

Microsoft

The Antivirus Defense-in-Depth Guide



MICROSOFT SOLUTIONS FOR SECURITY

ISBN: 0-7356-2155-1

The Microsoft Identity and Access Management Series, Extranet Access Management paper, release 2.0

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2004 Microsoft Corporation. All rights reserved.

Microsoft, Windows, Windows NT, Windows Server, Windows XP, ActiveX, Authenticode, MS-DOS, MSN, Outlook, SharePoint, and Visual Basic are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

Acknowledgments	v
------------------------	----------

Chapter 1

Introduction	1
Overview	3
Chapter 1: Introduction	3
Chapter 2: Malware Threats	3
Chapter 3: Antivirus Defense in Depth	3
Chapter 4: Outbreak Control and Recovery	3
Audience	3
Style Conventions Used in This Guide	4

Chapter 2

Malware Threats	5
Introduction	5
The Evolution of Computer Viruses	5
What Is Malware?	7
Trojan Horses	9
Worms	10
Viruses	10
Malware Characteristics	10
Target Environments	10
Carrier Objects	11
Transport Mechanisms	11
Payloads	13
Trigger Mechanisms	15
Defense Mechanisms	16
What Is Not Malware?	17
Joke Software	17
Hoaxes	17
Scams	17
Spam	18
Spyware	18
Adware	18
Internet Cookies	19
Antivirus Software	19
A Typical “In the Wild” Malware Timeline	21
Summary	22

Chapter 3

Antivirus Defense in Depth	23
Introduction	23
Malware Threat Vectors	24
The Malware Defense Approach	25
The Defense-in-Depth Security Model	25
Client Defenses	30
Client Antivirus Protection Steps	30
Client Application Antivirus Settings	36
Server Defenses	39
Server Antivirus Protection Steps	39
The Network Defense Layer	44
Network Antivirus Configuration	45
Physical Security	50
Policies, Procedures, and Awareness	50
Security Update Policy	52
Risk-based Policies	52
Automated Monitoring and Reporting Policies	54
User and Support Team Awareness	54
Summary	58

Chapter 4

Outbreak Control and Recovery	59
Introduction	59
Step 1: Infection Confirmation	60
Infection Reporting	61
Step 2: Incident Response	65
Emergency Outbreak Control	65
Preparing for Recovery	66
Step 3: Malware Analysis	68
Examine the Operating System Elements	68
Step 4: System Recovery	82
Clean or Rebuild?	82
System Cleaning	84
Restore or Reinstall?	85
Step 5: Post Recovery Steps	89
Post Attack Review Meeting	89
Post Attack Updates	90
Summary	90

Acknowledgments

The Microsoft Solutions for Security group (MSS) would like to acknowledge and thank the team that produced *The Antivirus Defense-in-Depth Guide*. The following people were either directly responsible or made a substantial contribution to the writing, development, and testing of this solution.

Author

Richard Harrison – Content Master Ltd

Editors

John Cobb – Volt Information Sciences

Steve Wacker – Volt Information Sciences

Testers

Gaurav Singh Bora – Infosys Technologies Ltd

Balkrishnan Venkiteswaran – Infosys Technologies Ltd

Security Content Review Board

Rich Benack, Security Support Engineer – Microsoft Product Support Services (PSS)

Matt Braverman, Program Manager – Microsoft Security Business and Technology Unit (SBTU)

Martin Fallenstedt, Development Lead – Microsoft Windows Security Core

Robert Hensing, Technical Lead – Microsoft Product Support Services (PSS)

Daryl Pecelj, Senior Antivirus Technician – Microsoft IT

Randy Treit, Program Manager – Microsoft SBTU

Jeff Williams, Security Privacy Officer – Microsoft PSS (Lead Reviewer)

Program Manager

Jeff Coon – Volt Information Sciences

Reviewers (in alphabetical order)

Ken Anderson, Security Solutions Technical Account Manager – Microsoft Consulting

Ignacio Ayerbe, Director of Strategic Alliances – Panda Software

Steve Clark, Systems Design Engineer – MSS

J.P. Duan, Group Manager of Antivirus Security Response – Microsoft SBU

Marius Gheorghescu, Software Design Engineer – Microsoft SBU

Yolanda Ruiz Hervas – Panda Software

Mikko Hypponen, Director of Antivirus Research – F-Secure Corporation

Maxim Kapteijns, Senior Program Manager – Microsoft Consulting

Mady Marinescu, Development Lead – Microsoft SBU

Brian May, Systems Design Engineer – MSS

Sami Rautiainen, Antivirus Researcher – F-Secure Corporation

Anil Francis Thomas, Development Manager – Microsoft SBU

Jessica Zahn, International Program Manager – Microsoft Publications

Contributors (in alphabetical order)

Eric Cameron, SCRB Program Manager – Volt Information Sciences

Philippe Goetschel – Product Unit Manager SBU

Joanne Kennedy, Group Program Manager – MSS

Kelly McMahon, User Experience – Content Master Ltd

Jeff Newfeld, Product Unit Manager – MSS

Rob Oikawa, Architect – MSS

Adrien Ransom, Business Development Manager – Microsoft SBU

Bill Reid, Group Product Manager – MSS

Bomani Siwatu, Test Lead – MSS

1

Introduction

Although many organizations have deployed antivirus software, new viruses, worms, and other forms of *malware* (malicious software) continue to rapidly infect large numbers of computer systems. There is no single reason for this apparent contradiction, but fundamental trends are apparent from feedback Microsoft has received from IT professionals and security staff in organizations whose systems have been infected, including such comments as:

- “The user executed the attachment from their e-mail even though we’ve told them again and again that they aren’t supposed to...”
- “The antivirus software should have caught this, but the signature for this virus hadn’t been installed yet.”
- “This never should have made it through our firewall; we didn’t even realize those ports could be attacked.”
- “We didn’t know our servers needed to be patched.”

The success of recent attacks illustrates that the standard approach of deploying antivirus software to each computer in your organization may not be sufficient. Recent outbreaks have spread with alarming speed, faster than the software industry’s ability to detect, identify, and deliver antivirus tools that are capable of protecting against attack. The techniques demonstrated by the latest forms of malware have also become substantially more advanced, enabling the most recent outbreaks to evade detection and propagate. These techniques include:

- **Social engineering.** Many attacks attempt to appear as if they originated from a system administrator or official service, increasing the likelihood that end users will execute them and infect their systems.
- **Backdoor creation.** The majority of recent outbreaks have attempted to open some form of unauthorized access to already infected systems, enabling a hacker to repeatedly access the systems. This repeated access is used to infect systems with new malware, using them as “zombies” in coordinated denial of service attacks, or to run any code a hacker may wish to run.

- **E-mail address theft.** E-mail addresses harvested from infected systems are used by malware programs to forward themselves to other victims and malware authors also may collect them. Malware authors can then use the addresses to send new malware variants, barter them with other malware authors for tools or virus source code, or sell them to others interested in using them to produce spam mail.
- **Embedded e-mail engines.** E-mail is the primary means for malware propagation. Many forms of malware now embed an e-mail engine to enable the malicious code to propagate much more quickly and with less likelihood of creating unusual activity that can be easily detected. Illicit mass-mailers now exploit backdoors in infected systems to capitalize on these opportunities to use such e-mail engines. As a result, it is believed the majority of spam produced last year was sent via such infected systems.
- **Exploiting product vulnerabilities.** Malware is capitalizing more frequently on product vulnerabilities to propagate, which enables the malicious code to spread much faster.
- **Exploiting new Internet technologies.** As new Internet tools become available, malware authors quickly examine them to determine how they might exploit them. Recently, Instant Messaging and peer-to-peer (P2P) networks have become attack vectors for such efforts.

These Malware terms and techniques are discussed in detail in the following chapters of this guide.

Microsoft remains strongly committed to securing the applications that it produces and to working with the company's partners to combat malware threats. Recent Microsoft efforts to reduce the impact of these threats include:

- Working closely with antivirus vendors to form the Virus Information Alliance (VIA). Alliance members exchange technical information about newly discovered malware so they can quickly communicate target, impact, and remediation information to customers. For more information about VIA, see the Virus Information Alliance (VIA) page on Microsoft® TechNet at: www.microsoft.com/technet/security/topics/virus/via.mspx.
- Researching new security technologies such as Active Protection Technology and Dynamic System Protection to help secure the Microsoft Windows® platform. For more information about these efforts, see Bill Gates' Remarks at the RSA Conference 2004 on Microsoft.com at: www.microsoft.com/billgates/speeches/2004/02-24rsa.asp.
- Releasing Windows XP Service Pack 2 with advanced security technologies to help protect your PC against hackers, viruses, and worms. For more information on this release, see Get Ready: Windows XP Service Pack 2 on Microsoft.com at: www.microsoft.com/windowsxp/default.mspx.
- Supporting legislation to eliminate spam and working with law enforcement officials and Internet service providers (ISP) to help prosecute spam operations. For information about an alliance dedicated to this effort, see America Online, Microsoft and Yahoo! Join Forces Against Spam on Microsoft.com at: www.microsoft.com/presspass/press/2003/apr03/04-28JoinForcesAntispamPR.asp.

- Announcing the Antivirus Reward Program and working closely with law enforcement agencies to reduce these threats from malware authors. For more information about the Antivirus Reward Program, see the Microsoft Announces Antivirus Reward Program page on Microsoft.com at: www.microsoft.com/presspass/press/2003/nov03/11-05AntiVirusRewardsPR.asp.

Microsoft has produced this security guidance to help you identify all the points in your infrastructure where you should consider implementing antivirus defenses. Information on how to remedy and recover from an infection if one occurs in your environment is also provided.

Overview

The Antivirus Defense-in-Depth Guide is composed of the following chapters:

Chapter 1: Introduction

This chapter presents a brief introduction to the guide, touches on malware terms and techniques, and includes an overview of each chapter, and its intended audience.

Chapter 2: Malware Threats

This chapter defines a variety of malware and specifies what types of programs are included — and not included — in this category. Information about malware characteristics, attack vectors, and means of propagation also is provided.

Chapter 3: Antivirus Defense in Depth

This chapter details considerations Microsoft recommends to establish a comprehensive antivirus defense for your clients, servers, and network infrastructure. User policies and other general security measures that Microsoft also recommends considering for your overall security planning are also discussed.

Chapter 4: Outbreak Control and Recovery

This chapter provides a step-by-step approach to resolving malware attacks, and then recovering from them based on industry best practices and internal operations at Microsoft.

Audience

This guide is primarily intended to help IT and security staff better understand the threats that malware poses, as well as how to defend against these threats, and respond quickly and appropriately when malware attacks occur.

While this guidance details considerations for antivirus defense that cover a wide variety of clients and servers, it is also applicable to organizations that run their

entire business on a single server. Each of the defense considerations is intended to protect your environment against a threat posed by some type of malware attack, thus making them relevant to any organization of any size. Some of the recommended measures, such as systems monitoring and management, may go beyond the scope or need of some organizations. However, the team that produced this guide firmly believes that it is in your interest to carefully reviewed them nonetheless to better understand the nature of the risks that malware poses to computer systems around the world today.

Style Conventions Used in This Guide

The following table notes the style conventions that are used in *The Antivirus Defense-in-Depth Guide*.

Table 1.1: Style Conventions

Element	Meaning
Bold	File names and user interface elements appear in bold.
<i>Italic</i> - or - <Italic>	<p>Italic is applied to characters that the user types and they may choose to change. Italic characters that appear within angled brackets represent variable placeholders where the user must supply specific values. Example: <Filename.ext> indicates that you should replace the italicized <i>filename.ext</i> with another filename that is appropriate for your configuration.</p> <p>Italic is also used to represent new terms. Example: <i>Digital identity</i> — The unique identifier and descriptive attributes of a person, group, device, or service.</p>
Screen Text font	This font defines output text that displays on the screen.
Monospace code font	This font is used to define code samples. Example: <code>public override void Install(IDictionary savedState)</code>
Monospace command font	This font is used to define commands, switches, and attributes the user types at a command prompt. Example: At the command prompt, type the following: <code>CScript SetUr1Auth.vbs</code>
%SystemRoot%	The folder in which the Windows operating system is installed.
Note	Alerts the reader to supplementary information.
Important	Alerts the reader to supplementary information that is essential to complete a task.
Caution	Alerts the reader that failure to take or avoid a specific action could result in the loss of data.
Warning	Alerts the reader that failure to take or avoid a specific action could result in physical harm to the user or hardware.

2

Malware Threats

Introduction

This chapter of *The Antivirus Defense-in-Depth Guide* provides a concise explanation of the evolution of computer viruses, from the first relatively simple viruses to the diverse assortment of malicious software or *malware* that exists today. The chapter defines an assortment of known malware types and techniques, and also provides information about malware propagation and the risks it poses to organizations of any size.

Because of the nature of this ever-evolving topic, this guide is not designed to capture and explain all malware elements and possible variations. However, it does provide a significant first step in trying to understand the nature of the various elements that comprise malware. The guidance also discusses and defines other things that are not malware, such as *spyware* (programs that conduct certain activities on a computer without obtaining appropriate consent from the user), *spam* (unsolicited e-mail), and *adware* (advertising that is integrated into software).

The Evolution of Computer Viruses

The first computer viruses were introduced in the early 1980s. These first attempts were largely experimental and relatively simple self-replicating files that would display simple taunts or jokes when executed.

Note: It should be noted that providing a definitive history of virus evolution is all but impossible. The illegal nature of malware means that it is in the interests of the perpetrators to hide the origins of the malicious code. This guidance distills the commonly accepted history of malware from virus researchers and the antivirus industry.

By 1986, the first viruses to attack Microsoft® MS-DOS® personal computers had been reported; the Brain virus was generally thought to be the first of these computer viruses. However, other firsts in 1986 included Virdem (the first file virus) and PC-Write (the first *Trojan horse*, a program that appears to be useful or harmless but that contains hidden code designed to exploit or damage the system on which it is run.) In the case of PC-Write, the Trojan horse masqueraded as a popular shareware Word Processor application of the same name.

As more people began exploring virus technology, the number of viruses, platforms being targeted, and virus complexity and diversity all began to increase substantially. Viruses focused on boot sectors for some time, and then began to infect executable files. In 1988, the first Internet *worm* (a type of malware that uses self-propagating malicious code that can automatically distribute itself from one computer to another through network connections.) appeared. The Morris Worm caused Internet communications to slow substantially. In response to this and the growing number of outbreaks, the CERT Coordination Center at: www.cert.org, was founded to help ensure the stability of the Internet by assisting in the coordination of responses to outbreaks and incidents.

In 1990, the Virus Exchange BBS went online as an exchange for virus writers to collaborate and share their knowledge. Also, the first book on virus writing was published, and the first *polymorphic* virus (commonly referred to as Chameleon or Casper) was developed. A polymorphic virus is a type of malware that uses an unlimited number of encryption routines to prevent detection. Polymorphic viruses have the ability to change themselves each time they replicate, which makes them difficult to detect by *signature*-based antivirus software programs that are designed to “recognize” viruses. Shortly thereafter, Tequila, the first major polymorphic virus attack, was released. Then in 1992, the first polymorphic virus engine and virus writing toolkits emerged.

Since then, viruses have become more sophisticated: viruses started accessing e-mail address books and sending themselves to contacts; macro viruses attached themselves to various office-type application files to and attack them; and viruses written specifically to exploit operating system and application vulnerabilities were released. E-mail, peer-to-peer (P2P) file-sharing networks, Web sites, shared drives, and product vulnerabilities are all exploited for virus replication and attack. *Backdoors* (secret or hidden network entry points introduced by malware) are created on infected systems to enable virus writers, or *hackers*, to return and run whatever software they choose. A hacker in the context of this guidance is a programmer or computer user who attempts illegal access to a computer system or network. Malware is discussed in detail in the next section of this chapter.

Some viruses come with their own embedded e-mail engines that enable an infected system to propagate the virus directly via e-mail, bypassing any settings in the user’s e-mail client or server. Virus writers have also begun carefully architecting

their attacks and using social engineering to develop e-mail messages with an authentic “look and feel.” This approach seeks to engage users’ trust to open the attached virus file, and dramatically increases the likelihood of a large-scale infection.

Throughout this malware evolution, antivirus software has continued to evolve as well. However, the majority of current antivirus software is almost entirely reliant on virus *signatures*, or the identifying characteristics of malicious software to identify potentially harmful code. An opportunity still exists between the initial release of a virus and the time when its signature files are broadly distributed by antivirus vendors. As a result, many viruses released today demonstrate a dramatically rapid infection rate in the first few days, and are then followed by a sharp decline once the signature files are distributed to counteract them.

What Is Malware?

This guide uses the term *malware* (an abbreviation of the phrase “malicious software”) as a collective noun to refer to viruses, worms, and Trojan horses that intentionally perform malicious tasks on a computer system.

So what exactly is a computer virus or a worm? How are these different from Trojan horses? And will antivirus applications only work against worms and Trojan horses or just viruses?

All these questions stem from the confusing and often misrepresented world of malicious code. The significant number and variety of existing malicious code makes it difficult to provide a perfect definition of each malware category.

For general antivirus discussions, the following simple definitions of malware categories apply:

- **Trojan horse.** A program that appears to be useful or harmless but that contains hidden code designed to exploit or damage the system on which it is run. Trojan horse programs are most commonly delivered to users through e-mail messages that misrepresent the program’s purpose and function. Also called Trojan code. A Trojan horse does this by delivering a malicious *payload* or task when it is run.
- **Worm.** A worm uses self-propagating malicious code that can automatically distribute itself from one computer to another through network connections. A worm can take harmful action, such as consuming network or local system resources, possibly causing a denial of service attack. Some worms can execute and spread without user intervention, while others require users to execute the worm code directly in order to spread. Worms may also deliver a payload in addition to replicating.

- **Virus.** A virus uses code written with the express intention of replicating itself. A virus attempts to spread from computer to computer by attaching itself to a host program. It may damage hardware, software, or data. When the host is executed, the virus code also runs, infecting new hosts and sometimes delivering an additional payload.

For the purpose of this guide, a *payload* is a collective term for the actions that a malware attack performs on the computer once it has been infected. These definitions of the various categories of malware make it possible to illustrate the differences between them in a simple flowchart. The following figure illustrates the elements that help to determine if a program or script falls into one of these categories:

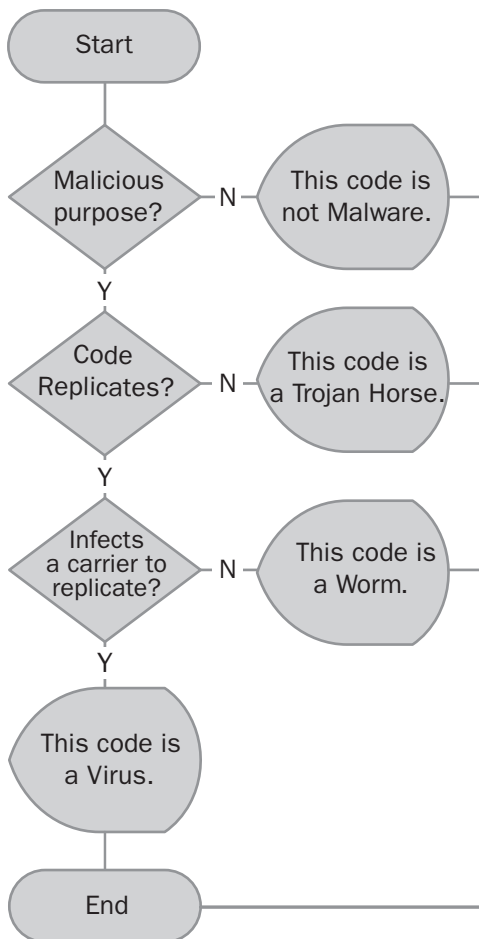


Figure 2.1

A malicious code decision tree

This figure makes it possible to distinguish between each of the common malicious code categories for the purposes of this guide. However, it is important to understand that a single attack may introduce code that fits into one or more of these categories. These types of attack (referred to as *blended threats* that consists of more than one type of malware using multiple attack vectors) can spread at rapid rates. An *attack vector* is a route that malware can use to mount an attack. For these reasons, blended threats can be especially difficult to defend against

In the following sections a more detailed explanation of each malware category is provided to help illustrate some of the key elements of each.

Trojan Horses

A Trojan horse is not considered a computer virus or worm because it does not propagate itself. However, a virus or worm may be used to copy a Trojan horse on to a target system as part of the attack payload, a process referred to as *dropping*. The typical intent of a Trojan horse is to disrupt the user's work or the normal operations of the system. For example, the Trojan horse may provide a backdoor into the system for a hacker to steal data or change configuration settings.

There are two other terms that are often used when referring to Trojan horses or Trojan-type activities that are identified and explained as follows:

- **Remote Access Trojans.** Some Trojan horse programs allow the hacker or data thief to control a system remotely. Such programs are called *Remote Access Trojans* (RATs) or backdoors. Examples of RATs include Back Orifice, Cafeene, and SubSeven.

For a detailed explanation of this type of Trojan horse, see the article "Danger: Remote Access Trojans" on Microsoft TechNet at: www.microsoft.com/technet/security/topics/virus/virusrat.mspx.

- **Rootkits.** These are collections of software programs that a hacker can use to gain unauthorized remote access to a computer and launch additional attacks.. These programs may use a number of different techniques, including monitoring keystrokes, changing system log files or existing system applications, creating a backdoor into the system, and starting attacks against other computers on the network. Rootkits are generally organized into a set of tools that are tuned to specifically target a particular operating system. The first rootkits were identified in the early 1990s, and at that time the Sun and Linux operating systems were the main targets. Currently, rootkits are available for a number of operating systems, including the Microsoft® Windows® platform.

Note: Be aware that RATs and some of the tools that comprise rootkits may have legitimate remote control and monitoring uses. However, the security and privacy issues that these tools can introduce raise the overall risk to the environments in which they are used.

Worms

If the malicious code replicates it is not a Trojan horse, so the next question to address in order to more clearly define the malware is: “Can the code replicate without the need for a carrier?” That is, can it replicate without the need to infect an executable file? If the answer to this question is “Yes,” the code is considered to be some form of worm.

Most worms attempt to copy themselves onto a host computer and then use the computer’s communication channels to replicate. For example, the Sasser worm relies on a service vulnerability to initially infect a system, and then uses the infected system’s network connection to attempt to replicate. If you have installed the latest security updates (to stop the infection), or enabled the firewalls in your environment to block the network ports the worm uses (to stop the replication), the attack will fail. In the case of Windows XP, once Service Pack 2 has been applied both the infection and replication methods are blocked. This is because the service vulnerability has been removed and the Windows firewall is enabled by default. Additionally, if the **Automatic Updates** option is set to **Automatic (recommended)** any future issues will be addressed as the updates become available.

Viruses

If the malicious code adds a copy of itself to a file, document, or boot sector of a disk drive in order to replicate it is considered a virus. This copy may be a direct copy of the original virus or it may be a modified version of the original. See the “Defense Mechanisms” section later in this chapter for more details. As mentioned earlier, a virus will often contain a payload that it may drop on a local computer, such as a Trojan horse, which will then perform one or more malicious acts, such as deleting user data. However, a virus that only replicates and has no payload is still a malware problem because the virus itself may corrupt data, take up system resources, and consume network bandwidth as it replicates.

Malware Characteristics

The various characteristics that each category of malware can exhibit are often very similar. For example, a virus and a worm may both use the network as a transport mechanism. However, the virus will look for files to infect while the worm will simply attempt to copy itself. The following section explains the typical characteristics of malware.

Target Environments

As malware attempts to attack a host system, there may be a number of specific components that it requires before the attack can succeed. The following are typical examples of what malware may require to attack the host:

- **Devices.** Some malware will specifically target a device type, such as a personal computer, an Apple Macintosh computer, or even a Personal Digital Assistant (PDA), although it should be noted that PDA malware is currently rare.
- **Operating systems.** Malware may require a particular operating system to be effective. For example, the CIH or Chernobyl virus of the late 1990s could only attack computers running Microsoft Windows® 95 or Windows® 98.
- **Applications.** Malware may require a particular application to be installed on the target computer before it can deliver a payload or replicate. For example, the LFM.926 virus of 2002 could only attack if Shockwave Flash (.swf) files could execute on the local computer.

Carrier Objects

If the malware is a virus, it will attempt to target a carrier object (also known as a host) to infect it. The number and type of targeted carrier objects varies widely among malware, but the following list provides examples of the most commonly targeted carriers:

- **Executable files.** This is the target of the “classic” virus type that replicates by attaching itself to a host program. In addition to typical executable files that use the .exe extension, files with extensions such as the following can also be used for this purpose: .com, .sys, .dll, .ovl, .ocx, and .prg.
- **Scripts.** Attacks that use scripts as carriers target files that use a scripting language such as Microsoft Visual Basic® Script, JavaScript, AppleScript, or Perl Script. Extensions for files of this type include: .vbs, .js, .wsh, and .pl.
- **Macros.** These carriers are files that support a macro scripting language of a particular application such as a word processor, spreadsheet, or database application. For example, viruses can use the macro languages in Microsoft Word and Lotus Ami Pro to produce a number of effects, ranging from mischievous (switching words around in the document or changing colors) to malicious (formatting the computer’s hard drive).
- **Boot sector.** Specific areas of computer disks (hard disks and bootable removable media) such as the master boot record (MBR) or DOS boot record can also be considered carriers because they are capable of executing malicious code. Once a disk is infected, replication is achieved if it is used to start other computer systems.

Note: If the virus targets both files and boot sectors for infection it may be referred to as a *multipartite* virus.

Transport Mechanisms

An attack can use one or many different methods to try and replicate between computer systems. This section provides information about a few of the more common transport mechanisms malware uses.

- **Removable media.** The original and probably the most prolific transmitter of computer viruses and other malware (at least until recently) is file transfer. This mechanism started with floppy disks, then moved to networks, and is now finding new media such as Universal Serial Bus (USB) devices and Firewire. The rate of infection is not as rapid as with network-based malware, yet the threat is ever present and hard to eradicate completely because of the need to exchange data between systems.
- **Network shares.** Once computers were provided a mechanism to connect to each other directly via a network, malware writers were presented with another transport mechanism that had the potential to exceed the abilities of removable media to spread malicious code. Poorly implemented security on network shares produces an environment where malware can replicate to a large number of computers connected to the network. This has largely replaced the manual method of using removable media.
- **Network scanning.** Malware writers use this mechanism to scan networks for vulnerable computers or randomly attack IP addresses. For example, the mechanism can send an exploit packet using a specific network port to a range of IP addresses with the aim of finding a vulnerable computer to attack.
- **Peer-to-peer (P2P) networks.** In order for P2P file transfers to occur, a user must first install a client component of the P2P application that will use one of the network ports that are allowed through the organization's firewall, such as port 80. The applications use this port to get through the firewall and transfer files directly from one computer to another. These applications are readily available on the Internet, and they provide a transport mechanism that malware writers can use directly to help spread an infected file onto a client's hard disk.
- **E-mail.** E-mail has become the transport mechanism of choice for many malware attacks. The ease with which hundreds of thousands of people can be reached via e-mail without the need for malware perpetrators to leave their computers has made this a very effective transport. It has been relatively simple to trick users into opening e-mail attachments (using social engineering techniques). Therefore, many of the most prolific malware outbreaks have used e-mail as their transport mechanism. There are two basic types of malware that use e-mail as a transport:
 - **Mailer.** This type of malware mails itself to a limited number of e-mail addresses, either by using mail software installed on the host (for example, Microsoft Outlook® Express), or using its own built-in Simple Mail Transfer Protocol (SMTP) engine.
 - **Mass mailer.** This type of malware searches the infected computer for e-mail addresses, and then mass mails itself to those addresses, using either mail software installed on the host or its own built-in SMTP engine.

- **Remote exploit.** Malware may attempt to exploit a particular vulnerability in a service or application in order to replicate. This behavior is often seen in worms; for example, the Slammer worm took advantage of a vulnerability in Microsoft SQL Server™ 2000. The worm generated a buffer overrun that allowed a portion of system memory to be overwritten with code that could run in the same security context as the SQL Server service. A buffer overrun is a condition that results from adding more information to a buffer than it is designed to hold. An attacker may exploit this vulnerability to take over a system. Microsoft identified and fixed this vulnerability months before Slammer was released, but few systems had been updated so the worm was able to spread.

Payloads

Once malware has reached the host machine via the transport, it will generally perform an action that is referred to as the *payload*, which can take a number of forms. Some of the more common payload types are identified in this section:

- **Backdoor.** This type of payload allows unauthorized access to a computer. It can provide full access but also may be limited to access such as enabling File Transfer Protocol (FTP) access via port 21 on the computer. If the attack was to enable Telnet, a hacker could use the infected computer as a staging area for Telnet attacks on other computers. As stated earlier, a backdoor is sometime referred to as a Remote Access Trojan.
- **Data corruption or deletion.** One of the most destructive types of payload can be malicious code that corrupts or deletes data, rendering the information on the user's computer useless. The malware writer has two choices here: the first option is to design the payload to rapidly execute. While potentially disastrous for the computer it infects, the malware design will lead to faster discovery from it and therefore limit the chance of it replicating undetected. The other option is to leave the payload on the local system (in the form of a Trojan horse) for a period (see the "Trigger Mechanisms" section later in this chapter for examples of this) to allow the malware to spread before an attempt is made to deliver the payload, and therefore alert the user to its presence.
- **Information theft.** A particularly worrying type of malware payload is one designed to steal information. If a payload can compromise the security of a host computer, it is possible for it to provide a mechanism to pass information back to the malware perpetrators. This can happen in a number of ways; for example, a transfer could be automated so that the malware simply captures local files or information such as keys the user is pressing (in the hope of obtaining a user name and password). Another mechanism is to provide an environment on the local host that allows the attacker to control the host remotely or gain access to the files on the system directly.

- **Denial of Service (DoS).** One of the simplest types of payload to deliver is a denial of service attack. A DoS attack is a computerized assault launched by an attacker to overload or halt a network service, such as a Web server or a file server. DoS attacks simply aim to render a particular service unusable for a period of time.
- **Distributed Denial of Service (DDoS).** These types of attacks typically use infected clients that are usually completely unaware of their role in such an attack. A DDoS attack is a type of denial of service attack in which an attacker uses malicious code installed on various computers to attack a single target. An attacker may use this method to have a greater effect on the target than is possible with a single attacking computer. The semantics of how an attack happens vary from attack to attack, but they usually involve sending large amounts of data to a particular host or Web site that causes it to stop responding (or become unable to respond) to legitimate traffic. This floods the available bandwidth to the victim site and effectively takes the site offline.
This type of attack can be extremely hard to defend against, because the hosts responsible for the attacks are in fact unwitting victims themselves. DDoS attacks are usually conducted by *bots* (programs that perform repetitive tasks), such as Internet relay chat (IRC) *Eggdrop* bots, which a hacker can use to control “victim” computers via an IRC channel. Once those computers are under the control of the hacker they become *zombies* that can affect a target on command from the attacker without the knowledge of the computers’ owners.

Both DoS and DDoS approaches can involve a number of different attack techniques, including:

- **System shutdowns.** If malware is able to shut down or crash the host system, it can succeed at disrupting one or more services. Attacking the host system requires the malware to find a weakness in an application or the operating system that can cause the system to shut down.
- **Bandwidth flooding.** Most services provided to the Internet are linked through a network connection of limited bandwidth that connects them to their clients. If a malware writer can deliver a payload that fills this bandwidth with false network traffic, it is possible to produce a DoS simply by stopping the clients from being able to connect directly to the service.
- **Network DoS.** This type of payload attempts to overload the resources available to the local host. Resources such as microprocessor and memory capacity have been overrun by *SYN flood* attacks, where an attacker uses a program to send a flood of TCP SYN requests to fill the pending connection queue on the server and deny legitimate network traffic to and from the host. *E-mail bomb* attacks also fill up storage resources to create a DoS attack in which an excessively large amount of e-mail data is sent to an e-mail address in an attempt to disrupt the e-mail program or to prevent the recipient from receiving further legitimate messages.

- **Service disruption.** This type of payload also can cause a DoS. For example, if an attack on a Domain Name System (DNS) server disables the DNS service this DoS attack technique would have been achieved. However, all other services on the system may remain unaffected.

Trigger Mechanisms

Trigger mechanisms are a characteristic of malware that the malicious software uses to initiate replication or payload delivery. Typical trigger mechanisms include the following:

- **Manual execution.** This type of trigger mechanism is simply the execution of the malware conducted directly by the victim.
- **Social engineering.** Malware will often use some form of social engineering to help trick a victim into manually executing the malicious code. The approach may be relatively simple, such as those used in mass mailing worms where the social engineering element focuses on selecting text in the subject field of the e-mail message that is most likely to be opened by a potential victim. Malware writers may also use e-mail *spoofing* to attempt to trick the victim into believing an e-mail is from a trusted source. Spoofing is the act of impersonating a Web site or data transmission to make it appear genuine. For example, the original Dumaru worm first seen in 2003 modified the **From:** field of e-mails to falsely claim it was sent from security@microsoft.com. (See the “Hoaxes” section in the next section of this chapter for more details on this characteristic).
- **Semi-automatic execution.** This type of trigger mechanism is started initially by a victim and then automatically executed from that point on.
- **Automatic execution.** This type of trigger mechanism requires no manual execution at all. The malware executes its attack without the need for a victim to run any malicious code on the target computer.
- **Time bomb.** This type of trigger mechanism performs an action after a certain period. This period may be a delay from the first execution of the infection or some pre-ordained date or date range. For example, the MyDoom.B worm would only start its payload routines against the Microsoft.com Web site on February 3, 2004, and against the SCO Group Web site on February 1, 2004. It would then stop all replication on March 1, 2004, although the time bomb’s backdoor component would still stay active after this time.
- **Conditional.** This type of trigger mechanism uses some predetermined condition as the trigger to deliver its payload. For example, a renamed file, a set of keystrokes, or an application starting up. Malware that uses this type of trigger is sometimes referred to as a *logic bomb*.

Defense Mechanisms

Many malware examples use some kind of defense mechanism to help reduce the likelihood of detection and removal. The following list provides examples of some of these techniques that have been used:

- **Armor.** This type of defense mechanism employs some technique that tries to foil analysis of the malicious code. Such techniques include detecting when a debugger is running and trying to prevent it from working correctly, or adding lots of meaningless code to make it difficult to determine the purpose of the malicious code.
- **Stealth.** Malware uses this technique to hide itself by intercepting requests for information and returning false data. For example, a virus may store an image of the uninfected boot sector and display it whenever an attempt is made to view the infected boot sector. The oldest known computer virus, called “Brain,” used this technique in 1986.
- **Encrypting.** Malware that uses this defense mechanism encrypts itself or the payload (and sometimes even other system data) to prevent detection or data retrieval. Encrypted malware contains a static decryption routine, an encryption key, and the encrypted malicious code (which includes an encryption routine). When executed, the malware uses the decryption routine and key to decrypt the malicious code. The malware then creates a copy of its code and generates a new encryption key. It uses that key and its encryption routine to encrypt the new copy of itself, adding the new key with the decryption routine to the start of the new copy. Unlike polymorphic viruses, encrypting malware always uses the same decryption routines, so although the key value (and thus the encrypted malicious codes signature) usually changes from infection to infection, antivirus software can search for the static decryption routine to detect malware that uses this defense mechanism.
- **Oligomorphic.** Malware that exhibits this characteristic uses encryption as a defense mechanism to defend itself and is able to change the encryption routine only a fixed number of times (usually a small number). For example, a virus that can generate two different decryption routines would be classified as *oligomorphic*.
- **Polymorphic.** Malware of this type uses encryption as a defense mechanism to change itself to avoid detection, typically by encrypting the malware itself with an encryption routine, and then providing a different decryption key for each mutation. Thus, *polymorphic* malware uses an unlimited number of encryption routines to prevent detection. As the malware replicates, a portion of the decryption code is modified. Depending on the specific malware code, the payload or other actions performed may or may not use encryption. Typically, there is a *mutation engine*, which is a self contained component of the encrypting malware that generates randomizes encryption routines. This engine and the malware are then both encrypted, and the new decryption key is passed along with them.

What Is Not Malware?

A variety of threats exist that are not considered malware because they are not computer programs written with malicious intent. However, these threats can still have both security and financial implications for an organization. For these reasons, you may wish to understand the threats they represent to your organization's IT infrastructure and the productivity of your IT users.

Joke Software

Joke applications are designed to produce a smile or, at worst, a waste of someone's time. These applications have existed for as long as people have been using computers. Because they were not developed with malicious intent and are clearly identified as jokes, they are not considered malware for the purposes of this guide. There are numerous examples of joke applications, producing everything from interesting screen effects to amusing animations or games.

Hoaxes

Generally, it is easier to trick someone into doing something for you than it is to write software that does it without their knowledge. Therefore, a large number of hoaxes are seen in the IT community.

Like some other forms of malware, a *hoax* uses social engineering to attempt to trick computer users into performing some act. However, in the case of a hoax there is no code to execute; the hoaxer is usually simply trying to trick the victim. Hoaxes have taken many forms over the years. However, a particularly common example is an e-mail message that claims a new virus type has been discovered and to warn your friends by forwarding the message. These hoaxes waste peoples time, take up e-mail server resources, and consume network bandwidth.

Scams

Virtually every form of communication has been used, at one time or another, by criminals in an attempt to trick people into acts that will provide the criminal some financial gain. The Internet, Web sites, and e-mail are no exception. An e-mail message that attempts to trick the recipient into revealing personal information that can be used for unlawful purposes (such as bank account information) is a common example. One particular type of a scam has become known as *phishing* (pronounced "fishing," and is also referred to as *brand spoofing* or *carding*).

Examples of phishing include cases in which senders mimic well-known companies such as eBay to try and gain access to user account information. Phishing scams often use a Web site that copies the look of a company's official Web site. E-mail is used to redirect the user to the fake site and trick them into entering their user account information, which is saved and used for unlawful purposes. These types of cases should be handled seriously and reported to local law enforcement authorities.

Spam

Spam is unsolicited e-mail generated to advertise some service or product. This phenomenon is generally considered a nuisance, but spam is not malware. However, the dramatic growth in the number of spam messages being sent is a problem for the infrastructure of the Internet that results in lost productivity for employees who are forced to wade through and delete such messages every day.

The source for the term spam is disputed, but regardless of its origin there is no doubt that spam has become one of the most persistent irritations in Internet-based communications. Many consider spam to be so significant an issue that it now threatens the health of e-mail communications around the world. However, it should be noted that except for the load endured by e-mail servers and anti-spam software, spam is not actually capable of replicating or threatening the health and operation of an organization's IT systems.

Malware has often been used by spam originators (so called *spammers*) to install a small SMTP e-mail server service on a host computer, which is then used to forward spam messages to other e-mail recipients.

Spyware

This type of software is sometimes referred to as *spybot* or *tracking software*. *Spyware* uses other forms of deceptive software and programs that conduct certain activities on a computer without obtaining appropriate consent from the user. These activities can include collecting personal information, and changing Internet browser configuration settings. Beyond being an annoyance, spyware results in a variety of issues that range from degrading the overall performance of your computer to violating your personal privacy.

Web sites that distribute spyware use a variety of tricks to get users to download and install it on their computers. These tricks include creating deceptive user experiences and covertly bundling spyware with other software users might want, such as free file sharing software.

Adware

Adware is often combined with a host application that is provided at no charge as long as the user agrees to accept the adware. Because adware applications are usually installed after the user has agreed to a licensing agreement that states the purpose of the application, no offense is committed. However, pop-up advertisements can become an annoyance, and in some cases degrade system performance. Also, the information that some of these applications collect may cause privacy concerns for users who were not fully aware of the terms in the license agreement.

Note: While the terms *spyware* and *adware* are often used interchangeably, it is only unauthorized adware that is on a par with spyware. Adware that provides users appropriate notice, choice, and control is not deceptive and should not be classified as spyware. You should also note a spyware application that claims to perform a particular function, while it is in fact doing something else, is acting like a Trojan horse.

Internet Cookies

Internet cookies are text files that are placed on a user's computer by Web sites that the user visits. Cookies contain and provide identifying information about the user to the Web sites that place them on the user computer, along with whatever information the sites want to retain about the user's visit.

Cookies are legitimate tools that many Web sites use to track visitor information. For example, a user might shop for an item in an online store, but once he or she has placed the item in their online shopping cart, they may want to move to another Web site for some reason. The store can choose to save the information about what products were in the shopping cart in a cookie on the user's computer so that when the user returns to the site, the item is still in the shopping cart and ready for the user to buy if he or she wishes to complete the sale.

Web site developers are only supposed to be able to retrieve information stored in the cookies they created. This approach should ensure user privacy by preventing anyone other than the developers of these sites from accessing the cookies left on the users' computers.

Unfortunately, some Web site developers have been known to use cookies to gather information without the user's knowledge. Some may deceive users or omit their policies. For example, they may track Web surfing habits across many different Web sites without informing the user. The site developers can then use this information to customize the advertisements the user sees on a Web site, which is considered an invasion of privacy. It is difficult to identify this form of targeted advertising and other forms of "cookie abuse," which makes it difficult to decide if, when, and how to block them from your system. In addition, the acceptable level of shared information varies among computer users, making it difficult to create an "anti-cookie" program that will meet the needs of all of the computer users in your environment.

Antivirus Software

Antivirus software is specifically written to defend a system against the threats that malware presents. Microsoft strongly recommends using antivirus software because it will defend your computer systems against all forms of malware, not just viruses.

There are a number of techniques that antivirus software uses to detect malware. This section discusses how some of these techniques work, including:

- **Signature scanning.** The majority of antivirus software programs currently use this technique, which involves searching the target (host computer, disk drive, or files) for a pattern that could represent malware. These patterns are generally stored in files referred to as signature files, which are updated by the software vendors on a regular basis to ensure the antivirus scanners recognize as many known malware attacks as possible. The main problem with this technique is that the antivirus software must already be updated to counteract the malware before the scanner can recognize it.
- **Heuristic scanning.** This technique attempts to detect both new and known forms of malware by looking for general malware characteristics. The primary advantage of this technique is that it does not rely on signature files to identify and counteract malware. However, heuristic scanning does have a number of specific problems, including:
 - **False positives.** This technique uses general characteristics, and is therefore prone to reporting legitimate software as malware if the characteristic is similar in both cases.
 - **Slower scanning.** The process of looking for characteristics is more difficult for the software to achieve than looking for a known malware pattern. For this reason, heuristic scanning can take longer than signature scanning.
 - **New characteristics may be missed.** If a new malware attack presents a characteristic that has not been previously identified, the heuristic scanner is likely to miss it until it is updated.
- **Behavior blocking.** This technique focuses on the behavior of a malware attack rather than the code itself. For example, if an application attempts to open a network port, a behavior blocking antivirus program could detect this as typical malware activity, and then flag the behavior as a possible malware attack.

Many antivirus vendors are now using a mixture of these techniques in their antivirus solutions in an attempt to improve the overall protection level of their customers' computer systems.

Antivirus software is available from a variety of Microsoft partners. For a complete and up-to-date list, see the Microsoft Antivirus Partners page on Microsoft.com at: www.microsoft.com/security/partners/antivirus.asp.

A Typical “In the Wild” Malware Timeline

A pattern has emerged to define the lifetime of new malware attacks that are available on public networks or when the malware goes *into the wild*. A review of this pattern can help you understand the risk new malware attacks pose after they are released.

A new timeline begins when malware is first developed and ends when all traces of it are removed from monitored networks. The timeline stages are defined as follows:

1. **Conceive.** Malware development often starts when a new method of attack or exploit is suggested and then shared among hacker communities. Over time these methods are discussed or explored until an approach is discovered that can be developed into an attack.
2. **Develop.** Malware creation used to require an understanding of both computer assembly language and the intricate working of the system being attacked. However, the advent of toolkits and Internet chat rooms has made it possible for almost anyone with malicious intent to create malware.
3. **Replicate.** After new malware has been developed and released into the wild, it typically has to replicate to potential host devices for some time before it can perform its primary function or deliver its payload.

Note: Although there are tens of thousands of known malware programs, only a tiny fraction currently exist in the wild. The vast majority of malware programs are never released to the public, and are often referred to as *Zoo viruses*.

4. **Deliver payload.** After malware has successfully infected a host it may deliver a payload. If the code has a conditional trigger for its payload, this stage is the point when the conditions for the delivery mechanism are met. For example, some malware payloads are triggered when a user performs a certain action or when the clock on the host machine reaches a particular date. If the malware has a direct action trigger it will simply start to deliver the payload at the point when the infection is complete. For example, in the case of data logging payloads the malware program will simply start recording the required data.
5. **Identify.** At this point in the timeline the malware is identified by the antivirus communities. In the vast majority of cases this step will occur before stage 4 or even before stage 3, but not always.
6. **Detect.** After the threat has been identified, antivirus software developers need to analyze the code to determine a reliable detection method. Once they have determined the method, they then update the antivirus signature files to allow existing antivirus applications to detect the new malware. The length of time this process takes is crucial in helping to control an outbreak.

7. **Removal.** After the update is available to the public, it is the responsibility of antivirus application users to apply the update in a timely manner to protect their computers against the attack (or to clean systems that are already infected).

Note: Failure to update local signature files in a timely manner can lead to the high-risk scenario of users believing they are protected by their antivirus product when in fact they are not.

As more users update their antivirus software the malware will slowly become less of a threat. This process rarely removes all instances of the malware in the wild, because some computers connected to the Internet with little or no antivirus protection remain in which the malware can reside. However, the threat from the attack as a whole is lessened.

Although this timeline repeats for each newly developed malware attack, it is not typical of all attacks. Many attacks are simply modified versions of an original portion of malware code. So the basic code and approach are the same, but small changes are made to help the attack avoid detection and therefore removal. Typically, a successful malware attack will spawn a number of revisions over the following weeks and months. This situation leads to a type of “arms race” in which malware writers attempt to avoid detection for their own gain whether the gain is for financial purposes, notoriety, or simply curiosity. The antivirus defenses are again updated, patched or changed as needed to mitigate the renewed threat.

Summary

Malware is a complex and constantly evolving area of computer technology. Of all the problems that are encountered in IT, few are as prevalent and costly as malware attacks and the associated costs of dealing with them. Understanding how they work, how they evolve over time, and the attack vectors that they exploit can help you deal with the issue proactively. And this in turn can provide you with a more efficient and effective reactive process when they do affect you or your organization.

As malware uses so many techniques to create, distribute, and exploit computer systems, it can be difficult to see how any system can be made secure enough to withstand such attacks. However, once the risks and vulnerabilities are understood it is possible to manage your system in a manner that makes the possibility of a successful attack highly unlikely.

The next step is to analyze the risks at various points in your IT infrastructure to design a suitable defense, which is the subject of the following chapter. Designing an effective recovery plan is the subject of the final chapter in this guide.

3

Antivirus Defense in Depth

Introduction

All organizations should develop an antivirus solution that will provide a high level of protection. However, many organizations still become infected, even after installing antivirus software. This guide proposes a different approach to the malicious software, or *malware*, problem. As with network security design, Microsoft recommends a defense-in-depth approach to antivirus solution design in order to help ensure that the design safeguards your organization adopts will be reliably maintained.

Such an approach is vital to the computer security of your organization, because unfortunately, regardless of how many useful features or services a computer system provides, someone (for whatever reason) will try to find a vulnerability to exploit for malicious purposes.

Working with consultants and systems engineers who have implemented Microsoft® Windows Server™ 2003, Windows® XP Professional, and Windows® 2000 in a variety of environments has helped to establish the latest best practices for securing clients and servers that run these operating systems against malware. This chapter provides you with this information.

This chapter also provides guidance to help you use a defense-in-depth approach when designing an antivirus security solution for your organization. The goal of this approach is to ensure that you understand each layer of the model and the specific threats that correspond to each layer so that you can use this information when implementing your antivirus defenses.

Note: Microsoft recommends including some of the steps in this guidance in your organization's general security procedures and policies. Where these occur, the guidance identifies them as a requirement for the security team in your organization to further define.

Important: If you suspect your organization is currently experiencing an attack, refer to Chapter 4, "Outbreak Control and Recovery," in this guide before reading this chapter.

If you are reading this guide after having experienced and recovered from a malware attack, the information provided is designed to help you prevent a recurrence and better understand how the previous attack took place.

Malware Threat Vectors

There are a number of methods through which malware can compromise an organization. These methods are sometimes referred to as *threat vectors* and represent the areas that require the most attention in your environment when designing an effective antivirus solution. The following list includes the areas in typical organizations that are subject to the most risk for malware attack:

- **External networks.** Any network that is not under the direct control of an organization should be considered as a potential source for malware. However, the Internet is by far the largest malware threat. The anonymity and connectivity that the Internet provides allows individuals with malicious intent to gain rapid and effective access to many targets to mount attacks using malicious code.
- **Guest clients.** As the use of laptops and mobile devices continues to expand in business, devices are regularly moved in and out of other organization's infrastructures. If guest clients do not have an effective antivirus defense in place, they represent a malware threat to the organization.
- **Executable files.** Any code that has the ability to execute can act as malware. This includes not only programs, but also scripts, batch files, and active objects such as Microsoft ActiveX® controls.
- **Documents.** As word processors and spreadsheet applications have become more powerful they have become targets for malware writers. Macro languages supported within many applications make them potential malware targets.
- **E-mail.** Malware writers can exploit both e-mail attachments and active Hypertext Markup Language (HTML) code within e-mail messages as attack methods.
- **Removable media.** File transfer via some form of removable media is an issue that organizations need to address as part of their antivirus defenses. Some of the more common removable media include:

- **CD-ROM or DVD-ROM discs.** The advent of cheap CD and DVD recording devices has made these media very accessible to all computer users, including those who write malware.
- **Floppy and Zip drives.** These media are becoming less prevalent due to their limited capacity and speed, but still remain risks if malware is physically able to access them.
- **USB drives.** These devices take on many forms, ranging from the classic key ring-sized device to a wrist watch. All these devices can be used to introduce malware if they can be inserted into the Universal Serial Bus (USB) port of a host.
- **Memory cards.** Digital cameras and mobile devices, such as PDAs and mobile phones, have helped establish digital memory cards. Card readers are becoming increasingly standard devices on computers, which makes it easier for users to transfer data on memory cards. Because this data is file-based, these cards can also transfer malware onto a host system.

The Malware Defense Approach

Before attempting to organize an effective defense against malware, it is essential to understand the various parts of the organization's infrastructure that are at risk and the extent of the risk to each part. Microsoft strongly recommends that you conduct a full security risk assessment before starting to design an antivirus solution. The information you need to optimize your solution design can only be obtained by completing a full security risk assessment.

For information and guidance on conducting a security risk assessment, see the *Microsoft Solution for Securing Windows 2000 Server* guide at: www.microsoft.com/technet/security/prodtech/win2000/secwin2k/default.mspx. This guide provides an introduction to the security risk management discipline (SRMD), which you can use to understand the nature of your organization's exposure to risk.

The Defense-in-Depth Security Model

Once you have discovered and documented the risks your organization faces, the next step is to examine and organize the defenses you will use to provide your antivirus solution. The defense-in-depth security model is an excellent starting point for this process. This model identifies seven levels of security defenses that are designed to ensure that attempts to compromise the security of an organization will be met by a robust set of defenses. Each set is capable of deflecting attacks at many different levels. If you are not familiar with the defense-in-depth security model, Microsoft recommends reviewing the Security Content Overview page on Microsoft TechNet at:

www.microsoft.com/technet/security/bestprac/overview.mspx.

You can also find additional information and practical design examples for this process in the Security Architecture chapter of the Windows Server System Reference Architecture guidance on TechNet at:

www.microsoft.com/technet/itsolutions/techguide/wssra/raguide/Security_Architecture_1.mspx.

The following figure illustrates the layers defined for the defense-in-depth security model:

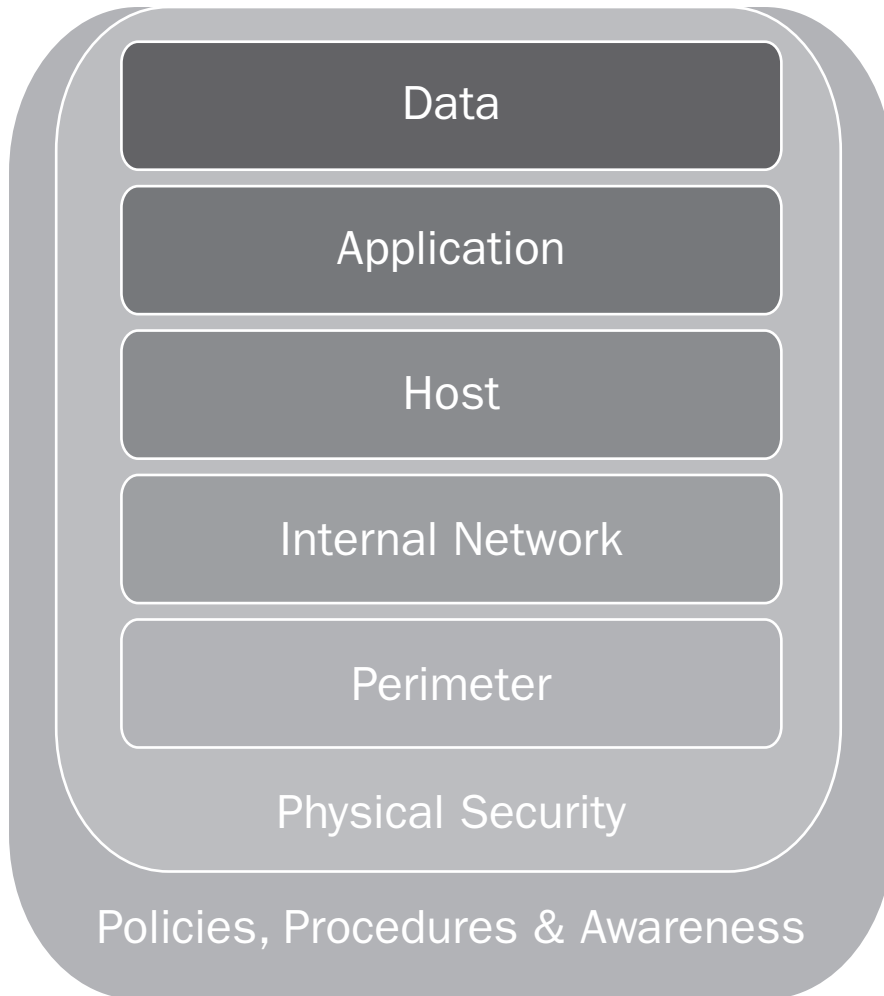


Figure 3.1
The layers of the defense-in-depth security model

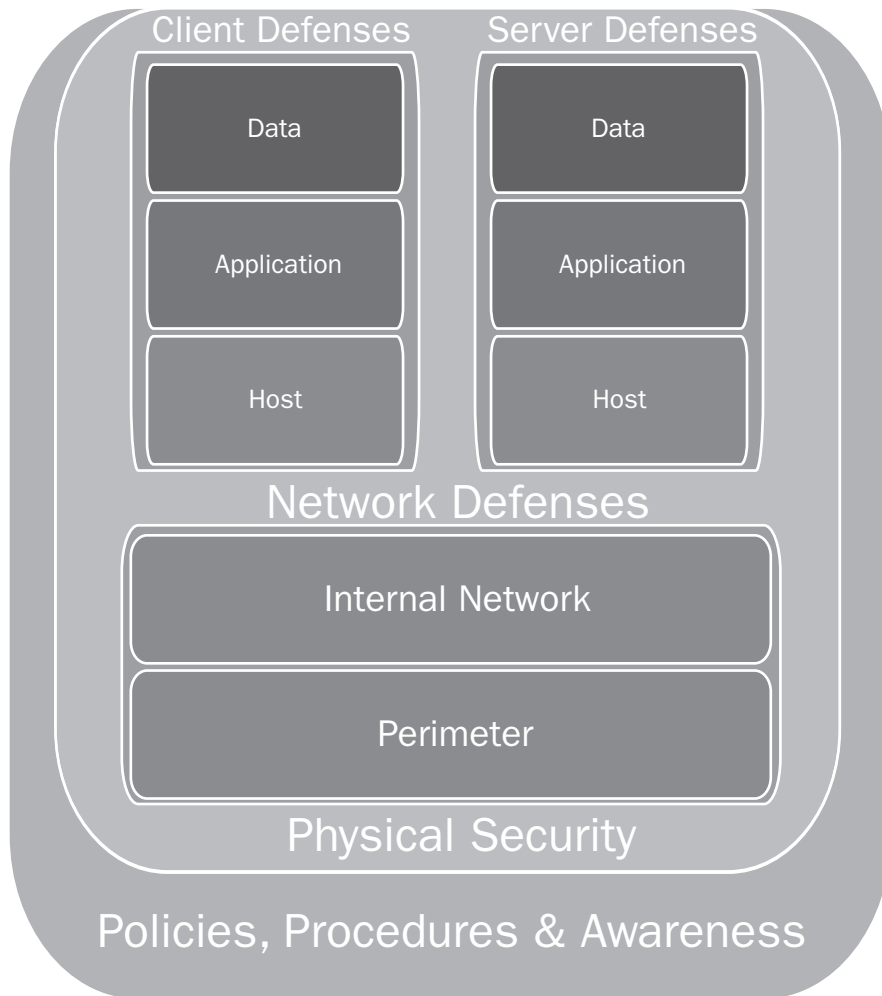
The layers in the figure provide a view of each area in your environment you should consider when designing security defenses for your network.

You can modify the detailed definitions of each layer based on your organization's security priorities and requirements. For purposes of this guidance, the following simple definitions define the layers of the model:

- **Data.** Risks at the data layer arise from vulnerabilities an attacker could potentially exploit to gain access to configuration data, organization data, or any data that is unique to a device the organization uses. For example, sensitive data such as confidential business data, user data, and private customer information stores should all be considered part of this layer. The primary concerns for the organization at this layer of the model are business and legal issues that may arise from data loss or theft, and operational issues that vulnerabilities may expose at the host or application layers.
- **Application.** Risks at the application layer arise from vulnerabilities an attacker could potentially exploit to access running applications. Any executable code a malware writer can package outside of an operating system could be used to attack a system. The primary concerns for the organization at this layer are access to the binary files that comprise applications, access to the host through vulnerabilities in the application's listening services, or inappropriate gathering of specific data from the system to pass on to someone who can use it for their own purposes.
- **Host.** This layer is typically targeted by vendors who provide service packs and hotfixes in order to address malware threats. Risks at this layer arise from attackers exploiting vulnerabilities in the services that the host or device offers. Attackers exploit these in a variety of ways to mount attacks against the system. A buffer overrun, which is a condition that results from adding more information to a buffer than it was designed to hold, is a good example. The primary concerns for an organization at this layer are preventing access to the binary files that comprise the operating system, as well as access to the host through vulnerabilities in the operating system's listening services.
- **Internal Network.** The risks to organizations' internal networks largely concern the sensitive data transmitted via networks of this type. The connectivity requirements for client workstations on these internal networks also have a number of risks associated with them.

- **Perimeter Network.** Risks associated with the perimeter network layer (also known as the DMZ, demilitarized zone, or screened subnet) arise from an attacker gaining access to wide area networks (WAN) and the network tiers that they connect. The primary risks at this layer of the model focus on available Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ports that the network uses.
- **Physical Security.** Risks at the physical layer arise from an attacker gaining physical access to a physical asset. This layer encompasses all the previous layers because physical access to an asset can in turn allow access to all of the other layers in the defense-in-depth model. The primary concern at this layer of the model for organizations using antivirus systems is to stop infected files from bypassing the perimeter and internal network defenses. Attackers may attempt to do this simply by copying an infected file directly to the host computer via some physical removable media, such as a USB disk device.
- **Policies, Procedures and Awareness.** Surrounding all of the security model layers are the policies and procedures your organization needs to put in place to meet and support the requirements for each level. Finally, it is important for you to promote awareness in your organization to all interested parties. In many cases, ignorance of a risk can lead to a security breach. For this reason, training also should be an integral part of any security model.

Using the security layers of the model as the basis for your antivirus defense-in-depth approach allows you to refocus your view to optimize them into groupings for the antivirus defenses in your organization. How this optimization occurs in your organization is entirely dependent on the priorities of your organization and the specific defense applications it is using. The important point is to avoid an incomplete and weakened antivirus design by ensuring that none of the security layers are excluded from the defenses. The following figure shows a more focused antivirus defense-in-depth view:

**Figure 3.2**

Focused antivirus defense-in-depth view

The Data, Application, and Host layers can be combined into two defense strategies to protect the organization's clients and servers. Although these defenses share a number of common strategies, the differences in implementing client and server defenses are enough to warrant a unique defense approach for each.

The Internal Network and Perimeter layers can also be combined into a common Network Defenses strategy, as the technologies involved are the same for both layers. The implementation details will differ in each layer, depending on the position of the devices and technologies in the organization's infrastructure.

Client Defenses

When malware reaches a host computer, the defense systems must focus on protecting the host system and its data and stopping the spread of the infection. These defenses are no less important than the physical and network defenses in your environment. You should design your host defenses based on the assumption that the malware has found its way through all previous layers of defense. This approach is the best way to achieve the highest level of protection.

Client Antivirus Protection Steps

There are a number of approaches and technologies you can use for client antivirus configurations. The following sections provide details that Microsoft recommends for consideration.

Step 1: Reduce the Attack Surface

The first line of defense at the application layer is to reduce the attack surface of the computer. All unnecessary applications or services should be removed or disabled on the computer to minimize the number of ways an attacker could exploit the system.

You will find the default settings for Windows XP Professional services on the Default settings for services page of the Windows XP Professional Product Documentation on Microsoft.com at: www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sys_srv_default_settings.mspx.

Once the attack surface has been minimized without affecting the required functionality of the system, the primary defense to use at this layer is an antivirus scanner. The scanner's primary role is to detect and prevent an attack, and then to notify the user and perhaps the system administrators in your organization as well.

Step 2: Apply Security Updates

The sheer number and variety of client machines that may be connected to an organization's networks can make it difficult to provision a fast and reliable security update management service. Microsoft and other software companies have developed a number of tools you can use to help manage this problem. The following patch management and security update tools are currently available from Microsoft:

- **Windows Update.** For small organizations or individuals, the Windows Update service provides both a manual and an automatic process to detect and download the latest security and feature modifications for the Windows platform. Issues with this approach for some organizations include the lack of support for testing prior to deploying updates from this service, and the amount of network bandwidth that the clients may consume in the organization when downloading the same package at the same time. Information on using this service is available on the Windows Update home page at: www.windowsupdate.com.

- **Software Update Service.** This service was designed to provide a security update solution for Windows clients in the enterprise. The service addressed both of the Windows Update shortcomings for larger organizations by allowing both internal testing and distributed security update management. However the Software Update Service is being replaced by the Windows Update services which provides a broader set of functionality (see the next bullet point). Information on using this service to develop a solution for your organization is available on the Software Update Service home page on Microsoft.com at: www.microsoft.com/windowsserversystem/sus/.
- **Windows Update Services.** These services are designed to replace the Software Update Service to provide a higher level of functionality across a wider range of Microsoft software. Windows Update Services reduces the cost and risk associated with update management while providing the flexibility to address a broad range of update management scenarios. Information on using these services for your organization is available on the Windows Update Services home page on Microsoft.com at: www.microsoft.com/windowsserversystem/wus/.
- **Systems Management Server 2003.** Microsoft Systems Management Server 2003 is a complete enterprise management solution that is capable of providing comprehensive security update services and much more. For more information about this solution, see the Systems Management Server home page on Microsoft.com at: www.microsoft.com/smsserver/default.asp.

Each of these Microsoft security update tools has specific strengths and goals. The best approach is likely to use one or more of them. To help you evaluate the security update solutions for your organization, see the features comparison provided on the Choosing a Security Update Management Solution page on Microsoft.com at: www.microsoft.com/windowsserversystem/sus/suschoosing.msp.

Step 3: Enable a Host-based Firewall

The host-based or personal firewall represents an important layer of client defense that you should enable, especially on laptops that users may take outside your organization's usual physical and network defenses. These firewalls filter all data that is attempting to enter or leave a particular host computer.

Windows XP includes a simple personal firewall called the Internet Connection Firewall (ICF). Once enabled, the ICF monitors all communication aspects that pass through it. The ICF also inspects the source and destination address of each data packet it handles to ensure that each communication is allowed. For more information on ICF, see the Windows XP Help system and also the Use the Internet Connection Firewall page on Microsoft.com at: www.microsoft.com/windowsxp/pro/using/howto/networking/icf.asp.

Windows XP Service Pack 2 enables the personal firewall by default and introduces a number of significant enhancements to that firewall (now called the *Windows*

Firewall) as well as other security-oriented improvements. A service pack is a tested, cumulative set of all hotfixes, security updates, critical updates, and updates created for defects found internally since the release of a product. Service packs may also contain a limited number of customer-requested design changes or features. For information about this update for Windows XP, see the Windows XP Service Pack 2 page on Microsoft TechNet at:

www.microsoft.com/technet/prodtechnol/winxppro/maintain/winxpsp2.mspx.

Versions of Windows before Windows XP did not come with a built-in firewall. Third-party host-based firewall solutions are available that can be installed to provide firewall services on earlier versions of Windows. For information about these firewall products see the Frequently Asked Questions About Internal Firewalls page on the Microsoft Protect Your PC Web site at:

www.microsoft.com/security/protect/firewall.asp.

Step 4: Install Antivirus Software

Many companies produce antivirus applications, each of which attempts to protect the host computer with minimal inconvenience to and interaction with end users. Most of these applications have become very effective in providing this protection, but they all require frequent updates to keep up with new malware. Any antivirus solution should provide a rapid and seamless mechanism to ensure that updates to the required *signature files* for dealing with new malware or variants are delivered as soon as possible. A signature file contains information that antivirus programs use to detect malware during a scan. Signature files are designed to be regularly updated by the antivirus application vendors and downloaded to the client computer.

Note: Such updates present their own security risk, because signature files are sent from the antivirus application's support site to the host application (usually via the Internet). For example, if the transfer mechanism uses File Transfer Protocol (FTP) to obtain the file, the organization's perimeter firewalls must allow this type of access to the required FTP server on the Internet. Ensure your antivirus risk assessment process reviews the update mechanism for your organization, and that this process is secure enough to meet your organization's security requirements.

Due to rapidly changing malware patterns and techniques, some organizations have adopted an approach that recommends requiring certain "high risk" users to run more than a single antivirus package on the same computer to help minimize the risk of malware going undetected. The following user types typically fall into this category:

- Webmasters or anyone who administers content on the Internet or an intranet.
- Release lab workers or anyone who produces electronic media such as CD-ROMs.
- Development team members who create or compile compressed files or other product software.

It should be noted that running antivirus applications from a number of different application vendors on the same computer may cause problems due to interoperability issues between the antivirus applications. System issues that can result from running more than one antivirus application in your environment at the same time include:

- **Memory overhead.** Many antivirus applications use active agents that stay resident in memory, reducing the amount of available system memory.
- **System crashes or stop errors.** Such crashes and errors can be caused by antivirus applications attempting to simultaneously scan the same file.
- **Performance loss.** As antivirus applications scan files for malicious code, system performance may decrease. Scans are repeatedly performed when multiple applications are used, which may lower your system performance to an unacceptable level.
- **Loss of system access.** Antivirus applications attempting to run concurrently may cause the system to halt during startup. This problem is more common in older versions of Windows, such as Microsoft Windows® NT and Windows 9x.

For these reasons, the use of multiple antivirus applications on the same computer is not a recommended approach and should be avoided if possible.

An alternative approach to consider is to use antivirus software from different vendors for the client, server, and network defenses in the organization. This approach provides consistent scanning of these different areas of the infrastructure with different scanning engines, which should help reduce the risk to your overall antivirus defenses if a single vendor's product fails to detect an attack.

For more information about antivirus vendors, see the Microsoft Antivirus Partners on Microsoft.com at:

www.microsoft.com/security/partners/antivirus.asp.

For more information about antivirus software designed for Windows XP, see the Microsoft Windows Catalog Antivirus page on Microsoft.com at:

<http://go.microsoft.com/fwlink/?LinkId=28506>.

Step 5: Test with Vulnerability Scanners

Once you have configured a system, you should check it periodically to ensure that no security weaknesses have been left in place. To assist you with this process, a number of applications act as scanners to look for weaknesses that both malware and hackers may attempt to exploit. The best of these tools update their own scanning routines to defend your system against the latest weaknesses.

The Microsoft Baseline Security Analyzer (MBSA) is an example of a vulnerability scanner that is capable of checking for common security configuration issues. The scanner also checks to ensure that your host is configured with the latest security updates.

For more information about this free configuration tool, see the Microsoft Baseline Security Analyzer V1.2 page on TechNet at: www.microsoft.com/technet/security/tools/mbsahome.mspx.

Step 6: Use Least Privileges Policies

Another area that should not be overlooked among your client defenses is the privileges assigned to users under normal operation. Microsoft recommends adopting a policy that provides the fewest privileges possible to help minimize the impact of malware that relies on exploiting user privileges when it executes. Such a policy is especially important for users who typically have local administrative privileges. Consider removing such privileges for daily operations, and instead using the **RunAs** command to launch the required administration tools when necessary.

For example, a user who needs to install an application that requires administrator privileges could run the following setup command at a command prompt to launch the setup program with appropriate privileges:

```
runas /user:mydomain\admin "setup.exe"
```

You can also access this feature directly from Microsoft Windows Explorer, in Windows 2000 or later systems, by performing the following steps:

► To run a program with administrative privileges

1. In Windows Explorer, select the program or tool you want to open (such as a Microsoft Management Console (MMC) snap-in or Control Panel).
2. Right-click the program or tool and select **Run As**.

Note: If **Run As** does not appear as an option, press and hold the **SHIFT** key while you right-click the tool.

3. In the **Run As** dialog box, select **The following user:** option.
4. In the **User name** and **Password** boxes, type the user name and password for the administrator account you want to use.

Step 7: Restrict Unauthorized Applications

If an application is providing a service to the network, such as Microsoft Instant Messenger or a Web service, it could, in theory, become a target for a malware attack. As part of your antivirus solution, you may wish to consider producing a list of authorized applications for the organization. Attempts to install an unauthorized application on any of your client computers could expose all of them and the data they contain to a greater risk of malware attacks.

If your organization wishes to restrict unauthorized applications, you can use Windows Group Policy to restrict users' ability to run unauthorized software. How to use Group Policy has already been extensively documented, you will find

detailed information about it at the Windows Server 2003 Group Policy Technology Center on Microsoft.com at:
www.microsoft.com/windowsserver2003/technologies/management/grouppolicy/.

The specific area of Group Policy that handles this feature is called the Software Restriction Policy, which you can access through the standard Group Policy MMC snap-in. The following figure displays a Group Policy MMC screen showing the path to where you can set Software Restriction Policies for both your computers and users:

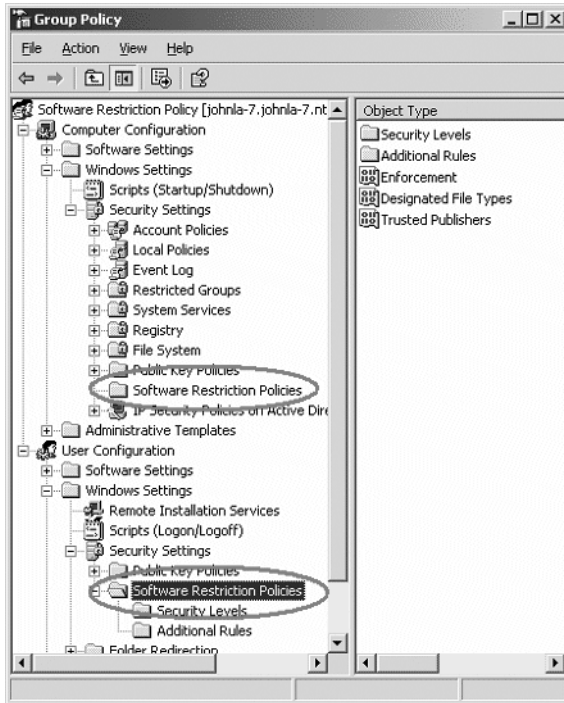


Figure 3.3

The path to the Software Restriction Policies folders in the Group Policy MMC snap-in

To access this snap-in directly from a Windows XP client, complete the following steps:

1. Click **Start** and then **Run**.
2. Type `secpol.msc`, then click **OK**.

A detailed explanation of all the setting possibilities is beyond the scope of this guide. However, the article “Using Software Restriction Policies to Protect Against Unauthorized Software” on TechNet at:

www.microsoft.com/technet/prodtechnol/winxppro/maintain/rstrplcy.mspx will provide you with step-by-step guidance on using this powerful feature of the Windows XP Professional operating system.

Warning: Group Policy is an extremely powerful technology that requires careful configuration and a detailed understanding to implement successfully. Do not attempt to change these settings directly until you are confident you are familiar with the policy settings and have tested the results on a non-production system.

Client Application Antivirus Settings

The following sections provide guidelines for configuring specific client applications that malware may target.

E-mail Clients

If malware does manage to make it past your antivirus defenses at the network and e-mail server levels, there may be a few settings that you can configure to provide additional protection for the e-mail client.

Generally, the ability of a user to open e-mail attachments directly from an e-mail message provides one of the major ways for malware to propagate on the client. If possible, consider restricting this ability in your organization's e-mail systems. If this is not possible, some e-mail clients allow you to configure additional steps that users will have to perform before they can open an attachment. For example, in Microsoft Outlook® and Outlook Express you have the ability to:

- Use Internet Explorer security zones to disable active content in HTML e-mail messages.
- Enable a setting so that users may only view e-mail messages in plain text.
- Prevent programs from sending e-mail messages without specific user approval.
- Block unsafe e-mail message attachments.

For information on how to configure these features, see the Microsoft Knowledge Base article "291387 - OLEXP: Using Virus Protection features in Outlook Express 6" at:

<http://support.microsoft.com/?kbid=291387>.

Additionally Windows XP Service Pack 2 has added extra security focused functionality to Outlook Express. For information on how Windows XP Service Pack 2 has changed the functionality of Outlook Express, see the Changes to Functionality in Microsoft Windows XP Service Pack 2 Part 4: E-mail Handling Technologies page on TechNet at:

www.microsoft.com/technet/prodtechnol/winxppro/maintain/sp2email.msp

Microsoft Outlook 2003 includes additional features to protect against malware and junk (or spam) e-mail messages. You will find information about configuring these features on the Customizing Outlook 2003 to Help Prevent Viruses page on Microsoft.com at:

www.microsoft.com/office/ork/2003/three/ch12/.

Desktop Applications

As desktop office applications have become more powerful they have also become targets for malware. Macro viruses use files created by the word processor, spreadsheet, or other macro-enabled applications to replicate themselves.

You should take steps wherever possible to ensure that the most appropriate security settings are enabled on all applications in your environment that handle these files. For information about securing Microsoft Office 2003 applications, see the Best practices for protection from viruses page on Microsoft.com at: <http://go.microsoft.com/fwlink/?LinkId=28509>.

Instant Messaging Applications

The instant messaging phenomenon has helped improve user communications across the world. Unfortunately, it has also provided another application with the potential to allow malware to enter your system. Although text messages do not pose a direct malware threat, most instant messenger clients provide additional file transfer capabilities to enhance the users' communication abilities. Allowing file transfers provides a direct route into an organization's network for potential malware attacks.

Network firewalls can block these file transfers by simply filtering the ports used for this communication. For example, Microsoft Windows and MSN® Messenger clients use a range of TCP ports between 6891 and 6900 for to transfer files, so if the perimeter firewall blocks these ports, file transfer via Instant Messenger cannot take place. However, mobile client computers will only be protected while they are on the organization's network. For this reason, you might want to configure the host-based firewall on your clients to block these ports, as well to provide protection for the mobile clients in your organization when they are outside of your network defenses.

If your organization cannot block these ports because other required applications use them or because file transfer is required, you should ensure all files are scanned for malware before being transferred. If your client workstations are not using a real-time antivirus scanner, you should configure the Instant Messaging application to automatically pass transferred files to an antivirus application for scanning as soon as the file has been received. For example, you can configure MSN Messenger to automatically scan transferred files. The following steps demonstrate how to enable this security feature:

Note: The Windows Messenger application that shipped with Windows XP does not support this feature. A real-time antivirus scanner should be used for this application.

► To scan files transferred by MSN Messenger

1. In the main MSN Messenger window, click the **Tools** menu, and then click **Options**.

2. Click the **Messages** tab.
3. Under **File Transfer**, select the **Scan for viruses using** check box.
4. Click **Browse**, select the antivirus scanning software that you are using, and then click **OK**.

Note: Finding the correct executable file to use and the command parameter to include here may require additional input from your antivirus scanning software vendor.

Once you have completed these steps, your antivirus software will automatically scan all files received via MSN Messenger on the client.

Note: Your antivirus scanning tool may require additional setup steps. Check the instructions for with your antivirus scanning software for more information.

Web Browsers

Before you download or execute code from the Internet, you want to ensure that you know that it is from a known, reliable source. Your users should not just rely on site appearance or the address of the site because both Web pages and addresses can be faked.

There are a number of different techniques and technologies that have been developed to help a user's Web browser application determine the reliability of the Web site he or she is browsing. For example, Microsoft Internet Explorer uses Microsoft Authenticode® technology to verify the identity of downloaded code. The Authenticode technology verifies that the code has a valid certificate, that the identity of the software publisher matches the certificate, and that the certificate is still valid. If all these tests pass, the chances of an attacker transferring malicious code to your system will be reduced.

Most major Web browser applications support the ability to restrict the level of automated access that is available to code that is executed from a Web server. Internet Explorer uses security zones to help restrict Web content from performing potentially damaging operations on the client. The security zones are based on the location (zone) of the Web content.

For example, if you are confident that anything downloaded within your organization's intranet is safe, you might set your clients' security settings for the local intranet zone to a low level to allow users to download content from your intranet with few or no restrictions. However, if the source of the download is in the Internet zone or the Restricted sites zone, you might want to configure the clients' security settings to a medium or high level. These settings will cause the client browsers to either prompt users with information about the content's certificate before they download it or prevent them from downloading it all.

Windows XP Service Pack 2 has added a significant number of security updates and enhancements to aid in the protection of the Web browsing experience for the user. For details of these updates, see the Changes to Functionality in Microsoft Windows XP Service Pack 2 Part 5: Enhanced Browsing Security page on TechNet at: www.microsoft.com/technet/prodtechnol/winxppro/maintain/sp2brows.mspx

For more information about security-related issues for Internet Explorer, see the Internet Explorer Security Center page on Microsoft.com at: www.microsoft.com/technet/security/prodtech/ie/.

Peer-to-Peer Applications

The advent of Internet-wide peer-to-peer (P2P) applications has made it easier than ever to find and exchange files with other people. Unfortunately, this situation has led to a number of malware attacks that attempt to use these applications to replicate files to other users' computers. Worms such as W32.HLLW.Sanker have targeted P2P applications such as Kazaa for replication purposes. There are many more malware examples that attempt to use other peer-to-peer applications, such as Morpheus and Grokster.

The security issues surrounding P2P applications have little to do with the client programs themselves. These issues instead have much more to do with the ability of these applications to provide direct routes from one computer to another through which content can be transmitted without the proper security checks.

If possible, Microsoft recommends restricting the number of clients in your organization that use these applications. You can use Windows Software Restriction policies that were discussed earlier in this chapter to help block users from running peer-to-peer applications. If this is not possible in your environment, be sure your antivirus policies take into account the greater risk the clients in your environment are exposed to because of these applications.

Server Defenses

The server defenses in your environment have a lot in common with your client defenses; both attempt to protect the same basic personal computer environment. The primary difference between the two is that there is generally a much higher expectation level placed on server defenses for reliability and performance. In addition, the dedicated roles that many servers play within an organization's infrastructure will often lead to a specialized defense solution. The information in the following sections focuses on the primary differences between server defenses and the previously discussed client defenses.

Server Antivirus Protection Steps

Server antivirus configurations vary greatly, depending on the role of the particular server and the services it is designed to provide. The process of minimizing

a server's attack surface is often referred to as *hardening*. Excellent guidance is available on hardening Windows Server 2003 when it is used in various typical roles in an organization. For more information on this topic, see the Server Security Index page on Microsoft.com at:

www.microsoft.com/security/guidance/topics/ServerSecurity.mspx.

Four of the basic antivirus steps to defend the servers in your organization are the same as those for your clients.

1. **Reduce the attack surface.** Remove unwanted services and applications from your servers to minimize their attack surface.
2. **Apply security updates.** Ensure all of your server computers are running the latest security updates, if possible. Perform additional testing as needed to ensure mission-critical servers are not adversely affected by new updates.
3. **Enable the host-based firewall.** Windows Server 2003 includes a host-based firewall you can use to reduce the attack surface on your servers, as well as remove unwanted services and applications.
4. **Test using vulnerability scanners.** Use the MBSA on Windows Server 2003 to help identify possible vulnerabilities in a server configuration. Microsoft recommends using this and other specialized vulnerability scanners to help ensure as robust a configuration as possible.

In addition to these common antivirus steps, consider using the following server-specific software as part of your overall server antivirus defenses.

General Server Antivirus Software

The primary difference between antivirus applications that are designed for client environments (such as Windows XP) and those designed for server environments (such as Windows Server 2003) has been the level of integration between the server-based scanner and any server-based services, such as messaging or database services. Many server-based antivirus applications also offer remote management capabilities to minimize the need for physical access to the server console.

Additional important issues that you should take into account when evaluating antivirus software for your server environment include:

- **CPU utilization during scanning.** In a server environment, CPU utilization is a critical component of the ability of the server to perform its primary role for the organization.
- **Application reliability.** A system crash on an important data center server has a far greater impact than a single workstation crash. Therefore, Microsoft recommends thoroughly testing all server-based antivirus applications to ensure your system reliability.
- **Management overhead.** The ability of the antivirus application to be self-managing could help reduce administrative overhead for the server management teams in your organization.

- **Application interoperability.** You should test the antivirus application with the same server-based services and applications that your production server will be running to ensure there are no interoperability issues.

For a list of antivirus applications that have been certified to work on Windows Server 2003, click the Business Solutions, Security page of the Windows Server Catalog at <http://go.microsoft.com/fwlink/?linkid=28510>.

Role-Specific Antivirus Configurations and Software

There are a number of specialized antivirus configurations, tools and applications now available for specific server roles in the enterprise. Examples of server roles that can benefit from this type of specialized antivirus defense:

- **Web servers** such as Microsoft Internet Information Services (IIS).
- **Messaging servers** such as Microsoft Exchange 2003.
- **Database servers** such as those running Microsoft SQL Server™ 2000.
- **Collaboration servers** such as those running Microsoft Windows SharePoint™ Services, and Microsoft Office SharePoint Portal Server™ 2003.

Application-specific antivirus solutions generally provide better protection and performance because they are designed to integrate with a specific service rather than try to function underneath the service at the file system level. All of the server roles discussed in this section are responsible for information that would not be accessible to an antivirus scanner working at the file system level. Information is also provided on each of these server roles, and how Microsoft recommends using specific antivirus configurations, tools, and applications with them.

Web Servers

Web servers in all types of organizations have been the target of security attacks for some time. Whether an attack comes from malware such as CodeRed or a hacker trying to deface an organization's Web site, it is important that the security settings on your Web servers are sufficiently configured to maximize your defenses against these attacks. Microsoft has produced guidance specifically for systems administrators tasked with protecting servers running IIS on the network in "Chapter 8 - Hardening IIS Servers" of the *Windows Server 2003 Security Guide* on Microsoft.com at:

www.microsoft.com/technet/security/prodtech/win2003/w2003hg/sgch08.mspx.

In addition to this guidance, there are some free tools you can download that will perform a number of security configurations automatically on IIS. For example, the IIS Lockdown Tool is available on Microsoft.com at:

www.microsoft.com/technet/security/tools/locktool.mspx.

This tool is used to tune the Web server to provide only those services required for its role, thereby reducing the attack surface of the server to any malware.

UrlScan is another security tool that restricts the types of HTTP requests that IIS will process. By blocking specific HTTP requests, UrlScan helps prevent potentially harmful requests from reaching the server. You can now cleanly install UrlScan 2.5 on servers running IIS 4.0 or later. For more information on UrlScan, see the UrlScan Security Tool page on Microsoft.com at:

www.microsoft.com/technet/security/tools/urlscan.mspx.

Messaging Servers

There are two goals to keep in mind when designing an effective antivirus solution for the e-mail servers in your organization. The first goal is to protect the servers themselves from malware. The second goal is to stop any malware from making its way through the e-mail system to the mailboxes of the users in your organization. It is important to ensure the antivirus solution you install on your e-mail servers is capable of achieving both these goals.

Generally speaking, standard file scanning antivirus solutions are not able to prevent an e-mail server from passing malware as attachments to clients. All but the most simple e-mail services store e-mail messages in a database of some type (sometimes referred to as the *message store*). A typical file scanning antivirus solution cannot access the content of such a database. In fact, a file scanning antivirus solution could possibly corrupt a message store if it is allowed to attempt scanning via a drive mapping (such as the M: drive on Exchange Server 5.5 and Exchange Server 2000).

It is important to match the antivirus solution to the e-mail solution in use. Many antivirus vendors now provide dedicated versions of their software for specific e-mail servers that are designed to scan the e-mail passing through the e-mail system for malware. Two basic types of e-mail antivirus solutions are generally available:

- **SMTP gateway scanners.** These Simple Mail Transfer Protocol (SMTP)-based e-mail scanning solutions are usually referred to as antivirus “gateway” solutions. They have the advantage of working with all SMTP e-mail services rather than being tied to a specific e-mail server product. However, these solutions are limited in some of the more advanced features they can provide due to their reliance on the SMTP e-mail protocol.
- **Integrated server scanners.** These specialized antivirus applications work directly with a particular e-mail server product. These applications do have a number of benefits. For example, they can integrate directly with advanced server features, and they are designed to use the same hardware as the e-mail server.

Microsoft Exchange provides a specific antivirus application programming interface (API) called the Virus API (VAPI), which is also referred to as the Antivirus API (AVAPI), or the Virus Scanning API (VSAPI). This API is used by specialized Exchange Server antivirus applications to help provide full messaging protection in a secure and reliable manner on Exchange e-mail servers. For more information on this API, see the Microsoft Knowledge Base article “328841 – XADM: Exchange and Antivirus Software” on Microsoft.com at:

<http://support.microsoft.com/?kbid=328841>.

Database Servers

There are four main elements to protect when considering the antivirus defenses for a database server:

- **Host.** The server or servers running the database.
- **Database services.** The various applications running on the host that provide the database service to the network.
- **Data store.** The data stored in the database.
- **Data communications.** The connections and protocols that are used between the database host and the other hosts on the network.

As the data inside the data store is not directly executable, it is generally believed that the data stores themselves do not require scanning. There are currently no major antivirus applications written specifically for data stores. However, the host, database services, and data communications elements of the database server should be carefully considered for antivirus configurations.

Host placement and configuration should be reviewed specifically for malware threats. As a general rule, Microsoft does not recommend placing database servers in the perimeter network of an organization's infrastructure, especially if the servers store sensitive data. However, if you must locate such a database server in your perimeter network, ensure that it is configured to minimize the risk of a malware infection.

If your organization uses SQL Server, see the following guidance for more information on specific malware attack configuration guidelines:

- Microsoft Knowledge Base article "309422 – INF: Consideration for a Virus Scanner On a Computer That Is Running SQL Server" on Microsoft.com at: <http://support.microsoft.com/?kbid=309422>.
- The Security Resources page for Microsoft SQL Server on Microsoft.com at: www.microsoft.com/sql/techinfo/administration/2000/security/.

The "Slammer" worm attack targeted SQL Server directly. This attack showed how important it is to protect your SQL Server database computers, regardless of whether they reside in your perimeter or internal network.

For information and software to help ensure your SQL Server systems are protected from the Slammer worm, see the Finding and Fixing Slammer Vulnerabilities page on Microsoft.com at: www.microsoft.com/security/slammer.asp.

Collaboration Servers

The very nature of collaboration servers makes them vulnerable to malware. When users copy files to and from the servers, they may expose the servers and other users on the network to a malware attack. Microsoft recommends protecting the collaboration servers in your environment (such as those running SharePoint Services and

SharePoint Portal Server 2003) with an antivirus application that can scan all files copied to and from the collaboration store. For detailed step-by-step information on protecting these services, see the Configuring Antivirus Protection page of the *Administrators Guide for Windows SharePoint Services* on Microsoft.com at: www.microsoft.com/resources/documentation/wss/2/all/adminguide/en-us/stse11.aspx.

For information about antivirus software specifically written to integrate with Windows SharePoint Services and SharePoint Portal Server 2003, see the Solutions Directory page on Microsoft Office Online at: <http://go.microsoft.com/fwlink/?linkid=13276>.

The Network Defense Layer

Attacks that are delivered across the network represent the largest number of recorded malware incidents. Typically, malware attacks will be launched to exploit weaknesses in network perimeter defenses to allow the malware to access host devices inside the organization's IT infrastructure. These devices could be clients, servers, routers, or even firewalls. One of the most difficult problems your antivirus defenses face at this layer is to balance the feature requirements of the IT systems' users with the limitations required to create an effective defense. For example, like many recent attacks, the MyDoom worm used an e-mail attachment to replicate itself. From an IT infrastructure perspective, blocking all incoming attachments is the simplest and most secure option. However, the requirements of your organization's e-mail users may not allow this to be a viable option. A compromise must be reached that will strike a balance between an organization's requirements and the level of risk it can accept.

Many organizations have adopted a multilayer approach to the design of their networks that uses both internal and external network structures. Microsoft recommends this approach because it directly conforms to the defense-in-depth security model.

Note: There is a growing trend to break the internal network into security zones to establish a perimeter for each one. Microsoft also recommends this approach because it helps reduce the overall exposure to a malware attack seeking to gain access to the internal network. However, for the purposes of this guide, only a single network defense is described. If you plan to use a perimeter and multiple internal networks, you can apply this guidance directly to each one.

The first network defenses for the organization are referred to as the perimeter network defenses. These defenses are designed to prevent malware from ever making it into the organization from an external attack. As discussed previously in this chapter, the typical malware attack focuses on copying files to a target computer. Accordingly, your antivirus defenses should work with the organization's general security measures to ensure that access to the organization's data is only available from properly authorized personnel in a secure manner (such as via an

encrypted virtual private network (VPN) connection). For more information about creating a secure perimeter network design, see the Windows Server System Reference Architecture guidance on TechNet at:

www.microsoft.com/technet/itsolutions/techguide/wssra/raguide/default.aspx.

Note: You should also consider any wireless local area networks (LANs) and VPNs as perimeter networks. If your organization has these technologies in place, it is important to secure them. Failure to provide this security could allow an attacker to gain direct access to your internal network (bypassing the standard perimeter defenses) to mount an attack.

For more information about securing WLANs, see the following articles on TechNet:

- “Planning a Secure Wireless LAN using Windows Server 2003 Certificate Services” at:
www.microsoft.com/technet/security/guidance/secmod167.aspx.
- “Securing Wireless LANs - A Windows Server 2003 Certificate Services Solution” at:
www.microsoft.com/technet/security/prodtech/win2003/pkiwire/swlan.aspx.

For guidance on securing VPN networks, see the following Windows Server System Reference Architecture Guide on Microsoft.com:

- Enterprise Design for Remote Access at:
www.microsoft.com/technet/itsolutions/techguide/wssra/raguide/Remote_Access_Services_SB_1.aspx.

In this guide, it is assumed that the network security design provides the organization with the required level of identification, authorization, encryption, and protection to defend against a direct intrusion from an unauthorized attacker. However, at this point the antivirus defenses are not complete. The next step is to configure the network layer defenses to detect and filter malware attacks that use permitted network communications, such as e-mail, Web browsing, and instant messaging.

Network Antivirus Configuration

There are many configurations and technologies that are specifically designed to provide network security for organizations. Although these are vital parts of an organization’s security design, this section will only focus on the areas that have a direct relationship with antivirus defense. Your network security and design teams should determine how each of the following techniques is used in your organization.

Network Intrusion Detection System

Because the perimeter network is a highly exposed part of the network, it is extremely important that your network management systems are able to detect and report an attack as soon as possible. The role of a network intrusion detection (NID) system is to provide just that: rapid detection and reporting of external attacks. Although a NID system is part of the overall system security design and not a

specific antivirus tool, many of the first signs are common for both system and malware attacks. For example, some malware uses IP scanning to find available systems to infect. For this reason, the NID system should be configured to work with the organization's network management systems to deliver warnings of any unusual network behavior directly to the organization's security staff.

A key issue to understand is that with any NID implementation, its protection is only as good as the process that is followed once an intrusion is detected. This process should trigger defenses that can be used to block an attack, and the defenses should be constantly monitored in real-time. Only then can the process be considered part of a defense strategy. Otherwise the NID system is really more of a tool for providing an audit trail after an attack has occurred.

There are a number of enterprise-class network intrusion detection systems available to network designers. These can be stand-alone devices or other systems that integrate into other network services, such as the firewall services of the organization. For example, the Microsoft Internet Security and Acceleration (ISA) Server 2000 and 2004 products contain NID system capabilities, as well as firewall and proxy services.

For a list of Microsoft ISA Server partners that offer additional NID services for ISA Server, see the Intrusion Detection page on Microsoft.com at: www.microsoft.com/isaserver/partners/intrusiondetection.asp.

Application Layer Filtering

Organizations are finding it not only useful but necessary to use Internet filtering technologies to monitor and screen network communications for illegitimate content, such as viruses. Traditionally, this filtering has been performed using the packet layer filtering provided by firewall services, which only allows filtering of network traffic based on a source or destination IP address, or a particular TCP or UDP network port. Application layer filtering (ALF) works at the application layer of the OSI networking model, so it allows the data to be examined and filtered based on its content. If ALF is used in addition to standard packet layer filtering, much greater security can be achieved. For example, using packet filtering may allow you to filter port 80 network traffic through your organization's firewall so that it can only pass to your Web servers. However, this approach may not provide sufficient security. Adding ALF to the solution would allow you to check all data passing to the Web servers on port 80 to ensure that it is valid and does not contain any suspicious code.

ISA Server can provide ALF on data packets as they pass through an organization's firewall. Web browsing and e-mail can be scanned to ensure that content specific to each does not contain suspicious data, such as spam or malware. The ALF capability in ISA Server enables deep content analysis, including the ability to detect, inspect, and validate traffic using any port and protocol. For a list of vendors who make

filters to enhance the security and interoperability for different protocols and Web traffic, see the Partner Application Filters page on Microsoft.com at: www.microsoft.com/isaserver/partners/applicationfilters.asp.

For a detailed description of how ALF works in ISA Server 2000, see the Introducing the ISA Server 2000 Application Layer Filtering Kit page at: www.isaserver.org/articles/spamalfkit.html.

Content Scanning

Content scanning is available as a feature in more advanced firewall solutions or as a component of a separate service, such as e-mail. Content scanning interrogates data that is being allowed to enter or leave an organization's network via valid data channels. If content scanning is performed on e-mail, it generally works with e-mail servers to check e-mail for particular characteristics, such as attachments. This technique can scan and identify malware content in real time as the data passes through the service. There are a number of partners who work with Microsoft to provide enhanced security features to both Microsoft Exchange Server and ISA Server, such as real-time antivirus content scanning.

For more details on partner antivirus products available for Microsoft Exchange Server 2003, see the Microsoft Knowledge Base article, "823166 "Overview of Exchange Server 2003 and Antivirus Software" on Microsoft.com at: <http://support.microsoft.com/?kbid=823166>.

For a list of Microsoft partners who have developed content scanning products for ISA Server, see the Partners page on Microsoft.com at: www.microsoft.com/isaserver/partners/.

URL Filtering

Another option that may be available to network administrators is URL filtering, which you can use to block problem Web sites. For example, you could use URL filtering to block known hacker Web sites, download servers, and personal HTTP e-mail services.

Note: The major HTTP e-mail service sites (such as Hotmail and Yahoo) provide antivirus scanning services, but there are many smaller sites that do not provide antivirus scanning at all. This is a serious problem for an organization's defenses, as such services provide a route directly from the Internet to clients.

Network administrators can use two basic approaches for URL filtering:

- **Block lists.** The firewall checks a predefined list of problem sites before allowing the connection. Users are allowed to connect with sites that are not specifically on the block list.
- **Allow lists.** This approach only allows communications with sites entered on a predefined list of Web sites that has been approved by the organization.

The first approach relies on an active process of identifying Web sites that may be a problem and adding them to the list. Because of the size and variable nature of the Internet, this approach requires either an automated solution or significant management overhead, and is generally only useful for blocking a small number of known problem sites instead of providing a comprehensive protection solution. The second approach provides greater protection because its restrictive nature makes it possible to control the sites available to users of the system. However, unless the correct research is done to identify all sites that users require, this approach may prove too restrictive for many organizations.

Microsoft ISA Server supports the manual creation of both of these lists using its Site and Content Rules. However, enhanced and automated solutions are available from Microsoft partners that work directly with ISA Server to ensure URLs can be blocked or allowed as required with a minimum of management overhead. A list of these solutions is available from the Microsoft Internet Security and Acceleration Server Partners URL Filtering page on Microsoft.com at: www.microsoft.com/isaserver/partners/accesscontrol.asp.

Both these approaches will only provide protection while a client is inside the organization's defenses. This protection will not be available when a mobile client connects directly to the Internet while out of the office, which means your network will be susceptible to a possible attack. If a URL filter solution is required for mobile clients in your organization, you should consider using a client-based defense system. However, this approach can lead to a significant management overhead, especially in environments with large numbers of mobile clients.

Quarantine Networks

Another technique you can use to secure networks is to establish a quarantine network for computers that do not meet your organization's minimum security requirements.

Note: This technique should not be confused with the quarantine feature available in some antivirus applications, which moves an infected file to a safe area on the computer until it can be cleaned.

A quarantine network should restrict, or even block, internal access to your organization's resources, but provide a level of connectivity (including the Internet) that will allow temporary visitors' computers to work productively without risking the security of the internal network. If a laptop from a visitor is infected with malware and connects to the network, its ability to infect the other computers on the internal network is restricted by the quarantine network.

An approach similar to this has been successfully applied to VPN-type remote connections for some time. VPN clients are diverted to a temporary quarantine network while system tests are performed. If the client passes the tests, for example

by having the required security updates and antivirus signature files, they are granted access to the organization's internal network. If the client does not meet these requirements they are either disconnected or allowed access to the quarantine network, which can be used to obtain the necessary updates to pass the tests. Network designers are now looking at this technology to help improve security on internal networks.

For more information on this technique, see the Planning for Network Access Quarantine Control page of the *Microsoft Windows Server 2003 Deployment Kit* on Microsoft.com at:

www.microsoft.com/resources/documentation/windowsservo/2003/all/deployguide/en-us/dnsbf_vpn_aosh.asp.

ISA Server Feature Pack

If your organization uses ISA Server 2000, Microsoft also recommends using the additional features provided in ISA Server Feature Pack 1. This free add-on provides additional security features that you can use to improve the security of communications (including e-mail) across the firewalls in your network defenses. The features that you can use to improve your antivirus network defenses include:

- **An enhanced SMTP filter.** This feature helps filter e-mail messages with increased reliability and security. The filtering is based on the name, size, or extension of an attachment, as well as the sender, domain, keyword, and any SMTP command and its length.
- **An enhanced Exchange remote procedure call (RPC) filter.** This feature protects Outlook e-mail communication to Exchange Server computers over untrusted networks without requiring you to set up a VPN. To achieve this, the following extra features are also included in ISA Server Feature Pack 1:
 - The ability for Administrators to enforce RPC encryption between Outlook and an Exchange Server.
 - The ability for outbound RPC communication to pass securely through ISA Server, which in turn permits Outlook clients connected to an ISA Server computer to access external Exchange Server computers.
- **UrlScan 2.5.** This tool helps stop malicious Web requests at the ISA Server computer before they can enter the network and access a Web server.
- **Outlook Web Access (OWA) Wizard.** You can use this wizard to quickly and easily configure ISA Server to help protect an OWA deployment.
- **RPC Filter Configuration Wizard.** You can use this wizard to only allow a precise level of access to RPC services on the internal network instead of all RPC traffic.

To obtain the feature pack, see the How to Obtain Feature Pack 1 page on Microsoft.com at:

www.microsoft.com/isaserver/featurepack1/howtogetfp1.asp.

For more information about using these features to secure a perimeter ISA Server firewall, see the ISA Server Feature Pack 1 page on Microsoft.com at:

www.microsoft.com/isaserver/featurepack1/.

Physical Security

Although physical security is more of a general security issue than a specific malware problem, it is impossible to protect against malware without an effective physical defense plan for all client, server, and network devices in your organization's infrastructure. There are a number of critical elements in an effective physical defense plan, including:

- Building security
- Personnel security
- Network access points
- Server computers
- Workstation computers
- Mobile computers and devices

Each of these elements should be evaluated in a security risk assessment for your organization. If an attacker compromises any of these elements, there is an increased level of risk that malware could bypass the external and internal network defense boundaries to infect a host on your network.

Protecting access to your facilities and your computing systems should be a fundamental element of your organization's overall security strategy. A detailed explanation of these considerations is beyond the scope of this solution. However, information about the basic elements of a sound physical security plan is available in the "5-Minute Security Advisor - Basic Physical Security" article on Microsoft TechNet at:

www.microsoft.com/technet/community/columns/5min/5min-203.mspix.

Polices, Procedures, and Awareness

Client, server, and network operational policies and procedures are essential aspects of the antivirus defense layers in your organization. Microsoft recommends consideration of the following policies and procedures as part of your organization's antivirus defense in depth solution:

- **Antivirus scanning routines.** Ideally, your antivirus application should support automated or real-time scanning. However, if this is not the case, you should

implement a process to provide guidance on when the users in your organization should run a full system scan.

- **Antivirus signature update routines.** Most modern antivirus applications support an automated method for downloading virus signature updates, and you should implement such a method on a regular basis. However, if your organization requires testing these updates prior to deploying them, you will generally not be able to use such methods. If this is the case, make sure your support staff identifies, downloads, tests, and updates signature files as soon as possible.
- **Policies on allowed applications and services.** A clearly communicated policy should exist to explain which applications are allowed on your organization's computers and others that access your organization's resources. Examples of applications that can cause problems include peer-to-peer network applications and applications that users may download directly from rogue Web sites.

At a minimum, Microsoft recommends the following policies and procedures for all devices in your organization's network defense layer.

- **Change control.** A key security process for network devices is to control changes that impact them. Ideally, all changes should be proposed, tested, and implemented in a controlled and documented manner. Spontaneous changes to devices in the perimeter network are likely to introduce configuration errors or flaws that an attack could exploit.
- **Network monitoring.** Correctly configuring your network devices to optimize them for security does not mean that other antivirus procedures can be neglected. Ongoing monitoring of all devices in the network is essential to detect malware attacks as soon as possible. Monitoring is a complex process that requires gathering information from a number of sources (such as firewalls, routers, and switches) to compile a "normal" behavior baseline you can use to identify abnormal behavior.
- **Attack detection process.** If a suspected malware attack is detected, your organization should have a set of clearly defined and documented steps to follow to ensure the attack is confirmed, controlled, and cleaned with minimum disruption to end users. See Chapter 4, "Outbreak Control and Recovery," for more information about this subject.
- **Home computer network access policy.** A set of minimum requirements should be established and met before an employee can connect a home computer or network to your organization's network via a VPN connection.
- **Visitor network access policy.** A set of minimum requirements should be established and met by visitors before they are allowed to connect to your organization's network. These requirements should apply to both wireless and wired connectivity.
- **Wireless network policy.** All wireless devices connecting to the internal network should meet minimum security configuration requirements before they can

connect. This policy should specify the required minimum configuration for the organization.

There are many more policies and procedures you could implement to improve the security of your network devices; the ones listed in this section should be considered as a good starting point. However, because additional policies provide general security settings rather than antivirus specific settings, they are outside the scope of this guide.

Security Update Policy

Client, server, and network defenses should all have some form of security update management system in place. Such a system could be provided as part of a wider enterprise patch management solution. The operating systems of hosts and devices should be checked for vendor-supplied updates on a regular basis. The security update policy should also provide the operating criteria for the process that is used to roll out security updates to your organization's systems. This process should consist of the following stages:

1. **Check for updates.** Some type of automated notification process should be in place to notify users of available updates.
2. **Download updates.** The system should be able to download updates with minimal impact on users and the network.
3. **Test updates.** If updates are for mission-critical hosts, you should ensure that each update is tested on a suitable non-production system before it is deployed in your production environment.
4. **Deploy updates.** Once an update has been tested and verified, a simple deployment mechanism should be available to help distribute it.

If the systems being updated in your environment do not require the testing phase of this list, your organization may wish to consider automating the entire process for its systems. For example, the **Automated Updates** option on the Microsoft Windows Update Web site makes it possible for your client computers to be notified and updated without user intervention. Using this option helps to ensure that your systems are running the latest security updates as soon as possible. However, this approach does not test the update before installing it. If this is a requirement for your organization, this option is not recommended.

Ensuring that your organization's systems are maintained with the latest security updates should become a routine part of your organization's system management.

Risk-based Policies

With so many clients, servers, and network devices connected at the perimeter and internal network layers of the antivirus defense-in-depth model, it can be difficult to create a single effective security policy to manage all of the requirements and

configurations in your organization. One approach you can use to organize your policy is to group the hosts in your organization into categories based on their type and exposure to risk.

To help determine the level of risk to assign to a host or device, consider conducting a risk assessment on each of them. A detailed set of guidance on performing such risk assessments is available in “Chapter 3 - Understanding the Security Risk Management Discipline” of the *Microsoft Solution for Securing Windows 2000 Server* on TechNet at:

www.microsoft.com/technet/security/prodtech/win2000/secwin2k/03secrsk.mspx.

Microsoft recommends consideration of the following configuration categories for your organization’s client focused risk assessment policies:

- **Standard client configuration.** This configuration category usually applies to office-based desktop computers that stay physically on site in an office building. These desktop clients are continuously protected by the existing external and internal network defenses, and they are secured within an organization’s buildings.
- **High-risk client configuration.** This configuration category is designed to meet the needs of mobile computer users and mobile devices such as PDAs and mobile phones. These devices often move outside the protection of the organization’s network defenses and are therefore at a higher level of risk.
- **Guest client configuration.** This configuration category is designed for client computers that your organization does not own or support. Managing the configuration of these computers may not be possible, because you are unlikely to have control over their configuration. However, you can set policies that will limit the ability of these computers to connect to your organization’s networks. Guest client computers are typically one of the following types:
 - Employee home computers.
 - Partner or vendor computers.
 - Guest computers.

Microsoft also recommends establishing risk categories for server roles, and the same risk assessment is recommended for servers as well as clients. As a starting point for your server policies, you could consider the following configuration categories:

- **Standard server configuration.** This configuration category is designed to be a common denominator for the majority of server configurations in your environment. It provides a minimum level of security, but without restricting commonly used services. You can then modify the high-risk and role-specific configuration category policies to cover all policy requirements at an appropriate level.

- **High-risk server configuration.** Servers that are in the perimeter network or exposed directly to external connections and files should be considered in this configuration category. For example, this category could include perimeter Web servers, firewall servers, and messaging servers. A server that contains particularly sensitive data, such as an HR database server, might also warrant this configuration regardless of its network location.
- **Role-specific configurations.** Your organization may also choose to organize specific server roles into different configurations to more closely match the requirements of your server applications. For example, you may choose to use role-specific configurations for messaging servers, database servers, or firewalls. You may elect to use this approach in addition to either the standard or high-risk configuration category as required.

The use of risk-based policies is ultimately the choice of the planning teams in your organization, and you can use the referenced configuration classifications as a basis for further development. Ultimately, the goal is to reduce the number of configurations your management systems must support. In general, a standardized approach is more likely to yield a secure configuration than configuring the security of each host in your environment independently.

Automated Monitoring and Reporting Policies

If your organization uses an automated monitoring system or an antivirus application that can report suspected malware infections to a central location, it is possible to automate this process so that any alert will automatically inform all of the users in your organization's IT infrastructure. An automated alert system will minimize the delay between an initial alert and users being aware of the malware threat, but the problem with this approach is that it can generate many "false positive" alerts. If no one is screening the alerts and reviewing an unusual activity reporting checklist, it is likely that alerts will warn of malware that is not present. This situation can lead to complacency, as users will quickly become desensitized to alerts that are generated too frequently.

Microsoft recommends assigning members of the network administration team the responsibility of receiving all automated malware alerts from all system monitoring software or antivirus packages that your organization uses. The team can then filter out the false positive alerts from the automated systems before issuing alerts to users. For this approach to be successful, the team needs to monitor for alerts 24 hours a day, 7 days a week to ensure all alerts are checked and, if required, released to network users.

User and Support Team Awareness

Team awareness and training should target the administration and support teams in your organization. Training for key IT professionals is a fundamental requirement in

all areas of IT, but for antivirus defense it is especially important because the nature of malware attacks and defenses may change on a regular basis. A new malware attack can compromise an effective defense system almost overnight, and your organization's defenses could be at risk. If the support personnel for these defenses are not trained in how to spot and react to new malware threats, it is only a matter of time before a serious breach in the antivirus defense system occurs.

User Awareness

User education is often one of the last considerations an organization makes when designing its antivirus defense. Helping users understand some of the risks associated with malware attacks is an important part of mitigating such risks, because everyone in the organization who uses IT resources plays a role in the security of the network. For this reason, it is important to educate your users about the more common risks that they can mitigate, such as:

- Opening e-mail attachments.
- Using weak passwords.
- Downloading applications and ActiveX controls from untrusted Web sites.
- Running applications from unauthorized removable media.
- Allowing access to your organization's data and networks.

As malware techniques change, antivirus defenses have to be updated. Regardless of whether an antivirus program's signature file or the program itself needs updating, it takes time to create and deploy updates. The amount of time it takes to create updates has been dramatically reduced over the last few years, and these updates are generally available in a matter of hours. However, in rarer cases, it can still take days from the time a new malware attack is released to make an effective antivirus defense available.

During this time the best defense your organization may have is users who are aware of malware and its risks. Providing your users with basic antivirus guidelines and training can help prevent a new malware strain that makes it past your IT defenses from propagating throughout your environment.

Training users does not have to be a complex process. Basic antivirus guidelines are largely based on common sense principles, but ensuring such guidelines are enforced and communicated clearly can be more of a challenge. The Windows XP Baseline Security Checklists available on Microsoft TechNet at: www.microsoft.com/technet/Security/chklist/xpcl.mspx can help you identify common antivirus and security related issues to communicate to your users.

Users responsible for mobile devices are likely to require additional training to help them understand the risks associated with taking a device outside of the organization's physical and network defenses. It is likely that additional defenses will be required specifically to safeguard these mobile devices. For this reason, you

may need to require additional configuration and training for users who manage these devices.

Note: There is some useful end user configuration information provided in the Protect your PC guidance on Microsoft.com at: www.microsoft.com/security/protect/. This site is a good information resource that can help your users educate themselves on how to secure their home computers and networks.

Support Team Awareness

The IT professionals responsible for the configuration and support of the servers, clients, and network devices of the organization will need antivirus training to help them ensure that their systems are optimally configured and maintained to stop malware attacks. Errors in the configuration of any of these computers or devices can open a route for a malware attack. For example, if a poorly trained firewall administrator opens all the network ports by default on a perimeter firewall device, a serious security and malware risk would be created. Administrators who are responsible for the devices that connect to your organization's perimeter network should receive specific security training to help them understand the range of attacks that can affect the network devices.

Many events, hands-on labs, and Webcasts on security topics are available directly from Microsoft. For more information about these topics, see Your Security Program Guide on Microsoft.com at:

www.microsoft.com/seminar/events/security.msp.

Security training and books are also available from Microsoft Learning. For more information about these publications, see the Microsoft Learning Security Resources page on Microsoft.com at:

www.microsoft.com/learning/centers/security.asp.

Obtaining User Feedback

Malware-aware users can provide an excellent early warning system if they are presented with a simple and effective mechanism to report unusual behavior on the systems they use. Such a mechanism can take the form of a telephone hotline number, e-mail alias, or a rapid escalation process from the organization's Helpdesk.

Proactive Internal Communications

If possible, members of the IT department should create a proactive antivirus response team that is responsible for monitoring external malware alert sites for early warnings of malware attacks. Good examples of such sites include:

- Antivirus application vendor Web sites.
- The Anti-Virus Information Exchange Network (AVIEN) Web site at:
www.avien.org.

- Antivirus alert services, such as the Antivirus Information Early Warning System (AVI-EWS) from AVIEN (you can subscribe to these services).
- The Microsoft Security Antivirus Information Web site on Microsoft.com at: www.microsoft.com/security/antivirus/.

Regular checking of reference sites like these should enable support staff to notify systems administrators and users of current malware threats before they penetrate your organization's network. The timing of these checks is crucial. Ensuring that system users receive a proactive warning before checking their morning e-mail can make the difference between managing the removal of a few suspicious e-mails and trying to contain a malware outbreak. If the majority of your system's users log on at 9 A.M., establishing a way to communicate new malware threats before this time would be considered best practice.

Internal Malware Alerts

Finding the most effective mechanism to inform all users of the potential for a malware attack in a timely and comprehensive way is crucial. Available communications systems vary greatly depending on the organization's infrastructure, and it is impossible to provide a malware alert system that will work for all organizations. However, this section provides the following examples of mechanisms that your organization may wish to consider for this purpose:

- **Organization notice boards.** A low-tech approach that should not be forgotten is to use internal office doors, notice boards, or paper-based information points that are obvious to employees. Although this process involves some overhead to maintain, it has the significant advantage of communicating vital information to your users when areas of the network are unavailable due to an attack.
- **Voice mail systems.** If your organization's voice mail system supports it, the ability to leave a single message for all users can be an effective mechanism to communicate a malware alert. However, it should be noted that this method relies on users accessing voice mail before e-mail to alert them of an e-mail threat.
- **Logon messages.** You can configure the Windows operating system to deliver a message directly to your users' screens during the logon process. This mechanism provides a good way to draw user attention to malware alerts.
- **Intranet portals.** A common intranet portal that users have set as their home page can be used to provide malware alerts. Users will need to be advised to view this portal before accessing their e-mail to make this alert mechanism effective.
- **E-mail systems.** Care should be taken when using an e-mail system to communicate malware alerts to your users. Because an attack could affect your e-mail servers, this mechanism may not be effective in all cases. Also, the nature of the inbox queuing process could deliver a malware warning after an e-mail containing malware has already been delivered to your users. For this reason, you may need to advise your users to first look for high priority malware warnings when they first log on to their computers before reviewing any e-mail messages.

Summary

Antivirus defense is no longer a matter of installing an application. The most recent malware attacks have proven that a more comprehensive defensive approach is required. This chapter has focused on how you can apply the defense-in-depth security model to form the basis of a defense-in-depth approach to create an effective antivirus solution for your organization. It is important to understand that malware writers are continually updating their methods to attack new IT technologies that your organization may be using, and that antivirus technologies are constantly evolving to mitigate these new threats.

The antivirus defense-in-depth approach should help ensure that your IT infrastructure will address all possible malware attack vectors. Using this layered approach makes it easier to recognize any weak points in the entire system, from the perimeter network to the individuals working at their computers throughout your environment. Failure to address any of the layers described in the antivirus defense-in-depth approach could leave your systems open to attack.

You should constantly review your antivirus solution so that you can update it whenever needed. All aspects of antivirus protection are important, from simple automated virus signature downloads to complete changes in operational policy.

Similarly, because the information provided in this guide is subject to updates, it is important to continually monitor the Microsoft Security Antivirus Information Web site on Microsoft.com at www.microsoft.com/security/antivirus/ to receive the latest antivirus information and guidance.

Microsoft recognizes how disruptive and costly malware can be, and has invested a great deal of effort into making it more difficult for those who create and distribute malware. Microsoft is also working to make it easier for network designers, IT professionals, and end users to configure systems to meet their security requirements with minimal impact to their business operations.

Although it may not be possible to completely eradicate malicious code, focusing consistent attention on the areas highlighted in this antivirus defense-in-depth approach will help minimize the effect a malware attack can have on your organization's business operations.

4

Outbreak Control and Recovery

Introduction

This chapter presents a detailed set of considerations that you can use to identify a malware infection or outbreak, contain it, and then remedy the effects it may have on the infected systems in your environment. The need for a consistent, straightforward approach to incident response and recovery cannot be understated; malware incidents tend to create a sense of urgency that is not conducive to instituting well thought out procedures that will remain effective and successful in the long term.

One additional important point needs to be made. As malware attacks have grown in complexity using many different payloads, no single process for removing it from your systems is any longer universally applicable. Each different malware attack is likely to require its own remediation. However, this does not lessen the value in defining a process for identifying, containing, and recovering from an attack that should remain consistent.

A high-level view of the steps in a malware outbreak recovery process includes:

1. Infection confirmation
2. Incident response
3. Malware analysis
4. System recovery
5. Post recovery steps

Step 1: Infection Confirmation

The ability to quickly determine whether a system has been infected will prove invaluable in your organization's ability to minimize the impact of infections. By quickly confirming an infection and identifying its suspect characteristics, the spread of the infection and its impact on your users can be reduced.

There are many different types of computer malfunctions that can be mistaken for virus-like behavior. Upon receiving a phone call or e-mail from a user that states "I think my system has a virus," the support staff must first determine if the behavior is likely to be caused by some kind of malicious code. The following list provides some examples of typical symptoms a user may report as "virus-like" behavior:

- "I opened an e-mail attachment and nothing happened; now my machine is acting funny."
- "I received e-mail replies from contacts asking why I have sent them an .exe, .zip or other attachment, which I never sent."
- "My antivirus software has stopped working and the computer keeps shutting down!"
- "My programs are not working properly, and they are all *very* slow!"
- "A bunch of files I have never seen before are all over the **My Documents** folder."
- "A number of my files won't open or have disappeared!"

Observations and feedback from your users is critical, because they will likely be the first to notice unusual activity. As the speed of malware outbreaks continues to increase, the window of time between the initial infection and the availability of an effective defense becomes increasingly important. Since most infections will occur during this period, your organization's ability to quickly identify and confirm an infection is crucial to minimizing both the spread of an outbreak and the damage it can cause.

The following section outlines a series of steps that will enable you to more quickly confirm if unusual behavior is indeed a malware attack or outbreak.

If a new type of malware infects a system, the user of that system may well be the first to notice unusual behavior. As discussed in Chapter 3, "Antivirus Defense in Depth" in this guide, there is generally a delay from the time new malware is released to the point an antivirus scanning application will be updated to detect and counteract it. The best way of providing an early warning system is to educate users to recognize the signs of a possible malware attack, and provide them with a fast communications link to report them as soon as possible.

Infection Reporting

After a call has been received or an alert has been generated about a possible new malware attack, it is usually beneficial for the Helpdesk to have a defined process for determining as quickly as possible whether the alert concerns a new attack. The following flow chart illustrates the major steps in this process:

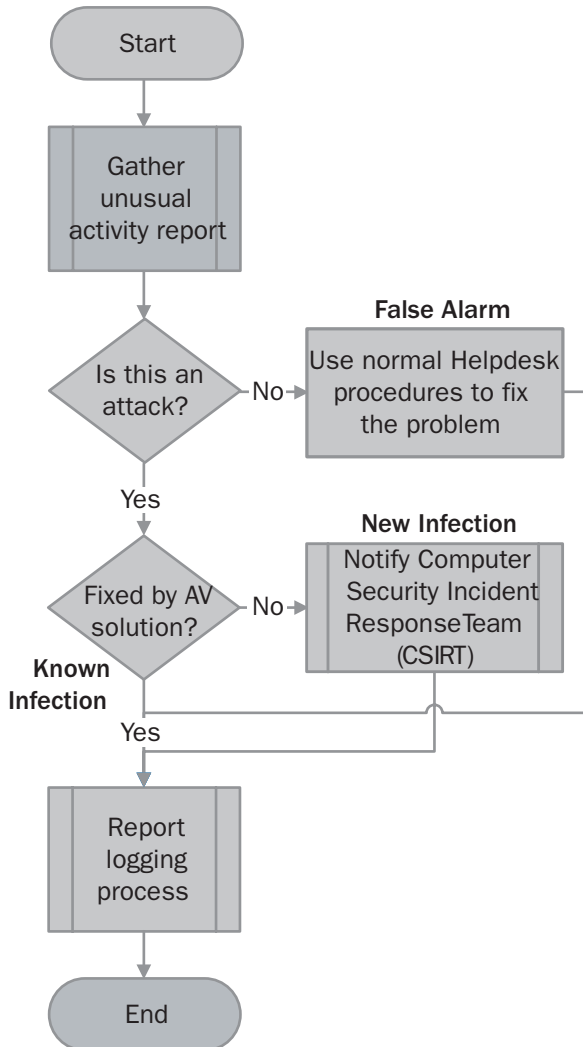


Figure 4.1
The malware infection reporting process

Unusual Activity Reporting

The following questions should be used to determine if the unusual activity that prompted the alert is likely a new malware attack. This guide makes the assumption that these questions should be directed towards a non-technical user by a member of the IT Helpdesk in your organization.

Gathering the Basic Information

The initial questions should be designed to produce answers that will help determine as quickly as possible the true nature of the alert, and the likelihood of it being a new malware attack. You can use the following sample questions as the starting point for this process; they should be modified to meet the requirements of your organization:

- What is the date and time of the report?
- What was the unusual activity that prompted the report?
- What activity was happening just prior to the unusual activity?
 - Have you visited any Web sites recently that are outside of your “normal” routine?
 - Has this system been outside the organization’s network recently (for example, in an airport, on a home network, at a Wi-Fi hotspot, or in a hotel)?
 - Have you seen any unusual pop-up windows or advertisements on the screen?
- What unusual or unexpected processes are currently running?
- Is the computer a workstation or server? What is its operating system and what security updates have been applied to it?
- Does the computer or any devices attached to it contain mission critical data?
- Is the user logged on with an account that has administrator privileges?
- Is the user using a strong password or passphrase?
- Has this system suffered a malware attack before?

This last question is important, as previous attacks often create vulnerabilities that can lead to subsequent attacks unless they are fixed. If the answer to this question is “Yes,” consider asking the following additional questions:

- When did the previous attack occur?
- Who handled the case and, if possible, what was the case number?
- Do you have information about what was done at that time?

Evaluating the Data

After the answers to these questions have been gathered, the support technician should evaluate the collected data against the following set of questions to help determine if a malware attack is a likely cause of the report:

- Could the report be the result of a legitimate new or updated characteristic of the system?
- Could it be explained by the activities of an authorized user (instead of a hacker/intruder)?
- Could it be explained by known system activity?
- Could it be explained by authorized changes to programs or systems?

Finally, a check should be made with external antivirus sources (identified in the “Proactive Internal Communications” section of Chapter 3, “Antivirus Defense in Depth” in this guide) to determine if this report matches some existing virus or worm alert.

Gathering the Details

At this point it may be possible to determine if a new malware attack is the likely cause of the problem. If not, a higher level of technical information may be required and a support technician may need to physically visit (or if possible, gain remote control to) the suspect system. You can use the following sample technical questions to gather more detailed information and determine, categorically, if the system has been attacked by a hacker or malicious code:

- Does the device have a firewall enabled on it or in front of it? If so, what ports are open to the Internet?
- If applications are crashing, contact the application vendor(s) right away to determine the root cause (for example, current Microsoft applications provide error reporting tools you can use to send in a crash report).
- Are there any security updates for this system that have been released but have not been installed?
- What kind of password policy does the system have? What is the minimum password length? What are the password complexity requirements?
- Are there any new or suspicious:
 - accounts on the local machine?
 - accounts in the administrators group?
 - services listed in the Services management console?
 - events in the event logs?
- Are there any network connections reported by the **netstat** utility to external IP addresses or suspicious IP addresses?

Unusual Activity Response

After the initial information has been gathered and used to determine the nature of the alert, it should be possible for the Helpdesk to make a decision about whether a false alarm, hoax, or real malware attack has occurred.

Creating a fake malware report is far easier than developing a virus or worm, which unfortunately assures the creation of many false malware alerts. These hoaxes and the calls and warnings they generate waste considerable time and money. Hoaxes also annoy your users and tend to make them question the value of reporting potential attacks. The following considerations should be made to ensure the alert is correctly handled.

- **False alarm.** If the report is a false alarm, the call information should be logged. Periodic review of this information may help determine if additional user training is required.
- **Hoax.** It is important to track and record false malware alerts as well as real malware activity, as they are still attack instances — they just do not use malicious code. Communicating information about false malware alerts as well as real malware threats to your users should be part of your organization’s regular antivirus communications. This information will help the users recognize hoaxes in advance and therefore reduce lost productivity.
- **Known infection.** If the system appears to be infected, the Helpdesk should take steps to determine if the infection is a known attack that can be handled with an existing antivirus application. The system’s antivirus application should be checked to ensure it is operational and up-to-date. A complete system scan should then be undertaken to attempt to clean the system. If this scan successfully identifies and cleans the infection, the call should be logged and a warning sent to all users to ensure their antivirus systems are running correctly and updated. If the scan fails to identify a specific form of malware, it should be considered a new infection and the guidance in the “Incident Response Process” section followed.
- **New infection.** If the system appears to be infected by a new malware attack, a number of initial actions should be followed to help ensure the problem is communicated in the correct manner. These initial actions are designed to help the IT support staff consistently follow a process that will ensure the correct course of action is followed. Answers to the initial questions listed earlier will help determine which of the following initial actions should be considered at this stage:
 - Contact the assigned member of the emergency response team with details of the alert.
 - If the suspect computer is a server, contact its administrator to discuss the implications of removing the computer from the network.
 - If the suspect computer is a workstation, contact its user(s) to discuss the implications of removing the computer from the network.
 - Consider triggering a high-level alert or warning to users of the IT system to warn of the detected attack.

At this point, the role of the Helpdesk is complete. Responsibility for the outbreak will move to the incident response process, and the members of the Computer Security Incident Response Team (CSIRT) will need to be notified.

Step 2: Incident Response

As discussed in Chapter 3, “Antivirus Defense in Depth” in this guide, the CSIRT will need to convene an emergency meeting as soon as possible to help organize the next stage of the organization’s incident response process. For more detailed explanations of how to create an incident response team and the processes for security and disaster recovery in general, refer to the same chapter in this guide.

For the purposes of this guide, it is assumed that the CSIRT is in place. The first goal of the team at this point should be to determine the immediate outbreak control mechanism. The following section provides information that will help determine the options for this mechanism and its components.

Emergency Outbreak Control

After the malware attack has been confirmed, the first step in controlling the outbreak is to ensure that the infected computers are isolated from other devices. Ensuring the isolation of infected computers is essential, because it prevents their ability to spread the malicious code. There are a number of different mechanisms for achieving this isolation that will all have an impact on the normal operations of the organization.

Important: If you believe your organization will be pursuing criminal or civil litigation, Microsoft recommends consulting with your organization’s legal representatives before taking further steps.

If the outbreak has been detected in the antivirus community, use the guidance provided by your antivirus vendor(s) to help you determine the severity of the outbreak.

If the outbreak is currently unknown in the wider antivirus community, you should report the incident to your antivirus vendor(s) as soon as possible. They may request that you send them examples of the malware in a compressed and password-protected file to allow them to analyze it. The process of finding these examples is not always straightforward and should, ideally, be prepared for in advance. See the section “Step 3: Malware Analysis” of this chapter for guidance in preparing malware examples.

The next course of action that should be followed is to contain the immediate attack. There are three basic options for you to consider:

- Disconnect the compromised system(s) from the local network.
- If possible, isolate the network(s) containing infected hosts.
- If the entire network is compromised or potentially can be compromised, disconnect the complete network from all external networks.

There are many more detailed technical steps that could be taken, such as monitoring the network to try and identify the network ports and IP addresses involved in the attack. However, if the detailed analysis of the malware has not been completed, the risk of missing an attack vector that could lead to wider infection is significant. The only mechanism available to your organization to help you determine whether this risk is acceptable would be a completed security risk assessment report. This report would enable you to determine the risks involved in failing to stop an attack and potentially infecting or unwittingly being used to launch an attack on customers or partner organizations. If you have not completed this risk analysis prior to an attack, it is recommended that your organization err on the side of caution and minimize the possibility of spreading an attack by selecting the highest level of isolation possible.

The options listed here are guidelines only. Your specific course of action may be different depending on such factors as business needs, locale, impact, severity, and other factors that may apply only to your organization and the circumstances of the outbreak.

Preparing for Recovery

After the outbreak control mechanism has been activated, you should start the process of active recovery. The overall aim of the recovery process is to ensure that the following goals are achieved:

- Minimal disruption to the organization's business.
- The fastest possible recovery time from the attack.
- The capture of information to support possible prosecution.
- The capture of information to allow for additional security measures to be developed, if required.
- Prevention from further attacks of this type for the recovered systems.

Unfortunately, the first two goals require a “rapid fix” approach while the remaining three require time to be spent in gathering information about the attack to fully understand it. To satisfy both — that is, to quickly resolve the issue and still capture all the relevant data required — consider using the process shown in the following figure. This process is designed to ensure an infected system is released for recovery as quickly as possible while at the same time making sure the required forensics data is not lost. This data is important, because your organization will use it to determine if the recovered systems will be safe from future attack, and it will also serve as evidence if future legal action is pursued.

The processes of system recovery and virus analysis should be run as parallel activities to ensure the fastest possible recovery time.

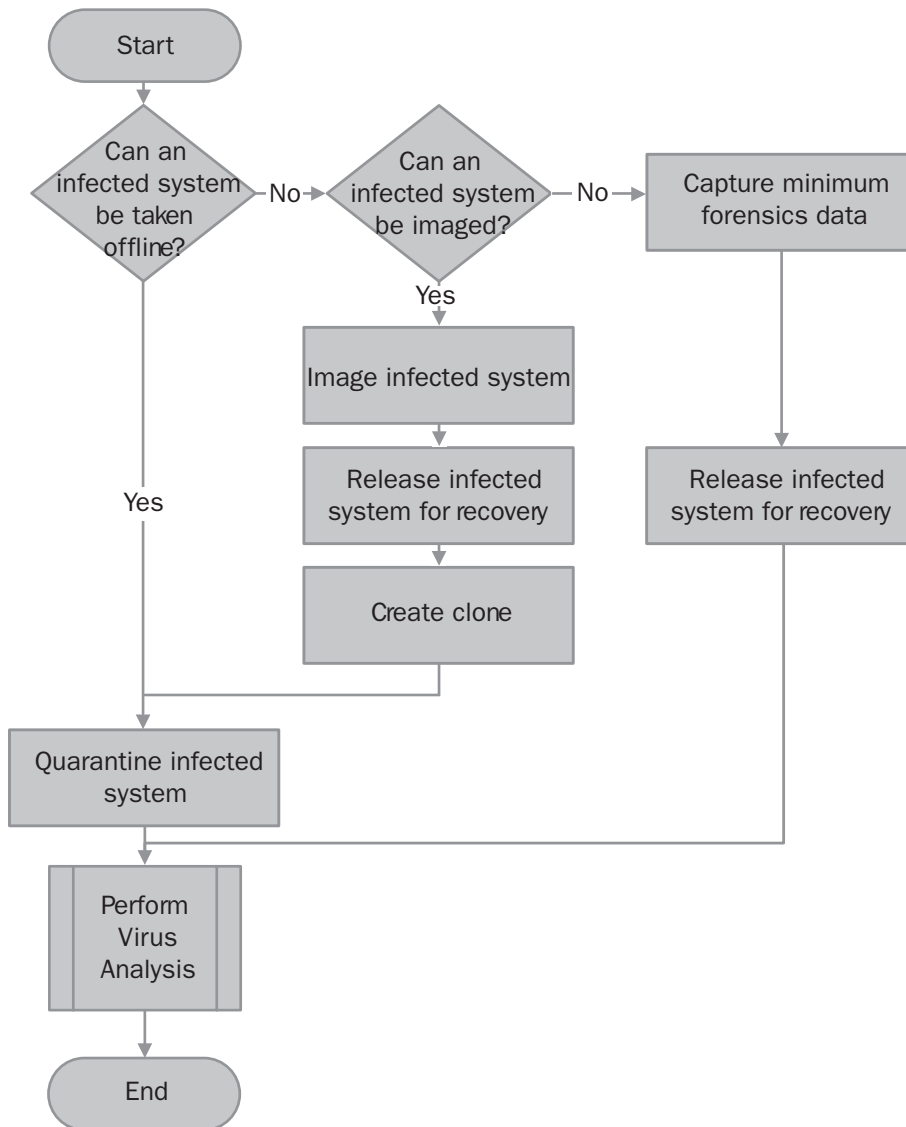


Figure 4.2
Recovery steps before analysis

The fastest way to allow all systems to be recovered is to determine if one infected system can be used for analysis. If so, this system should be quarantined and analyzed. (Guidance on this analysis process is provided in the following “Step 3: Malware Analysis” section of this chapter.) If quarantine and analysis are not possible, the next best option is to create a clone of the system using some type of imaging software. If this option is available, you can take an image of the system, release the original computer for recovery, and then create a clone system.

In cases where evidence will be gathered or more detailed analysis can be undertaken, it is especially important that affected computers are imaged as soon as possible (before remediation activities begin) so that the infection can be identified, triaged, and handled in the most expedient and appropriate manner.

Finally, if imaging is not possible a set of minimum forensic data should be gathered before the system is released for recovery. Ideally, your organization's security team should develop and maintain some form of incident response toolkit. You can use such a toolkit to gather both volatile and nonvolatile system data that will be useful for providing system forensic data. This toolkit could be a subset of a more complete malware analysis toolkit that will be used in the next section of this chapter to uncover and document all elements of the malware. However, the key differentiator of an incident response toolkit is that it should capture the minimum level of system information required in the fastest possible time to allow the system to be released for recovery as soon as possible.

Step 3: Malware Analysis

As soon as the spread of the malware attack has been contained, it is important to take some time to understand the nature of the outbreak and to perform a more detailed analysis of the malware. Failure to carry out this step can increase the likelihood of re-infection; failure to understand how the malware works will make it impossible to ensure that systems are cleaned and secured against further attacks.

Ideally, the malware analysis would be carried out by a member of the security team with a dedicated set of applications and utilities that can be used to gather the required information in as automated a fashion as possible. The following steps will help understand the nature of the attack.

Examine the Operating System Elements

Try to determine which operating system files were introduced or modified by the attack. As part of this analysis, look for changes in the following areas:

- Active processes and services.
- The local registry.
- Files in the Microsoft® Windows® system folders.
- New user or group accounts, especially with Administrator privileges.
- Shared folders (including hidden folders).
- Newly created files with normal looking file names but in unusual locations.
- Opened network ports.

Techniques that you can use for checking these operating system elements will be explained in the following sections.

Checking for Active Processes and Services

Infected systems are likely to have had new processes introduced into their memory.

The use of specialized process listing tools, such as PsTools and the Process Explorer freeware program is recommended to provide a more user friendly interface. These tools are available from the Sysinternals Web site at www.sysinternals.com, and make it possible to see not only the path to the image file but also the process tree.

To help minimize the number of entries in the process list and therefore help in the identification of any rogue processes, you should close all valid applications and any valid background applications such as Instant Messenger, e-mail monitors, or third-party utilities that stay memory resident.

If a specialized tool is not available, the Windows Task Manager tool in all Microsoft Windows systems can be used as a quick check for active processes running on the system. However, because Task Manager does not show the path to the image that launched the process, it would be impossible to determine whether a malware attack launched as “svrhost” would be a legitimate process or not.

Complete the following steps to analyze the active processes using Task Manager:

► To analyze active processes on a computer running Windows

1. Press **CTRL+ALT+DELETE** simultaneously to bring up the **Windows Security** window and select **Task Manager**.

Note: On Windows 9x computers you will see a list of running programs instead of the Task Manager application.

2. Click the **Processes** tab.
3. Resize the **Windows Task Manager** window to display as many of the active processes as possible on your screen.
4. Select the **View** option from the menu bar and click **Select Columns...**
5. Select the check boxes for the following columns:
 - PID (Process Identifier)
 - CPU Usage
 - CPU Time
 - Memory Usage
 - Peak Memory Usage
 - I/O Reads
 - I/O Writes
6. Click **OK** and resize the window to show as many of these columns as possible.

You can sort the order of the columns by clicking any column title. Use this sorting method for each of the listed columns and determine which processes are using which resources.

Note: To obtain a printout of this list for future reference, make Process Explorer or the Windows Task Manager the active window and press **ALT+PRT SCRN** on the keyboard. A screen shot of the list will be created in the computer's clipboard, which can be pasted into the Windows Paint application or Microsoft Word and printed.

The following figure shows the process details of the Blaster worm as an active process in the Microsoft Windows 2000® Server Task Manager.

Image Name	PID	CPU	CPU Time	Mem Usage	Peak Mem Usage	Page Faults	I/O Reads	I/O Writes
System Idle Process	0	99	1008:21:59	16 K	16 K	1	0	0
System	8	00	0:02:58	212 K	636 K	45,999	4	23,435
smss.exe	148	00	0:00:00	564 K	2,220 K	674	166	54
csrss.exe	172	00	0:00:04	1,860 K	2,520 K	3,184	24,758	0
winlogon.exe	192	00	0:00:07	3,392 K	7,912 K	70,305	14,750	14,371
services.exe	220	00	0:00:08	11,760 K	14,264 K	68,648	36,362,186	24,500,460
lsass.exe	232	00	0:00:06	1,448 K	4,476 K	734,646	170,967	133,533
svchost.exe	388	00	0:00:00	2,104 K	2,128 K	648	162	177
SPOOLSV.EXE	416	00	0:00:00	2,384 K	2,416 K	864	61	61
svchost.exe	480	00	0:00:01	5,928 K	6,008 K	16,628	8,534	4,138
regsvc.exe	552	00	0:00:00	816 K	824 K	235	3	3
mstask.exe	592	00	0:00:00	1,760 K	1,768 K	563	110	105
winmgmt.exe	660	00	0:00:03	364 K	4,828 K	5,497	193	24,638
igfxtray.exe	1904	00	0:00:00	3,008 K	3,008 K	755	150	150
regedit.exe	2308	00	0:00:01	228 K	3,724 K	1,108	55	26
msblast.exe	2428	00	0:00:00	1,932 K	1,932 K	486	35	32
taskmgr.exe	2508	00	0:00:00	1,776 K	1,776 K	455	0	0
mmc.exe	2672	00	0:00:00	584 K	4,448 K	1,273	10	3
explorer.exe	2716	00	0:00:04	3,424 K	9,084 K	18,635	1,356	181
hkcmd.exe	2756	00	0:00:00	2,560 K	2,568 K	642	104	104

Figure 4.3

The Windows 2000 Task Manager showing the active Blaster worm process

Note: Some malware may try and block Task Manager from starting as a form of defense. If this is the case, the **Tasklist** command line utility can be used on Microsoft Windows® XP and Windows Server™ 2003 computers (or the **Tlist** command line utility on Windows 2000 computers) to generate a simple text file list that can be copied to removable media for further analysis. Use the following command line syntax to generate a text file containing a list of all active processes:

```
tasklist /v >TaskList.txt
```

This command line will create a file called **TaskList.txt** in the current working directory.

Use the following tips to check processes on a computer that is suspected of running some form of malware:

- Check for any instances of running Telnet or File Transfer Protocol (FTP) services.
- If you are not sure of a process, use an Internet search engine such as Google to try and find some information about it.
- Check the path to the image file for processes whose image name you recognize.
- Look for both running and stopped services.

In addition to the msblast.exe process displayed in the previous figure, examples of other possibly suspicious processes include:

- ServuFTP
- Ocxdll.exe
- Kill.exe
- Mdm.exe
- Mdm.scr
- Mt.exe
- Ncp.exe
- Psexec.exe
- Win32load.exe

Note: This list is provided to illustrate examples of the type of file names that have been used in the past. Almost every attack will use a different name so it is important to be able to spot the unusual entries in the task list and to understand the naming techniques used by the malware writers.

Checking the Startup Folders

It is possible that the malware has attempted to launch itself by modifying the startup folders of the system.

Note: The precise path for these folders will change depending on the operating system being analyzed. The following information is for operating systems running Windows XP, Windows Server 2003, or Windows 2000.

There are two areas of the startup folder that you should check. The first is the **All Users** folder, which can be found at the following default location:

C:\Documents and Settings\All Users\Start Menu

The second area is the user profile path for the currently logged on account, although it is important to check all profiles that have been created on the system and not just the account that is currently logged on. You will find this information at C:\Documents and Settings*<UserName>*\Start Menu, where *<UserName>* is the logon ID of the defined users on the system being inspected.

Note: On Microsoft Windows® 95 and Windows® 98 systems it is possible for malware to rename the startup folder. For more information about this topic, see Microsoft Knowledge Base article “141900: Folder Other Than StartUp Launches Programs” on Microsoft.com at: <http://support.microsoft.com/?kbid=141900>

Check each of the entries in each startup folder to ensure no malware is attempting to start during a system startup.

Checking for Scheduled Applications

It is also possible (although rarer) that malware may try and use the Windows scheduler service to launch an unauthorized application. To confirm that this is not the case, a simple check of the scheduler queue should be performed by completing the following steps:

► To check the scheduler queue

1. Click **Start, Run**, type `cmd` to open a command prompt window.
2. At the command prompt type `at` and then press **ENTER**.
3. If there are any entries in the list check for any unauthorized or suspicious applications, create a report for future analysis using the following command:
 - a. Click **Start, Run**, type `at >C:\AT_Queue_Report.txt` and then press **ENTER**.

Executing this command will create a text file in the root of the C: drive, which should be moved to a removable disk for future analysis. Review the text file to determine if any unauthorized applications are scheduled in the queue.

Once a complete analysis of the active and scheduled processes has been completed, it may be possible to identify the process or processes that were introduced by the attack. Once these have been documented, a system reboot should be performed and the analysis repeated to determine if the attack managed to compromise other areas of the system and allowed the rogue processes to be launched at startup. If so, analysis of the system’s boot files and registry will have to be completed to find the mechanism used to maintain the rogue process or processes.

Analyzing the Local Registry

Because the completed system registry is a large and complex data store, it may be beneficial to create a copy of the entire system registry for a detailed analysis after the attack recovery process has been completed.

The Backup utility that is included with all versions of Windows can be used to back up and restore the entire registry. If you already use Backup to regularly back up your hard disk, you can easily include the registry in these backups. To back up the registry with the Backup application, select **System State** when you select the drives, files, and folders that you want to include in a backup set.

As the System State includes other system-specific information as well as the registry, these backup files can be hundreds of megabytes in size. Another option is to use the registry editor utilities that also come with all versions of Windows. These utilities are ideally suited to make copies of the registry. Windows XP and Windows Server 2003 have two registry editor tools, **Regedit.exe** and the command line tool **Reg.exe**.

Note: The Windows 2000 and Windows NT® operating systems use **Regedt32.exe** and require the **RegBack.exe** and **RegRest.exe** Resource Kit tools to provide the same functionality as **Regedit.exe** and **Reg.exe**. For more information about these tools, see the Backing up and Restoring the Windows 2000 Registry page of the *Windows 2000 Resource Kit* on Microsoft.com at:
www.microsoft.com/windows2000/techinfo/reskit/en-us/regentry/RegistryBackup.asp.

► **To make a backup copy of the registry using Regedit**

1. Click **Start, Run**, type `regedit` and then press **ENTER**.
2. In the left pane, select **My Computer**, and then from the **File** menu select **Export**.
3. In the **File name** box, type a name and location for the copy of the registry file.
4. Under **Export range**, click **All** and then click **Save**.

Detailed information on how to use **Regedit.exe** and **Reg.exe** can be found at the Registry Reference for Windows Server 2003 page of the Windows Server 2003 deployment guide at:

www.microsoft.com/resources/documentation/WindowsServ/2003/all/deployguide/en-us/RegistryBackup.asp.

Important: Because this disk will be exposed to the malware, take great care to ensure that it is not exposed to other systems until an effective method of control has been established.

Once a successful backup has been taken of the registry, check the following areas for any unusual file references:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SessionManager\KnownDLLs
HKEY_LOCAL_MACHINE\System\ControlSet001\Control\Session Manager\KnownDLLs
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current Version\Run
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current Version\RunOnce
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current Version\RunOnceEx
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Windows
("run="
    line)
HKEY_CURRENT_USER\Software\Microsoft\Windows\Current Version\Run
HKEY_CURRENT_USER\Software\Microsoft\Windows\Current Version\RunOnce
HKEY_CURRENT_USER\Software\Microsoft\Windows\Current Version\RunOnceEx
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices
HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows ("run="
    value)
```

These areas of the registry are often targeted by malicious code because they allow the malware to launch itself at system startup. For example, the W32@.Mydoom.G@mm worm added the following value:

```
"(Default)" = "%System%\<random_filename>"
```

to the following registry keys:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run  
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
```

Another area that has recently been targeted is the following key:

```
HKEY_CLASSES_ROOT\CLSID\{E6FB5E20-DE35-11CF-9C87-00AA005127ED}\InProcServer32
```

This key controls the .dll files that Microsoft Internet Explorer (**Explorer.exe**) loads. For example, the Mydoom worm and its variants would add an entry here to load a .dll file that would open a vulnerability and allow a backdoor attack.

The W32.Netsky.D@mm worm would delete this key and the following keys altogether:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\PINF  
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\WksPatch
```

Another tool that can be extremely useful for analyzing Windows XP and Windows Server 2003 based systems is the System Configuration Utility. Using this tool it is possible to view and modify a variety of startup and configuration information as well as review the current services list. More information on using this tool can be found in the Windows XP Professional Resource Kit. This information is also available online on the System Configuration Utility page on Microsoft.com at: www.microsoft.com/resources/documentation/Windows/XP/all/reskit/en-us/prmb_tol_dxth.asp

Note: You must be logged on as an administrator or a member of the Administrators group in order to use System Configuration Utility.

Checking for Malware and Corrupted Files

Most malware will modify one or more files on a computer's hard disk, and finding which ones have been affected may be a difficult process. If the system was created from an image, you may be able to compare the infected system directly with a fresh system created from this image.

If this option is not available, another method to determine which files have been changed is to use a system-wide search of all files that have changed since the malware was first introduced to the system. Such a search can be achieved using the Windows Search tool; the following screen shot shows how to narrow the search for infected files using the **Search Results** pane's advanced options.

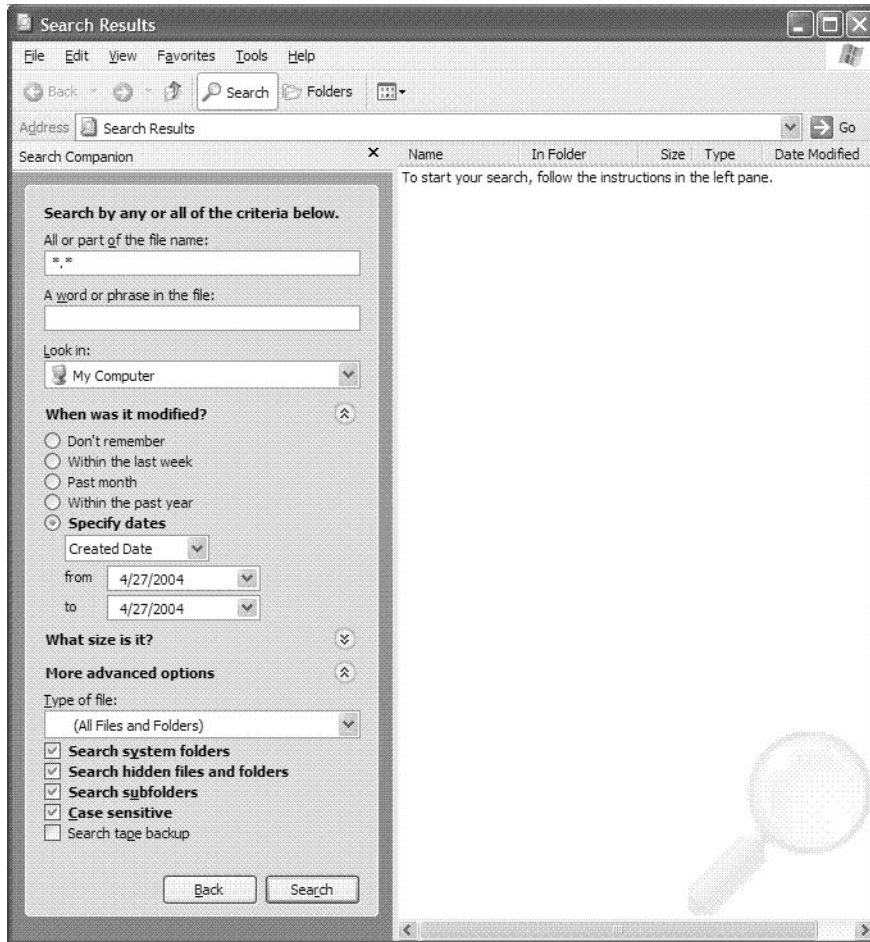


Figure 4.4
The Search Results advanced options dialog

With the options set as they are in this figure, all files that were created on the day the malware was introduced onto the host (in this example, April 27, 2004) will be listed.

It is also possible to create a text file containing a list of all files in the current directory and its subdirectories, although you should be aware that this could be a long list.

► **To create a listing of all files in a directory and its subdirectories**

1. Click **Start, Run**, type `cmd` and then press **ENTER**.
2. Change to the directory you wish to document.
3. At the command prompt, type `dir /s /-c /o:-d /t:c /q > FileList.txt` and then press **ENTER**.

Executing this command will create a text file called **FileList.txt** in the current directory, which should be copied to a removable media for further analysis.

Note: There are many other ways to create such a list using other tools and scripts. However, the aim of this section is to help gather information quickly using tools that are known to be available on the computer. If you have had time to prepare an emergency response toolkit that contains a more advanced script, use it instead of the procedure shown here.

After this search is completed, the search results can be sorted by type to help identify the executable files, which are generally the target for malware. The following list provides examples of some of the more common file types that can contain executable code:

*.exe	*.html	*.cmd	*.htm
*.bat	*.cpl	*.pif	*.pot
*.vbs	*.vbe	*.js	*.jse
*.scr	*.jpg	*.doc	*.xls
*.mdb	*.com	*.ocx	

Note: The search list may contain a large number of entries, and you may not have the time to review all modifications at this stage in the process. However, it is important to save or print a copy of this list for when you have sufficient time to review the likely target files.

The following files may indicate the presence of malware on the system:

- DLL16.ini
- DLL32.hlp
- DLL32NT.hlp
- Gates.txt

- Gg.bat
- Httpsearch.ini
- Seced.bat
- Xvpll.hlp
- Psexec.bat
- Lcp_netbios.dll

These files have been used historically by malware attacks, and are provided here to illustrate the naming techniques that have been used to attempt to hide malware files. If you are unsure of a particular file name, an Internet search can sometimes indicate the nature of a file and whether it has been linked to malware. However, it is important that such a search be performed on a system that is not infected, because Internet browsing behavior can be modified by a malware attack.

It is also important to be aware that a number of malware attacks have used valid system file names, but have placed the file in a different folder to avoid detection by the Windows File Protection service. For example, one file that has been used in the past by malware is **Svchost.exe**, which is normally installed and protected in the %WINDIR%\System32 folder. However, examples of malware creating a file of the same name directly in the %WINDIR% folder have been seen. It is important to check the full path as well as the file names.

Some of the common target areas for malware attacks to place and modify files include:

- **%Windir%**. This is a variable that is assigned to the Windows operating system default installation folder. This folder contains a number of important executable and configuration files. By default, this variable will point to the following folder paths:
 - C:\Windows (for Windows 95/98/ME/XP and Windows Server 2003 systems).
 - C:\Winnt\ (for Windows NT/2000 systems).
- **%System%**. This is a variable that is assigned to the system folder underneath the Windows operating system default installation folder. This folder contains the system files for the host operating system. By default, this variable will point to the following folder paths:
 - C:\Windows\System (for Windows 95/98/ME systems).
 - C:\Winnt\System32 (for Windows NT/2000 systems).
 - C:\Windows\System32 (for Windows XP and Windows Server 2003 systems).
- **%Temp%**. This is a variable that is assigned to the path used by applications to write temporary files. By default, this variable is assigned to the following paths:
 - C:\Windows\TEMP (for Windows 95/98/ME systems).

- C:\WINNT\Temp (for Windows NT/2000 systems).
- C:\Document and Settings*<UserName>*\Local Settings\Temp (for Windows XP and Windows Server 2003).
- **%Temporary Internet Files%**. This is a variable that is used by Internet browser applications to store temporary files during Web browsing. By default, this variable will point to the following paths:
 - C:\Windows\Temporary Internet Files (for Windows 95/98/ME systems).
 - C:\Document and Settings*<UserName>*\Local Settings\Temporary Internet Files (for Windows NT/2000/XP and Windows Server 2003 systems).

If your analysis of the files on the system uncovers any infected files, you should copy the files to removable media for future analysis. Obviously, because these files are infected, steps should be taken to ensure they are not available for anything other than the intended process. Some of the steps you might consider to help protect these copies include:

- **Changing the file name extension.** By changing the file name's extension to something unknown to the operating system, it will not be able to execute the file by an accidental click. For example, consider replacing the last letter of the file **Avirus.exe** with an underscore to make it **Avirus.ex_**.
- **Store the infected files in a protected archive.** Consider zipping the files that are infected and using a password to protect the zipped file.
- **Specialized media.** Ensure the removable media are physically identifiable from standard media by using colored disks or non-standard labels.
- **Lock files in a safe place.** Physically secure all malware sample media in a safe or some other secure storage facility.
- **Only e-mail protected archives.** If you need to send suspected malware through e-mail (for example, to an antivirus vendor), always send a password-protected archive file of the malware. E-mail gateways will be able to scan and detect the malware if it is sent as a typical unprotected attachment.

Note: Some malware attacks have used protected archives to escape antivirus scanning techniques. As a result, a number of organizations have blocked or quarantined all inbound archived files. Check that this mechanism will work for your intended recipient before sending the file.

Checking Users and Groups

Some malware attacks will try to elevate the privileges of existing users on the system or add new accounts in groups that have administrator privileges. Check for the following unusual settings:

- Odd user accounts and groups.

- User names that do not appear to fit.
- Groups that contain invalid user membership.
- Invalid user rights.
- Recently elevated privileges for any user or group accounts.
- Finally, confirm all Administrator group members are valid.

Use the Local Users and Groups Microsoft Management Console (MMC) snap-in to check for any unusual additions to the local administrators group. Also check the security log of the local computer for any unusual entries. For example, “Account Management” category entries such as event 636 indicate a new member has been added to a local group. These logs will also provide you with the date and time that the change took place.

If the system being examined is a Windows server, use the Active Directory Users and Groups MMC snap-in to examine the domain group memberships as well. For more information about default users and groups for Windows 2000, see the Default User Accounts and Groups page on Microsoft TechNet at:

www.microsoft.com/technet/prodtechnol/windows2000serv/evaluate/featfunc/07w2kadb.mspx. And the Knowledge Base article “243330: Well Known Security Identifiers in Windows Server Operating Systems” that provides information on well-known security identifier (SID)s and their associated user and group information, on Microsoft.com at:

<http://support.microsoft.com/?kbid=243330>.

Note: Although the articles describe Windows 2000, it is also relevant to Windows 2003 because the same basic default groups have not changed. However, additional default groups have been introduced by Windows Server 2003, such as the Network Service and Local Service special groups. Check your default system configuration for details.

Checking Shared Folders

Another common symptom of malware is the use of shared folders to spread infection. Check the state of the shared folders on the infected system using the Computer Management MMC snap-in or via the command line using the *NetShare* command. The following tables illustrate the default shares on Windows clients and servers.

Note: By default, Windows 9x computers do not share files or folders unless file sharing has been enabled. Also, Windows 9x clients do not have “admin\$” or equivalent hidden shares; only those folders or volumes that are specifically shared are available via the network (barring the system being compromised some way or some remote-control software being installed on it).

Table 4.1: Windows XP Default Folder Shares

Shared folder	Shared path	Comment
ADMIN\$	C:\Windows	Remote Admin
C\$	C:\	Default share
<n>\$	<n>:\	Represents a share for the root of each fixed drive on the system.
SharedDocs	C:\Documents and Settings \All Users\Documents	Will be added if local file sharing has been enabled.

Table 4.2: Windows Server 2003 and Windows 2000 Server Default Folder Shares

Shared folder	Shared path	Comment
ADMIN\$	C:\Windows	Remote Admin
C\$	C:\	Default share
<n>\$	<n>:\	Represents a share for the root of each fixed drive on the system.
SharedDocs	C:\Documents and Settings \All Users\Documents	Will be added if local file sharing has been enabled.
Wwwroot\$	C:\inetpub\wwwroot	Will be set up if Internet Information Services (IIS) has been installed as a Web server.

You can also examine the permissions on these shares with the **SrvCheck** command line tool from the Microsoft Windows Server 2003 Resource Kit Tools page online Microsoft.com at <http://go.microsoft.com/fwlink/?LinkId=4544>.

Other third-party utilities such as **Dumpsec**, which you can obtain from the SystemTools.com Web site at: www.somarsoft.com, can also be used for generating these reports.

Checking for Opened Network Ports

Many malware attacks attempt to weaken a compromised system to make it easier to attack in the future. One technique that is often used is to open network ports on the host that will then be used by the malware attacker to gain an additional route to the host.

There are a number of tools that can be used to export a list of the current network port settings, including **PortQRY** from the Microsoft Windows Server 2003 Support Tools. For more information about this tool, see Knowledge Base article “832919: New features and functionality in PortQry version 2.0” on Microsoft.com at: <http://support.microsoft.com/?kbid=832919>.

Another tool is the **FPort** command line utility from Foundstone available at: www.foundstone.com. Additionally if the computer is using a personal firewall, such

as Windows Firewall or Zone Labs ZoneAlarm®, you should check with the documentation that came with the firewall, as many of them can also show listening ports and the applications that are listening on them.

Finally you can use the **NetStat** command line utility that comes with Windows to document the state of current network connections and network ports that are listening. This tool can be used to obtain a complete printout of the network connections and port status.

► To create a NETSTAT report

- On the infected host, click **Start, Run**, type `Netstat -an >c:\netstat_report.txt` and press **ENTER**.

Note: If you are running Netstat on Windows XP or later you may wish to use the following command, which will also list the associated process identifier (PID) in your report:

```
Netstat -ano >c:\netstat_report.txt
```

A text file called **netstat_report.txt** (you may also wish to add the date to the file name) will be created in the root of the C: drive. This file should be saved to a removable media for future analysis.

Using a Network Protocol Analyzer

A network protocol analyzer tool can be used to create a network traffic log of data being transmitted to and from the infected host. The network trace file should be saved as part of the set of information files for future analysis.

Examples of network protocol analyzers that could be used for creating these network trace files include the Network Monitor component of Microsoft Systems Management Server (SMS), or other third party tools such as the Ethereal analyzer that is available from the Ethereal Web site at: www.ethereal.com.

Checking and Exporting System Event Logs

It may be possible to use the Windows system event logs to spot a wide range of unusual behavior that could be used to identify both the changes malware has made and when they were made. Use the Event Viewer management console to save each type of event log file (Application, Security, and System) to removable media for further analysis. By default, these files are stored in the `C:\Winnt\System32\Config\` directory and are called **AppEvent.evt**, **SecEvent.evt**, and **SysEvent.evt**. However, while the system is active these files are locked and should be exported using the Event Viewer management tool.

The following tips provide information on how these logs can be used to help determine the effects of a malware attack:

- Look for any changes at the time of the suspected attack.
- Compare event log times with file creation and modification times.
- Look for accounts that were created or had a password changed around the time of a suspected intrusion.

At the end of the malware analysis process it may be possible to consider reconnecting the isolated networks, depending on the nature of the malware. For example, if the analysis determines the malware spreads only via a particular peer-to-peer (P2P) application, changing the perimeter firewall filters to block the network ports used by this application would allow the networks and other services to be restored. Such a remedy would allow the organization to return to some level of normal communications while the system recovery process was undertaken.

Step 4: System Recovery

After you have collected the required information about the attack and understand its full nature, you can start the process of removing the malware and recovering any corrupted data from the infected computers.

Important: Even if you have an antivirus application that can recognize and clean a malware attack from a computer, Microsoft recommends spending some effort to determine the date and time of the infection as well as how the infection occurred. Without this information it is difficult to determine which other systems, backup media, or removable media were possibly exposed to the attack.

How you complete this process will largely depend on the nature of the particular malware attack. However, you can use the following high-level process to ensure a complete recovery of both data and your computer systems:

1. Restore missing or corrupted data.
2. Remove or clean infected files.
3. Confirm your computer systems are free of malware.
4. Reconnect your computer systems to the network.

Confirming the system is free of malware is a crucial step that should not be overlooked. Many malware threats are designed to remain undetected for extended periods. In addition, backup images or system restore points could contain infected system files, which would cause another infection if an infected backup image is your recovery source. For these reasons, it is vital to ascertain the date and time of the first instance of the malware attack if at all possible. Once you have a time stamp as a benchmark, you can determine through the dates of your backup images as to whether any of them are likely to contain the same malware corruption.

Clean or Rebuild?

Two choices are available to you when considering how to recover your system. The first option is to clean your system, which relies on the known characteristics of the attack to systematically undo the damage inflicted by each. The second choice is frequently referred to as rebuilding or *flattening* a system. However, deciding which option to use is not a simple choice.

You should only choose to clean your system if you are extremely confident that all elements of the attack have been well documented, and that the cleaning procedure will remedy every element of the attack successfully. An antivirus vendor will usually provide the documentation you need, but it may take the vendor several days to fully understand the nature of the attack. Cleaning the system is often preferred because it returns the system to its clean state with applications and data intact. This approach typically results in a faster return to normal operations than rebuilding the system. However, without a detailed analysis of the malware code, cleaning the system may not entirely remove the malware.

The fundamental risk of cleaning a system is the possibility that either an undocumented element of the initial infection — or potentially a secondary infection or attack — may not have been discovered or documented, leaving your system still infected or susceptible to some malware mechanism. Because of this risk, many organizations choose to simply rebuild their infected systems to absolutely ensure that they are free of malware.

In general, whenever a system has suffered an attack where a backdoor or rootkit was installed, Microsoft recommends rebuilding the system. For more information about these kinds of attacks, see Chapter 2, “Malware Threats” in this guide. The various components of these types of attacks are difficult to detect reliably, and will frequently recur after attempts to eradicate them. These attacks are often used to open unauthorized access to a compromised system, which may enable them to initiate additional attacks on the system to escalate their privileges or install their own software. For these reasons, the only way to be absolutely sure that your computer systems are free of these malware attacks is to rebuild them from trusted media and configure them to remediate the weakness that allowed the attack in the first place, such as a missing security update or weak user password.

This process also requires carefully capturing and measuring all the necessary user data from the infected system, fixing anything corrupted, scanning it to ensure the data does not contain any malware, and finally restoring the clean data back to the newly rebuilt system.

Rebuilding a system also requires reinstalling all of the applications previously available on the system, and then configuring each one appropriately. Therefore, rebuilding provides the highest degree of assurance of eliminating the infection or attack, but generally is a much larger task than cleaning.

The primary consideration in choosing which option to use on your system should depend on your level of confidence in the one you select to completely eliminate and resolve the infection or attack. The down time required during the repair should be a secondary consideration compared to ensuring the integrity and stability of the system.

Table 4.3: Advantages and Disadvantages of System Cleaning and Rebuilding

Cleaning	Rebuilding
Simple process, if cleaning tools are available.	More complex process, especially if a backup and recovery solution is not in place prior to the infection.
Fewer steps to ensure data is clean.	More steps necessary to capture, backup, clean, scan, and restore data.
Fewer resources required to use removal tools than to rebuild entire systems.	The rebuilding process is likely to consume a significant amount of time and resources to complete.
Risk of system still being infected.	Little risk of system still being infected if restored from clean media and adequately managed data.

Note: If you choose to clean an infected system, your organization's management and legal teams should perform a risk analysis to determine if they are willing to accept the increased risk of a future attack if the cleaning process misses part of the malicious code.

System Cleaning

You should only consider system cleaning as a viable option if the attacks and behavior of the malware are well documented and the cleaning procedures have been tested and proven. Thoroughly documented steps administrators can follow or automated tools that clean the infection from your system may be available from either Microsoft or antivirus vendors. Both options are intended to carefully undo each of the actions performed during the infection and return your system to its original operational state. These procedures generally only become available to address major viruses or worms, and typically only several days after the initial malware infection.

Note: Since many malware attacks are released in waves, for example MyDoom@A, MyDoom@B, and so on, it is very important to only use cleaning procedures or tools to clean the specific version of the malware from your system.

If an automated tool is not available to address the malware you are dealing with, the basic steps to consider if you opt to manually clean it from your system include the following:

1. Stopping the malware execution processes. You must terminate any currently running malware related process, as well as any auto-run entries or scheduled tasks associated with the malware you remove.

2. Removing the introduced malware files. This step will require a detailed analysis of the files on the host hard disk drives to determine which files were affected by the malware.
3. Applying the latest security updates or patches to mitigate the vulnerabilities that the original attack exploited. This step may require a number of reboots and visits to the Windows Update Web site to ensure that all security updates are applied.
4. Changing any passwords (domain or local) that may have been compromised, or ones that are weak and easily guessed. For guidance on setting strong passwords see the Strong Passwords page on Microsoft.com at:
www.microsoft.com/resources/documentation/WindowsServ/2003/enterprise/proddocs/en-us/windows_password_tips.asp.
5. Undoing any system changes the malware introduced. This step could involve restoring the local hosts file and firewall configurations on the computer.
6. Restoring user files modified or deleted by the malware.

If you decide to manually undertake these steps, you should only rely on them as a remedy for the infection if you can later compare them with published cleaning procedures to ensure that you have performed all of the necessary steps. Or, if your organization has an antivirus support team, it will also need to ensure that the inspection and remediation procedures it uses to identify and mitigate all possible attack vectors are adequate. Failure to ensure your procedures are adequate could lead to a rapid re-infection.

Restore or Reinstall?

If you determine the best approach is to rebuild your system, you can either restore it using a previous image or system backup you are certain is clean or reinstall the system from original media.

If you choose to restore the system from a previous image, consider attempting to salvage the latest user data on the infected system to avoid losing changes created or updated between the time of the backup and the present. If you rebuild the system from original media rather than a backup, your only option to prevent data loss is to preserve the data from the infected system before backing it up.

Recovering Data from the Infected System

The most valuable asset of your system is most likely the data that resides on it. For this reason, it is crucial to carefully consider how to save, restore or repair the data, back it up, and then restore it on the system after it has been rebuilt.

Be sure to capture all of the following types of data appropriately to completely restore your system:

- **Operating system configuration data.** This data includes all configuration settings required to restore the host operating system to its original state to enable all services of the host to function correctly.
- **Application data.** This data includes all data that is used and stored by the applications that are installed on the host device.
- **User data.** This data includes all configuration data, such as user profiles and user generated files.

Note: This preserved data obviously presents a serious risk of being infected itself. A high level of care should be taken when working with this data until a reliable method of checking the data for the malware has been identified.

Back up all of the data to a safe medium or location where it cannot be executed or accessed by unauthorized users or systems. If necessary, use whatever tools or other means are available to restore the data, and then safely store it until you can restore it on the system after it has been rebuilt.

Restoring From an Image or Backup

To restore data from an image or backup, you must have previously captured it using a recovery tool before the infection compromised your system. A wide variety of tools are available that may dramatically simplify the task of backing up and recovering data from your systems. These tools provide a high level of insurance to protect your systems against not only malware infections, but also hardware failures and other potential threats to your system. Configuring a complete disaster recovery infrastructure is not within the scope of this guide. However, a few key technologies in this area that you can use to address antivirus-related issues are discussed in the following sections.

Windows System Restore

Windows System Restore (WSR) protects critical system and application files by monitoring, recording, and in some cases backing up these files before they are modified. It is important to know if your antivirus application supports WSR, because WSR can create a restore point that could become infected with malware if you used it to clean a system any time after the initial malware attack. If this is the case, it is possible that the malware could be re-introduced to the system from the infected restore point. Fortunately, a WSR-aware antivirus application will detect the malware during a restore process. If any infected files are detected, the antivirus solution will attempt to modify, move, or delete them. If the files are successfully cleaned, WSR will restore the files in question. However, if a file cannot be cleaned and is deleted or quarantined, the restoration process will fail because isolating a file

results in an inconsistent restore state. If this is the case, WSR will revert the system back to its previous state (before the restore operation began).

For more information about how antivirus applications can work with this service, see the Knowledge Base article “831829: How antivirus software and System Restore work together,” on Microsoft.com at: <http://support.microsoft.com/?kbid=831829>.

Note: As virus signature files are updated to cover a malware attack, a restore that failed days before may now succeed (after the antivirus application is updated). Conversely, if you restore to a point that succeeded before but a new signature file enables the detection of an attack on a backed up file that cannot be cleaned, the restore process could possibly fail.

For more information about Windows System Restore, see the How to Restore Windows XP to a Previous State page on Microsoft.com at: www.microsoft.com/windowsxp/pro/using/itpro/managing/restore.asp.

Automated System Recovery

Automated System Recovery (ASR) provides a simple means to quickly back up both the boot volumes and system volumes on your computer, which will enable you to more rapidly restore your system in the event of an infection or failure. However, just like other backup media, it is possible that the ASR backup files could become infected by the malware. For more information about ASR and how you can use it in your organization, see the “How ASR Works” white paper on Microsoft.com at: www.microsoft.com/resources/documentation/WindowsServ/2003/all/deploymguide/en-us/sdcbc_sto_axho.asp.

The Windows Backup Solution

The backup solution that is supplied as part of the Windows family of operating systems provides a simple backup solution for departmental or small- to medium-sized business environments. However, just like WSR and ASR, the backup files themselves can contain infected malware. For this reason, ensure that you do not restore the malware to your system and restart the malware attack if you use this solution. All backup files should be checked and scanned with an updated antivirus application that is capable of detecting and removing the malware before you use the backup image to restore your system. You will find detailed documentation on disaster recovery, including backup and restore operations, in the Planning for Disaster Recovery section of the *Windows Server 2003 Deployment Kit* on Microsoft.com at: www.microsoft.com/resources/documentation/WindowsServ/2003/all/deploymguide/en-us/sdcbc_sto_gqda.asp.

Reinstalling the System

Once you know the backup data for your system is trustworthy, you can start the process of rebuilding your system. This point in the process is the best time to reformat drives, change partition sizes, and perform other system maintenance as required to ensure the optimal performance of your system after it is restored. If possible, rebuild your servers using a fully updated slipstreamed share. More information on creating slipstreamed installs of Windows can be found in:

- The “Combination Installation” section of the *Microsoft Windows XP Hotfix Installation and Deployment Guide* on Microsoft.com at:
www.microsoft.com/WindowsXP/pro/downloads/servicepacks/sp1/hfdeploy.asp#the_combination_installation_gxsi.
- The “Installing Windows 2000 with the Service Pack and Hotfixes” section of the *Windows 2000 Hotfix Installation and Deployment Guide* on Microsoft.com at:
www.microsoft.com/windows2000/downloads/servicepacks/sp3/HFDeploy.htm#installing_windows_2000_with_hotfixes_ykot.

If you cannot rebuild from a slipstreamed source, the risk is that the system will get infected from network-based malware before you can connect to the Windows Update Web site to download critical service packs and security updates. If this is the case, use the following steps to reinstall:

1. Disconnect from the network. Physically unplugging the computer from the network is the safest approach.
2. Install the operating system from the original system installation media. During this process it is imperative that you create strong local administrator passwords for each machine. These passwords should be unique to each machine.
3. Start the system and log on using the local administrator account.
4. Activate a host-based firewall on the system, such as the Windows XP Internet Connection Firewall (ICF).

Note: In Windows XP Service Pack 2, ICF has been renamed to the *Windows Firewall* and is enabled by default on all network connections. If the system is based on Windows 2000 or earlier, it is recommended that a third-party host-based firewall be installed.

5. Reconnect the system to the network. At this point it is important to perform the following steps as quickly as possible to minimize the risk to the system being rebuilt.
6. Update the freshly installed system with the latest software updates. It’s important to note that not all security updates are offered by the Windows Update Web site. Only core operating system security updates are offered by Windows Update; updates to other products such as SQL Server™, Front Page, Commerce

Server, and so on, will not be offered by Windows Update. For this reason, you should visit the Microsoft Security Bulletin Search site on Microsoft TechNet at: www.microsoft.com/technet/security/current.asp to check for product-specific updates.

7. Install an antivirus package, ensure it is using the latest version of the virus signature file, and perform a complete antivirus scan of the system.
8. Harden the configuration of the system using the latest hardening guidelines for the organization. See Chapter 3, “Antivirus Defense in Depth” of this guide for information on this process.
9. Check the system for any remaining vulnerabilities using a vulnerability scanner such as the Microsoft Baseline Security Analyzer (MBSA). This free tool is available for download from Microsoft.com at: www.microsoft.com/technet/security/tools/mbsahome.msp.

After you have rebuilt the system and scanned it to confirm there are no longer any infected files on it, it is safe to restore the user data.

Step 5: Post Recovery Steps

This section provides guidance on specific steps you should take after controlling and recovering from the initial malware attack. It is important to complete this stage to help strengthen your organization’s overall policies for people, processes, and technologies.

Post Attack Review Meeting

This meeting should include all affected parties and call for a free exchange of lessons learned for the benefit of all. Specifically, participants should seek to:

- Work with legal counsel to determine whether your organization should pursue legal steps against the attack perpetrators.
- Work with legal counsel to determine whether your organization should report the attack to the authorities if sensitive data was compromised. For example, credit card information.
- Assign a monetary value to the damage the attack caused for internal reporting that includes the following elements:
 - The hours spent on the recovery.
 - The cost to repair damaged equipment.
 - Revenue loss.
 - The cost or damage to customer and partner relations.
 - The amount of lost productivity from affected workers.
 - The value of any lost data.
- Try to identify any system vulnerabilities the attack used to exploit your systems.

- Recommend changes to your organization's antivirus defense-in-depth policy.
- Recommend changes to your organization's security policy, including:
 - A refined default password policy.
 - Audit policies.
 - Security updates policy.
 - Firewall policies.

Post Attack Updates

Review and evaluate whatever recommendations result from the meeting, and then ensure that they are implemented as soon as possible across your organization. Once a particular vulnerability has been exposed, there are often a number of approaches you can use simultaneously to mitigate it.

It is important to understand that these changes are likely to affect the people, processes, and technologies of your organization. Reviewing the estimated cost of the attack to the organization should serve to underscore the future cost benefit your organization can realize by proactively working to prevent a reoccurrence of the attack.

At this point, if your organization has not already implemented an antivirus defense in depth approach, see Chapter 3, "Antivirus Defense in Depth" in this guide to review which elements of this approach will benefit your organization the most.

Summary

This chapter provided guidelines and recommendations that you can use to recover from a malware attack in a considered and consistent manner. It is important to follow the suggested steps consistently, as failure to do so may leave your organization open to further attack from malware. Failure to do so may also make it difficult or impossible for your organization to take legal action against the perpetrator of the attack.

If your organization has implemented an antivirus defense-in-depth solution, the number of times you will need to mitigate attacks with it will likely be kept to a minimum. However, failure to plan on how to address worst-case scenarios in advance will leave your organization open to making serious errors if an attack succeeds in breaching your antivirus defenses.

You should prepare for this in advance by training security staff to understand common malware techniques, such as those covered in this chapter. Also consider creating a malware analysis toolkit that contains some of the tools described in this chapter, as well as any scripts or other utilities that can be used to quickly capture and document vital information from infected systems. This preparation will help

reduce the impact on your business operations if systems become subject to a malware attack.

Each new attack may introduce different methods to compromise or corrupt your systems. Therefore, Microsoft strongly recommends monitoring the Microsoft Security Antivirus Information Web site at: www.microsoft.com/security/antivirus/. This site will provide you with up-to-date antivirus information and guidance on how to address the latest malware attacks. Using the resources in this chapter will help you to effectively control the impact a malware outbreak may have on your organization, and to recover from it in an efficient and reliable way.

