Combating Malware Threat using Hybrid Security Model

P. R. Lakshmi Eswari
Centre for Development of Advanced Computing (C-DAC)
Hyderabad, India

N. Sarat Chandra Babu
Centre for Development of Advanced Computing (C-DAC)
Bangalore, India

Abstract

Malware threat is continuously growing with sophistication. Though multiple layers of defense are provided at perimeter, network, host, application and data levels, it is still becoming a challenge to address malware related problems. They have grown in number as well as complexity and are responsible for attacks ranging from denial-of-service to compromising online banking accounts. In the recent times, blended attacks are popular with high severity of damage and are difficult to address using signature based anti-malware solutions. Signature based anti-malware solutions are not able to detect zero-day malware. Though heuristic based anti-malware solutions are able to increase the detection rate, their false positive rate is high. Positive security model is effective but creates rigidity on environment. Through this paper we analyzed positive as well as negative security models and proposed hybrid security model for combating malware threat, considering the nature of Information Technology (IT) systems and their business objective.

1. Introduction

In the last two decades malware related threats on Information Technology (IT) systems are growing continuously with lot of sophistication. Malware has evolved from virus to botnets, is responsible for attacks ranging from denial-of-service to compromising online banking accounts. Blended threats are on the raise, which use a combination of malware attack vectors against different vulnerabilities to penetrate targets and carry out the malicious activity. These threats are used by cyber criminals to attack specific target user, by sending a socially engineered message in the form of an email or instant message that lures the user to click on a link. The link can be a pdf or doc file which contains malware that exploited vulnerabilities in software such as Adobe Reader or Microsoft Office. Malware payload in these exploited files silently executes on the target's computer. In this way cyber criminal takes control of targeted computer and collects data from the network through the compromised computer. Attacks exploiting software vulnerabilities are growing due to the proliferation of software intensive systems. These attacks are targeted on information assets of high-profile organizations to steal intellectual property or to cause any other possible damage. Recent malware like Stuxnet, Duqu and Flame are very complex and are different from everyday malware. These malware have exploited zero-day vulnerabilities and were designed specifically to target industrial control systems. They used fake digitally signed components in order to appear as trustworthy applications. Stuxnet alone exploited four different zero-day vulnerabilities when attacking its targets [1]. It is initially spread through infected USB flash drives, and then it uses other exploits and techniques such as peer-to-peer RPC to infect and update other computers inside private networks which are not directly connected to the Internet. In the case of Flame, it is developed using SQLite, SSH, SSL and LUA libraries which helps it to look like a business database software than a piece of malware [2]. Duqu has many similarities to Stuxnet like modular structure, injection mechanisms and using fake digital signature [3].

Web-based malware infections are emerging as another major threat for websites and web users. These malware operate from flash based advertisements, HTML and in many forms of Javascript functions. Social networks and dynamic content have raised the delivery of sophisticated web threats. With Web 3.0 becoming the critical force for the future of cloud computing and increase in services offered through subscription-based, there is a huge malware threat for these environments. Also, with the increase in usage of ubiquitous mobile devices such as smart phones and tablets, mobile malware is on the raise. Cyber criminals view these devices as highly-sensitive targets and are developing new ways to compromise personal data on mobile devices and breach privacy of individuals by tracking people etc. With the wider proliferation of Google Android, attacks exploiting vulnerabilities are increasing [4].

These malware threats are throwing new challenges for the anti-malware industry in developing effective anti-malware technologies. Signature-based approach tries to detect different malware by searching for known patterns of data within executable code. Although it is a commonly used technique for malware detection, it cannot detect unknown malware and it is also susceptible

to evasion such as packing and junk code insertion. With the growing number of new vulnerabilities noticed every day and the unknown number of undisclosed vulnerabilities, zero-day attacks makes the malware signature set always incomplete [5]. Also since these solutions require a separate signature for each malware variant, database of signatures grow exponentially and becomes unmanageable.

Heuristic based approach increases the detection rate of malware by using generic signatures to detect the variants of known malware. But its false positive rate is high. In recent times, anti-malware industry is looking at whitelisting, which allows only approved applications to run. Since whitelisting allows only known good, it is an effective way to address malware threat but it creates a rigid environment. Therefore, each of the different approaches used in anti-malware solutions have their own advantages and disadvantages. In order to choose appropriate solution and provide practical security, there is a need to consider the nature of IT systems as well as their business objective. In this paper, the positive and negative security models are analyzed in detail and a Hybrid Security Model is proposed to combat malware threat on IT systems. Section 2 gives detailed overview on positive and negative security models. Section 3 covers details of proposed Hybrid Security Model to provide practical security for IT systems. Conclusions are given in Sections 4 and references are provided at the end.

2. Positive and Negative Security Models

Positive security model whitelist and allow only known good where as negative security model blacklist and block only known bad as depicted in Fig. 1. When no rule is defined negative security model would grant access to everything and when exploits are discovered rules are created and added. Adding more rules in negative security model increases the blocking behavior, thereby decreasing the threat through unknown and allowed, as security gets tightened. When no rule is defined in positive security model, everything is blocked. Once the resources / behavior are trusted, rules are created and added.
Adding more rules in positive security model increases allowed behavior.

Over the years, positive as well as negative security models are being used in order to secure our networks and systems. These models are implemented through popular mechanisms such as anti-virus, intrusion detection system / intrusion prevention system (IDS/IPS), network firewall etc. Known good in the form of whitelist of applications, IP addresses, port numbers, and application behavior are maintained and enforced in positive security model like Access Control Lists (ACLs) in firewalls [6]. Whereas, blacklist of known bad is maintained and enforced using negative security model like signatures in antivirus scanners, anti-spam engines etc.

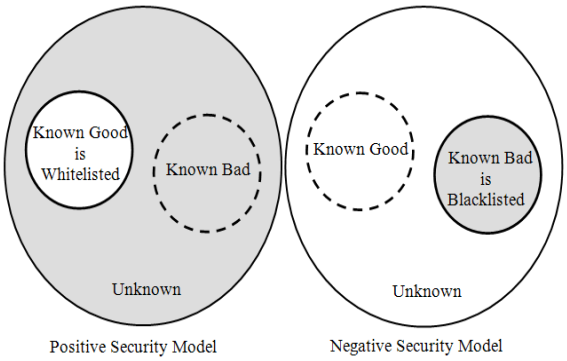

Figure 1. Positive Vs Negative Security Models

Traditionally technology used to address malware issue is negative security model. Trusted third parties like antivirus companies maintain the blacklist of malware signatures and via Internet everyone is able to automatically update the blacklist and enforce them through antivirus scanner. When the known bad list is small, it is administratively easy to maintain blacklist and stop it [7]. But currently the number of malware software is growing rapidly and it is becoming administratively difficult to maintain and enforce the blacklist of signatures. Also this approach will not be able to detect malware which is bad but unknown like zero-day threats, which is a malware without corresponding antivirus signature. Unknown vulnerabilities in continuously evolving software systems are exploited and zero-day attacks are made. There is a vulnerability window that exists during the time between when vulnerability is first exploited and when software developers start to develop and publish a counter mechanism to that threat. It takes time for the developers to discover the existence of a new attack, analyze the attack to plan remedy, develop a defensive attack signature and distribute the same for enforcing at various endpoints. No matter how efficient the antivirus software is, the attack will always spread quickly before the vulnerability window can be closed. Also, if we analyze carefully it is very difficult to build any software like web application, database application and embedded application, which is completely vulnerability free. Skillful attackers would continuously make efforts to attack through unknown vulnerabilities and bypass signatures.

Antivirus technology was originally designed for scanning files when they are created on PCs and not for addressing the threat coming through Internet. This technology is not effective against attacks that penetrate via active code executed in browsers or transferred through USB devices [8]. Currently, the malware that penetrate through the browser show the highest growth rate among all malware classes. Similarly, infected USB devices are another major channel for malware penetration as they are rarely scanned for malware when connected to a PC. Latest malware like Stuxnet and Conficker are using USB device as one of their methods for propagation. Though antivirus software tries to supplement signature scanning with behavior heuristics which looks for anomalies while processing and increases the detection rate but their false positive rate is high.

With the increasing number of free software applications with hidden malicious behavior available on Internet, end users install these unauthorized software and corrupt their systems and networks. So it is important to use effective defense mechanism in order to protect our end points from growing malware threat. Also it is very clear that antivirus solutions are not effective for zero-day, Internet and USB device attacks. In this scenario, where malware attacks are using sophisticated techniques and also since their count is overtaking the genuine software, it is difficult to address the problem using negative security model alone. So there is a need to analyze the potential of positive security model. It is very clear that antivirus solutions are not effective for zero-day, Internet and USB device attacks.

Rather than detecting and blocking threats, positive security model blocks all and only trusted items are allowed. In order to effectively implement positive security model, every part of IT infrastructure has to be discovered and whitelisted including hardware, software and configuration like servers, desktops, printers, USB devices, applications, behavior of applications etc. It is required to create whitelisting rules for all trusted items of the network.

Application whitelisting is an approach where only known or trusted applications are whitelisted and only those applications are allowed. This is effective in addressing the threat through unknown applications [9]. Though application whitelisting is a promising technique, it is also vulnerable to attacks. Any malware injected at runtime and operating from a whitelisted application cannot be detected. To address this problem, application behavior also needs to be whitelisted and enforced. One major approach to whitelist behavior of application is anomaly detection. In this approach, profile of application is captured and any deviations from this are flagged as suspicious. This approach is susceptible to false positives and mimicry attacks. Also it becomes difficult to capture the normal behavior of complex applications. Another approach for whitelisting behavior is specification based detection, in which all events from application to the operating system are mediated by a policy or specification. These policies are application specific and indicate whether the events are allowed or denied. Specification-based approach is flexible and has lower false positives. Though positive security model is effective, it is administratively challenging to maintain whitelist of known good and the granularity at which it has to be created and enforced. This approach would be relevant to systems which are less frequently changed or patched. Therefore in order to provide practical security, it is required to consider the nature of IT systems and secure them by using a combination of positive and negative security models.

3. Practical Security Framework using Hybrid Security Model

In order to combat malware threat in different IT systems, a practical security framework using Hybrid Security Model is proposed as given in Fig. 2. In this framework, IT systems are classified into home environments, corporate environments, online web services, cloud environments, embedded systems and mission critical environments. Each of these environments has different requirements of security and end user flexibility. Based on these requirements, security framework is proposed using a combination of positive and negative security models.

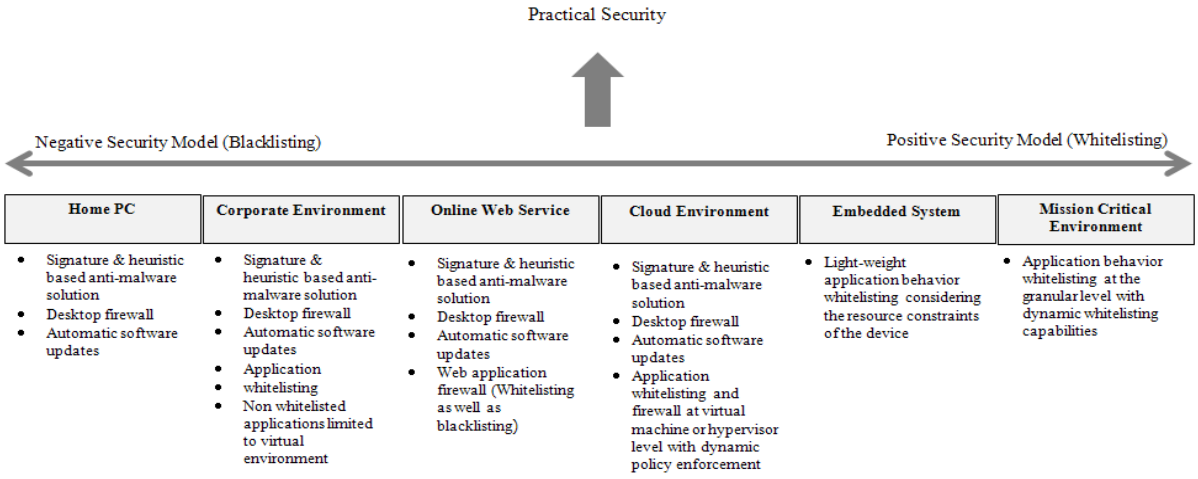

Figure 2. Practical Security Framework using Hybrid Security Model

Home environments are normally used for general purpose computing and entertainment. These environments do not have any critical business goal and user wants lot of flexibility in downloading and deploying applications of their choice. So, signature as well as heuristic based anti-malware solutions along with desktop firewall would be appropriate to secure these environments. Desktop firewall helps to allow only required ports and block all other ports in order to stop external infiltration. Also it is very important to configure these environments for automatic software updates.

Signature as well as heuristic based anti-virus solutions, desktop firewall and configuring the end system for automatic software updates are also relevant in other IT systems like corporate environment, online web servers and cloud environment. These IT systems also require additional security solutions.

Every corporate environment has a specific business goal. These environments have applications accessed by different stakeholders. In order to secure these environments, access to enterprise applications should be given only to authorized stakeholders. Also only enterprise specific applications are allowed to run in those environments. To enforce these security requirements, application whitelisting can be used. All enterprise specific applications can be whitelisted and only those applications are allowed. In order to check the integrity of whitelisted applications details like hash value, publisher etc can verified while loading the application for execution. In case if the organization want to allow users to deploy any other applications of their choice, then their access should be limited to only virtual environment without giving access to enterprise services.

IT systems providing online web services have focused role of giving specific service online through web interface. Rules based on negative security model to cover common attacks like SQL injection, cross site scripting etc need to be supported. Dynamism and rapid evolution of new web technologies left security vulnerabilities at server side as well as client side components of web based applications. Since this is the root cause for growing web based attacks, it is very important to address this threat by monitoring HTTP protocol messages and whitelisting valid messages exchanged between client and web application. Therefore web application firewall also uses positive security model, which allows only web traffic that complies with web application behavior.

These days' services are being offered through cloud environments. These environments support scaling of applications due to their elastic nature. In these environments, different services are offered through different Virtual Machines (VMs) on the same machine or single service offered through different VMs across multiple machines and it is dynamically controlled. End point security deployed in these machines would not be able to provide VM level security. Also in most of the cases all machines of a particular cloud have similar software setup. In such a scenario, if unknown vulnerability is exploited in any one machine of cloud, with in no time it can bring down all other machines of the cloud. So, it is very important to harden virtualization layer. Application whitelisting, firewall and IDS/IPS functionalities have to be deployed and enforced with dynamic policies at virtual machine or hypervisor level. This also protects from the spread of threat from one VM to another VM.

Most of the times, focus is not given to security aspects in embedded systems used at home, automobiles, controllers etc, which leaves them vulnerable to various security threats. Security can be easily compromised in these devices, like DNS settings on wireless routers were updated by malware and computers connected through these devices to Internet are compromised. Since embedded systems are ubiquitous, it is very difficult to make security related updates after deployment even if vulnerabilities are detected later. So, it would be better to support whitelist based security to embedded devices and lock down the behavior of the embedded system. Since these devices have limited resources and created for specific purpose, specification based application behavior whitelisting using a light weight model would be an appropriate method to combat malware threat [10]. This reduces the likelihood of malware exploiting vulnerabilities and carrying out malicious activity.

Mission critical IT systems like supervisory control and data acquisition (SCADA) systems have focused applications with critical business objective. Any vulnerability exploitation in these systems is a serious threat and it can bring down the operations. The functionality of critical applications in specifically defined way should be allowed for execution and any unknown or bad behavior should be blocked. Stuxnet malware was designed to propagate and target specific Supervisory Control and Data Acquisition System (SCADA). It has altered the control functions in the programmable logic controller (PLC) to physically destroy the equipment. So in these type of mission critical environments, it is always better to whitelist or allows known good at the granular level with dynamic whitelisting capabilities. Specification based application behavior whitelisting should be used at the granular level by whitelisting only those resources and operations specific to the behavior of critical application in the network. Positive security model has greater relevance in mission critical environments [11]. Therefore, it is very important to consider the business requirements of IT systems while implementing the practical security.

Also with the growing attacks through unknown vulnerabilities, it is very important to give assurance to end user that functionality is implemented correctly and software provides only the desired features. So, it is important to provide assurance in software. Positive Security model can also play a vital role in providing assurance to end user. Assurance model can be evolved based on application whitelisting and its behavior whitelisting rules. Specification based application behavior whitelisting will enforce the whitelisted behavior and can also provide formal security assurance of IT systems [12]. Effective Application behavior whitelisting techniques at granular level have huge potential in providing security assurance of IT systems. Many research efforts are being made to come out with an effective light-weight behavior model by using static as well as dynamic analysis techniques [13][14] .

4. Conclusions

It is becoming difficult to address today's malware threat, using signature and heuristic based anti-malware solutions. With the growing zero-day attacks and also malware outnumber the genuine software, a combination of positive and negative security models is more appropriate to protect our IT systems. Practical security requirements of each IT system are different and depend on its business objective. Through this paper we analyzed positive and negative security models and proposed a Hybrid Security Model for IT systems to combat malware threat. As we move from home PC towards embedded system and mission critical environment, need for positive security model grows. Since home PC requires more flexibility negative security model would be appropriate. In mission critical environment, since any security threat can completely bring down the operations, positive security model is more appropriate. Specification based application behavior whitelisting is more effective in these environments. Also with the growing attacks by exploiting software vulnerabilities, it is required to give assurance to end user that functionality is implemented correctly and software provides only the desired features. Positive Security model can play a vital in providing assurance to end users. Research efforts in coming out with effective application behavior whitelisting techniques would help to provide security and also formal security assurance of IT systems.

5. References

[1] Ryan Naraine, "Stuxnet attackers used 4 Windows zero-day exploits," *http://www.zdnet.com/blog/security/stuxnet-attackers-used-4-windows-zero-day-exploits/7347*, Sep 14, 2010.

[2] Mikko Hypponen, *Why Antivirus Companies Like Mine Failed to Catch Flame and Stuxnet*, http://www.wired.com/threatlevel/2012/06/internet-security-fail/, Jan 6, 2012.

[3] Boldizsár Bencsáth, Gábor Pék, Levente Buttyán, Márk Félegyházi, *Duqu: A Stuxnet-like malware found in the wild*, http://www.crysys.hu/publications/files/bencsathPBF11duqu.pdf, Oct 14 2011.

[4] Kaspersky Lab, *Cyberthreat forecast for 2012*, http://www.kaspersky.com/images/Kaspersky%20report-10-134377.pdf, 2012.

[5] Kevin Townsend, "Does it Matter if it's Black or White (listing)?," *http://www.infosecurity-magazine.com/view/20083/does-it-matter-if-its-black-or-whitelisting-/*, Aug 18, 2011.

[6] Alan Murphy and KJ (Ken) Salchow, "Applied Application Security— Positive & Negative Efficiency," *http://www.f5.com/pdf/white-papers/applied-app-security-wp.pdf*, Oct, 2007.

[7] Bruce Schneier, "Whitelisting vs. Blacklisting," *http://www.schneier.com/blog/archives/2011/01/whitelisting_vs.html*, Jan 28, 2011.

[8] Ogren Group, *Endpoint Security: Moving Beyond AV*, http://www.preventia.co.uk/resources/white_papers/lumension/Endpoint-Security-Moving-Beyond_AV-Application-Whitelisting.pdf, Jul 2009.

[9] Shyni, "Whitelist and Blacklist Entries," *http://strangerinmypc.com/108/*, Feb 16, 2012.

[10] Dave Shackleford, "Application Whitelisting: Enhancing Host Security," *http://www.sans.org/reading_room/analysts_program/McAfee_09_App_Whitelisting.pdf*, Oct, 2009.

[11] Brian Contos, "Host Security for SCADA and ICS Systems Part 2," *http://blogs.mcafee.com/enterprise/security-connected/host-security-for-scada-and-ics-systems-part-2*, Feb 22, 2012.

[12] Patrice Godefroid, Michaeal Y. Levin, David Molnar, *SAGE: Whitebox Fuzzing for Security Testing*, Communications of the ACM, March 2012.

[13] Peng Li, Hyundo Park, Debin Gao, Jianming Fu, *Bridging the Gap between Data-flow and Control-flow Analysis for Anomaly Detection*, Annual Computer Security Applications Conference 2008.

[14] Sandeep Bhatkar, Abhishek Chaturvedi, R. Sekar, *Dataflow Anomaly Detection*, IEEE Symposium on Security and Privacy, 2006.

