Study of Malware and Malware Detection Techniques

Manoj Kumar Chauhan
Assistant Professor, RBS Management Technical Campus , Agra-282002
Email Id- manojchauhan72@gmail.com

Dr. Pankaj Saxena[*]
Professor , RBS Management Technical Campus , Agra-282002
Email Id- pankajrbsmtc@gmail.com

[*] Corresponding Author

***Abstract*— All over the world, thousands of people created malware. Because of this we are experiencing an ever growing security problem with malicious code and consequently seeing the malware analysis research as a new scientific field.**

Malware has been used by cybercriminals as weapons in compromising security of system. Everyday new variants of malware are created by malware authors to evade detection by anti-malware engines. It became a serious threat for the information security from the past decade. Despite so many corrective measures, the threat is increasing in an unprecedented rate which is a motivation for the researchers to work on it.

Today's it is clear that how difficult for anti-malware companies to tackle attacks and also releases there new updates within a limited time to prevent their customers from malware infection. It must we clear Malware protection is a very important task of anti-malware companies as we know a huge data and money can be lost just because of one single attack.

In this paper, researchers first provide a brief overview on malware and needs on malware detection. This study presents a systematic and detailed survey of the malware detection mechanisms using data mining techniques. In this study, besides presenting the full and comprehensive literature on malware definitions, types, various detection techniques and methods, our objective is to give an idea about various techniques that are used for representing the malware samples.

The objective of the study is to provide a reference, which could be suitable for further studies to develop malware detection system.

***Index Terms*—Malware, Virus , Machine Learning .**

I. INTRODUCTION

Viruses were the first instances of malware. Virus-like program appeared on microcomputers in the 1980s. The first viruses on microcomputers were written on the Apple-II, circa 1982. Fred Cohen,s initial research with computer viruses in 1984 concluded that the computer virus problem is ultimately an integrity problem [1] .

What is Malware

Malicious software, commonly known as malware, is any software used to

-disturb operation of computer ,

- Unauthorized access of sensitive information , or

-gain access to private computer systems.

Any software that does something that causes harm to a user, computer, or network can be considered malware including viruses , Trojan horses , worms, rootkits, scareware , and spyware [3] .

McGraw and Morrisett [4] define malicious code as "any code added, changed, for removed from a software system in order to intentionally cause harm or subvert the intended function of the system."

Malware (short for malicious software), is a generic term widely used to denote all different types of unwanted software programs.

What is Malware Analysis

Malware analysis is the art of dissecting malware to understand how it works, how to identify it, and how to defeat or eliminate it [5] .

Malware analysis is a very important process that will determine the purpose and functionality of a given malware sample (such as a virus, worm, or Trojan horse)

When analyzing suspected malware, our goal will typically be to determine exactly what a particular suspect binary can do, how to detect in our system, and how to measure and contain its damage.

Malware analysis has some common patterns that can be learned easily . There are several techniques that researcher use to reach their ultimate goal .

Malicious Code Environment

When performing malware analysis, we should know the kinds of things that malware usually does. We need to be able to find the environment in which they operate.

The figure 1 shows environment that each layer might create new dependences (such as Vulnerability) for malicious code.

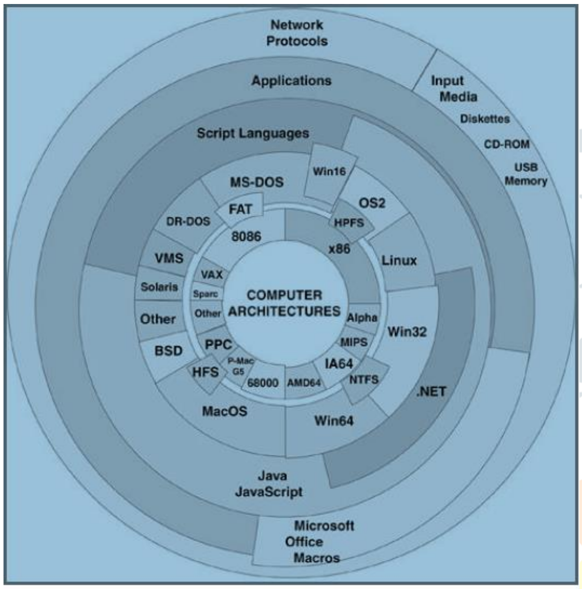


Figure 1 [Peter Szor 2005]

Type of Malware [Michael Sikorski and Andrew Honig , 2012**]**

Malware can be classified into different categories based on how they try to infect or based on their behavior's. Try to define a unified nomenclature for malware is almost as old as computer viruses themselves. CARO(Computer Antivirus researchers Organization) designed a computer virus naming scheme for use in Antivirus products .

Here are the some categories that most malware falls into:

Backdoor : Malicious code installs itself onto a computer. It allows someone that is aware of it to gain access without going through the usual security access procedures. Backdoor attacker connects to the computer with little or no authentication and executes commands on that system.

Botnet : It is similar to a backdoor because it also allows the attacker to access the system, and all computers infected with the same botnet controlled by the same instructions from a single command-and-control server.

Downloader : It is generally used for download other malicious code. Downloader program are installed by attackers when they want to gain access to a system. It will download and install additional malicious code for gain access to a system.

Information-stealing malware: Malicious code that gathers information from a victim's computer and sends it to the attacker. Examples include keyloggers , sniffers and password hash grabbers. This type of malware is generally used to gain access of online accounts such as bank or email.

Launcher : Malicious code that is used for launch other malicious programs is known as launcher. Launchers launch other malicious programs in order to ensure stealth or access to a system.

Rootkit: Malware that is designed to conceal the existence of other code. Rootkits paired with other malware, such as a backdoor, to gain remote access to the attacker and make the code difficult for the victim to detect.

Scareware: software designed to frighten an infected user into buying something. It informs infected users that there is virus on their system and that the only way to get rid of it is to buy their "software," but in reality, this software does nothing more than remove the scareware.

Spam-sending malware: Malware that uses infects machine to send spam. This malware is used to sell spam-sending services.

Worm or virus : Malicious code that can copy itself. A computer virus is a malware program that, when executed, replicates by inserting copies of itself into other computer programs, data files, or the boot sector of the hard drive

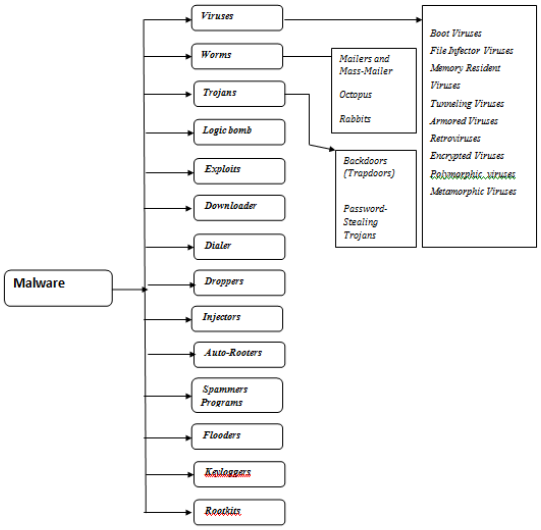


Figure 2 [malware taxonomy]

In this study, our aim to providing various detection techniques and methods that are used for representing the malware samples, as we will conduct a survey on some representations that are based on the major malware detection techniques.

II. ANALYSIS OF MALWARE DETECTION

Malware analysis is like a cat and mouse game.

Malware detectors are used to detect malwares and antivirus scanners are one of the way to detect some of them but with progression of malware development techniques, malware detectors use a number of techniques to avoid the disastrous effects of these software. Due to the limitation of the existing malware detection techniques, the machine learning and data mining methods are combined with existing detection methods to add the efficiency in the detection process

Two fundamental approaches of malware analysis are: static and dynamic.

Static analysis :

Static analysis examines the malware without running it. Static analysis describes the process of analyzing the code or structure of a malicious code to determine its function. The program itself is not run at this time.

In Static analysis syntax or structural properties of the program (static)/process (dynamic) used under inspection (PUI) to determine its maliciousness. [Nwokedi Idika , Aditya P. Mathur ,2007]

Advanced static analysis involves of reverse-engineering the malware's internals by loading the executable into a disassembler and monitoring of the program instructions in order to discover what the program does.

Disassemble/Debugger tools like IDA Pro and OllyDbg displays the malware's code as Intel ×86 assembly instructions, which provide a lot of insight into what the malware is doing and provide patterns to identify the attackers.

The detection pattern can be extracted in static analysis like string signature, Windows API calls, opcode (operation codes) frequency and byte sequence n-grams[6]

Dynamic analysis:

Dynamic analysis is an efficient way to identify unknown malware. Dynamic analysis techniques involve running the malware and observing its behavior on the environment in order to identify the infection, produce effective signatures, or both.

In this analysis, suspicious files are executed and monitored in a controlled environment like virtual machine, simulator, emulator, sandbox etc. for analyzing the behavior of a malicious code .

Dynamic analysis can identify known and unknown malware. Dynamic analysis techniques are the second step in the malware analysis process. Dynamic analysis is also an efficient way to detect malware functionality.

Techniques that can be applied to perform dynamic analysis include function parameter analysis, information flow tracking, function call monitoring, instruction traces and auto start extensibility points etc. [7].

Reviewing the various surveyed papers, API and system calls are largely employed in malware dynamic analysis as well as file system and Windows registry.

Hybrid Analysis:

Both analyses have their own advantages and limitations. Static analysis is fast and safer compared to dynamic analysis. On the other hand, dynamic analysis is reliable and can beats obfuscation techniques. However, malware evade it by using obfuscation techniques.

Hybrid analysis is such an approach that combines the respective advantages of both static and dynamic analysis. For example, the packed malware can first go through a dynamic analyzer, where the hidden-code bodies of a packed malware instance are extracted by comparing the runtime execution of the malware instance with its static code model. When the hidden-code are uncovered, a static analyzer can continue the analysis of the malware program.

Hybrid analysis collect malware information from static analysis and dynamic analysis. Security researchers are using hybrid analysis to gain the benefits of both static and dynamic analyses.

Advantage and disadvantages of static and dynamic analysis

Analysis type	Advantage	Disadvantage
Static analysis	Fast and safe but low level of false positives Good at analyzing multipath malware	Difficulty analyzing unknown malware
Dynamic analysis	Good at detecting unknown malware	Slow and unsafe
Hybrid analysis	Combines aspects of both static and dynamic analysis	Difficulty analyzing multipath malware

Malware Detection Techniques

Techniques used for detecting malware can be broadly categorized into two categories: anomaly-based detection and signature-based detection.

Figure 3 shows the relationship between the various types of malware detection techniques. Each of the malware detection techniques can employ one of three different approaches: static, dynamic, or hybrid

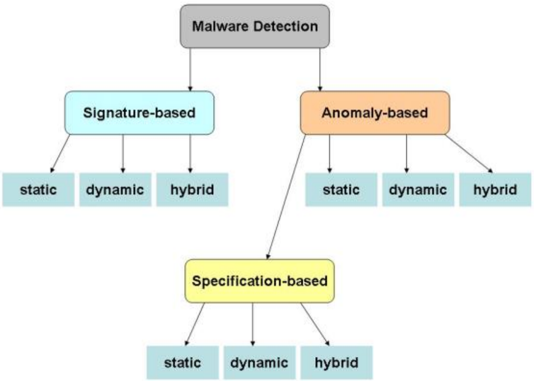


Figure 3 **[Elhadi et al, 2015]**

Signature-based detection

Signature-based detection uses its signature (characterization) of what is known to be malicious for deciding the maliciousness of a program under inspection. This characterization of the malicious behavior is the key to a signature-based detection method's effectiveness.

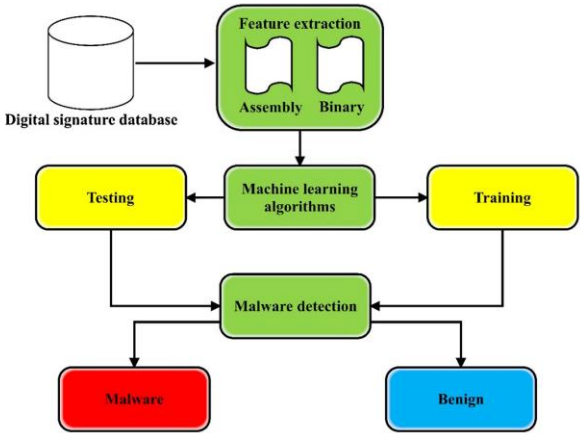
After identifying the malicious code, the identified signature is added to the existing database as the recognized malware. The database include huge number of the various signatures that classify malicious objects. Various qualities of signature-based malware are fast identification, easy to run, and broadly accessible.

Most of the available antivirus software use signature based approach. In this approach unique signature extracts from captured malware file and this signature use to detect similar malware

Malware writers have created another challenge for signature-based approach by using obfuscation techniques. This obfuscation technique includes register reassignment, instruction substitution, dead code insertion, and code manipulation [8].

In signature-based malware detection, two main methods use for applying malware detection approach in machine learning methods are assembly features and binary features.

Figure 4 illustrates a standard signature-based malware detection framework using data mining approaches. [9]



Anomaly-based detection

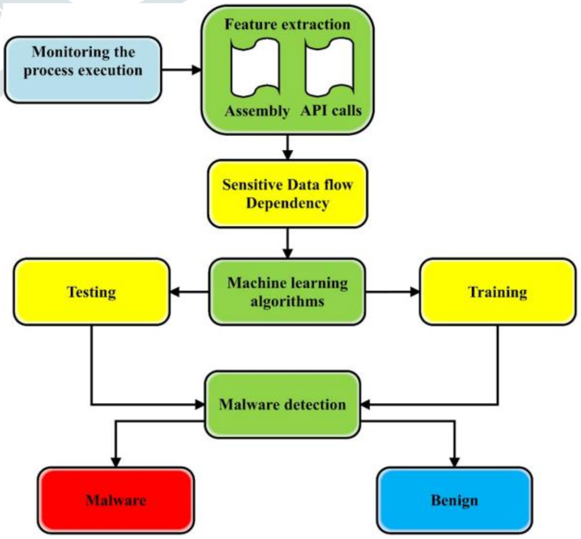
Anomaly -based is also known as heuristic or behavior based detection an anomaly-based detection technique uses its knowledge of what constitutes normal behavior to decide the maliciousness of a program under inspection.

In anomaly-based detection, the activities performed by malicious code during runtime are analyzed in a training (learning) phase. After training phase, the program file is labeled as malicious or legitimate file during a testing (monitoring) phase based on extracted pattern.

Behavior-based approach is used to detect both unknown malware and malware that uses obfuscation techniques. However, the major drawbacks of this approach are a large false positive rate (FP) and excessive monitoring time

Heuristic-based commonly depends on data mining techniques in order to understand the behaviors of running files, such techniques include Support Vector Machine, Naïve Bayes, Decision Tree and Random Forest.

Figure 5 depicts a standard behavior-based malware detection approach using data mining algorithms.


Monitoring the process execution
Feature extraction
Assembly
API calls
Sensitive Data flow Dependency
Testing
Machine learning algorithms
Training
Malware detection
Malware
Benign

III LITERATURE REVIEW

Researcher reviewed relevant surveys on malware and summarized their contributions in Table .
We categorized the surveyed papers into anomaly-based detection and signature-based detection approaches.

Detection Approaches	**Malware Study**	Classification approach	analysis method	Dataset used	Data set qty	Accuracy %
signature-based	API malware detection [17]	Naive Bayes and Decision Tree—SVM	Dynamic	Google play store	7000	95
signature-based	Polymorphic Malware Detection [18]	K-means	Dynamic	ClamAV, VirusTotal	2876	**99**
signature-based	N-grams malware detection [19]	SVM	Dynamic	Google play store	658	97
signature-based	Opcode sequences [20]	K-nearest neighbors and SVM	Hybrid	VxHeavens website	2000	92.9
signature-based	Signature and Heuristic-based Malware Detection [21]	SVM, J48, KNN, Decision tree and Random tree	Hybrid	M0DROID website	500	99.81
signature-based	Hybrid pattern based text mining approach [22]	ANN, malicious sequential pattern based malware detection	Hybrid	Viruses hair and Virus-Total websites	8000	98.89
behavior-based	Hybrid analysis malware [23]	Bayesian network, Naive Bayes, Lazy K-Stare	Hybrid	Selected randomly from malware repository of APA, the security research laboratory at Shiraz University	3000	95.27
behavior-based	Behavioral Malware [24]	Regression, SVM, J48	Dynamic	Web data commons library in VirusSign and VXHeaven	7000	98.3
behavior-based	Malicious code based on API [25]	Decision tree, SVM and random forest	Dynamic	API hooking library in VirusSign	2000	96.89
behavior-based	Hybrid analysis malware [26]	RF, SVM	Hybrid		1368	98.7

IV. CONCLUSION

In this literature Researchers have presented a series of techniques and topics within the domain of Malware detection. In this study, researchers surveyed the different types of malware and malware detection system. Researchers have reviewed certain malware detection techniques such as techniques based on machine learning, data mining and string representation. Researchers have also pointed out the various methods that are used in malware analysis whether been static, dynamic or hybrid.

Researchers have discussed the use of some tools that could be used in the representation of the malware sampled collected. Researchers also presented findings with respect to the various malware detection representation techniques and corresponding methods.

The objective of the study is to provide a reference, which could be suitable for further studies to develop malware detection system.

V. JUSTIFICATION

From the review, it may be that the methodologies previous studies employed did not adequately explain the phenomenon. A universal metric for malware detection ability needs to be developed.

An overview on different methods that were proposed for malware detection is given still has a non-zero false positive rate. Malware detection using machine learning will not replace the standard detection methods used by anti-virus vendors, but will come as an addition to them.

Hence the proposed methodologies are not adequate due to complex nature of malware.

APPENDIX

Tools for malware Analysis

VirusTotal (http://www.virustotal.com) : – VirusTotal generates a report that provides the total number of engines that marked the file as malicious , the malware name and additional information about the malware.

BinDiff - BinDiff is a powerful binary comparison plug-in for IDA Pro that allows you to quickly compare malware variants. BinDiff lets you pinpoint new functions in a given malware variant and tells you if any functions are similar or missing. If the functions are similar, BinDiff indicates how similar they are and compares the two

Capture BAT -Capture BAT is a dynamic analysis tool used to monitor malware as it is running. Capture BAT will monitor the filesystem, registry, and process activity.

IDA Pro -IDA Pro is the most widely used disassembler for malware analysis.

Netcat -Netcat, known as the "TCP/IP Swiss Army knife," can be used to monitor or start inbound and outbound connections. Netcat is most useful during dynamic analysis for listening on ports that you know the malware connects to, because Netcat prints all the data it receives to the screen via standard output.

OllyDbg -OllyDbg is one of the most widely used debuggers for malware analysis.

PE Explorer -PE Explorer is a useful tool for viewing the PE header, sections, and import/export tables. It is more powerful than PEview because it allows you to edit structures. PE Explorer contains static unpackers for UPX-, Upack-, and NsPack-compressed files.

Python -The Python programming language allows you quickly code tasks when performing malware analysis. Throughout the book and labs, we use Python. IDA Pro and Immunity Debugger have built-in Python interpreters, allowing you to quickly automate tasks or change the interface. We recommend learning Python and installing it on your analysis machine. Download Python for free from http://www.python.org/.

Snort -Snort is the most popular open source network intrusion detection system (IDS).

TCPView - TCPView is a tool for graphically displaying detailed listings of all TCP and UDP endpoints on your system. This tool is useful in malware analysis because it allows you to see which process owns a given endpoint

WinDbg -WinDbg is the most popular all-around debugger, distributed freely by Microsoft. You can use it to debug user-mode, kernel-mode, x86, and x64 malware. WinDbg lacks OllyDbg's robust GUI, providing a commandline interface instead.

Wireshark -Wireshark is an open source network packet analyzer and useful tool for dynamic analysis. You can use it to capture network traffic generated by malware and to analyze many different protocols. Wireshark is the most popular freely available tool for packet capturing and has an easy-to-use GUI. We discuss Wireshark usage in Chapter 3. You can download Wireshark from http://www.wireshark.org/.

VMware Workstation -VMware Workstation is a popular desktop virtualization product. There are many alternatives to VMware, but we use it in this book due to its popularity. Chapter 2 highlights many VMware features, such as virtual networking, snapshotting (which allows you to save the current state of a virtual machine), and cloning an existing virtual machine. You can purchase VMware Workstation from http://www.vmware.com/ or download VMware Player (with limited functionality) for free from the same site.

Sandboxes: The Quick-and-Dirty Approach- A sandbox is a security mechanism for running untrusted programs in a safe environment without fear of harming "real" systems. Sandboxes comprise virtualized environments that often simulate network services in some fashion to ensure that the software or malware being tested will function normally. Many malware sandboxes—such as Norman SandBox, GFI Sandbox, Anubis, Joe Sandbox, ThreatExpert, BitBlaze, and Comodo Instant Malware Analysis— will analyze malware for free

ACKNOWLEDGMENT

We would like to express our deep gratitude Professor Dr. Sandeep Kumar Jain, Department of Computer Science , DR. Bhim Rao Ambedkar University , Agra , and Dr. Ashu Sharma , MALWARE ANALYST , Mindtree, Hyderabad , for their patient guidance and enthusiastic encouragement of this research work.

REFERENCES

1. Peter Szor , "The Art of Computer Virus Research and Defense" Addison Wesley Professional ,Feb.03.2005

2 YANFANG YE ,TAO LI, DONALD ADJEROH,S. SITHARAMA IYENGAR, "A Survey on Malware Detection Using Data Mining Techniques" ACM Computing Surveys, Vol. 50, No. 3, Article 41, Publication date: June 2017

3 Nwokedi Idika , Aditya P. Mathur "A Survey of Malware Detection Techniques" , Purdue University, West Lafayette , 2007

4. G. McGraw and G. Morrisett. Attacking malicious code: A report to the infosec research council. IEEE Software, 17(5):33–44, 2000.

5 . Michael Sikorski and Andrew Honig "PRACTICAL MALWARE ANALYSIS" , No Starch Press, Inc. :2012

6 Rami Sihwail, Khairuddin Omar, K. A. Z. Ariffin, "A Survey on Malware Analysis Techniques: Static, Dynamic, Hybrid and Memory Analysis" , IJASEIT : Vol.8 (2018) No. 4-2

7. Ekta Gandotra, Divya Bansal , Sanjeev Sofat "Malware Analysis and Classification: A Survey" Journal of Information Security , 2014,5,56-64

8 I. You and K. Yim, "Malware obfuscation techniques: A brief survey," in Proceedings - 2010 International Conference on Broadband, Wireless Computing Communication and Applications, BWCCA 2010, 2010, pp. 297–300.

9. Souri and Hosseini "A state‑of‑the‑art survey of malware detection approaches using data mining techniques" , *Hum. Cent. Comput. Inf. Sci. (2018) 8:3*

10 Smita Ranveer , Swapnaja , " Camparative Analysis of Feture Extraction Methods of Malware detection" IJCA : Vol120-No5 June 2015

11Bat-Erdene M, Park H, Li H, Lee H, Choi MS (2017) Entropy analysis to classify unknown packing algorithms for malware detection. Int J Inf Secur 16(3):227–248.

12 Hellal A, Romdhane LB (2016) Minimal contrast frequent pattern mining for malware detection. Comput Secur 62:19–32.

13Wang P, Wang Y-S (2015) Malware behavioural detection and vaccine development by using a support vector model classifier. J Comput Syst Sci 81:1012–1026

14 P. V. Shijo and A. Salim, "Integrated static and dynamic analysis for malware detection," in Procedia Computer Science, 2015, vol. 46, pp. 804–811.

15 Eskandari M, Khorshidpour Z, Hashemi S (2013) HDM-Analyser: a hybrid analysis approach based on data mining techniques for malware detection. J Comput Virol Hacking Tech 9:77–93.

16. Galal HS, Mahdy YB, Atiea MA (2016) Behavior-based features model for malware detection. J Comput Virol Hacking Tech 12:59–67.

17. Fan CI, Hsiao HW, Chou CH, Tseng YF (2015) Malware detection systems based on API log data mining. In: 2015 IEEE 39th annual computer software and applications conference, pp 255–260

18. Fraley JB, Figueroa M (2016) Polymorphic malware detection using topological feature extraction with data mining. In: SoutheastCon 2016, pp 1–7

19. Boujnouni ME, Jedra M, Zahid N (2015) New malware detection framework based on N-grams and support vector domain description. In: 2015 11th international conference on information assurance and security (IAS), pp 123–128

20. Santos I, Brezo F, Ugarte-Pedrero X, Bringas PG (2013) Opcode sequences as representation of executables for datamining- based unknown malware detection. Inf Sci 231:64–82.

21. Rehman Z-U, Khan SN, Muhammad K, Lee JW, Lv Z, Baik SW, Shah PA, Awan K, Mehmood I (2017) Machine learningassisted signature and heuristic-based detection of malwares in Android devices. Comput Electr Eng.

22. Malhotra A, Bajaj K (2016) A hybrid pattern based text mining approach for malware detection using DBScan. CSI Trans ICT 4:141–149.

23. Eskandari M, Khorshidpour Z, Hashemi S (2013) HDM-Analyser: a hybrid analysis approach based on data mining techniques for malware detection. J Comput Virol Hacking Tech 9:77–93.

24. Norouzi M, Souri A, Samad Zamini M (2016) A data mining classification approach for behavioral malware detection.J Comput Netw Commun 2016:9.

25. Galal HS, Mahdy YB, Atiea MA (2016) Behavior-based features model for malware detection. J Comput Virol Hacking Tech 12:59–67.

26. Wang P, Wang Y-S (2015) Malware behavioural detection and vaccine development by using a support vector model classifier. J Comput Syst Sci 81:1012–1026.