International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 3 | ISSN : 2456-3307
DOI : 10.32628/CSEIT195379

# Malware attack and Malware Analysis : A Research

**Soumen Chakraborty**
Department of Information Technology, MCKV Institute of Engineering, MAKAUT, West Bengal, India
Email : csoumen88@gmail.com

**Abstract**:

Malware analysis is the manner of performing evaluation of the malware and knowledge its moves and conduct. It is of two types- static and dynamic evaluation. Static analysis is carried out by staring at the supply code of the malware and drawing conclusions primarily based on it. Dynamic analysis is the analysis achieved through executing the piece of code and noting its actions. Malware evaluation is an essential and relevant undertaking, for the advanced forms of malware these days are often not even detectable through generally available anti-virus software program. In the present paper, the authors have made a scientific have a look at on one of a kind problems in malware and analysis of malware. One of the most full-size threats to cyber security in nowadays's world of limitless Internet get right of entry to is malware. In latest times, the malware being designed are polymorphic and metamorphic, with the ability to transform their code and to cover quietly within the structures of the unsuspecting customers.

**Keywords** : Malware, Viruses, Static Analysis, Dynamic Analysis, Classification, Security.

## I. INTRODUCTION

With the rise of broadband Internet access, simple computer viruses, which were first of all written as mere pranks, have given way to malicious software program which, in turn, has taken the form of a massive epidemic. In recent times, malware has been written and designed maintaining earnings in thoughts. Since 2003, the general public of worms and viruses, which have grow to be enormous, had been written so that it will perform illicit activities on users' computer systems. McAfee [2] catalogs over a hundred,000 new malware samples each day means approximately 69 new threats each minute or about one risk according to 2nd. Preliminary results from Symantec Malware, abbreviated from malicious software program, is any form of software that "deliberately fulfills the harmful reason of an attacker." Aycock (2006) defined malware as "software program whose rationale is malicious, or whose effect is malicious". [26] It can have any quantity of vindictive functions which include disrupting the normal computer operations, accumulating sensitive and exclusive information from an unwitting user, gaining access to personal pc networks and showing unwanted advertisements or unsolicited mail. The time period 'malware' was coined by Yisrael Radai in 1990, however, preceding that these forms of software had been prevalently referred to as pc viruses [22]. Malware is a preferred time period, used to refer to a ramification of intrusive and dangerous software program, together with laptop viruses, Trojan horse, worms, spyware, and adware. [25] Malware may be stealthy portions of software, quietly stealing non-public information or they will reason direct damage like sabotage and even extort price from the unsuspecting victim.

## II. METHODS AND MATERIAL

DIFFERENT TYPES OF MALWARE

The following paragraphs are totally here for the reason of a short advent to the extraordinary terminology related to malware. The various styles of malicious software are very shortly discussed. It is to be referred to that the instructions which can be going to be referred to here are not together different and a piece of malware can be displaying traits and traits of multiple such classes. In depth discussions of malicious code can be determined in Szor (2005).

Viruses: “A laptop program typically hidden inside every other reputedly harmless program that produces copies of itself and inserts them into different applications or documents, and that generally plays a malicious action (including destroying information)”.[6]

Worms: Spafford (1989) defines a trojan horse as “a program that could run independently and might propagate a totally working version of itself to different machines.” This type of dangerous code is predominantly established in networks along with the Internet.

Trojan horses: The time period is derived from the Ancient Greek story of the timber horse that helped Greek troops invade the metropolis of Troy by way of stealth and deception. Like the name, Trojan horses are basically programs that misrepresent themselves as beneficial software, along with plug-ins or downloadable video games, while secretly perform malicious sports in the heritage.

Spyware: Spyware refers to software that get entry to confidential information from the person and skip it directly to some other entity with out informing the consumer of this motion. Thus it takes manipulate over the consumer’s machine with out asking for his/her consent regarding the problem.

[3] posted in 2008 counseled that "the discharge fee of malicious code and other undesirable applications can be exceeding that of legitimate software program packages." F-Secure [4] mentioned how the amount of malware produced in 2007 turned into as a lot as inside the preceding twenty years. This, but, has visible a rise in tremendously specific and superior tools to counter such malware. In a reason-impact scenario, this, in turn, has led to the brand new generation cyber security threats and attacks to be extra centered, unknown, stealthy, personalized and 0 day. This is in stark assessment to the traditional malwares which had been huge, acknowledged, open and one

protections. After set up, they call their command and manage servers for further instructions, which will be to thieve facts, infect different machines, and permit reconnaissance. [1]

method is by enforcing stolen certificates to disable anti-virus protection. This is executed by means of positive spyware; technical remedies are available to address such spyware.

## III. STATIC MALWARE ANALYSIS

Analyzing malicious software with out executing it, but with the aid of merely inspecting this system is referred to as static or code evaluation. It is typically performed with the aid of taking each a part of the binary file and analyzing every aspect very well with out absolutely executing it. The different detection patterns used in case of static analysis [1] consist of string signature, byte-collection n-grams, syntactic library call, manipulate go with the flow graph and

opcode (operational code) frequency distribution and so forth.

Basic static evaluation entails the subsequent steps [7]. First, the suspicious executable is run via specific anti-virus software and answers to hit upon commonplace malware. The programs detected by the anti-virus are noted down as is the facts of what the anti-virus detects the malware as. Once the scanning of malware using anti-virus is completed, the second one step of simple static evaluation is finished. The malware is opened up in a hex editor to see what type of software program it's far and whether or not it's far using a few kind of packer utility (together with Ultimate Packer for Executables or UPX). A packer including UPX makes use of a simple records compression set of rules which permits for decompression of the record in a few hundred bytes of code. On unpacking the malware, other gear can be run against it. A replica of the malware need to be made, for if the unpacking is achieved incorrectly, the malware can be not able to characteristic. Microsoft has a beneficial utility called Strings, which tests for ASCII, Unicode or both characters in a document. Strings searches the executable even as ignoring context and formatting and for that reason allows in finding out protocols, ports, IP addresses and different such facts that offer vital clues approximately the functionality of the malware. Strings searches for a three-letter or greater collection of ASCII and Unicode characters, accompanied through a string termination character. Once the Strings seek is achieved, it's time to disassemble the malware. The executables are opposite compiled the use of debuggers or disassemblers. The disassembled, decrypted malware code affords excellent insight into the working of the software program. Current systems to locate malicious code (most prominently, virus scanners) are in large part based totally on syntactic signatures. That is, those structures are geared up with a database of regular expressions that designate byte or instruction sequences which are considered malicious. A software is declared malware while one of the signatures is diagnosed within the software's code. [5] Malware authors use an expansion of techniques to reveal the vulnerabilities in extraordinary internet offerings, working systems, browsers, or in variations of browser plug-ins and exploit those weaknesses. Some commonly used techniques carried out encompass useless code insertion, sign up reassignment, subroutine reordering, instruction substitution, code transposition and code integration to stay away from detection with the aid of traditional defenses like firewalls, antivirus and gateways which generally use signature primarily based strategies and are not able to come across the formerly unseen malicious executables. Syntactic signatures forget about the semantics of instructions and consequently the syntactic homes of code are in large part unnoticed, resulting in such malware being resilient against common defence mechanisms. Commercial antivirus companies aren't able to provide on the spot safety for zero day malwares as they need to research these to create their signatures. [1]

Thus with the usual signature based defence mechanisms failing inside the face of superior modern-day malware, the want for malware evaluation is abundantly clear. Malware evaluation may be of types – static and dynamic. Static malware evaluation refers to studying malicious software program without executing, by using merely observing and inspecting the strategies of the malware. Dynamic malware evaluation includes analyzing a given application whilst it is being done.

Backdoor: The method of evading ordinary authentication procedures, in particular over connections including the Internet, is referred to as backdoor. One or extra backdoors can be established into a machine without the user's understanding to make the gadget liable to outdoor assaults.

Rootkits: It is critical for a chunk of malicious software to stay concealed once it has been hooked up in a device. This is to avoid detection, which defeats the cause of malware. Rootkits are software program packages that useful resource inside the hiding of malware in the gadget with the aid of enhancing the user's operating system making sure that the software stays concealed.

In recent times, a widespread portion of malware makes use of a number of strategies to avoid detection. The most ordinary evasion method implemented in in recent times is by means of fingerprinting the environment as quickly as the malware is carried out. Another common P a g e

Often malware writers use diverse obfuscation techniques to prevent the above reverse engineering and as a result make static evaluation unpredictable, unreliable and high-priced. Binary obfuscation turns malware binaries into self-compressed and uniquely based binary documents. Furthermore, vital facts about the malware is misplaced at the same time as trying to work with such binary executables and subsequently the technique of malware analysis is made greater hard. [1] Moser et al. Discussed the drawbacks and shortcomings of static analysis. In their paper, they introduced a scheme primarily based on code obfuscation that proven how static evaluation on my own isn't sufficient to detect malware. They additionally proposed that dynamic malware analysis at the side of static analysis is important to make the system much less susceptible to code obfuscation strategies and approach.

## IV. DYNAMIC MALWARE ANALYSIS

Once static malware analysis is completed, it is time to transport directly to dynamic evaluation. As referred to earlier than, for proper analysis of malware, dynamic analysis should complement the technique of static evaluation. In dynamic analysis, the procedure is completed and the system is discovered whilst the modifications that arise are referred to. It is to be kept in mind that the malware needs to be performed in a safe environment, ideally in a digital gadget. It is likewise clever to take a picture of the digital gadget earlier than the malware binaries are achieved in an effort to ensure that the safe nation can be lower back to and the right changes may be noted. Before appearing dynamic analysis, it's miles to be ensured that the digital gadget networking is not connected to another networks other than the host, for there may be a superb threat of introducing the malware to other networks if this step is not accomplished. Other examples of managed environments where the malware may be completed are simulators, emulators and sandboxes. Also, earlier than executing the malware, sure monitoring equipment along with Process Monitor [8] and Capture BAT [9] (for document device and registry monitoring), Process Explorer [10] and Process Hackerreplace [11] (for procedure tracking), Wireshark [12] (for network tracking) and Regshot [13] (for machine alternate detection) are installed and activated. [1] The various techniques which can be employed to perform dynamic analysis are mentioned beforehand P a g e

Volume four, Issue 10, October 2016 pg. 22-30

Function Call Monitoring: Functions utilized in applications encompass codes that carry out a selected task, inclusive of sorting a fixed of information. Functions are useful in case of programming, for they help in code reusability and also make the code easier to preserve. One characteristic of features is that they provide an aspect of abstraction in carrying out the project. For

example, a sorting function would make certain that the number one difficulty is the suitable taken care of output, with out indulging within the details of the algorithm this is hired to attain the aforementioned result. It does not truely be counted within the large photo where the algorithm used is a merge sort or a bubble sort so long as the result reached by the characteristic is accurate. These abstractions assist in developing a summary of the conduct and sample of the program at the same time as analyzing code. One way to useful resource such analysis is to intercept the calls to such functions. The time period 'hooking' is used to describe this manner of intercepting function calls. The code is analyzed and alongside the desired functions, a 'hook' feature is invoked which helps to enforce the respective analysis functionality, together with analyzing input parameters or preserving information of invocations to log files. [14] System calls are utilized by consumer-mode software to request the working device to perform certain functions on its behalf. Usually, malware invokes gadget calls to interact with kernel space even as executing inside the person space. This makes the dynamic analysis of this interface quite thrilling.

Function Parameter Analysis: In static evaluation, function parameter analysis attempts to infer the sorts of the parameters or the set of their values in a static manner. However in dynamic evaluation, the real values of the parameters, which can be surpassed while a characteristic is called, are of concern. For instance, the go back fee of a CreateFile gadget name can be used later for a WriteFile call, and this correlation is of first-rate importance in dynamic malware analysis. [14]

Information Flow Tracking: The purpose of records float monitoring/evaluation is to problematic at the go with the flow of 'exciting' information throughout the machine, whilst this system manipulating it's miles being completed. Dynamic taint analysis helps in tracking the waft of data between the 'source' and the 'sink'. Any fee within the application that relies upon on computation the usage of statistics from a 'tainted' supply is referred to as 'tainted' records. Yin et al [15] proposed a method called whole-device fine-grained taint evaluation which uses a whole-system emulator to capture the intrinsic residences of a diffusion of malware and therefore presenting a great quantity of development to automatic malware detection and analysis. As cited in their paper, their proposed gadget showed how the method helped in detecting a massive range of malware of different training consisting of backdoors.

Instruction tracing: Instruction tracers, or tracers, are referred to for recording every unmarried practise and related nation while executing a chunk of code. This tracing is then analyzed by a hint analyzer for extraction of relevant data. Bangerter et al [16] delivered a brand new tracer called Helios which involved a lot of optimizations along with automatically skipping beside the point code components which are also computationally luxurious. The simple approach behind Helios is simple in that it interrupts the execution go with the flow whenever manage switch coaching (CTI) takes place, records the training among two CTIs, and then incorporates on with the execution from the vacation spot cope with of the CTI.

Apart from this, there are numerous different strategies like autostart extensibility points which are also part of dynamic malware evaluation. It ought to be stated that while dynamic analysis is some distance greater efficient than static malware analysis, it is also lots more time extensive and useful resource consuming, therefore main to expanded scaling problems.

## V. TOOLS USED FOR MALWARE ANALYSIS

Before discussing the one of a kind malware analysis equipment to be had popularly, some terms need to be clarified. Malware evaluation uses a device called sandbox typically, so as to run the unauthorized and in all likelihood dangerous piece of code or software with out harming the host gadget. In laptop safety, a sandbox refers to a precise, separate environment, analogous to a container, with strict regulations and permissions, where pc code can run without being able to inflict any harm or purpose infection. Anything outside the sandbox is beyond the reach of the suspicious computer code. It is to be stated that a sandbox and a digital gadget are not the equal element. When a application runs in a sandbox, it has the permission to execute as although it was not in a sandbox. Any modifications tried through the utility are lost whilst the application stops going for walks. In evaluation, whatever modified or created through the utility is permitted to remain in a virtual gadget, and all movements stay inside it.

 ISSN: 2321-7782 (Online) Impact Factor: 6.047 25 4, Issue 10, October 2016 pg. 22-30

The foremost job of the sandbox is to enable "users to automate the pattern submission procedure; absolutely analyze any risk; and speedy act to protect touchy facts". [17]

Several on-line automated tools exist for the purpose of dynamic analysis of malware. Only some of them will be mentioned here.

Norman SandBox [18]: "The Norman SandBox Analyzer is a utility intended to automate, simplify, and speed up the facts amassing system while studying malware." [19] The sandbox affords a notably controlled environment and can hence be taken into consideration as a specific shape of virtualization. Sandboxes are therefore frequently used to test codes with malware, without inflicting harm to the host laptop. They characteristic with the aid of limiting the assets used by the execution of the piece of malicious code. One of the numerous benefits of the usage of Norman SandBox is that because the malware receives done in a simulated system, obfuscation can not avoid the manner of malware evaluation itself. This aids in the detection of viruses and worms spread over electronic mail or thru P2P networks. Alongside this, a widespread malware detection algorithm is likewise run in order to seize different types of malicious software.

Anubis: Anubis is developed by the International Secure Systems Lab and is able to studying both files and URLs. Unknown binaries are analysed in an emulated surroundings of a Windows XP operating system on this project. The evaluation is executed through monitoring the machine calls and Windows API features. Function parameters also are tracked and monitored in this malware evaluation assignment [20].

CWSandbox: CWSandbox is a tool for malware analysis that satisfies the three design situations of automation, effectiveness and correctness. Dynamic analysis of malware is achieved to achieve automation. The software or code is done in a simulated environment, a sandbox. Effectiveness is ensured with the aid of the usage of the method of API hooking. The calls to the Windows application programmers' interface (API) are despatched to the monitoring software for analysis before the actual API code is called. API hooking ensures that all nuances of malware conduct are referred to for which the API calls are hooked. Correctness of the tool is accomplished through implementing the technique of DLL code injection. Briefly, DLL code injection may be defined to allow API hooking to be applied in a reusable and modular way. It is worth of

point out that although this tool can touch upon the seen movements of the malware, it cannot describe how the malware turned into programmed. However, regardless of this disadvantage, the information amassed from executing malware using a CWSandbox is treasured and more regularly than no longer, sufficient to diagnose the risks related to the precise malware [21].

## III. RESULTS AND DISCUSSION

It is with alarm that experts are noting how malware appears to be getting increasingly more sophisticated with the aid of the day. A as a substitute bleak photograph is without difficulty painted, thinking about that compared to the advancements inside the generation behind malware, running systems and internet browsers are not released that fast.

One recent trend in the upward push of malware assaults is noticeable inside the subject of on the spot messaging. As the various systems begin permitting interaction between them, the quantity of malware attacks on them will growth. A similar pattern is cited amongst Massive Multiplayer Online Roleplaying Games (MMORPG), where malware authors take manipulate of the debts of the unsuspecting users and use them to their own malicious advantage. Top companies nowadays are typically involved approximately Trojan Horses. Using the help of cautiously positioned keyloggers or display-scraping software, cybercriminals have taken to attacking unique computer systems, assisting in their vindictive hobbies of industrial espionage or comparable financially motivated crimes. Such focused assaults regularly remain undetected, for popular software are not able to identifying them. Recent reports be aware many new and stepped forward assaults. There became a malware specially written to steal intellectual assets. What became anomalous approximately the malware became its capability to crawl thru specific file types (Excel, PDF, and so on), encrypt the intellectual records and ship the stolen facts to a faraway server. Another method used by cybercriminals nowadays is known as “hack-back”, as stated with the aid of Gunter Ollmann, vice-president of studies for Damballa. This feature detects if and while a researcher is reading the malware,

 ISSN: 2321-7782 (Online) Impact Factor: 6.047 26 four, Issue 10, October 2016 pg. 22-30

and right now compromises the researcher’s device. Similar steps are taken by numerous different botnet malware which spark off

DDoS – denial of carrier – attacks on researchers, if they come too close to the C&C (command-and-control) machine. The Conficker computer virus really blacklists users investigating the malware, who try to get admission to the botnet server.

Although statistics display that there are about 30% more CVEs (common vulnerabilities and exposures) in Microsoft Office in place of PDFs, it is seen that the number of assaults on PDFs hugely overshadows the quantity of attacks on MS Office. [23] Prior to October, 2007, PDF attacks were non-existent; but, at the quit of September that 12 months, CVE-2007-5020 turned into launched, exposing a major vulnerability in Adobe’s PDF software program. And accordingly started out the unfairness of protection assaults on PDFs over MS Office. Attackers use three foremost ways to compromise PDFs – mass mailing, power-by downloads and focused attacks. Mass mailing includes sending malicious PDFs via e mail and using social engineering so that you can tempt users to open the record. On the other hand, pressure-by downloads silently supply the PDFs to the unsuspecting customers’ machines after they visit malicious websites. Targeted assaults are like mass

mailings, except that the malicious PDFs aren't sent in bulk, however are sent to a particular man or woman or agency.

Malicious PDFs can be improved in three distinct approaches through the attackers. The first is it containing one precise exploit. The 2nd one is the possibility of a malicious PDF having several exploits. And the 0.33 approach of infection includes detecting the version of PDF software program installed on the pc this is focused. That manner handiest an appropriate take advantage of will be used. Lesser the range of exploits, greater is the risk of a a hit assault.

Example of a de-obfuscated malicious JavaScript code showing how the PDF software version is detected:

```
feature PDF(admwn.P collab)
var lv=Pdf1.GetVersions();
var fi=/EScript=([^,]+),/;
lv=lv.Healthy(fi)[1].Break up('.');
lv=parseInt(lv.Be part of(' '));
if(lv<=812)
SHOWPDF(collab);
else
SHOWPDF(admwnp);
```

This piece of code identifies the type of PDF software and can provide a malicious PDF for that reason.

When analysing malicious PDFs, we can commonly classify them into two categories: JavaScript based totally and Non JavaScript based totally. Because of its flexibility and ease of use, JavaScript is widely used in malicious PDF, exploiting a inclined JavaScript API and putting in the PDF reader program's memory with malicious code. Although the general public of malicious PDFs determined in the wild use JavaScript, different strategies also are observed. One usual method is to embed Flash objects in the PDF. In addition to diverse distribution techniques, attackers have additionally improvised extraordinary techniques to stay away from detection of the malicious content material. Some of the malicious PDFs comprise junk parts of code to throw off antivirus software. Others crash the PDF reader so one can supply the person the illusion that they're corrupted files, whilst silently wearing out the vicious sports. Furthermore, diverse elements of a PDF can be obfuscated, as a result not permitting detection. Obfuscation of PDF report format is tons easier compared to other record formats along with MS Word, etc. A quite simple obfuscation method with the aid of which a vulnerable API call string is damaged into smaller strings is demonstrated below.

```
Tryeval("thi"+"s.M"+"ed"+"ia"+".N"+"ew"+"Pl"+"ay"+
"er(n"+"ull)");

catch(e)
```

To avoid being infected, a person can carry out the subsequent precautions. Users can disable JavaScript guide where possible. They need to maintain up to date with all of the software patches available for the PDF reader software program. Antivirus and IPS definitions need to be updated, and eventually, users should always exercise caution at the same time as commencing PDFs from an untrustworthy supply.

Another principal exploitation fashion was spotted in Java as these days as 2012. [24] According to Oracle, 1.1 billion desktops run Java, and subsequently vulnerability in Java is sure to have a good sized impact on consumer safety. The vulnerability in query have been constant before the primary appearance of the malware. However a researcher disclosed details of the vulnerability on his web page, which turned into shortly accompanied through the monitor of the malware. Shortly afterwards, it

become cited that the exploitation of this vulnerability took over all different pre-existing Java vulnerabilities. This vulnerability is currently the number one vector for all pressure-with the aid of exploits. Categorizing beyond Java vulnerabilities, 4 classes are cited typical: type confusion, good judgment error, reminiscence corruption and argument injection. Type protection is a totally vital feature of Java safety. Type protection is ensuring that a variable with a positive records type isn't always treated as a distinctive facts kind in a application. On the failure of type protection, type confusion takes location. It is analogous to identification theft inside the real international. An example of a beyond kind confusion vulnerability in Java element is CVE-2012-0507: Atomic Reference Array kind confusion vulnerability. Logic errors can reside inside Java device code. CVE-2011-3544: Java Rhino Script Vulnerability is an instance of common sense errors wherein Security Manager is disabled. Memory corruption problems, even as now not a trend, have took place before, together with CVE-2010-0842: Sun Java Runtime Environment MixerSequencer. Argument injection may be very popular with Java plug-ins, as become mentioned in CVE-2010-0886: Java Deployment Toolkit Component. To analyse such Java vulnerabilities, specific static and dynamic research tools are required for Java binaries and the platform. CVE-2012-0507 is presently the maximum commonplace vulnerability for pressure-with the aid of exploits.

## IV.CONCLUSION AND FUTURE SCOPE

The thoughts of malware, the one of a kind types of malware and malware analysis were discussed in details here. It is inferred from the facts accumulated that dynamic evaluation is a higher approach of malware analysis than static evaluation. Although dynamic analysis has the plain flaw of studying simplest one execution of the malware, static evaluation is rather hard to do well, for in maximum instances, the source code isn't visible. And although the supply code turned into available, it can by no means be ensured that no changes have been done to the binary executables which remained undocumented through the supply. Hence, because of many such drawbacks, dynamic evaluation is favored over static malware analysis. On reading malware analysis equipment, it can be concluded that the sandbox surroundings is in particular conducive towards studying malware. Recent trends in malware attacks show extraordinarily superior strategies being implemented to secure sensitive statistics. It is with awesome alarm that experts are noting how malware assaults are being more and more state-of-the-art in a brief period of time, whereas working systems and different person software are not produced in such brief durations.

## V. REFERENCES

[1]. Gandotra, E., et al. (2014) Malware Analysis and Classification: A Survey. Journal of Information Security, five, 56-sixty four. Http://dx.Doi.Org/10.4236/jis.2014.52006

[2]. (2013) Infographic: The State of Malware. Http://www.Mcafee.Com/in/security-recognition/articles/nation-of-malware-2013.Aspx three"Symantec Internet Security Threat Report: Trends for July–December 2007 (Executive Summary)" (PDF). XIII. Symantec Corp. April 2008: 29. Retrieved 11 May2008.

[3]. "F-Secure Reports Amount of Malware Grew by using one hundred% during 2007" (Press launch). F-Secure Corporation. Four December 2007. Retrieved 11 December 2007

[4]. Andreas Moser, Christopher Kruegel, and Engin Kirda, Limits of Static Analysis for Malware Detection, Secure Systems Lab Technical University Vienna

[5]. “What are viruses, worms, and Trojan horses?". Indiana University. The Trustees of Indiana University. Retrieved 23 February 2015.

[6]. Http://resources.Infosecinstitute.Com/malware-analysis-basics-static-evaluation/eight(2014) Process Monitor. Http://technet.Microsoft.Com/en-us/sysinternals/bb896645.Aspx nineCapture BAT. Https://www.Honeynet.Org/node/315

[7]. (2014) Process Explorer. Http://technet.Microsoft.Com/en-us/sysinternals/bb896653.Aspx elevenProcess Hackerreplace. Http://processhacker.Sourceforge.Net/

[8]. Wireshark. Http://www.Wireshark.Org/

[9]. Regshot. Http://sourceforge.Net/tasks/regshot/

[10]. Egele, M., Scholte, T., Kirda, E. And Kruegel, C. (2012) A Survey on Automated Dynamic Malware-Analysis Techniques and Tools. Journal in ACM Computing Surveys, forty four, Article No. 6.

[11]. Whole-gadget Fine-grained Taint Analysis for Automatic Malware Detection and Analysis Heng Yin hyin@cs.Wm.Edu College of William and Mary Dawn Song dawnsong@cmu.Edu Carnegie Mellon University http://bitblaze.Cs.Berkeley.Edu/papers/malware-hit upon.Pdf

[12]. Efficient and stealthy practise tracing and its programs in computerized malware evaluation: Open problems and demanding situations Endre Bangerter, Stefan B¨uhlmann, and Engin Kirda Bern University of Applied Sciences, Switzerland endre.Bangerter@jdiv.Org Bern University of Applied Sciences and Joe Security, Switzerland stefan.Buehlmann@bfh.Ch Northeastern University, USA ek@ccs.Neu.Edu, http://dl.Ifip.Org/db/conf/ifip11 - 4/inetsec2011/BangerterBK11.Pdf

[13]. Http://cwsandbox.Org/

[14]. Norman Sandbox. Http://sandbox.Norman.No

[15]. Gadhiya et al., International Journal of Advanced Research in Computer Science and Software Engineering three(four), April - 2013, pp. 972-975

[16]. Anubis. Analysis of unknown binaries. Http://anubis.Iseclab.Org

[17]. Toward automatic dynamic malware evaluation the usage of CWSandbox. Http://dl.Acm.Org/citation.Cfm?Id=1262675

[18]. Christopher Elisan (five September 2012). Malware, Rootkits & Botnets A Beginner's Guide. McGraw Hill Professional. Pp. 10–. ISBN 978-zero-07-179205-nine

[19]. Karthik Selvaraj and Nino Fred Gutierrez, The Rise of PDF Malware, Symantec Security Response.

[20]. Jeong Wook (Matt) Oh (jeongoh@microsoft.Com), Recent Java exploitation trends and malware, Black Hat USA 2012 Las Vegas.

[21]. Imtithal A Saeed, Ali Selamat and Ali M A Abuagoub. Article: A Survey on Malware and Malware Detection Systems. International Journal of Computer Applications 67(16):25-31, April 2013. Full textual content to be had.

[22]. Verma, Aparna, M.S.Rao, A.K.Gupta, W. Jeberson, and Vrijendra Singh. "A Literature Review On Malware And Its Analysis." International Journal of Current Research and Review five (2013), 71-82.

**Cite this article as :**

Soumen Chakraborty, "Malware attack and Malware Analysis : A Research", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 3, pp. 268-272, May-June 2019.

Journal URL : http://ijsrcseit.com/CSEIT195379