Global Information Assurance Certification Paper

Stopping Malicious Code at the Desktop

Anthony Tulio

SANS Security Essentials
GSEC Practical Assignment
Version 1.2f

November 6, 2001

Contents

Introduction

Of all the threats to computer systems today, the damage caused by malicious code may be one of the greatest. An InformationWeek article put the damage caused by the ILoveYou virus at $6.7 billion during the first five days of the incident[1]. Another article by Reuters puts the damage for all viruses at over $10 billion so far this year[2]. Whether or not you agree with the some of the numbers that are put forth on costs of this and similar incidents, one fact remains: If a piece of malicious code slips by a desktop's defenses, it has the potential for significant damage. Similarly, if the code has the ability to replicate in the enterprise, it could take considerable time to bring the entire system back to the pre-incident state and may result in a sizeable loss of productivity for the business.

To better understand how to protect the desktop from malicious code, we will discuss methods for identifying and intercepting such code before it causes any damage. And since there is no magic pill that cures all malicious code ills, we will examine how these different methods respond against some of the different types of attack mechanisms that are available. But first, lets take a brief look at some of the ways in which malicious code tries to circumvent protection products and establish a foothold on a local computer.

Malicious Software

Malicious software has been around for as long as most PC users can remember. Code to steal access credentials, destroy data, acquire information, and just about anything else that you can think of has probably been developed and deployed on more platforms than most people ever knew existed. For a look at the early days of malware, the reader may want to look at the short book "The History of Computer Viruses[3]". For a more modern perspective on the topic, read an article entitled "A History of Viruses"[4] at the SecurityFocus web site.

When we think of malicious code that can attack the desktop, most of us will first think of computer viruses. Indeed, these are probably the most prevalent type of malicious code that the average desktop user will encounter. The comp.virus newsgroup FAQ defines a virus as "… a self-replicating program containing code that explicitly copies itself and that can "infect" other programs by modifying them or their environment such that a call to an infected program implies a call to a possibly evolved copy of the virus." Put simply, a virus is a piece of parasitic code that attaches to a host executable file to spread. Sometimes the virus's only mission is to spread. Other times, it can be meant to modify or destroy data. Sometimes, it's meant to display a message or perform some other nuisance task. Although technically not a virus, a worm is a similar type of malicious code that spreads itself throughout available systems. For our discussion, the main difference between a worm and a virus is the worm's ability to spread on it's own without the need to attach to a host file.[5]

Another prevalent class of malware out there is the Trojan. A Trojan, or Trojan horse, is an executable piece of code that masquerades as a benign piece of software. This

software usually represents itself as doing something that is desirable to the end user, such as a game or utility. When the software is launched, it will usually perform an action that has a negative impact on the security of the system. Trojans have been written to destroy data, steal information, and transfer control of a PC to a remote host. And with the plethora of executable types available today on Windows systems, a Trojan, or for that matter any type of malicious code, need not be an exe file.

Sometimes, a simple batch file may be all that is needed to meet a nefarious objective. There are also a number of scripting languages available, including Visual Basic, which is available on the vast majority of PC's in use today. These scripting languages are not difficult to learn, and people have developed scripting front ends that allow almost anyone to sit down and write a virus or Trojan with little effort.

Fortunately, much of this software follows certain patterns of coding or behavior. This makes it possible to detect and stop many actions before they adversely affect the user. Anti-virus software, which can detect previously unknown malicious actions, as well as more traditional types of viruses, is the first line of defense against the malicious code threat. Anti-virus vendors are constantly examining code from the wild for new threats from which they can update their protection software. And writers of malicious code are constantly looking for ways to make their creation invisible to the current version of detection software. This process of the malware writers and the anti-virus vendors leapfrogging one another occurs continuously. Almost as soon as a new virus is introduced, new software to combat it is released. And almost as soon as new protection software is released, new ways of defeating it are discovered.

Detection Software

Software to detect the malicious code threat at the desktop generally falls into one of three categories: signature matching, behavior analysis, or CRC matching. Many vendors who publish malware protection software include multiple techniques for protecting a PC. One of the most common methods of protection is signature scanning.

Signature Scanning

Signature scanning works on the premise that a virus will present a known pattern that can be matched against a list and identified. Signature scanning can check for these patterns in a file, in memory, or in boot sectors of a disk. This pattern can be in the body of a piece of executable code that is unique to a particular virus and is not normally found in uninfected software. It can be a file that is identified by a particular name. It can even be a line in a script that is the hallmark of a particular virus. Signature scanning is probably one of the most widely used methods of detection. Most major anti-virus products today use some form of signature scanning. But, signature scanning is a reactive response. A signature cannot be written into a vendor's virus definition, or dat, file, until a new virus actually appears. Once the vendor updates the dat file, the user still needs to get the updated dat file into their anti-virus software. And that's where many installations fail when it comes to stopping the malicious code threat. In a recent Computerworld article, John Pescatore from think-tank Gartner, Inc. was quoted as saying "At Gartner, we're declaring signature-based antiviral [protection] at the desktop to be dead. It's providing near-zero value today,

mainly because of the lag in updating the signatures."[6]

Even if the user is vigilant in their efforts to keep their protection up to date, authors of malicious code are constantly looking for news ways to beat the signature scan. One of the more common methods of passing a Trojan through a signature scanner is packing. A packer is a program that compresses an executable permanently, allowing the Trojan to run in its executable state, and changing its signature in the process. And this process can be repeated several times, creating an almost infinite number of variants.[7]

To bypass a signature scan for viruses, some authors have resorted to a variable encryption method call polymorphism. A polymorphic virus is one in which the virus is encrypted using a different key each time. This change is made possible by including an encryption program with the virus called a mutation engine, which sends out a different encryption string with each copy of the virus that it propagates.[8] The varying encryption causes a different signature for each instance of the virus, making a signature match more difficult.

An advanced subset of the signature detection method used by some of the major AV vendors is called heuristics. Heuristic scanners examine the programming logic of a file for suspicious code that may have the possibility of causing damage if executed.[9]The heuristics process takes longer than a signature scan, and can result in false positives. This has the potential to turn the heuristic scanner into a nuisance for the end user.

And as the number of viruses grows, the number of signatures to be compared against grows as well. The Mcafee Virus Information Library[10] currently lists over 58,000 virus variants. This increase in false alerts combined with an increase in the amount of resources needed may cause some users to put malicious code protection second to production and the protection software runs the risk of being ignored or disabled completely.

Signature scanning products come from most AV vendors. A good source for these vendors is West Coast Labs at http://www.check-mark.com. Many of the major AV vendors also offer heuristic scanning capabilities in their products.

Behavior Analysis

Another way to stop malicious code is to watch how suspect code functions. I remember a software product in the late 80's called Symantec Antivirus for Macintosh, or SAM for short. SAM used traditional signature scanning, but also looked for any programs that made changes to files, to the boot sector, or anything else that may have seemed suspicious. When it detected this kind of activity, it simply denied access to the program performing action and notified the user. This approach fell by the wayside of the mainstream vendors as systems became more complex and signature scanning matured, but it's starting to make a comeback.

The biggest appeal for behavior-based detection is the ability to detect unknown viruses. Since these products look for behavior, a program that tries to overwrite an executable file can be detected, even if the program is a virus or Trojan that has never before been seen. And in most cases, the software needs to be updated only for bug

fixes or performance enhancements, lessening the dependence on the end user for the software to remain effective.

Unfortunately for behavior-based software efforts, not all software catches all activities. Some software will excel at looking for malicious code directed towards the OS, while others will concentrate on Internet access or email.[11,12] Not surprisingly, malware authors have also found ways around behavior-based blocking. One of the most effective is tunneling. In this method of avoidance, malicious code tries to bypass higher-level system calls and install itself under the protection software thereby stopping any evidence of its actions from reaching the protection software.[13] Since the behavior method does not look for signatures in executable files, which may allow a signature-based product to catch this type of threat, a tunneling program has the chance to execute and install itself before the behavior-based scanner can detect any suspicious activity.

Another problem with most behavior analysis software is that it only monitors and blocks suspicious activity. It doesn't disinfect. You usually need a signature-based product for that.

Two of the major behavior-based scanning product publishers are Indefense and Finjan Software. Signature-based products such as Norton AntiVirus from Symantec are also starting to add some limited behavior-based detection to their products.

CRC matching

A less commonly used method of protection is CRC matching. Sometimes called checksumming or vaccination, this method will create a CRC checksum of selected files and store this information. It can then compare this information with subsequent scans of a file and determine if the file has been changed. This method is more helpful in detecting an attack after it has occurred, and this method by itself does not offer any way to restore the file to it's original condition. The types of files scanned using this method will vary from product to product. Not many protection products on the market today offer this kind of protection.

Conclusions

As you have probably guessed, no one product can stop all of the malicious code threats out there. Relying on one product to protect your desktop may not be the best approach. You may want to combine products to get the best of signature scanning alongside of the benefits of behavior analysis. But even this approach is not without risk. Aside from the fact that a malware author will eventually write something that will get through your defenses, behavior and signature based products can sometimes conflict. It can cause a situation similar to tunneling where each product tries to get below the other in the system resources, and eventually one or both may fail, or your system may crash altogether. Sometimes a process of trial and error may be needed to obtain a stable, secure configuration.

Another mistake is to rely on the default configurations of protection products. Many product installations will only install certain components. For example, Mcafee VirusScan v4.5, which is still in use in many corporate environments, does not install tools to scan email or Internet access in its default installation mode. Whenever

possible, you should choose to do a custom install and make sure that you know what you are getting, and what you could be missing, when you install a protection product. And install a much as you think that the PC can handle. Most new PC's can handle a full install with little problem. If you don't have a newer PC, it's easier to remove a component of your protection software than it is to remove a virus that got through.

The default configuration warning also goes for settings. Open up the settings and look at how they are set. And read the help file or the manual to find out what settings are best for you. When in doubt, set to maximum, and then work your way back down if you are not satisfied with the results.

For signature-based products, find a product that has automatic update capability built in. And then use it. Remember the biggest reason that signature scanners fail? Your investment in this software will be wasted if you do not update your signatures. Many anti-virus vendors can release signature updates weekly. An update process that you don't have to think about is the one that usually works best.

Of course, another factor in the fight against malicious code that wasn't mentioned above, but that is just as important, is you. You should exercise care when you use your system. Don't go to a website that's questionable and download software to run on your PC. Don't use floppy disks from another system without scanning for viruses first. Don't open email attachments from people that you don't know. You may not want to open an email attachment from someone you do know unless you are expecting it.

And spend some time keeping the rest of your system up to date. There are always updates that will enhance your system security against the malicious code threat. And especially if you are a Windows user, go to http://windowsupdate.com and look at the critical updates. Security patches are posted there regularly.

If you are in a corporate environment, the best defense at the desktop may be to stop malicious code at your network's border. Although it's beyond the scope of this paper, many protection products are available that function on mail servers, proxies, firewalls, and just about any other border device out there. And unless you have an omnipotent enterprise management system, you may want to look at the management offerings of some of the anti-virus vendors, such as Mcafee or Symantec. I recently deployed one of these systems at my company, and was surprised at how much our perception of our current level of protection differed from reality. Fortunately, these management systems also let you bring your systems up to date with relative ease. And don't forget your biggest security asset (or risk, depending on your situation), your end users. A carefully crafted and managed system of user education can do wonders to help curtail the malicious code threat.

The most important point to remember is to employ a strategy of defense in depth. Whether you have one desktop or one thousand, a layered system will give you the best chance of stopping malicious code from invading the desktop.

[13] Hulme, George. “Security Problems Widespread In 2000.” InformationWeek. 3 Jan 2001
URL: http://www.informationweek.com/story/IWK20010103S0002 (1 Sep 2001)

[2] Sullivan, Andy. “Computer Virus Costs Reach $10.7 Billion This Year.” Reuters. 6 Nov 2001
URL: http://biz.yahoo.com/rf/011106/n06338921_6.html (6 Nov 2001)

[3] Slade, Robert. “The History of Computer Viruses.” 1992
URL: http://www.bocklabs.wisc.edu/~janda/sladehis.html (6 Nov 2001)

[4] Paquette, Jeremy. “A History of Viruses.” Security Focus. 17 Jul 2000
URL: http://www.securityfocus.com/infocus/1286 (6 Nov 2001)

[5] Fitzgerald, Nick. “Frequently Asked Questions on Virus-L/comp.virus”. Safetynet. 9 Oct 1995
URL: http://www.safetynet.com/support/kbvfaq.asp (5 Nov 2001)

[6] Scheier, Robert L. “Managing the Virus Threat” Computerworld 7 May 2001
URL: http://www.computerworld.com/cwi/story/0,1199,NAV47_STO60208,00.html (5 Nov 2001)

[7] “Security Overview” Finjan Software
URL: http://www.finjan.com/mcrc/overview.cfm (5 Nov 2001)

[8] Nachenburg, Carey. “Understanding and Managing Polymorphic Viruses”
URL: http://www.norton.com/avcenter/reference/striker.pdf (5 Nov 2001)

[9] “Mobile Code Attack and Recovery” 14 April 2000
URL: http://enterprisesecurity.symantec.com/article.cfm?articleid=65&PID=na (5 Nov 2001)

[10] Mcafee Virus Information Library
URL: http://vil.nai.com/vil/default.asp (6 Nov 2001)

[11] “Achilles Shield: A Technology White Paper” June 2000
URL: http://www.indefense.com/downloads/whitepaper.pdf (5 Nov 2001)

[12] “Frequently Asked Questions” Finjan Software
URL: http://www.finjan.com/mcrc/faq.cfm (5 Nov 2001)

[13] “Virus Glossary” Mcafee
URL: http://www.mcafeeb2b.com/naicommon/avert/avert-research-center/virus-glossary.asp (5 Nov 2001)

---

[1] Hulme, George. “Security Problems Widespread In 2000.” InformationWeek. 3 Jan 2001
URL: http://www.informationweek.com/story/IWK20010103S0002 (1 Sep 2001)

[2] Sullivan, Andy. “Computer Virus Costs Reach $10.7 Billion This Year.” Reuters. 6 Nov 2001
URL: http://biz.yahoo.com/rf/011106/n06338921_6.html (6 Nov 2001)

[3] Slade, Robert. “The History of Computer Viruses.” 1992
URL: http://www.bocklabs.wisc.edu/~janda/sladehis.html (6 Nov 2001)

[4] Paquette, Jeremy. “A History of Viruses.” Security Focus. 17 Jul 2000
URL: http://www.securityfocus.com/infocus/1286 (6 Nov 2001)

[5] Fitzgerald, Nick. “Frequently Asked Questions on Virus-L/comp.virus”. Safetynet. 9 Oct 1995
URL: http://www.safetynet.com/support/kbvfaq.asp (5 Nov 2001)

[6] Scheier, Robert L. “Managing the Virus Threat” Computerworld 7 May 2001
URL: http://www.computerworld.com/cwi/story/0,1199,NAV47_STO60208,00.html (5 Nov 2001)

[7] "Security Overview" Finjan Software
URL: http://www.finjan.com/mcrc/overview.cfm (5 Nov 2001)

[8] Nachenburg, Carey. "Understanding and Managing Polymorphic Viruses"
URL: http://www.norton.com/avcenter/reference/striker.pdf (5 Nov 2001)

[9] "Mobile Code Attack and Recovery" 14 April 2000
URL: http://enterprisesecurity.symantec.com/article.cfm?articleid=65&PID=na (5 Nov 2001)

[10] Mcafee Virus Information Library
URL: http://vil.nai.com/vil/default.asp (6 Nov 2001)

[11] "Achilles Shield: A Technology White Paper" June 2000
URL: http://www.indefense.com/downloads/whitepaper.pdf (5 Nov 2001)

[12] "Frequently Asked Questions" Finjan Software
URL: http://www.finjan.com/mcrc/faq.cfm (5 Nov 2001)

[13] "Virus Glossary" Mcafee
URL: http://www.mcafeeb2b.com/naicommon/avert/avert-research-center/virus-glossary.asp (5 Nov 2001)

Last Updated: May 30th, 2017

Upcoming Training

Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event