of malware detection and How might the selected dataset affect the outcomes of the classifier. The dataset is taken based on the publicly accessible data from IEEE Data port the result shows that the accuracy 98.91%.

In [33] the authors used Random Forest By utilizing several machine learning models; the goal of this study is to identify the best machine learning method to determine whether or not a file is harmful. To choose the best machine learning method; in order to evaluate if a file is harmful using several machine learning models. Any internet repository may be used to collect a dataset repository the result shows that the accuracy 97.99%.

The study in [34] suggests an effective a strong malware detection strategy that makes use of machine learning classification algorithms, there employed signature-based ngram an attributes and behavior-based API (Application Programming Interface) call a succession of malware analysis. The malicious records are gathered from VX Heaven and the original files are gathered from web resources: Download.com and Softpedia.com. The model accuracy is 98.6%.

The research in [35] used bag-of-words (BoW) NLP model to formulate the behavioral statistics reveals. In this research, the authors suggested an investigative methodology for malware detection and family attribution that is adaptable, effective, and still efficient. The fundamental idea is to use the bag of words model to simulate behavioral reports. Then, we use cutting-edge NLP and ML methods to create discriminative machine learning ensembles. Using the datasets from Malgenome, Drebin, and MalDozer, MalDy obtains over 94% f1-score in the Android detection test and over 90% in the attribution task.

Research in [36] used a fresh strategy based on the MLPdf multilayer perceptron neural network model for the identification of malware that uses PDFs. In further detail, the MLP model employs a back propagation algorithm with stochastic gradient descent look for the model update 95.12% that was attained while retaining a relatively low FPR of 0.08%.

In research [37] authors CNN employed a deep learning strategy. Use a normal 2d CNN at first and train it using the metadata that the hypervisor has collected for each process within a virtual machine (VM). Findings revealed a notable increase in accuracy of 90% for 3d CNN.

Research in [38] Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) are two Machine Learning (ML) algorithms that were implemented and evaluated for feature set classification into either benign or malicious applications (apps). Malicious and benign applications on a dataset named M0Droid2 had average accuracy rates of 79.08% and 80.50%, respectively, with average true positive rates of over 67.00% and 80.00% using SVM and KNN, respectively.

In [39] research study To classify malware, a deep Eigen space learning technique was utilized samples were collected using Virus Total 2 Threat Intelligence the result of technique for accurately detecting malware of 98.37% and a precision rate of 98.59%.

To minimize image size, non-local methods of denoising for noise reduction and discrete wavelet transform are used were two crucial approaches. Four sizable datasets are frequently used to locate and classify malware in the field of malware categorization and detection. BIG2015, MalImg, Malicia, and Malevis are these datasets. The authors employed a malware dataset based on memory forensics that consists of 4294 malicious and clean dump files in total. Via the development of a virtual Windows 10 environment, these dump files were retrieved. With Prodcump A dump file is created by extracting the system's memory image, the result reveals that the performance indicators of 97.82% for accuracy, 97.66% for precision, 97.25% for recall, and 97.57% for f1-score are obtained.

Table 3: The comparison results papers that applied machine learning for malware detection approaches

Reference	Method	Detection accuracy	F1 score	TP rate	Dataset samples
[1]	CNN-LSTM	99%.	1		
	DT	98%			
	SVM	95%			
	LSTM	99%			
[2]	CNN	99.83%			VirusTotal
[3]	AdaBoost	98.84%.			Evasive-PDFMal2022
[4]	R-CNN	98.12%			Microsoft Malware Classification on the Kaggle website
[5]	DT	99%			Canadian Institute for Cyber Security
	CNN	98.76%			
	SVM	96.41%			
[6]	ML and DL	98.91%			IEEE Dataport
[7]	Random Forest(RF)	97.99%			
[8]	Machine learning classification	98.6%.			VX Heaven Download.com and Softpedia.com
[9]	(BoW) NLP	94%			Malgenome, Drebin, and MalDozer
[10]	MLP (df)	95.12%			
[11]	CNN	90%			
[12]	SVM	79.08%		67.00 %	M0Droid2
	KNN	80.50%		80.00 %	
[13]	Deep Eigen space learning	98.37%		98.59 %.	VirusTotal 2
[14]	Non-local means denoising	97.82%	97.57 %	97.25 %	BIG2015, MalImg, Malicia, and Malevis

VI. Results and Discussion

Several security specialists believe that malware's future is still up in the air. The future of malware creation faces a variety of difficulties that we think security firms and researchers should take into account. Automating the production of malware variants is the first issue.

By researching attackers can create automated programs that can generate tens of thousands of different malware samples daily utilizing machine learning and the latest malware detection techniques. Second, malware groups could rent or sell malware automation technologies, allowing less experienced groups and amateur hackers to get into the realm of malware.

Third, malware changes swiftly both in terms of structure and usability. The majority of the approaches investigated learned and tested behaviors using a single malware dataset (classifier). Despite their high probability of detection, the outcomes will vary when the approaches are used on newly released malware. Lastly, future malware is anticipated to be more advanced. Attackers may employ cutting-edge encryption or obfuscation techniques to prevent malware from being detected and analyzed.

The traditional method used by antivirus software to catch malware is to look for a common signature regrettably; this method can simply be avoided with a simple perturbation technique [41]. Static and dynamic analytics also have some restrictions. Memory Analysis, on the other hand, offers a thorough malware analysis. The code of malware can be successfully disguised within the computer system. However, malware

To finally complete its job, it has to run its own code from memory. RAM that is volatile maintains its data until it is switched off. As a result, RAM analysis can provide information on system activity.

The active network connections, sockets, ports, running processes, DLLs, files, registry keys, services, and valuable live information are all stored in memory [42]. Memory analysis is thus a potential filed. Technology that, along Using data mining and machine learning approaches, malware detection is anticipated to gain popularity.

Collecting malware samples is crucial for researchers in order to investigate malicious tactics and strategies. One method of sample collection is the use of attraction loci, this is a special device used to lure attackers to pick up their assault methods [43]. Researchers can also utilize URLs that are known to be harmful. The malware dataset may also be acquired via anti-malware agent websites such the Virus Share malware repository, Malware DB, Malwr, MalShare, VX Paradise, the Zoo, and Malware DB. Moreover, several specialist businesses and research endeavor teams occasionally offer their malware dataset collections. A 500GB data collection of recognized malicious files was submitted to the Grand Challenge competition in 2015 by Micros 7oft. [44].

VII. Conclusion

Malware causes a serious a risk to our data internet, systems, and computers. Antivirus software and security researchers have faced many difficulties as a result of the difficulties posed by malware writers who routinely produce complicated software changes its signature to evade detection and releases more sophisticated posed by malware writers who routinely produce complicated software. In this study, we have made a brief survey of Malware kinds and techniques for detecting them. We also looked at three different methods for analyzing malware: static, dynamic, and hybrid. Techniques that malware Obfuscation, attack, and anti-analysis strategies were also examined as ways to avoid discovery. Lastly, this article studies the key sources of the malware dataset as well as the projected trend in malware creation.

REFERENCES

[1] Saja Abu-Zaideh, Mohammad Abu Snober and Qasem Abu Al-Haija. “Smart Boosted Model for Behavior-Based Malware Analysis and Detection”.

[2] Mozammel Chowdhury(&), Azizur Rahman, and Rafiqul Islam, Malware Analysis and Detection Using Data Mining and Machine Learning Classification.

[3] Tian, R., Islam, R., Batten, L., Versteeg, S.: Differentiating malware from clean ware using behavior al analysis. In: International Conference on Malicious and Unwanted Software: MALWARE 2010, pp. 23–30 (2010).

[4] Dinh Viet Sang, Dang ManhCuong, Le Tran BaoCuong. “An Effective Ensemble Deep Learning Framework for Malware Detection”, Conference Paper • December 2018. SoICT 2018, December 6–7, 2018, Danang City, Viet Nam. DOI: 10.1145/3287921.3287971.

[5] Manuel Egele, Theodoor Scholte, Engin Kirda, and Christopher Kruegel. 2012. A survey on automated dynamic malware-analysis techniques and tools. ACM computing surveys (CSUR) 44, 2 (2012).

[6] Vidyarthi, D., Kumar, C. R. S., Rakshit, S., & Chansarkar, S. (2019). Static malware analysis to identify ransom ware properties. International Journal of Computer Science Issues (IJCSI), 16(3), 10-17, DOI:10.5281/zenodo.3252963.

[7] Shijo, P. V., & Salim, A. J. P. C. S. (2015). Integrated static and dynamic analysis for malware detection . Procedia Computer Science , 46, 804-811.https://doi.org/10.1016/j.procs.2015.02.149.

[8] Yunus, Y. K. B. M., &Ngah, S. B. (2020, February). Review of hybrid analysis technique for malware detection. In IOP conference series: materials science and engineering (Vol. 769, No. 1, p. 012075). IOP Publishing. DOI 10.1088/1757-899X/769/1/012075.

[9] Talukder, S. (2020). Tools and techniques for malware detection and analysis. arXiv preprint arXiv:2002.06819.

[10] Damodaran, A., Troia, F. D., Visaggio, C. A., Austin, T. H., & Stamp, M. (2017). A comparison of static , dynamic, and hybrid analysis for malware detection . Journal of Computer Virology and Hacking Techniques, 13, 1-12. DOIhttps://doi.org/10.1007/s11416-015-0261-z.

[11] Han, W., Xue, J., Wang, Y., Huang, L., Kong, Z., & Mao, L. (2019). MalDAE: Detecting and explaining malware based on correlation and fusion of static and dynamic characteristics . & security, 83, 208-233.https://doi.org/10.1016/j.cose.2019.02.007.

[12] Fleshman, W., Raff, E., Zak, R., McLean, M., & Nicholas, C. (2018, October). Static malware detection & subterfuge: Quantifying the robustness of machine learning and current anti -virus. In 2018 13th International Conference on Malicious and Unwanted Software (MALWARE) (pp. 1-10). IEEE.DOI: 10.1109/MALWARE.2018.8659360.

[13] Rami Sihwail, Khairuddin Omar, K. A. Z. Ariffin. “A Survey on Malware Analysis Techniques: Static, Dynamic, Hybrid and Memory Analysis“. International Journal on Advanced Science Engineering and Information Technology, September 2018. DOI: 10.18517/ijaseit.8.4-2.6827.

[14] D. Ucci, L. Aniello, and R. Baldoni, “Survey on the Usage of Machine Learning Techniques for Malware Analysis,” arXiv Prepr.arXiv1710.08189, pp. 1–67, 2018.https://doi.org/10.1016/j.cose.2018.11.001.

[15] E. Gandotra, D. Bansal, and S. Sofat, “Malware Analysis and Classification: A Survey,” J. Inf. Secur., vol. 05, no. 02, pp. 56–64, 2014.DOI:10.4236/jis.2014.52006.

[16] Abou-Assaleh, T., Cercone, N., Keselj, V., & Sweidan, R. (2004, September). N-gram-based detection of new malicious code . In Proceedings of the 28th Annual International Computer Software and Applications Conference, 2004. COMPSAC 2004. (Vol. 2, pp. 41-42). IEEE.DOI: 10.1109/CMPSAC.2004.1342667.

[17] H. Hashemi and A. Hamzeh, “Visual malware detection using local malicious pattern,” Journal of Computer Virology and Hacking Techniques, pp. 1–14, 2018.

[18] Y. Cheng, W. Fan, W. Huang, and J. An, “A Shellcode Detection Method Based on Full Native API Sequence and Support Vector Machine,” in IOP Conference Series: Materials Science and Engineering, 2017, vol. 242, no. 1, pp. 1–7.DOI 10.1088/1757-899X/242/1/012124.

[19] H. S. Galal, Y. B. Mahdy, and M. A. Atiea, “Behavior-based features model for malware detection,” J. Comput. Virol. Hacking Tech., vol. 12, no. 2, pp. 59–67, 2016.

[20] Y. Ki, E. Kim, and H. K. Kim, “A novel approach to detect malware based on API call sequence analysis,”

Int. J. Distrib. Sens. Networks, vol. 2015, no. 6: 659101, pp. 1–9, 2015.

[21] C.-I. Fan, H.-W. Hsiao, C.-H. Chou, and Y.-F. Tseng, “Malware Detection Systems Based on API Log Data Mining,” in 2015 IEEE 39th Annual Computer Software and Applications Conference, 2015, pp. 255–260.

[22] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, “A survey on automated dynamic malware-analysis techniques and tools,” ACM Comput. Surv., vol. 44, no. 2, pp. 1–42, 2012.

[23] M. Sikorski and A. Honig, Practical malware analysis: the hands-on guide to dissecting malicious software. No starch press. 2012.

[24] Chen, X., Andersen, J., Mao, Z. M., Bailey, M., &Nazario, J. (2008, June). Towards an understanding of anti -virtualization and anti -debugging behavior in modern malware . In 2008 IEEE international conference on dependable systems and networks with FTCS and DCC (DSN) (pp. 177-186). IEEE.DOI: 10.1109/DSN.2008.4630086.

[25] M. Eskandari, Z. Khorshidpour, and S. Hashemi, “HDM-Analyser: a hybrid analysis approach based on data mining techniques for malware detection,” J. Comput. Virol. Hacking Tech., vol. 9, no. 2, pp. 77–93, 2013.DOIhttps://doi.org/10.1007/s11416-013-0181-8.

[26] Faruk, M. J. H., Shahriar, H., Valero, M., Barsha, F. L., Sobhan, S., Khan, M. A., & Wu, F. (2021, December). Malware detection and prevention using artificial intelligence techniques. In 2021 IEEE International.

[27] Fettaya, R., & Mansour, Y. (2020). Detecting malicious PDF using CNN. arXiv preprint arXiv:2007.12729. N, https://doi.org/10.48550/arXiv.2007.12729.

[28] Al-Haija, Q. A., Odeh, A., & Qattous, H. (2022). PDF Malware Detection Based on Optimizable Decision Trees. doi: 10.20944/preprints202209.0103.v1

[29] Deore, M., &Kulkarni, U. (2022). Mdfrcnn: Malware detection using faster region proposals convolution neural network . International Journal of Interactive Multimedia and Artificial Intelligence, Vol. 7, No 4, DOI: https://doi.org/10.9781/ijimai.2021.09.005.

[30] Akhtar, M. S., & Feng, T. (2022). Malware Analysis and Detection Using Machine Learning Algorithms . Symmetry, 14(11), 2304. School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China. https://doi.org/10.3390/sym14112304.

[31] Singh, H. K., Singh, J. P., & Tewari, A. S. (2022). Static Malware Analysis Using Machine and Deep Learning. In Proceedings of International Conference on Computing and Communication Networks (pp. 437-446). Springer, Singapore. DOI:10.1007/978-981-19-0604-6_41.

[32] Buradkar, M. U., Anajani, P. K., Reddaiah, B., Madduri, A., Thakkar, D., Bobade, S. D., &Subbulaks Tools and techniques for malware detection and analysis. arXiv preprint arXiv:2002.06819.hmi, T. (2022). Static Malware Analysis Using Optimal Machine Learning Algorithm for Malware Detection. Neuro Quantology, 20(10), 4128-4141.

[33] Chowdhury, M., Rahman, A., & Islam, R. (2017, June). Malware analysis and detection using data mining and machine learning classification. In International conference on applications and techniques in cyber security and intelligence (pp. 266-274). Edizionidella Normale, Cham.DOI: 10.1007/978-3-319-67071-3_33.

[34] Karbab, E. B., & Debbabi, M. (2019). MalDy: Portable, data-driven malware detection using natural language processing and machine learning techniques on behavioral analysis reports . Digital Investigation , 28, S77-S87 DOI:10.1016/j.diin.2019.01.017.

[35] Zhang, J. (2018). MLPdf: an effective machine learning based approach for PDF malware detection. arXiv preprint arXiv:1808.06991., https://doi.org/10.48550/arXiv.1808.0699.

[36] Abdelsalam, M., Krishnan, R., Huang, Y., &Sandhu, R. (2018, July). Malware detection in cloud infrastructures using convolutional neural networks. In 2018 IEEE 11th International conference on cloud computing (CLOUD) (pp. 162-169). IEEE.DOI: 10.1109/CLOUD.2018.00028.

[37] Kakavand, M., Dabbagh, M., &Dehghantanha, A. (2018, November). Application of machine learning algorithms for android malware detection. In Proceedings of the 2018 International Conference on Computational Intelligence and Intelligent Systems (pp. 32-36).https://doi.org/10.1145/3293475.3293489.

[38] Azmoodeh, A., Dehghantanha, A., &Choo, K. K. R. (2018). Robust malware detection for internet of (battlefield) things devices using deep eigen space learning. IEEE transactions on sustainable computing , 4(1), 88-95.

[39] Shah, S. S. H., Jamil, N., & Khan, A. U. R. (2022). Memory Visualization-Based Malware Detection Technique. Sensors, 22(19), 7611, https://doi.org/10.3390/s22197611.

[40] A.Moser, C. Kruegel, and E. Kirda, “Limits of static analysis for malware detection,” in Proceedings - Annual Computer Security Applications Conference, ACSAC, 2007, pp. 421–430.

[41] J. Okolica and G. Peterson, “A compiled memory analysis tool,” in IFIP Advances in Information and

Communication Technology, 2010, vol. 337 AICT, pp. 195–204.https://hal.inria.fr/hal-01060619.

[42] Atluri, A. C., & Tran, V. (2017). Botnets threat analysis and detection . Information Security Practices : Emerging Threats and Perspectives , 7-28.DOI: 10.1007/978-3-319-48947-6_2.

[43] Endgame, “Ember,” 2018. [Online]. Available: https://www.endgame.com/blog/technical-blog/introducing-emberopen-source-classifier-and-dataset.

Citation of this Article:

Naz faith M Jameel, Muna M. T. Jawhar, “A Survey on Malware Attacks Analysis and Detected” Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 5, pp 32-40, May 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.705005

****