IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-3, Issue-1 E-ISSN: 2347-2693

Survey on Malware and Rootkit Detection

Aswana Lal[1*], M. Azath[2] and Miss Sony[3]

[1*,2,3]*Department of Computer Science and Engineering, Met's School of Engineering, Mala, India*
aswanalal@gmail.com, mailmeazath@gmail.com,sonythekkekara1271988@gmail.com

www.ijcaonline.org

Received: Dec /26/2014 Revised: Jan/8/2015 Accepted: Jan/20/2015 Published: Jan/31/2015

Abstract— Malwares are malicious software, designed to damage computer systems without the knowledge of the owner. Rootkit is also malicious software which hides the existence of certain processes or programs from normal methods of detection and enables continued privileged access to a computer. Now a day the impact of malware and rootkit is getting worst. Their detection is difficult because malicious program may be able to subvert the software that is intended to find it. Detection methods uses an alternative and trusted operating system, signature scanning behavioral-based methods, difference scanning, and memory dump analysis etc. Malware and rootkit detectors are the primary tools in defense against malicious programs. The quality of such a detector is determined by the techniques used by it. There are mainly two types of techniques to detect the malwares, signature based and anomaly based techniques. Signature-based detection is a malware detection approach that identifies a malware instance by the presence of at least one byte code pattern present in a database of signatures from known malicious programs. If a program contains a pattern that already exists within the database, it is deemed. In anomaly based detection malwares are classified according to some heuristics and rules. This survey study about signature based and anomaly based malware detection and list their strengths and limitations. It also compares techniques and helps to choose a desirable technique for secure system.

Keywords— Anomaly based malware, rootkit, malware detection malcode, malicious code, malicious software, signature-based, behavior based.

I. INTRODUCTION

II.

Rootkits and malwares are the programs used by cybercriminals, hackers and nation states to disrupt computer operations, steal personal or professional data, bypass access controls and also cause harm to the host system[1]. Appearing in the form of executable code, active content, scripts or other software variants, there are many different classes of malware which vary in means of infecting machine and propagate themselves. The term 'Malware' is created from merging the words 'malicious' and 'software'. Once the malware or rootkit finds its way into the system, it scans for vulnerabilities in operating system and perform un-intended actions on the system finally slowing down the performance of the system[2]. Malware has ability to infect other data/system files, executable code, boot partitions of drives, and create excessive traffic on network leading to denial of service.

Security products such as virus scanners or rootkit scanners look for characteristics byte sequence to identify malicious code. The quality of the detector is determined by the techniques employed for detection. A good malware detection technique must be able to identify malicious code that is hidden or embedded in the original program and should have some capability for detection of yet unknown malware. Malware detectors take two inputs. One is its knowledge of the malicious behavior. The other input is the program under inspection. A commercial virus scanner has very low resilience to new attacks because malware writers continuously make use of new obfuscation methods so that the malware could evade detections.

III. MALWARE TYPES

Malware is commonly divided into a number of classes, depending on the way in which it is introduced into the target system and the sort of policy with which it is intended to cause

A. Virus: A computer virus is code that replicates itself by inserting into other programs[3]. From an infected system the virus can spread through network, or medias like USB drive CD etc. Viruses' cause systems failure, wasting computer resources, corrupting data and files, increases maintenance costs, etc.

B. Worms: A computer worm is a standalone program which replicates itself by executing its own code independent of any other program[4]. Worms spread via network connections with the goal of infecting as many computer systems connected to the network as possible. It might delete files, encrypt files in as crypto viral extortion attack or send junk email and also cause traffic conjunction in network.

C. Trojan horses: A Trojan horse is malware embedded by its designer in an application or system. The application or system seems to perform some useful function, but is performing some unauthorized actions[5].

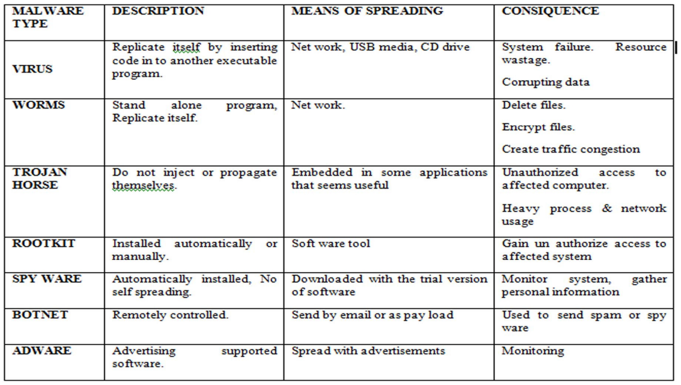
A Trojan often acts as a backdoor that can contact a controller which can then have unauthorized access to the affected computer.Computers may seem to run slower due to heavy processor or network usage.

D. Rootkit: A rootkit is a set of software tools that enable an unauthorized user to gain control of a computer system without being detected. Rootkit installation can either be automated, or an attacker can install it once they've obtained root or Administrator access.Rootkit detection is difficult because a rootkit may be able to subvert the software that is intended to find it.

E. Spyware: Spyware is a program which monitors and gathers information about a person or organization without their knowledge and pass the information to another entity. It check the pages frequently visited by the user, email address, credit card number, key pressed by user etc. It also gain the control of the system without the customers knowledge. It generally enters a system when free or trial software is downloaded.

F. Botnet: A botnet is remotely controlled software or a collection of autonomous software robots. Botnets are usually used to send spam or spyware remotely.

G. Adware: Adware or advertising-supported software downloads plays or displays advertisements to a computer after malicious software is installed or application is used without the users knowledge. This piece of code is usually embedded into free software. The problem is, many developers abuse ad supported software by monitoring Internet users' activities .The most common adware programs are free games, peer-to-peer clients etc.



MALWARE TYPE	DESCRIPTION	MEANS OF SPREADING	CONSIQUENCE
VIRUS	Replicate itself by inserting code in to another executable program.	Net work, USB media, CD drive	System failure. Resource wastage. Corrupting data
WORMS	Stand alone program, Replicate itself.	Net work.	Delete files. Encrypt files. Create traffic congestion
TROJAN HORSE	Do not inject or propagate themselves.	Embedded in some applications that seems useful	Unauthorized access to affected computer. Heavy process & network usage
ROOTKIT	Installed automatically or manually.	Soft ware tool	Gain un authorize access to affected system
SPY WARE	Automatically installed, No self spreading.	Downloaded with the trial version of software	Monitor system, gather personal information
BOTNET	Remotely controlled.	Send by email or as pay load	Used to send spam or spy ware
ADWARE	Advertising supported software.	Spread with advertisements	Monitoring

Table 1: Types of malwares and its properties.

IV. MALWARE DETECTOR

Malware detector software is computer software which prevents, detects and removes malicious softwares from computer. It is also called antivirus softwares. The malware detector may or may not reside on the same system it is trying to protect. Anti-virus detects and blocks attempts by the bad guys to infect a computer. The malware detector performs its protection through the manifested malware detection techniques.[6] The goal of testing a malware detector is to find out false positive, false negative and hit ratio.

a) False positive:
A false positive is a mistake that happens occasionally the antivirus thinks a download is harmful when it's actually safe. But malicious people may try to trick you into downloading malware with this assurance.

b) False negative:
A false negative occurs when the scanning software does not find a virus that in fact exists on the system. User of course won't have any way of knowing that this has occurred, until the virus manifests itself in some obvious way on the system. No virus scanner is perfect, and some viruses will be missed by any of these programs, although good ones will miss relatively few.

c) Hit ratio:
A hit occur when a detector detects a malware. Hit ratio will be high if it detects all the malwares in the system.

V. MALWARE DETECTION TECHNIQUES

Malware detection techniques are broadly classified in to two classes. Anomaly-based detection and signature-based detection. An anomaly-based detection technique uses its knowledge of what constitutes normal behavior to decide the maliciousness of a program under inspection. A special type of anomaly-based detection is referred to as specification-based detection. Specification-based techniques leverage some specification or rule set of what is valid behavior in order to decide the maliciousness of a program under inspection. Programs violating the specification are considered anomalous and usually, malicious. Signature based detection uses the knowledge of what is considered as malicious to finds out the maliciousness of the program under inspection.

a) Signature-Based Malware Detection Technique:

Commercial antivirus scanners look for signatures which are typically sequence of bytes within the malware code to declare that the program scanned is malicious in nature. Ideally, a signature should be able to identify any malware exhibiting the malicious behavior specified by the signature.[6][7] There will be a repository which represents all of the knowledge the signature-based method has. Signatures are being searched in this repository. Currently, we primarily rely on human expertise in creating the signatures that represent the malicious behavior exhibited by programs. Once a signature has been created, it is added to the signature-based method's knowledge (i.e. repository). One of the major drawbacks of the signature-based method for malware detection is that it cannot detect a zero-day attack that is an attack for which there is no corresponding signature stored in the repository. Figure 1 illustrates the major disadvantage of signature-based methods.

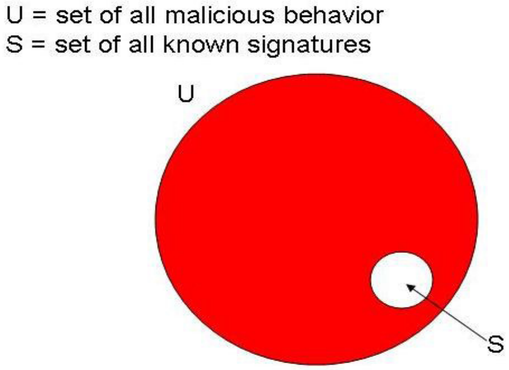

U = set of all malicious behavior
S = set of all known signatures
U
S

Figure 1: An illustration of why signature-based detection is insufficient

Since the set of possible malicious behaviors, U, is infinitely large, there are no known techniques for accurately representing U via signatures[7]. Furthermore, a repository of signatures is a weak approximation to U.

Another drawback of signature-based methods is that human involvement/expertise is usually needed to develop the signatures. This not only allows for the introduction of human error, but takes considerably more time than if signature development was completely automated. Given that some malware has the capability to spread extremely fast, the capability to quickly develop an accurate signature becomes paramount.

In signature-based method we have to create unique signature for each malware. So there will be a number of malware signature stored in the repository. Although currently, storage is not an issue, over time, storage could potentially become a serious one as this will have direct affects on the time complexity of the malware detector.

Basically there are three kinds of malwares: basic, polymorphic, metamorphic malwares. In basic malware the program entry point is changed such that the control is transferred to malicious payload.[6] Detection is relatively if the signature can be found for the viral code. Polymorphic viruses mutates while keeping the original code intact. A polymorphic malware consists of encrypted malicious code along with the decryption module. To enable the polymorphic virus the virus has got polymorphic engine somewhere in the virus body. The polymorphic engine generates new mutants each time it is executed. Signature based detection for such a virus is difficult because each variant new signature is generated which makes signatures based detection difficult. Strong static analysis based on API sequencing is used for polymorphic virus detection.

Metamorphic malware can reprogram itself using certain obfuscation techniques so that the children never look like the parent. Such malwares evade the detections from the malware detector since each new variant generated will have different signature, hence it is impossible to store the signatures of multiple variants of same malware sample. In order to prevent detection, a metamorphic engine has to be implemented with some sort of disassembler in order to parse the input code. After disassembly, the engine will transform the program code and will produce new code that will retain its functionality and would still look different from the original code.

The main problem with the signature based detection method is as follows:

- Signature extraction and distribution is a complex task.
- The signature generation involves manual intervention and requires strict code analysis.
- The signatures can be easily bypassed as and when new signatures are created.
- The size of signature repository keeps on growing at an alarming rate.

b) Anomaly based Detection

An Anomaly-Based malware or intrusion detection system is a system for detecting computer malwares and intrusion by monitoring system activity and classifying it as either normal or anomalous.[8] The classification is based on heuristics or rules, rather than patterns or signatures, and attempts to detect any type of misuse that falls out of normal system operation. This is as opposed to signature based systems which can only detect attacks for which a signature has previously been created. Anomaly based detection usually occurs in two phases, a training (learning) phase and a detection (monitoring) phase. During the training phase the detector attempts to learn the normal behavior. The detector could be learning the behavior of the host or the PUI or a combination of both during the training phase. A key advantage of anomaly-based detection is its ability to detect zero-day attacks. Similar to zero-day exploits, zero-day attacks are attacks that are previously unknown to the malware detector. However, the use of anomaly detection in practice is hampered by a high rate of false alarms.

Another major drawback of anomaly detection is defining its rule set. The efficiency of the system depends on how well it is implemented and tested on all protocols. Rule defining process is also affected by various protocols used by various vendors. Apart from these, custom protocols also make rule defining a difficult job.

c) Specification based Detection:

Specification-based detection is the derivate of anomaly based detection[9]. Instead of approximating the implementation of a system or application, specification-based detection approximates the requirements of application or system. In specification-based system there exists a training phase which attempts to learn the all valid behavior of a program or system which needs to inspect. The main limitation of specification based system is that it if very difficult to accurately specify the behavior the system or program. One such tool is Panorama which captures the system wide information flow of the program under inspection over a system, and checks the behavior against a valid set of rule to detect malicious activity. Even though specification based detection have less false positive, it is not as effective as anomaly based detection

d) Behavior based Detection:

Behavior based detection differs from the surface scanning method in that it identifies the action performed malware rather than the binary pattern [10]. The programs with dissimilar syntax's but having same behavior are collected, thus this single behavior signature can identify various samples of malware. This types of detection mechanisms helps in detecting the malwares which keeps on generating new mutants since they will always use the system resources and services in the similar manner.

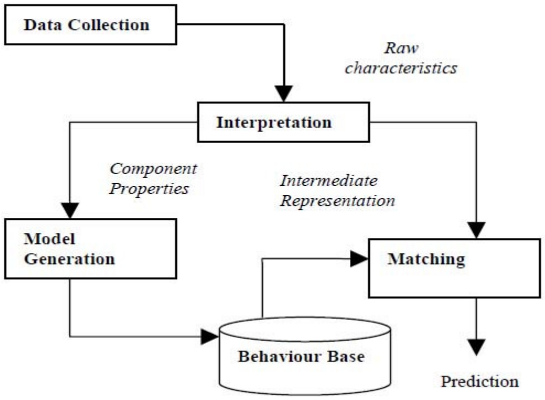


Fig. 2 Behavior detector [11]

DETECTION TECHNIQUES	DISCRIPTION	DE MERITS
SIGNATURE Based Technique	Detector checks the unique signature whether matches with signatures in the repository	Human errors can occur. Cannot detect zero-day attack. Storage keeps growing
ANOMALY Based Technique	Classification is based on heuristics rule. Ability to detect zero day attack. Detect malwares whose signatures are not present.	High false alarm rate. Defining rule set is complex task.
SPECIFICATION Based Technique	Manually developed specifications are used to characterize the behavior. Less false alarm	Difficult to specify the behavior. Not efficient as anomaly based.

Table 2: Comparison of different malware detection techniques

VI. CONCLUSION

In this survey a series of malware detection techniques have been presented. The problems related to the traditional signature based detection method are also highlighted. Rate of new malware's causing destructions to systems worldwide is increasing at alarming rate. Detection of malware's changing their signatures frequently has posed many open research issues. In summary, it can be concluded that automatic behavior-based malware analysis and the use of machine learning techniques could detect malware quite effectively and efficiently.

REFERENCE

[1]https://www.cert.gov.uk/wpcontent/uploads/2014/08/An-introduction-to malware.pdf

[2]http://www.ukessays.com/essays/computer-science/the-introduction-to-malicious-software-computer-science-essay.php

[3]http://en.wikipedia.org/wiki/Computer_virus

[4]http://en.wikipedia.org/wiki/Computer_worm

[5]http://en.wikipedia.org/wiki/Trojan_horse_(computing)

[6]" Survey on Malware Detection Methods" Vinod P. Department of Computer Engineering, Malaviya National Institute of Technology, Jaipur, Rajasthan

[7]"A Survey of Malware Detection Techniques"NwokediIdika,AdityaPMathur.Department of Computer Science Purdue University, West Lafayette, IN 47907.

[8]" A Survey on Techniques in Detection and Analyzing Malware Executables" Kirti Mathur M.Tech. Scholar, Department of CSE Rajasthan Technical University, India.

[9]"A Specification-based Intrusion Detection System for AODV" Chin-Yang Tseng, Poornima Balasubramanyam, Calvin Ko,Rattapon Limprasittiporn,Jeff Rowe,Karl Levitt,Computer Security Laboratory University of California, Davis.

[10]http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.138.7174

[11] Greoigre Jacob,Herve Debar,Eric Fillol,"Behavioral detection of malware:from a survey towards an established taxonomy",Springer-Verlag France 2008