



realtimepublishers.com®

*The Administrator
Shortcut Guide™ To*



Email Protection

Paul Robichaux

Introduction to Realtimerepublishers

by Don Jones, Series Editor

For several years, now, Reptime has produced dozens and dozens of high-quality books that just happen to be delivered in electronic format—at no cost to you, the reader. We’ve made this unique publishing model work through the generous support and cooperation of our sponsors, who agree to bear each book’s production expenses for the benefit of our readers.

Although we’ve always offered our publications to you for free, don’t think for a moment that quality is anything less than our top priority. My job is to make sure that our books are as good as—and in most cases better than—any printed book that would cost you \$40 or more. Our electronic publishing model offers several advantages over printed books: You receive chapters literally as fast as our authors produce them (hence the “realtime” aspect of our model), and we can update chapters to reflect the latest changes in technology.

I want to point out that our books are by no means paid advertisements or white papers. We’re an independent publishing company, and an important aspect of my job is to make sure that our authors are free to voice their expertise and opinions without reservation or restriction. We maintain complete editorial control of our publications, and I’m proud that we’ve produced so many quality books over the past years.

I want to extend an invitation to visit us at <http://nexus.realtimerepublishers.com>, especially if you’ve received this publication from a friend or colleague. We have a wide variety of additional books on a range of topics, and you’re sure to find something that’s of interest to you—and it won’t cost you a thing. We hope you’ll continue to come to Reptime for your educational needs far into the future.

Until then, enjoy.

Don Jones

Introduction to Realtimepublishers..... i

Chapter 1: Email Content Dangers1

Is the Sky Falling?1

Just the Facts, Ma’am2

Virus 101.....3

 What Is a Virus?.....3

 Virus Types.....4

 Virus Names.....6

 Why Create a Virus?.....6

 A Brief History of Viruses7

 What Does the Future Hold?.....10

 Are Viruses Dangerous?10

 Hoaxes and Urban Legends11

Email-Based Viruses.....12

 Melissa13

 Happy9913

 PrettyPark.....14

 Mylife.....14

 Nimda.....15

Scanning for Viruses.....15

 A Matter of Trust16

An Ounce of Prevention.....17

Summary18

Chapter 2: Protection at the Client Level.....19

Virus Entry Points.....19

Securing the Desktop—Client Protection 10120

 Choosing Antivirus Client Software.....20

 Common Vendors of Antivirus Client Software21

 General Recommendations22

 Netscape Mail22

 Qualcomm Eudora22

Outlook Security Update.....23

 Understanding the Outlook Security Update23

Demoting Attachments to Level-2.....29

Installing the Office Update for Outlook 2000.....30

Desktop Configuration Best Practices31

Outlook Macro Security.....31

Quelling Active Content33

 The Preview Pane33

 Viewing the Message.....34

 Reading Messages as Plain Text.....38

Centralizing Some Client Security Features39

 AD to the Rescue!.....39

 Using the Outlook Security Features Administrative Package.....40

 Where Are My Custom Settings?49

 Final Thoughts About Outlook Security Settings.....49

 Using Exchange Server to Control Outlook Client Versions49

Summary52

Chapter 3: Server-Side Antivirus Protection53

Basics for Protecting Your Organization.....53

 Properly Configuring the Firewall53

 Avoid Directly Publishing Exchange Server Resources.....54

 Employing a Multi-Layered Virus Protection Strategy55

 Differing Scanning Policies56

 Updating the Software57

 Applying Exchange Restrictions.....58

 Protecting Mail-Enabled Groups58

 Restricting Message Size and Recipient Count59

 Limiting Mailbox Storage.....60

 Scanning and Blocking File Attachments.....61

The Exchange Antivirus API.....63

Designing a Server-Based Protection Scheme.....64

 Exchange-Aware Antivirus Vendors65

 File-Based Virus Scanners.....65

Monitoring Exchange Server Virus Protection.....66

 Monitoring Windows Performance Counters66

Examining the Windows Application Event Log68

Containing a Virus Outbreak71

 Locking Users Out of the Exchange Server.....72

 Cleaning Up Queue Directories72

 Cleaning the MTADData directory74

 Getting Rid of the Virus from the Stores74

Best Practices for Virus Protection.....80

Summary80

Copyright Statement

© 2006 Realtimepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimepublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimepublishers.com, Inc or its web site sponsors. In no event shall Realtimepublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimepublishers.com and the Realtimepublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimepublishers.com, please contact us via e-mail at info@realtimepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library. All leading technology guides from Realtimepublishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 1: Email Content Dangers

On March 26, 1999, email administrators were awakened to a new threat to their systems. A macro virus called Melissa was spreading rapidly through their email servers, clogging queues and eating up network bandwidth. By March 29, Melissa had reached more than 100,000 computers. Macro viruses were nothing new by this time, yet this one was different. Melissa was the first major virus to use the Exchange global address list (GAL) and the users' contacts as a mechanism to email itself to others. Though almost any virus can be spread through email if the *dropper* (a program that is infected or spreads the virus) is emailed to another person, this incident was the first time that a virus actually started emailing itself to others.

With an estimated 772 million mailboxes worldwide (Anti-Virus, Anti-Spam, and Content Filtering Market Trends 2003-2007 Report from The Radicati Group), email is an excellent delivery vehicle for computer viruses and malicious content. Knowing how to protect your email systems is critical, and doing so begins with a good understanding of what viruses are, how they're written, how they spread, and how they can be blocked.

Is the Sky Falling?

During the Melissa outbreak, some computer media journalists and industry analysts screamed "The sky is falling! Viruses are the number one threat to computer security! Email servers are going to start crashing worldwide!" There's no doubt that viruses are costly. *Computer Economics* estimated in 2000 that damages relating to virus outbreaks were in excess of US\$17 billion (yes billion!) and that the Love Bug virus alone has cost organizations more than \$8.75 billion dollars to fight and eradicate. Some experts call these numbers conservative while others dismiss these numbers as wild guesses. Are these claims valid? Tallying the actual cost of an outbreak of any virus is difficult, if not impossible, and an exact cost of any type of virus outbreak is impossible to gauge.

What is fact is that outbreaks of email-based viruses such as the Love Bug, Anna Kournikova, BugBear, and SirCam have caused email systems operated by the United States government, Microsoft, EDS, and others to be shut down, inbound and outbound SMTP to be halted, and IT staff to spend countless hours devoted to clean up. The cost of not having email available is compounded by the fact that for many organizations, the mail system incorporates users' calendars, to-do lists, contacts, faxing functionally, pager gateways, and more. In addition, many email-based viruses not only replicated throughout the internal organization, but also to customers and vendors. In many cases, these viruses have continued to propagate for hours or days after their discovery as a result of email administrators' basic lack of virus knowledge.

Is the sky falling? Not by a long shot. But when the industry analysts, pundits, and media commentators all expound on the topic, even if they don't all agree, it provides evidence that the threat of email-based viruses must be taken seriously. Connecting any email system, especially a system with Microsoft Outlook clients, to the Internet would be foolhardy and irresponsible without a good virus-protection plan in place.

An administrator must be prepared as viruses are taking new forms all the time. “Traditional” viruses spread through floppy disks, then later through email, and now through Web services. “New-fangled” viruses are already finding themselves at home on PDAs, wireless devices, and instant messaging clients. Worms such as Nimda are exhibiting characteristics of both worms and viruses by finding multiple ways to infect a computer, then spreading the infection to other computers. Most Windows and Exchange administrators have only a cursory understanding of viruses, how they spread, and—most important—how to protect their organizations from the differing strains and types of viruses out “in the wild.” Fighting viruses is a little like fighting in a darkened room with an opponent who has night vision goggles—you don’t know from which direction the next attack will come nor how you will be attacked, you just know that there will be another attack.

Just the Facts, Ma’am

Although there are many unknowns in the world of viruses, there are at least a few knowns. The following list highlights a few of these facts (and commonly held opinions masquerading as facts) about viruses and some recent virus outbreaks—perhaps you can use them to justify the purchase of virus protection for your organization:

- The first IBM PC-compatible virus, Brain emerged in 1986 from Lahore, Pakistan; it was written by two brothers, Basit and Amjad Alvis. This virus was only 3.5KB and infected the boot sectors of 360KB floppies. By comparison, the Slammer worm was smaller than 400 bytes.
- In 1990, it is estimated that there were fewer than 500 DOS/Windows viruses in existence, including exceptionally rare viruses. By 1994, there were more than 5000 viruses, and in mid-2003, Symantec estimates that its antivirus software detects more than 64,000 viruses.
- Antivirus software company Sophos estimated in 2002 that there were more than 70,000 viruses: 26.1 percent were macro viruses, 26.1 percent were Trojan horses, 19.2 percent were executable, and 6.8 percent were script viruses. The remaining 21.8 percent were boot sector, worms, file, UNIX, and Macintosh viruses.
- The Cooperative Association for Internet Data Analysis (CAIDA—<http://www.caida.org>), who analyzes and develops tools for the Internet infrastructure, estimates that the Code Red worm affected more than 359,000 hosts during the first 14 hours of its spread. At its peak, it was spreading at approximately 2000 hosts each minute. As if that weren’t scary enough, CAIDA estimates that the Slammer worm at its peak was scanning 55 million systems *per second*. It is interesting to note that both the Code Red and the SQL Slammer worms exploited well-known vulnerabilities in software for which fixes had been published *months* before the worms were released.
- Sophos estimates that it analyzes about 1200 new viruses each month.
- The 2002 SQL Slammer worm spread worldwide in approximately 10 minutes. In the early stages of infection, it was doubling the number of infected hosts every 8.5 seconds.
- The British version of the magazine *PC Today* distributed a copy of its magazine with a floppy disk to more than 50,000 people. The floppy accidentally contained a copy of the DiskKiller virus.

- Some parent viruses can spawn more than 50 varieties of child viruses that cannot be detected by the same virus signature as their parent. Many virus protection companies consider these children to be a separate virus and, thus, these companies report larger numbers of detected viruses.
- The SANS Institute (<http://www.sans.org>), one of the world's leaders in security research and education, estimated in late 2001 that more than 86,000 hosts on the Internet had been compromised and had helped spread the Nimda worm. About 43 percent of these hosts were in the United States.
- SecurityPortal.com, a company that tracks piracy and safety news for both individuals and companies, estimate that more than 40 variations of the Love Bug virus existed within a year after it was unleashed on the Internet. Macro viruses, such as the Love Bug, include their source code, so it is easy for a recipient to modify them and release the modified version into the wild.
- The British Broadcasting Corporation (BBC) estimated that the Love Bug virus affected at least 45 million users.
- A San Francisco FBI Computer Intrusion Squad survey summarizes that 273 companies that responded to the survey had *quantifiable* losses in excess of \$265 million as a result of computer viruses.
- Market research company The Radicati Group (<http://www.radicati.com>) estimates that malicious code (viruses, Trojans, and worms) will cost more than \$28 billion in 2003. By 2007, that cost will nearly triple to an estimated \$75 billion. A 2002 survey by the company found that protecting against viruses was the number one priority of email systems administrators (reducing spam was second).
- Viruslist.com (<http://www.viruslist.com>) estimated that the undisputed leader in virus threat sources was email with a whopping 96.4 percent of all infections. The Internet accounted for 2.3 percent and portable media accounted for 1.3 percent.

Virus 101

So you know that viruses are a very real threat, but perhaps you don't know an email-based virus from a polymorphic logic bomb. If you don't know how viruses are developed or even how they propagate, don't worry—this section is for you.

What Is a Virus?

Like viruses in carbon-based life forms, a computer virus is capable of replicating itself within the host computer and finding a method to move from one host to another. Viruses reproduce and spread without users' consent, and usually without their knowledge.

Viruses are programs that are usually small and difficult for the layman to detect. When executed or accessed, a virus loads itself into the computer's memory, overwrites another program, attaches itself to another program, or writes itself to the master boot record of the computer. The virus might take other actions that can be malicious, innocent but annoying, or innocent but accidentally destructive.

Viruses differ from worms in an important way: virus infections are persistent because the virus writes itself to permanent storage, like files or documents on disk or messages in an Exchange mailbox database. Worms are transient; if you turn off an infected computer and reboot it, it won't be infected again (as long as you've fixed the security hole that let it become infected in the first place).

In the early days of viruses, viruses were usually written with low-level programming tools such as assembler or C. As a result, viruses could be especially small and efficient, and were able to infect components of the computer that a script could not. Most of the more annoying viruses recently have been written using some variant of a scripting language, such as VBScript or JavaScript, although worms are usually still written in low-level languages. The fact that many viruses are now script-based also means that less-skilled programmers can write viruses.

After the introduction of Microsoft Office 95, macro viruses became increasingly prevalent. These viruses use the built-in Office scripting language (now called Visual Basic for Applications—VBA) to do their dastardly work. The most common macro-type viruses were the Concept family of viruses and Wazzu; both of these viruses were Word macro viruses (they attach themselves as macros to Word documents). On each new computer, these macro viruses copy themselves to the master template, normal.dot. From that point forward, any document created or edited on that computer will contain the macro virus as well.


Usually document content was not altered by macro viruses; these viruses were little more than an annoyance, but oh, what an annoyance! Increased document sizes were occasionally problematic as was document corruption; however, the biggest annoyance (and embarrassment!) was to accidentally send an infected document to a customer or vendor. By the late 1990s, infected Word documents (and data files created by other Microsoft Office suite of applications) were a daily occurrence for IT personnel. I even received a CD-ROM from Microsoft that had documents with the Concept virus. Office 2000 and later versions have some significant protections against macro viruses, which we'll explore in a later chapter.


Virus Types

Viruses are classified in several ways. Many of the “viruses” in the wild today are not truly viruses but worms or Trojan horses; many worms in the wild have characteristics similar to viruses. (The term *in the wild* refers to viruses that are currently spreading to production systems rather than a virus that has only been identified in a lab environment.) This section has some basic definitions for virus types; they are not universally agreed upon by all virus experts, so don't be surprised if you come across different definitions from antivirus software companies:

- *Malware* is a broad term used to describe any type of malicious code. This code could be in the form of a virus, worm, Trojan horse, hostile Java code, hostile Web script, or hostile ActiveX control. Malware may disrupt the computer's operation, replicate itself to other computers, steal information, or initiate a denial-of-service (DoS) attack.
- A *Trojan horse* is not usually a virus but rather a program that appears to do one thing but in reality is doing some nefarious task behind the scenes. The program may claim to be a fun game, and in reality, it is searching your hard drive for important data files and mailing them to an outside person. Users are often tricked into running a Trojan horse. Trojans may be just as destructive as a virus, but they don't have the ability to reproduce themselves. The most common use of Trojans is to allow an attacker remote access or control over your computer.


- A *worm* is a program that usually has its own built-in method of replication. It is not associated with any single host program and actively searches for systems to infect. Nimda and Code Red are examples of worms.
- A *macro virus* is not a standalone program but rather code (usually scripts) embedded in a data file such as a Microsoft Word document or Excel spreadsheet. When the document is opened, the macro is run. Macro viruses are generally easier to write and spread quickly. In 1998, it is estimated that there were about 1000 macro viruses. Today, it is estimated that about 75 percent of all viruses are macro viruses. Well-known macro viruses include the Concept virus and the infamous Melissa virus.
- A *blended threat* is malware that combines characteristics of worms, viruses, and Trojan horses. By using a combination of methods and techniques, this type of malware can spread more quickly. Many of the virus/worms that have caused the most harm in the past few years have been blended threats.
- A *dropper* is a program that is responsible for depositing a virus' malicious code on a target system. The dropper itself may be a virus or it may be a Trojan horse.
- A *stealth virus* is a virus that attempts to conceal itself. Most viruses are stealth to at least some degree. Some viruses will even protect the area of the disk on which the virus is stored.
- A *polymorphic virus* is a virus that goes to great lengths to conceal itself by changing itself every time it replicates. These viruses make it more difficult for antivirus software to detect and remove them.
- *Adware* and *spyware* cannot really be considered viruses, but they are often annoying and can offer potential security risks. Adware is a term used for a program that resides on your computer and sends pop-up advertising to the console; the adware program may also install spyware. Spyware is generic term for software that is designed to sit on your computer and monitor your computer usage. The software then reports that information back to the organization that originally installed the software. Although the violation of privacy is enough to concern most computer users, the potential for even greater harm exists. Any program that runs in the background on your computer has the potential to scan for sensitive information, collect keystrokes, and make modifications to your computer. Spyware-type programs are not currently spread through email; they are downloaded when a user loads something from a Web site. File-sharing giant Kazaa is notorious for placing Spyware that changes default start pages in the Web browser of users' computers and sends pop-up advertising. These programs can also contribute to computer performance problems and can actually break some real software if the Spyware is removed. Spyware that does so is often referred to as *scumware*.
- *Companion viruses* rename an existing executable file to something else, then put themselves into that program's place. When the program is executed, the companion virus first executes, then runs the intended program.
- *Keystroke loggers* are not specifically worms or viruses. A keystroke logger can either be hardware or software. The keystroke logger captures all keystrokes that a user types when using a computer, including anything typed in to a document or spreadsheet as well as usernames and passwords.

 The evolution of spyware and adware is just getting rolling. Expect to see this type of threat become more prevalent on both personal home computers as well on the corporate desktop computer. You can find more information about spyware and adware at <http://www.adware.info> and <http://www.cexx.org/adware.htm>.

 Lavasoft is a small company that provides a tool called Ad-Aware that can detect and remove spyware and adware from your computer. You can find more information at <http://www.lavasoft.de>.

Virus Names

You might be confused about why you hear viruses, worms, and Trojan horses referred to by different names. For example, the SQL Slammer worm was also known as the W32.SQLExp.Worm, DDOS.SQLP1434.A, W32/SQLSlammer, Sapphire, and W32/SQLSlam-A. Contrary to popular opinion, the virus author often does not name the virus. The reason viruses are often named so differently is that each antivirus software vendor assigns the virus a name. The companies rarely pick a name that has much to do with the name that the virus author might have assigned to the virus; they usually pick a name that will help to identify the virus, the platform it affects, and the type of virus. Therefore, tracking a virus can become confusing if you subscribe to notification lists from more than one vendor. Alternative names for viruses can be found on many antivirus vendors' Web sites.

 A good location to cross-reference virus names is the vgrep page at <http://www.virusbtn.com/resources/vgrep>.

Why Create a Virus?

The question that begs to be answered is why would someone write a virus? Most of the viruses that I have come in to contact with were written by someone with a good degree of programming and computer skills. Why would a skilled programmer waste time writing a virus that inconveniences hundreds or even millions of computer users?

I suspect that most viruses have been nothing more than a challenge for a bored programmer. Virus programmers might want to see whether they can achieve a particular result by writing something a certain way. Or they might want to see how far the virus will spread. The virus writer might want to exploit a weakness in a particular platform or application.

Other virus writers have a more mischievous perspective. They don't want to cause any real harm but see their creations as pranks or practical jokes. The financial costs associated with cleaning up their mess are probably lost on these programmers. (For an example of such a virus artist, see the sidebar "You've Lost that Lovin' Feeling.")

You've Lost that Lovin' Feeling

Onel de Guzman, a 24-year-old college dropout in Manila, "accidentally" unleashed the email-based Love Bug (ILOVEYOU) computer virus on the world and attributed it to "youthful exuberance." Considering that the Love Bug spread worldwide in less than 24 hours and cost an estimated \$10 billion to clean up, this virus creator has a gift for understatement. Though he was never prosecuted, those of us who spent many hours cleaning up the aftermath of the Love Bug (saturated WAN links, tens of thousands of queued copies of the message, and so on) while users pounded on the data center doors would like to see him expend some of that youthful exuberance elsewhere.

Some virus programmers have set out to make a political statement; one Microsoft Word macro virus displayed the message "Stop all French nuclear testing in the Pacific!" Virus writing as activism is thankfully rare, although some public pronouncements by the US government indicate that they're worried about the use of targeted malware that attacks critical infrastructure. The recent BugBear worm functioned as a dropper that installed a keystroke logger on target machines, provided that they were on the network at a number of specifically targeted banks.

Fairly few viruses truly cause direct harm to the computers they infect. There are no recorded instances of viruses causing harm directly to the hardware of a computer, though some viruses may be able to damage a computer's CMOS. The real harm associated with viruses is usually indirect. Some of the dangers and direct or indirect costs include:

- Viruses that deposit Trojan horse programs or that install keystroke loggers offer a potentially significant security breach.
- Sophisticated viruses (worms and Trojans included) can leave agents that mine information from infected computers and send that information to someone outside of the organization.
- Computers infected with viruses may operate more slowly as a result of increased memory usage or problems due to a virus being attached to the program or a document.
- Email viruses and worms can cause increased WAN bandwidth usage.
- Viruses can chew up a significant amount of disk space, especially viruses that spread themselves via email.
- Labor costs to clean up a virus outbreak can be significant if there are many servers or desktop computers that must be inspected.
- Loss of user productivity.

A Brief History of Viruses

Computer viruses have been around almost as long as computers themselves, though we did not really call them viruses until the mid-1980s. Viruses have evolved over the years to become more complex and more dangerous. Viruses are becoming more and more difficult to detect and eradicate, the method that viruses (and worms) are using to spread is increasingly more efficient, and the increased dependency on personal computers and corporate desktops are giving complex viruses the ability to potentially compromise security.

Even early IBM 360/370 machines had primitive, self-replicating, self-propagating programs. According to Eugene Kaspersky of antivirus company Kaspersky Labs, in the late 1960s there were programs on mainframes that cloned themselves and occupied system resources. These programs were called ‘the rabbit.’ In 1981, a 9th grader in Pennsylvania named Rich Skrenta wrote a virus called Elk Cloner that infected Apple II floppy disks and displayed a harmless message. There were also viruses that were developed as “proof-of-concept” viruses, including one developed for the UNIX-based VAX machine by Fred Cohen and Len Adleman, who coined the term *computer virus*.

The first IBM PC-compatible virus is generally accepted as being the Brain virus that was allegedly developed by two brothers in Pakistan. This virus, like most of the other viruses in the early days of computer viruses, was a boot-sector infector. It embedded itself in to the master boot record of a floppy disk, then infected any computer that used that disk. This virus even had stealthy characteristics: if a program attempted to read a disk sector of the master boot record in which the virus was located, Brain would move that sector to another place within the master boot record.

Shortly after the Brain virus appeared in 1986, the Virdem virus appeared along with a Trojan horse program masquerading as the PC-Write word processor for DOS. This nasty little Trojan destroyed the disk’s file allocation table and initiated a low-level format.

By 1988, there were more boot-sector infector viruses (including the pervasive Stoned virus that hung around for many years) as well as file infector viruses and Macintosh viruses. November 1, 1988, saw the “accidental” release of the first major Internet worm—the Robert Morris worm. This worm caused many computers on the Internet to be isolated or shut down. Of course, there were relatively few computers (about 60,000) on the Internet.

The year 1991 saw the arrival of the first network aware virus, the GP1 virus, which attempted to steal Novell NetWare passwords. The first polymorphic virus, Tequila, also arrived on the scene in 1991.

In 1992, the Michelangelo virus appeared amidst media hysteria and fears of massive damage. A lot of print was wasted for what amounted to almost nothing. In 1995, the first Microsoft Word *macro virus*, Concept, appeared. The first Microsoft Excel macro virus, called Laroux, was released in 1996.

1996 and 1997 saw the release of the Staog and Linux.Bliss viruses, showing that even the Linux platform was vulnerable. In December of 1997, a new type of worm emerged called an mIRC worm. These worms took advantage of an early version of the Windows Internet Relay Chat (IRC) client.

In 1998, the Strange Brew virus gained the dubious distinction of being the first Java virus. Also released in 1998 was the Back Orifice Trojan, which, if executed, could allow an intruder to take remote control of a computer on which the Back Orifice infector had been run.

March of 1999 saw the first email-based virus; this Word macro virus was called Melissa. It could use either Outlook or Outlook Express to send itself to recipients in the Exchange GAL or the user’s address book. Melissa actually required the user to open the attachment. Bubbleboy, released later that year, executed if someone merely opened a message in Outlook (or even previewed the message in the Outlook Express Preview Pane). Corner, also found in 1999, was the first Microsoft PowerPoint-based virus.

As the 21st century dawned, new types of viruses have been introduced. These include viruses that spread through Adobe PDF files, file-sharing systems, IRC worms, and instant messaging (viruses that send themselves to everyone in a victim's buddy list), and viruses or Trojans that infect PDAs.

In 2001, a whole new breed of virus/worm was introduced, including SirCam, Code Red, and the infamous Nimda. Although Nimda was a worm, there were reported cases in which it was spread through email. Nimda primarily used a vulnerability in the Microsoft Internet Information Server (IIS) 4.0 or 5.0 server to infect an IIS server. Once that server was infected, it would then begin scanning IP ranges looking for other IIS servers to infect. Nimda could also infect Web pages so that if someone opened an infected Web page, the user's computer would become infected and begin attempting to spread the infection. Finally, if Nimda discovered an Outlook or Outlook Express address book, it would attempt to mail itself to others.

 An excellent explanation of Nimda can be found at <http://www.cknow.com/vtutor/vtnimda.htm>.

The year 2002 introduced even more “innovative” approaches to spreading viruses and worms. This generation includes the SQL Slammer worm that used a known vulnerability (and often neglected fix) of Microsoft SQL Server; this worm went worldwide in about 10 minutes. The LFM-926 virus infected Macromedia Shockwave files. The Perrun virus attaches itself to JPG files. New worms, such as Zircon C and I-Worm.Cervinec, that propagate through email were introduced. Figure 1.1 shows a breakdown of the most widespread computer viruses for the year 2002.

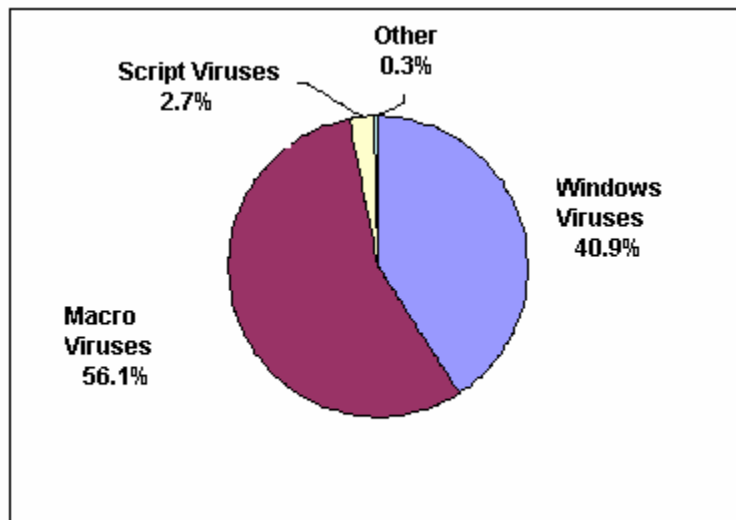


Figure 1.1: The most widespread virus types for 2002 (from <http://www.viruslist.com>).

What Does the Future Hold?

The folks writing viruses, worms, and Trojans are getting more and more creative. Nimda gave us a great example of just how determined a virus programmer can be to get a virus out in the wild and spreading; Slammer showed just how quickly a well-engineered piece of malware can spread world-wide. Even with warnings in every technology media outlet, email newsletter, and even CNN, email viruses continue to propagate. Users and administrators are becoming more diligent and more restrictions are placed on email content all the time, but the race continues.

Worms will continue to spread as newer and more sophisticated technologies are introduced into the workplace. Almost certainly, viruses and worms that take advantage of wireless networking, PDAs, and cell phones will continue to emerge. The methods that the virus writers use will continue to become more ingenious, creative, and deceptive. Table 1.1 shows a list of the 10 most widespread malware programs of the year 2002 as collected by Viruslist.com. Note that most of these malicious programs were worms.

Rank	Name
1	I-Worm.Klez
2	I-Worm.Lentin
3	I-Worm.Tanatos
4	I-Worm.BadtransII
5	Macro.Word97.Thus
6	I-Worm.Hybris
7	I-Worm.Bridex
8	I-Worm.Magistr
9	Win95.CIH
10	I-Worm.Sircam

Table 1.1: Viruslist.com's 2002 most widespread malware programs.

Are Viruses Dangerous?

Are computer and email viruses truly dangerous? Is the perceived threat much ado about nearly nothing? If you have ever had to remove a virus, you can sympathize with the folks that claim financial loss. Although most do not, there are certain viruses that can (and do!) destroy data on a computer's hard drive. And loss of productivity is difficult to quantify for more organizations. Virus outbreaks can also increase the need for data storage as files get larger or messages get queued for delivery. When these messages begin to be delivered outside your organization, the cost of the bandwidth that will be consumed can inflate the damage. In addition, a virus infection that attacks your organization may be dangerous for other reasons. If an email virus is sent from your company to many of your customers, clients, and vendors, your organization may suffer a significant loss of credibility or public embarrassment.



For some organizations, such as law firms, accounting firms, and healthcare organizations, disclosure of confidential information could result in law suits. For an organization such as a government or military, accidental disclosure of information could get someone killed.

Several virus experts have commented over the years that the industry is far more obsessed with viruses and virus protection than they are with topics that present more dangerous and immediate threats to IT operations such as disaster recovery and business continuity. This commentary certainly does not mean that we should ignore the threat of viruses; only that virus protection is one cog in a well-oiled disaster prevention machine. The following list highlights the main potential dangers that can result from viruses:

- Damage to computers' applications or operating systems (OSs)
- Loss of important data
- A carefully crafted and targeted virus could be used to glean information from inside the targeted organization and send the information to an outsider (for example, the recent BugBear virus targeted a number of financial institutions)
- Public embarrassment—imagine having a virus infection that sends infected email to all your clients
- Loss of credibility with customers, vendors, and the public
- Downtime and loss of productivity
- Failing to meet deadlines or customer expectations
- Financial costs associated with cleaning a virus

Hoaxes and Urban Legends

Not a week goes by that I don't get an email from a relative or friend warning me of the latest virus threat. More often than not, these messages are false alarms; in fact, a number of commercial spam filters include an option to automatically screen out common virus hoax messages. Many of these messages instruct me to delete a file out of my computer's Windows directory. Less sophisticated computer users are more than willing to believe that their computer has a virus, and they are more than willing to forward such notices on to many others. Although the people who forward these messages are well meaning, often the files these messages instruct you to delete can do real harm to your OS. Users should be trained that if they receive an alert about a virus, they should verify that alert with their antivirus vendor's Web page or their IT department.

These hoaxes play on our fears of computer viruses. Usually they sound quite convincing and make things seem like doom is certain unless you follow their instructions. These hoaxes almost always ask you to forward the message to as many of your friends and family as you possibly can.

The Good Times hoax is probably the most successful hoax in virus history. From some perspectives, Good Times accomplished the same things as modern email based viruses—it managed to get passed from mailbox to mailbox. This hoax originated in 1994, but is still being passed around the Internet today. The author of this hoax instructed anyone who received a message with the subject "Good Times" to immediately delete the message; interestingly enough, if people were following its advice, they would have read only one of these messages. The message claimed that the Good Times virus would send a message to everyone in your address book, then proceed to trash the computer. They then urge you to forward the message on to your friends and local system users.



In 1995, an Australian virus group created a real virus called Good Times that included some of the text from the Good Times hoax message in the source code of the real virus.

The Internet is full of hoaxes, myths, and urban legends. In the early days of the publicly accessible Internet, Darwin awards (some were true, some were simply Internet folklore), warnings of kidney theft rings, and stories of Neiman Marcus cookie recipes occasionally hit my mailbox.



You can find more information about hoaxes, myths, and urban legends at the following Web sites:



<http://www.vmyths.com>



<http://www.sophos.com/virusinfo/hoaxes>



<http://www.snopes.com>



<http://www.stiller.com/hoaxes.htm>

Email-Based Viruses

Traditionally, viruses were spread only by infected floppy disks. Public email systems, such as the Internet, have given viruses a new method of replicating themselves. Any virus can be spread through email if a user inadvertently sends an infected program or document to another user. However, specific email-based viruses generally have to be written to understand the email client that it will use to propagate itself. When an email virus is executed, it scans the user's address book and sends a copy of itself to everyone in the user's address list. In the case of clients such as Microsoft Outlook or Outlook Express, an email-based virus may send itself to the entire Outlook GSL as well as the contacts in user's personal address book and contact folders. In the case of Outlook Express, the virus sends itself to the recipients in the Windows Address Book. In some organizations, a single user opening an email virus could cause a message to be sent to tens or hundreds of thousands of mail recipients.



Not all viruses simply use the Exchange GAL as a source for email addresses. Some viruses now look at the Outlook Inbox and Sent Items folders and gather email addresses from messages you have sent and received. One virus even scans your hard drive looking for HTML files that contain email addresses.

Most email-based viruses work by sending themselves as an attachment to an email. Most unsuspecting users will open this attachment, and the virus begins its infecting and spreading phase once again. The attachment may be an executable file, script, or some other type of file that might be able to automatically execute a macro or perform some action as a result of the file being loaded. Even registry files could possibly be used to help a virus spread; however, there have been viruses that could execute even if the message was merely displayed in the Outlook preview pane. To help better understand how email-based viruses propagate, I will discuss in more detail a few of the more common email viruses.


Melissa

Melissa (also known as W97M.Melissa) is a Microsoft Word 97 macro virus, but it has some of the characteristics of a worm. Melissa got its name from a stripper in Florida that the author, David Smith, knew. In the code of the virus, the author identified himself as Kwyjibo; Kwyjibo is a word that Bart used in a game of Scrabble in an episode of the TV show *The Simpsons*.

```
"WORD/Melissa written by Kwyjibo
Works in both Word 2000 and Word 97
Worm? Macro Virus? Word 97 Virus? Word 2000 Virus? You Decide!
Word -> Email | Word 97 <-> Word 2000 ... it's a new age! "
```

Melissa arrived with a subject of “Important Message from” and the sender’s name. The attachment initially was a file called LIST.DOC, but the virus can be spread by any file attachment. When the user opened the file, a macro executed that looked for Outlook address books. The virus would then send a copy of itself to the first 50 people in your address book.

The *Simpsons* reference would occasionally be included in the currently open Word document if the document was opened when the day of the month matches the current minutes value on the clock (for example 15th of the month and 15 minutes after the hour). In this case, the user would see Bart’s quote after he used the word Kwyjibo to win a game of scrabble, “Twenty-two points, plus triple-word-score, plus 50 points for using all my letters. Game’s over. I’m outta here.” There are now a few dozen variants of the Melissa virus.

 If you are interested in learning more about Melissa, visit http://www.softpanorama.org/Antivirus/AV_Secrets/Vgallery/melissa.shtml.

Happy99

Happy99 (Happy99.Worm, Trojan.Happy99, I-Worm.Happy, W32-Ska, and Happy00) is officially an email-based worm, but it has characteristics of a virus and a Trojan Horse. The program (Happy99.exe) arrives in an email message. When opened, it displays “Happy New Year 1999!!” and fireworks (see Figure 1.2). The fireworks are merely a diversion while the virus installs itself to the local hard drive, configures itself to run each time the computer restarts, and emails itself to addresses in the address book or Exchange GAL. Happy99 is one example of a worm that will reload itself after a computer reboots.

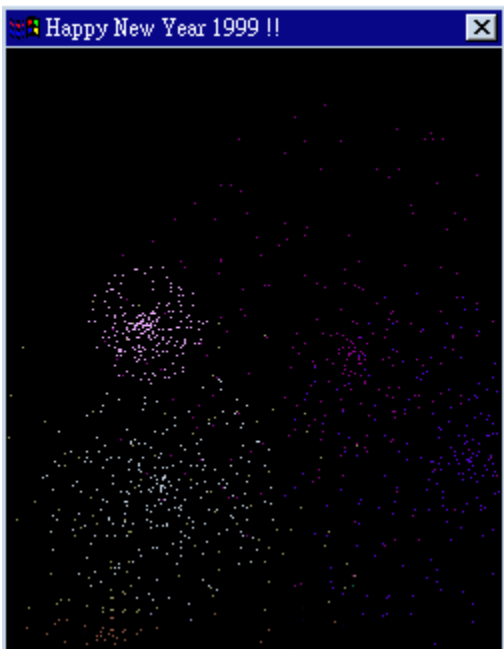


Figure 1.2: Happy99's seemingly harmless output.

PrettyPark

PrettyPark is one of the new breed of hybrid viruses that have worm characteristics. It behaves similar to the Happy99 worm in that it is able to reload itself after a reboot. This virus arrives in a user's mailbox with the attachment PrettyPark.exe. When the file is executed, it might display the Windows 3D Pipes screen saver. It also creates a new Files32.vxd in the \Windows\System folder, and modifies the registry so that this file is included as a Windows shell option. It then attempts to email itself to addresses in the Outlook address book every 30 minutes, and connects to an IRC channel once every 30 minutes. The worm can then transmit information about your computer to this IRC channel and can receive instructions from someone in the IRC channel.

Mylife

The Mylife worm/virus caught a lot of even the most attentive administrators off guard. It was first discovered in April 2002. This message arrived with the innocuous message indicating a funny screen saver is attached. When the attachment, usa.scr, was opened, it displayed the graphic that Figure 1.3 shows.



Figure 1.3: The Mylife caricature seems harmless.

The caricature graphic was not even particularly funny, so most users would close it and move on, assuming that it was just another boring email attachment. In the background, though, the .scr attachment added itself to the registry, copied itself to the Windows system directory, and mailed a copy of itself to all of the addresses in the Outlook address book and the MSN Messenger contact list, if available. When the Mylife scr file is run from the Windows system folder, it deletes all the files and folders on the C: drive.

Nimda

Nimda (admin backwards, in case you were wondering) arrived on the scene in September 2001; it or its variants have been making our lives difficult ever since. Nimda is classified as both a virus and a worm, but it behaves more like a worm. This worm introduces a whole new range of ways that a worm can spread itself. Copycats have been developing worms ever since, attempting to use the same techniques.

The worm attempts to use a known (and patched) bug in Microsoft IIS 4.0 and 5.0 that allows malformed URLs to access files and folders that are located anywhere on the same logical drive as the Web server folders. This vulnerability allows the worm to infect the Web server, replace Web pages, or even add itself to Web page content. The worm then uses that Web server as a platform for scanning IP address ranges, looking for other Web servers with the same vulnerability.

After unsuspecting clients connect to an infected Web server, the clients are prompted to download an .eml file (email file). This file contains the worm as an attachment. When the clients download and open the message, their computers become infected and start looking for other computers to infect. In addition, the worm will scan Outlook address books and any HTML files on computers looking for email addresses to which it can send itself. Nimda has a built-in SMTP server that sends outbound messages, so it can bypass some of Outlook's built-in antivirus features. Nimda also scans the network looking for other computers with open shared folders. If it finds the folders, it attempts to transfer the file RICHED20.DLL (which is used by Outlook and Word when creating email) to the remote shared folders. If a user opens a message and the new version of RICHED20.DLL is in the path, Outlook or Word uses the new version of the file and begins to infect computers with the worm. For a 57KB virus, Nimda has some impressive features.

Scanning for Viruses

Depending on the vendor, virus scanning software can take one of a couple of different approaches to finding and repairing viruses. However, at the heart of all virus scanning systems are the virus signatures. The *virus signatures* are a database of all known viruses as well as something unique about each of those viruses. For most vendors, the publicly released version of this database changes weekly; during periods of high virus activity or when multiple new strains of a virus are being discovered, updated versions of the signatures database may be released several times a week. Internally, the database may change several times a day, as virus specialists reverse engineer the viruses that are discovered daily.

Secondary to the virus signature database is the scanning engine itself. The scanning engine is the technology that actually scans for viruses. Keeping the scanning engine up-to-date is almost as important as keeping the signatures updated. As new viruses are discovered and viruses get harder to detect, older scanning engines will not be able to detect viruses accurately.

Virus scanning software uses the signature database as it scans files on the hard drive to look for signs that a virus has infected a file. Most scanners simply open the designated files and search for the patterns of known viruses. Virus scanners can be configured to scan the hard disk for all files or only files with specific extensions. Virus scanners also scan the computer's memory and master boot record looking for viruses. A virus scanner should also be capable of real-time scanning of the file system (hard drives, removable drives, and floppy disks). When a user accesses files on the file system, the virus scanner should scan the file.


Virus scanners also employ a number of techniques beyond just simple file scanning to detect viruses. More sophisticated scanners can examine executable files looking for something out of the ordinary in the file, such as an instruction at the beginning of the executable that instructs the computer to immediately go to the end of the program. Scanners with *heuristic detection* capabilities work on the assumption that a virus will attempt to conceal itself (such as stealth or polymorphic viruses). Client scanners are also email-aware. These scanners can detect when an email client opens a message, then scan that email message for viruses.

 Do you want to test your virus scanner? The European Institute for Computer Anti-Virus Research (EICAR) has a test file called eicar.com. It is not a virus, but most virus scanners will recognize and detect this file. It can be found at <http://www.eicar.org>.

A Matter of Trust

Just as if your machine had been attacked and compromised by a skilled hacker, once a sophisticated virus (or worm or Trojan horse) has taken over your machine, you may not be able to completely trust the machine again. The virus may have left components that are not yet detected and cleaned up by antivirus scanners. The component may lie dormant for some period of time, reactivate, and re-infect the computer.

For this reason, integrity checking products are becoming more popular for critical computers such as servers. An integrity checking program works by scanning the hard disk of a computer that you know can be trusted. The program calculates a checksum for each of the executables and dynamic link libraries (DLLs) on the server's disk and records this information to a log or database file. If the machine is ever compromised, a checking program can analyze the hard drive and determine whether any other critical OS or applications have been compromised of which you are not aware. The limitation of integrity checking programs is that each time a service pack or OS update is released, the integrity checking program must recalculate the checksums of the files.

 A good explanation of integrity checking and what to look for in an integrity checking program can be found at <http://www.cknow.com/vtutor/vtintegrity.htm>.

An Ounce of Prevention

In the early days of viruses, there was not a lot that users could do to protect themselves from viruses. About all you could do was never download files, never install software that was not from a completely trusted source (and this was no guarantee!), and never share floppy disks with anyone else. By 1989, there were a couple of commercially available antivirus software programs, which changed the face of virus protection.

Keeping user's computers—not only their desktop computers at work, but also their home computers—virus free is very important. (Many users work from home occasionally and could introduce malware from their home computers to a corporate computer system.) The following list provides important steps that you and your end users can take to help ensure that you remain virus-free:

- Purchase and install an antivirus scanner that does real-time antivirus scanning of the computer's file system and detect floppies when they are inserted.
- Keep the antivirus software and virus signature database up-to-date. Doing so might require updating the signature database a couple of times per week.
- If you receive files that you do not expect via email, don't open them.
- If you are using Microsoft Outlook or Microsoft Outlook Express, download the latest version or the security updates.
- If a message or attachment from someone seems to be out of character for that person, don't open it. For example, your boss is hopefully not going to send you a message with the subject ILOVEYOU.
- If you are sending a message with an attachment, send the message and attachment, then send a second message letting the user know that you sent them a legitimate file. Use S/MIME digital signatures to further guarantee the authenticity of your message.
- Do not forward junk mail or chain messages.
- If a Web site you are visiting prompts you to download software or run a program, don't do it! Only download software that you have specifically requested.
- Do not download software from the Internet if you do not have an antivirus program running on your computer.
- Exercise extreme caution when using Internet file-sharing services. These services are a major source of viruses, Trojan horses, spyware, and adware.
- Keep regular backups of your important data.
- Verify the sender of emails. If Microsoft, Apple, IBM, Amazon, your bank, your credit union, or any other entity that you may have an online relationship with sends you an email and advises you to download something, view that messages with a good degree of skepticism. If you choose to follow the link, make sure that it takes you to that vendor or company's Web site.
- OS updates and fixes are usually installed by an organization's IT department; avoid following the advice of email messages telling you that your computer is insecure and need updates. Especially on your company's network.

References and More Information

When I was writing this chapter, several excellent references reinforced my knowledge and taught me new facts. If you want additional information, these Web sites are great starting points.

Carnegie Mellon University's Computer Emergency Response Team Coordination Center (CERT/CC) (<http://www.cert.org>) is one of the best places to look for information about security, viruses, Trojan horses, and worms.

The WildList Organization International (<http://www.wildlist.org>) has a great Web site that includes information about which viruses are currently spreading and which are fading from the radar.

The SANS Institute (<http://www.sans.org>) is an excellent source of information not only about viruses and worms but also about security information in general.

Computer Knowledge's Virus Tutorial (<http://www.cknow.com/vtutor>) is an excellent starting place for learning more about the basics of viruses as well as for articles written by industry experts.

Current virus alerts, virus news, statistics, advice, and information about hoaxes can be found at Viruslist.com (<http://www.viruslist.com>).

EICAR helps to coordinate the activities of private and public organizations, security experts, and government agencies when fighting computer viruses. You can find the organization on the Internet at <http://www.eicar.org>.

The Internet USENET newsgroups include alt.comp.virus. A culmination of information from these newsgroups is organized in the group's FAQ at <http://www.faqs.org/faqs/by-newsgroup/alt/alt.comp.virus.html>. If you plan to join any of the virus-related newsgroups, you should read the FAQ first.

Want to know how your antivirus program stacks up? Visit AV-Test (<http://www.av-test.org>). This project is a combined effort of the Business-Information Workgroup at the Institute of Technical and Business Information Systems at the Otto Von Guericke University in Magdeburg, Germany.

Summary

Computer viruses have evolved dramatically over the past 20 years. They are sneakier, more powerful, and spread more quickly. In addition, virus writers will continue to make viruses more and more difficult to detect and more powerful. These writers will undoubtedly continue to find new ways to spread their malware. Emerging technologies such as wireless communications, PDAs, tablet computers, cell phones, and other handheld devices will undoubtedly become targets for virus writers.

As we've explored in this chapter, the challenge of virus protection is going to continue. Hopefully, the background information about viruses as well as knowing which types exist and how they work will help you protect your organization's desktop computers, file servers, database servers, Web servers, and email servers.

Chapter 2: Protection at the Client Level


In the first chapter, we explored many of the risks to which networked computers are exposed. Even if a computer is not networked, if the users of that computer share media (floppies, CD-ROMs, DVDs, or tapes) that have infected files, the computers can be infected by viruses, worms, or Trojan horses.

Your virus protection strategy is not complete until you have protected not only the email server, but also the client. In this chapter, I'll cover how to protect Windows-based clients that are using Microsoft Outlook or Outlook Express and connecting to a Microsoft Exchange Server. Why? Protecting clients and leaving the server unprotected gives you a reasonable degree of security, but the reverse isn't true. Client protection is the fundamental first step toward building a strong, layered defense against viruses. Many of the techniques discussed in this chapter are specific to Exchange environments, while others are generic and the principles can be applied to most any environment.

Virus Entry Points

Before I delve into client protection, let's first explore some of the different ways that malware (viruses, worms, and Trojan horses) can make their way into your organization. Although many of these entry points are the result of users introducing a virus to the environment, some of them are a result of carelessness on the part of the systems or network administrators. The widespread perception is that the most common way viruses are transported is via an infected email message or an infected attachment. However, probably the most common way that viruses are spreading in 2003 is by exploiting vulnerabilities in OSs and Web servers. Nimda, Code Red, Klez, and the SQL Slammer prove that such blended virus/worm threats can be devastating. However, there are many ways for malware to spread:

- Through users who accidentally introduce viruses, worms, and Trojan horses by introducing outside, unscanned media (floppy, CD-ROM, DVD, or tape)
- By way of users who download viruses, worms, and Trojan horses from Internet Web sites either unknowingly or by downloading shareware or infected software from file sharing services or 'warez sites
- Via users who use their external POP3, IMAP4, or Web mail accounts to retrieve personal email—without it being scanned for malware—from outside of the organization
- Through users that connect to the network remotely from computers that are unprotected, such as a laptop or home computer

 Some antivirus software vendors allow a user to install a copy of the company software on a single home computer—check your software's license agreement.

Penny-Wise and Pound Foolish

Company X had purchased antivirus software for its Exchange Server system and the company's firewall had a virus scanning component. Only a few computers actually had antivirus client software installed, and the software was what had shipped with the client computers. In an effort to save money, the IT department reasoned that client software was not necessary on the company's desktop computers. They rationalized this decision by stating that viruses entered the organization through the email system.

During the first year after this server-protection-only strategy was implemented, numerous viruses entered the organization via documents on floppy disks and by way of users who brought their notebooks in and out of the office. Often, the only time a virus was detected was when a user emailed a document infected with a macro virus. The internal network was also infected with the Nimda worm/virus twice due to users plugging infected notebook computers into the company network.

Securing the Desktop—Client Protection 101


Users are remarkably ingenious when it comes to figuring out how to completely mess up their desktop computers. Even the most technically inept user can (unintentionally) figure out a way to require a visit from the Help desk. The more technically savvy the user is, the more difficult the problem can be to track down. In the process of exchanging documents, adding software, changing settings, and reading email from outside sources, users can introduce viruses from outside extremely easily. The desktop client must be secured in order to prevent viruses from being introduced into your organization.

Choosing Antivirus Client Software

There are many antivirus client software vendors targeting the desktop market. Each of these products has strengths and weaknesses. You must evaluate the client software based on your specific needs and decide which vendors meet your requirements for desktop scanning. The following list highlights a few guidelines for picking client software for an organization with more than 100 seats. The ideal client should be:

- Capable of automatically updating its virus signatures on a schedule you set (or at least on a daily basis)
- *Email aware*, meaning that it understands email client software (such as Outlook and Outlook Express); for SMTP, POP3, and IMAP4 clients, the software must be able to scan inbound and outbound message traffic
- Configurable to scan different types of files—not just executables
- Capable of scheduled weekly hard disk scans
- Capable of detecting blended threats and other forms of malware (in addition to viruses)
- Able to allow centralized administration of client configuration and scanning rules
- Automatically configured to scanning removable media for viruses when inserted
- Able to perform real-time scans of the file system so that if hostile content is written to or read from the file system, the scanner will detect it

- Optimized and designed for organizations of your size and needs—some antivirus clients are designed for organizations with centralized administration, while others are designed for small offices or home use
- Designed for a corporate environment, including reporting software that allows the administrator to monitor client software version, signatures, and outbreaks


 Client software that can quickly update its signatures during a virus outbreak is very helpful. A desktop client that can only access signatures once per day or once per week will be dangerously exposed during more virulent and fast-spreading virus outbreaks. Some blended threats, such as Nimda and SQL Slammer, infected computers world-wide within 15 minutes of their initial release into the wild. McAfee estimates that there are approximately 300 new viruses per month. Of course, you probably won't want to pull down multiple updates per day in normal operation, but you want the ability to do so when a fast-spreading virus is about.

Common Vendors of Antivirus Client Software

You should give careful consideration to whichever antivirus client software you are planning to use. It should meet both your security and administrative needs. Table 2.1 lists some of the more common vendors of antivirus software. This list is alphabetical and is not an endorsement of a specific product nor is it all-inclusive.

Vendor	Web address
AVG Anti-virus	http://www.grisoft.com
BitDefender	http://www.bitdefender.com
Computer Associates	http://www.ca.com
FRISK Software	http://www.f-prot.com
Kaspersky Lab	http://www.kaspersky.com
Network Associates	http://www.mcafee.com
Norman Virus Control	http://www.norman.com
Panda Software	http://www.pandasoftware.com
Sophos	http://www.sophos.com
Symantec	http://www.symantec.com
Trend Micro	http://www.trendmicro.com

Table 2.1: Common antivirus vendors.

 For comparisons and tests of various antivirus products, visit <http://www.av-test.org>. This Web site is a joint project of the University of Magdeburg and GEGA IT-Solutions. Products are tested on a variety of Windows and non-Windows platforms to determine performance and detection for not only viruses found in the wild but also for “zoo” (lab) viruses. In April of 2003, PC Magazine published a review of corporate antivirus software solutions; you can find this review at <http://www.pcmag.com/article2/0,4149,978683,00.asp>.

General Recommendations


This chapter focuses mostly on the use of Microsoft Outlook 2000 and later as an email client. Many of the suggestions for tightening security for Outlook (such as restricting Internet Explorer—IE—Security Zones) applies to Outlook Express as well as Outlook. Outlook and Outlook Express present interesting targets for malware authors because of the number of different programmatic interfaces that are available and the proliferation of these clients.

However, this attraction to Outlook and Outlook Express by malware authors does not mean that other email clients are not at risk. Although the rest of this chapter does focus on these Microsoft platforms, I wanted to take a brief opportunity to provide some solid suggestions for protecting other email client software. These tips (some of them are pretty obvious) apply whether you are using a Macintosh, Linux, or other desktop platform and regardless of the email client you are using:

- Regularly fix bugs and close known vulnerabilities
- Update the Web browser—the version should be fairly recent and you should have all the latest fixes; if the vendor is no longer releasing fixes for the version of the browser you are using, it is time to upgrade
- Update the email client—most email clients (including many of the public domain clients and shareware clients) are updated periodically with security fixes and feature updates
- Disable an email client's scripting interface (regardless of the email client that you are using) as a good security precaution against virus outbreaks—different clients have different recommendations for making the client software more secure


Netscape Mail

Disabling the scripting interface in Netscape Mail will help tighten the security of that email client. To disable scripting, launch the Netscape Mail client, select Edit, then Preferences. Select Category, then Advanced, clear the *Enable Javascript for Mail and News* check box, and click OK to close the dialog box.

 You can find more information about the Netscape Mail program on the Web at <http://channels.netscape.com/ns/browsers/mail.jsp>.


Qualcomm Eudora


Qualcomm makes a few recommendations for tightening security and improving virus protection in the Eudora mail client. First the company recommends that you disable allowing executables in the mail client. To do so, select Tools, Options—Viewing Mail, and clear the *Allow Executables in HTML Content* check box. The company also recommends that you configure Eudora with a different directory than the default for the Attach directory.

 For documentation regarding the directory change check out <http://www.eudora.com/techsupport/kb/2020hq.html>. You can also find more information about Eudora at (<http://www.eudora.com> and <http://www.eudora.com/security.html>).

Outlook Security Update


Over the past several years, Microsoft has made quite a few security improvements in Outlook. One of the most significant of these changes is the Outlook Security Update, originally released for Outlook 98 and Outlook 2000. If you are running Outlook 98 or Outlook 2000, you should apply this update to your desktop clients as soon as possible.

 You can find the update for Outlook 2000 at <http://office.microsoft.com/downloads/2000/Out2ksec.aspx> (note that Outlook 2000 SR1 must be installed). For more information about the update for Outlook 97, 98, and 2000, see the Microsoft article “Outlook E-mail Attachment Security Update.”

 The Office 2000 Service Pack 2 (SP2) and SP3 both include the security features that were introduced in the Outlook Security Update. Outlook 2002 and Outlook 2003 also include the security fixes.

Understanding the Outlook Security Update

The Outlook Security Update made a number of changes to Outlook to improve Outlook’s resistance to viruses. These attachment security features are an attempt to protect users from accidentally opening malware attachments and to protect users’ address books against attacks by macro viruses that try to mail themselves to everyone in an address list. These measures can help to slow the spread of email-based viruses within an organization.

 Notify your user community of the new restrictions that the Outlook Security Update or a newer version of Outlook will impose. Some users may need to be able to access restricted files via their email, so they’ll need to think of other ways to accomplish this task.

The security update has changed slightly between the original version and Outlook 2000 SP3; I am going to discuss the features in the later version, which I strongly recommend that you apply.

By default, when the update is installed, the security features are all enabled. Later in this chapter, I will review some features that will allow you to centrally control these restrictions using the Outlook Security Features Administrative Package from the Office Resource Kit.

These new security features include:

- Defining a specific set of attachments as potentially unsafe and preventing users from either opening the attachment or saving it to disk. These attachments are known as Level-1 attachments.
- Warning users when they attempt to send an attachment categorized as a Level-1 attachment that the attachment might be potentially unsafe.
- Giving users the ability—through the registry editor—to “demote” specific Level-1 attachment types to Level-2 attachments. Level-2 attachments can be saved to the hard disk, but not opened directly through Outlook.

- Implementing add-in and third-party application security; if an application attempts to programmatically access Outlook using the Outlook object model, Simple MAPI, or Collaborative Data Objects (CDO), the add-in can automatically be denied access to Outlook or the user can be prompted to allow the add-in to access Outlook messages, features, and the address book.
- Allowing specific add-ins and extensions to be defined as safe; thus, the user will not be warned each time the add-in accesses Outlook. This feature is useful when the add-in is used for each message sent or received.

Prior to installing the security update for all of your users, run some tests on custom forms and Outlook add-ins in your environment. It is entirely possible that this patch will change the behavior of these forms and add-ins or that the users may see additional prompts that they had not seen prior to the update being installed. If you have custom forms that contain scripts, you might need to change the behavior of these forms and make sure that they are published to the organization forms library.

The patch defines for Outlook two types of attachments: Level-1 and Level-2. Users are prohibited from opening or saving attachments in the Level-1 attachment list. Though the attachments are still in the message, the user sees a note directly below the button bar indicating that the message contains a potentially unsafe attachment (see Figure 2.1).

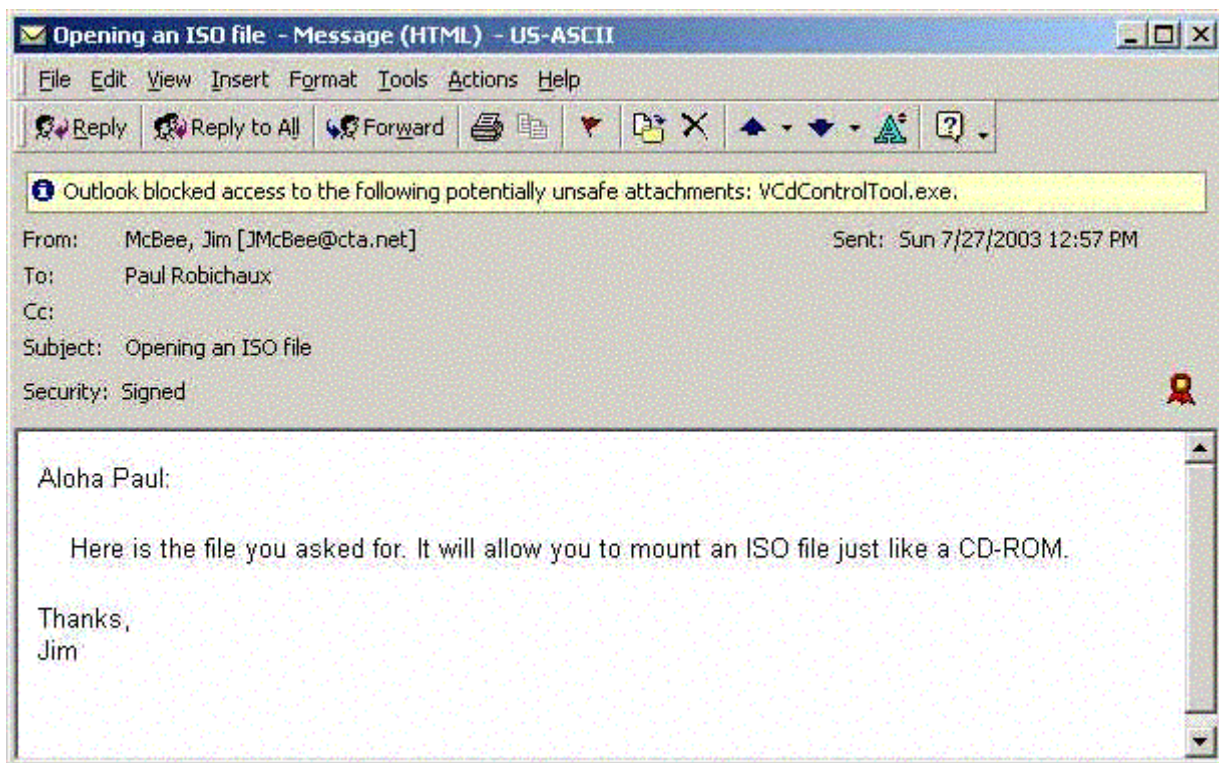


Figure 2.1: Once the Outlook Security Update is installed, users no longer have access to unsafe attachments.

If the user attempts to forward the message, the potentially unsafe attachment will be stripped from the forwarded message. However, Outlook will allow the user to send the message as long as they click Yes in the warning dialog box that Figure 2.2 shows (the warning dialog box will appear slightly different with different versions of Outlook and depending on whether the Outlook Assistant is enabled).

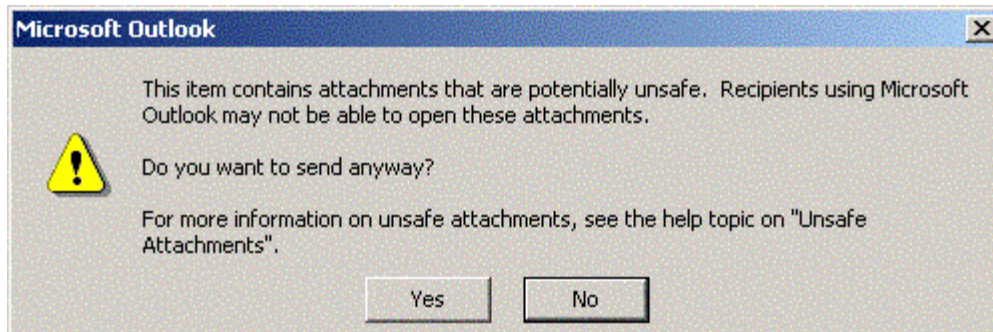


Figure 2.2: Outlook warning message for unsafe attachments.

As I mentioned earlier, users or administrators can demote some attachments from Level-1 to Level-2 attachments. Once an attachment is considered a Level-2 attachment, the user can save the attachment to disk, but they will still receive a warning indicating that the attachment is potentially unsafe if they attempt to open it directly from within Outlook. Figure 2.3 shows this warning (this image is from Outlook 2002).



Figure 2.3: Warning a user that he or she cannot open a potentially unsafe attachment.

☞ In many email environments, some attachments are automatically blocked from being sent and received by the firewall, SMTP scanner, or the mail-server antivirus scanning software. A very good practice is to instruct users to rename their potentially unsafe attachments—such as renaming VBS attachments to VB_ or to use a zip utility. Some antivirus and scanning software may still detect the attachment type if it has been renamed or compressed, so this solution will not work for everyone.

So what is considered a blocked attachment? Table 2.2 provides the Level-1 attachment types that are blocked.

 You can find a comprehensive list of the blocked attachment types in Appendix B of the Microsoft whitepaper “Outlook 98/2000 Email Security Update” at <http://www.microsoft.com/office/ork/2000/download/OutSecWP.doc>.

Attachment Extension	Description
Ade	Microsoft Access project extensions
Adp	Microsoft Access project
App	Microsoft Visual FoxPro application
Asx	Windows Media audio or video shortcut
Bas	Visual Basic module
bat	Batch file
chm	Compiled HTML help file
cmd	Windows command script
com	MS-DOS program
cpl	Control Panel applet
crt	Security certificate
csch	KornShell script
exe	Executable program
fxp	Microsoft Visual FoxPro compiled program
hlp	Windows Help file
hta	HTML program
inf	Setup information file
ins	Internet Naming Service
isp	Internet communication settings
js	JavaScript file
Ink	Windows shortcut link
jse	JScript Encoded Script file
ksh	KornShell script file
Ink	Shortcut
mda	Microsoft Access add-in program
mdb	Microsoft Access program
mdt	Microsoft Access workgroup information
mdw	Microsoft Access workgroup information
mde	Microsoft Access MDE database
mdz	Microsoft Access wizard program
msc	Microsoft MMC document
msi	Windows Installer package
msp	Windows Installer patch
Mst	Visual Test source files
ops	Office XP settings
pcd	Photo CD image

pif	Program information file for DOS programs
prf	Microsoft Outlook profile settings (Outlook 2002 only)
prg	Microsoft Visual FoxPro program
pst	Microsoft Outlook Personal Folders file
reg	registry entries
scf	Windows Explorer command (Outlook 2002 only)
scr	Screensaver file
sct	Windows Script component
shb	Shell scrap object
shs	Shell scrap object
url	Internet shortcut
vb	Visual Basic script
vbe	Visual Basic script encoded script file
vbs	Visual Basic script
wsc	Windows Script component
wsf	Windows Script file
wsh	Windows Scripting Host settings file

Table 2.2: Level-1 attachments blocked by Outlook 2000 SP3 and later.

This list of attachments can be modified by the administrator. As additional attachments present a risk to your organization, the administrator can add attachments to the Level-1 or Level-2 list.

Alternatively, in a standalone environment, or when you need to allow specific users access to certain attachment types, you can download an excellent piece of shareware from Slovak Technical Services that allows you to easily change an attachment from a Level-1 attachment to a Level-2 attachment (see Figure 2.4).

 You can find Outlook 2003 Options and Attachment Security Customizations tool on the Web at <http://www.slovaktech.com/attachmentoptions.htm>.

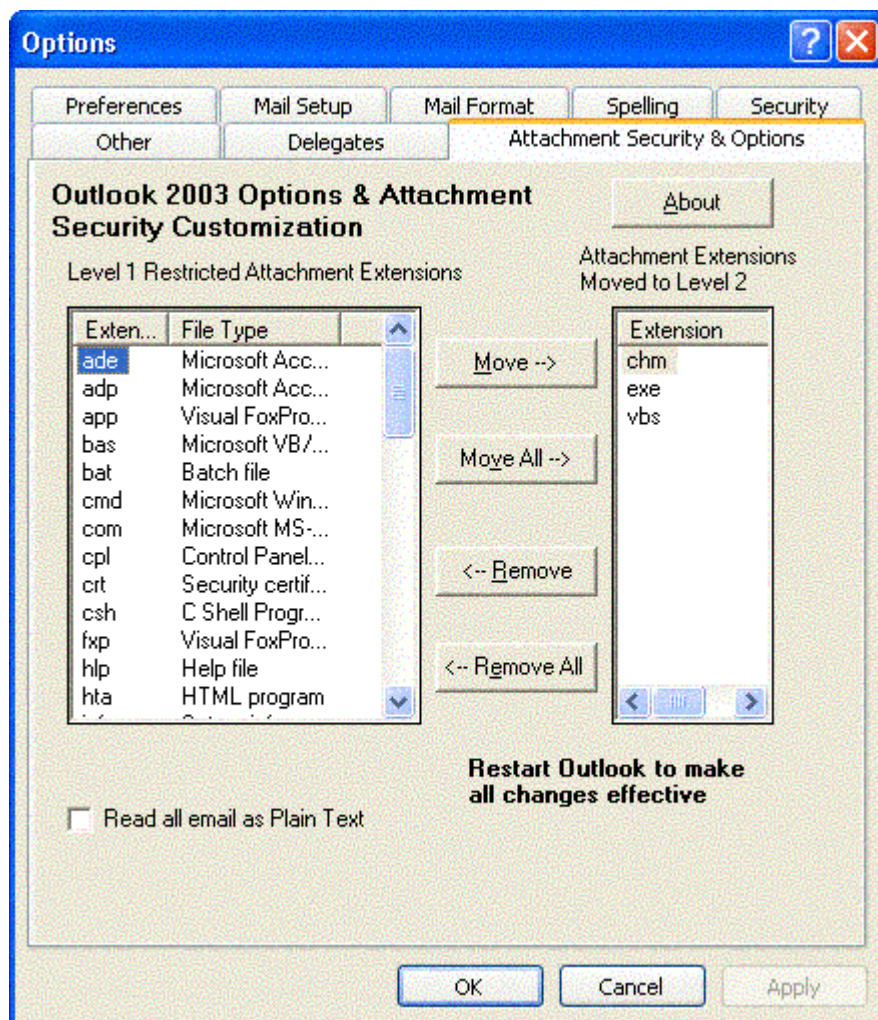


Figure 2.4: Outlook 2003 Options & Attachment Security Customization from Slovak Technical Services.

If users are allowed to move some attachments from the Level-1 category to the Level-2 category, then save the attachments to disk, there is still some risk that the attachment contains hostile content. This risk emphasizes the need not only for client-based antivirus software, but also for the need to train users to recognize the signs of potentially hostile content:

- If you don't know the sender, don't open the attachment.
- Do not open attachments if you are not expecting an attachment from that user.
- If you receive an attachment from a user that usually does not send you programs or other files that require that the attachment first be saved to the file system, check with that person first before running it.
- Avoid opening attachments that arrive with a message that is trying to entice you to open the attachment (for example, "This is really funny" or "Thought you would enjoy this").

- Be very careful about attachments that look like documents. By default Windows will “hide” extensions of known file types. A malicious attachment could say something like PerformanceReview.doc.exe. The last extensions (EXE, in this case) would be hidden from Windows. When you save this file to the hard disk, then click on it to open it, the executable would run.
- When in doubt, call the Help desk.

Demoting Attachments to Level-2

If a user has read/write access to the correct registry keys, the user can edit the values that control whether a potentially dangerous attachment is considered a Level-1 or Level-2 attachment. However, this ability is only possible with Outlook 2000 SP3 and later; earlier versions of the Outlook Security Update do not support making these changes. To move a Level-1 attachment to Level-2, locate the following registry key (substitute X for 9.0 for Outlook 2000, 10.0 for Outlook 2002, and 11.0 for Outlook 2003):

HKEY_CURRENT_USER\Software\Microsoft\Office\X\Outlook\Security.

Create a new value named Level1Remove of type REG_SZ. Enter into that value the extensions (separated by semicolons) of attachments that should not be Level-1 attachments. For example, VBS;CMD;BAT would allow VBScripts, batch files, and command files to be saved to the file system. Figure 2.5 illustrates this registry modification.

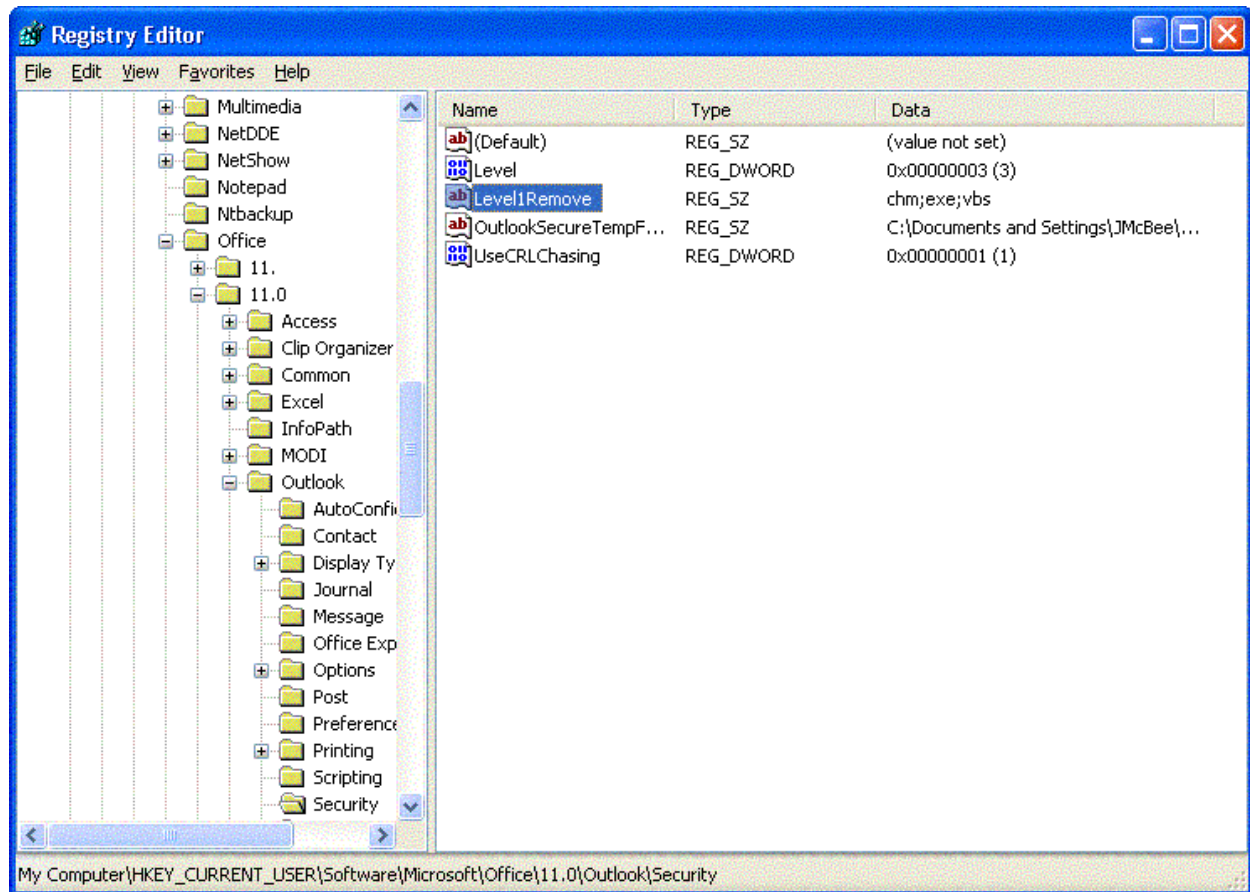



Figure 2.5: Removing CHM, EXE, and VBS file attachments from Level-1.

 You might find tips on the Internet suggesting you can get around the Outlook Security Update features by hex-editing outlib.dll or replacing that DLL with one from an earlier version. This workaround is dangerous and if problems arise, Microsoft will not support you.

Installing the Office Update for Outlook 2000

No matter which version of Outlook you're using, I strongly recommend installing the latest service pack. You can confirm that the security fix is in place in Outlook by clicking Help, About. The About Microsoft Outlook dialog box (see Figure 2.6) should indicate Corporate or Workgroup—Security Update directly below the version information. You'll see the same text in the About Microsoft Outlook dialog box of Outlook XP and Outlook 2003.


 Microsoft publishes a fairly comprehensive set of documents about deploying Outlook 2000 and later, including information about how to create an administrative shared folder that can be used for mass deployments. You can find this information, along with a link to all of the files necessary (including the Windows installer patch file), at <http://www.microsoft.com/office/ork/xp/journ/o2ksp3a.htm>.



Figure 2.6: The About Microsoft Outlook screen indicates whether the security update has been applied.

Desktop Configuration Best Practices

Regardless of the client OS or desktop platform, there are several things that you as an administrator should always do. These include:

- Use client-side antivirus software from a different vendor than you use on your Exchange Server systems. Doing so gives you a better chance of catching new viruses because you've now got two (or more) sets of signatures in use during scans.
- Users should never log on to their computers as either a Domain Admin or local Administrator (of course, you should tightly restrict who has these privileges in the first place!).
- Install the current service pack for your version of Internet Explorer (IE—SP2 for IE 5.5, or SP1 for IE 6.0) plus all current hotfixes. Regularly use Windows Update, the Microsoft Baseline Security Analyzer, the Microsoft Software Update Service, or another patch-management tool to keep your systems up to date.
- Only administrators should have NTFS permissions to modify the \Windows and \Program Files folders.
- Use a firewall or scanner that can check the content of incoming attachments.
- Prohibit the use of file-sharing services such as Kazaa, Grokster, and Morpheus on your network.
- Prevent workstations from downloading POP3 or IMAP4 mail from personal ISPs; it may be easiest to do so at the network firewall.
- Configure workstations so that they cannot establish outbound virtual private network (VPN) sessions or inbound VPN/RAS sessions.

Outlook Macro Security

Many of the very first email-based viruses as well as many of the annoying viruses that spread through Windows during the mid-1990s were actually macros. The Microsoft Office family allows developers (and unfortunately virus writers) to write macros using Visual Basic for Applications (VBA). These macros can be embedded into documents, spreadsheets, presentations, and email forms. However, Outlook has three levels of Macro security that let you restrict which macros may be run. You can view these settings in Outlook by clicking Tools, Macro, Security (Figure 2.7 shows the Security dialog box from Outlook 2000—the only change in this screen for Outlook 2003 is that the property tab Trusted Sources is now called Trusted Publishers).

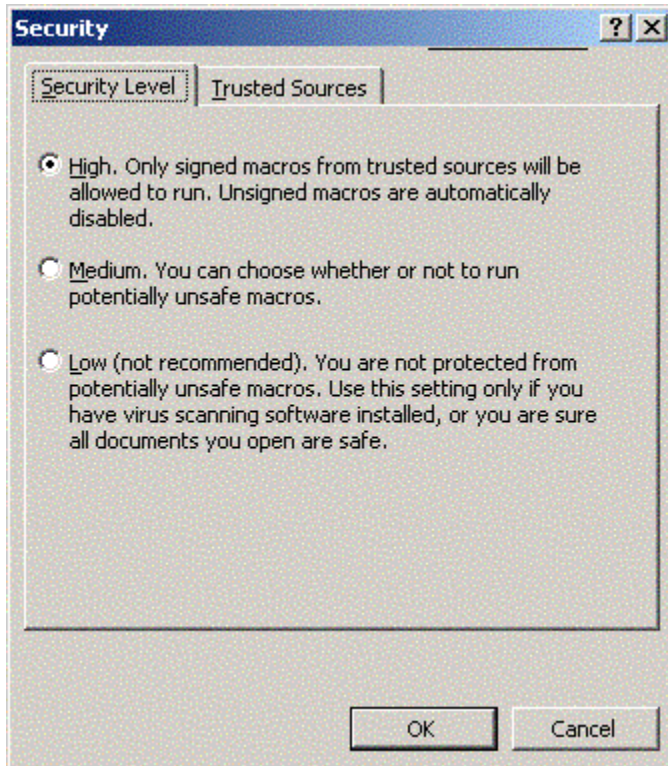


Figure 2.7: Outlook Macro security settings.

Even though the default setting is High, inevitably a user will attempt to run a form that contains an untrusted macro and get an error message. The user will then navigate to the macros security settings and drop the configuration to Low, which is not a good idea and may expose the user to future hostile content. Most users are probably not in a position to determine whether a macro should really be trusted, further they are not in a position to determine trusted publishers.

☞ The Outlook macro security settings only apply to VBA macros, not to VBScripts embedded in custom forms.

If your organization is using VBA macros, you should get your macro digitally signed and added to the Trusted Sources list. Macros that can be placed in the Trusted Sources list must be digitally signed. Outlook will not allow you to publish trusted sources that are not digitally signing their macros. The Exchange organization forms library is considered a trusted source for VBA macros.

☞ The Outlook macro security settings can be enforced through an AD Group Policy Object (GPO). To do so, you will need the administrative templates from the Office Resource Kit (ORKTOOLS.EXE) found at <http://www.microsoft.com/office/ork/xp/appndx/appc00.htm>.

Quelling Active Content

I first saw the term *active content* used by security guru Russ Cooper on his Web site (<http://www.ntbugtraq.com>). He used the term active content with respect to an email message that contains anything besides a plain-text message. This content could be Microsoft Rich Text Formatting, HTML messages, scripts, forms, or ActiveX objects and can be embedded in a message. This content can easily be used for malicious purposes—not only for viruses, but also for worms and Trojan horses, all of which are frequently spread due to the ability of Microsoft's Outlook and Outlook Express clients to support active content.

The Preview Pane

A common question is whether the Preview Pane (called the Reading Pane in Outlook 2003) is vulnerable. The preview pane (if enabled) is the portion of the Outlook window that shows the currently highlighted messages (as Figure 2.8 shows). The short answer is, yes, messages with hostile content can execute if viewed in the preview pane. Viruses and worms such as Klez, Zoher, and HLLW.Pluto can take advantage of the fact that a message is viewed in the Preview Pane.

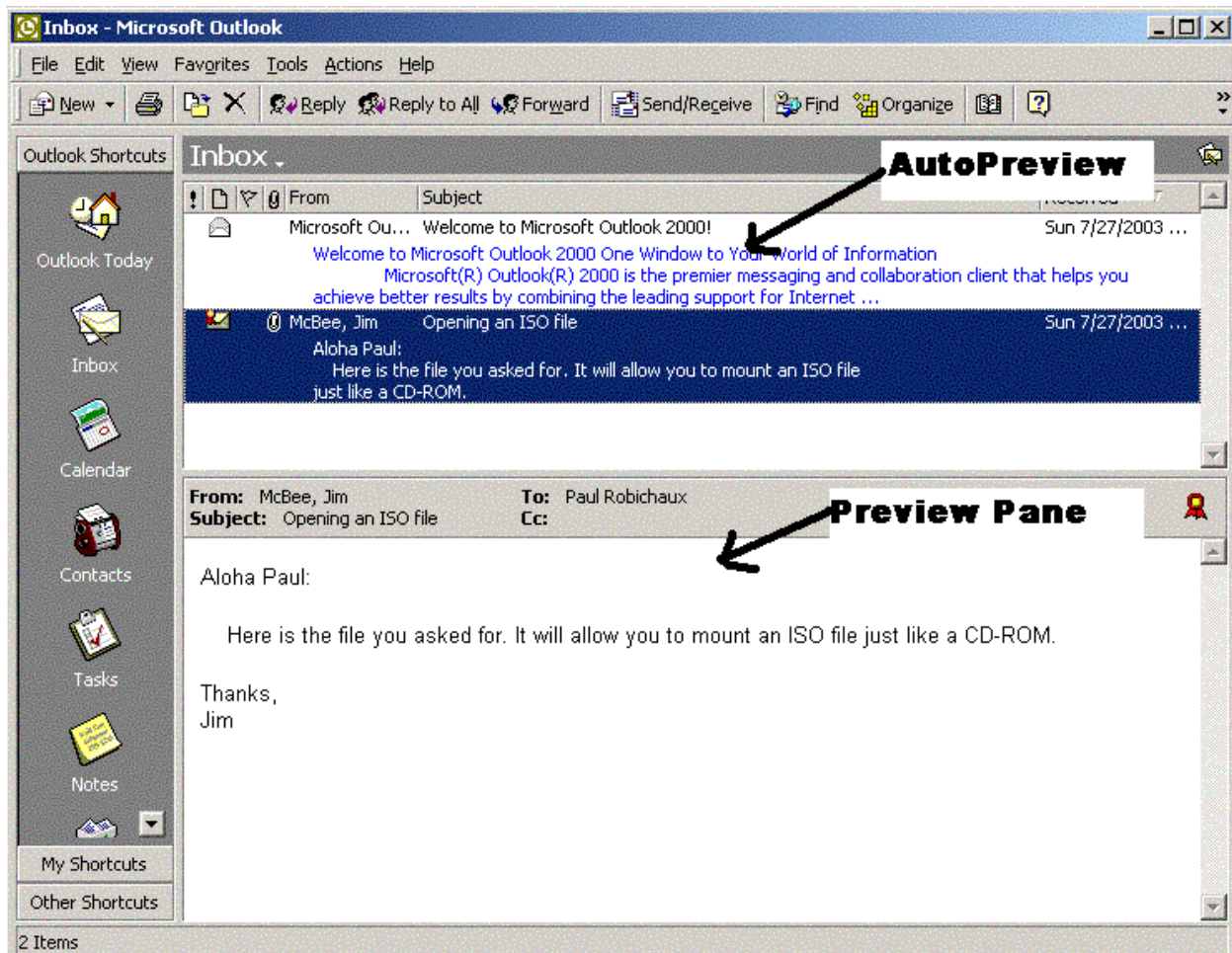


Figure 2.8: The Outlook AutoPreview feature and the Preview Pane.

With Outlook Express and Outlook 98, active content in the preview pane was executed based on the security settings in the IE security zones. Starting with Outlook 2000, ActiveX controls and active scripting items are considered to be disabled in the Preview Pane even if they are enabled in the security zone settings. This configuration will tighten security on potentially harmful message content in the Preview Pane. In certain cases, Outlook will display a dialog box indicating that items in the message can't be displayed in the Preview Pane and telling you to open the message to see all its contents.

Is the Outlook AutoPreview Feature Vulnerable?

The Outlook AutoPreview feature (which you access by selecting AutoPreview from the Outlook View menu) enables all versions of Outlook to display the first three lines of read (or unread) messages. This text normally appears in blue. When Outlook examines the message, it takes the first three lines of the message text and displays them. It makes no effort to interpret HTML codes, forms, or execute scripts; thus, the AutoPreview pane is safe and not subject to some of the dangers of active content.

 If you are running Outlook 2002 or Outlook 2003, the Preview (Reading) Pane can be disabled through an AD GPO.

Viewing the Message

Once a message with active content is opened, the active content (scripts, formatting, links, and so on) are executed and the message is viewed the way the sender intended for it to be viewed. The IE security zone settings enforce how the active content will be executed; the security zone settings also control how IE handles potentially unsafe Internet content. You can configure Outlook to use one of two IE zones when opening messages with active content: the Internet zone or the restricted zone.

For Outlook 2000 with the email security update and later, the default zone is the restricted sites zone. You can change this setting on the Security tab of the Outlook Options dialog box (which you access by selecting Tools, Options, Security), though it is recommended that you leave the zone as restricted sites. Figure 2.9 shows the Outlook 2003 Security options.

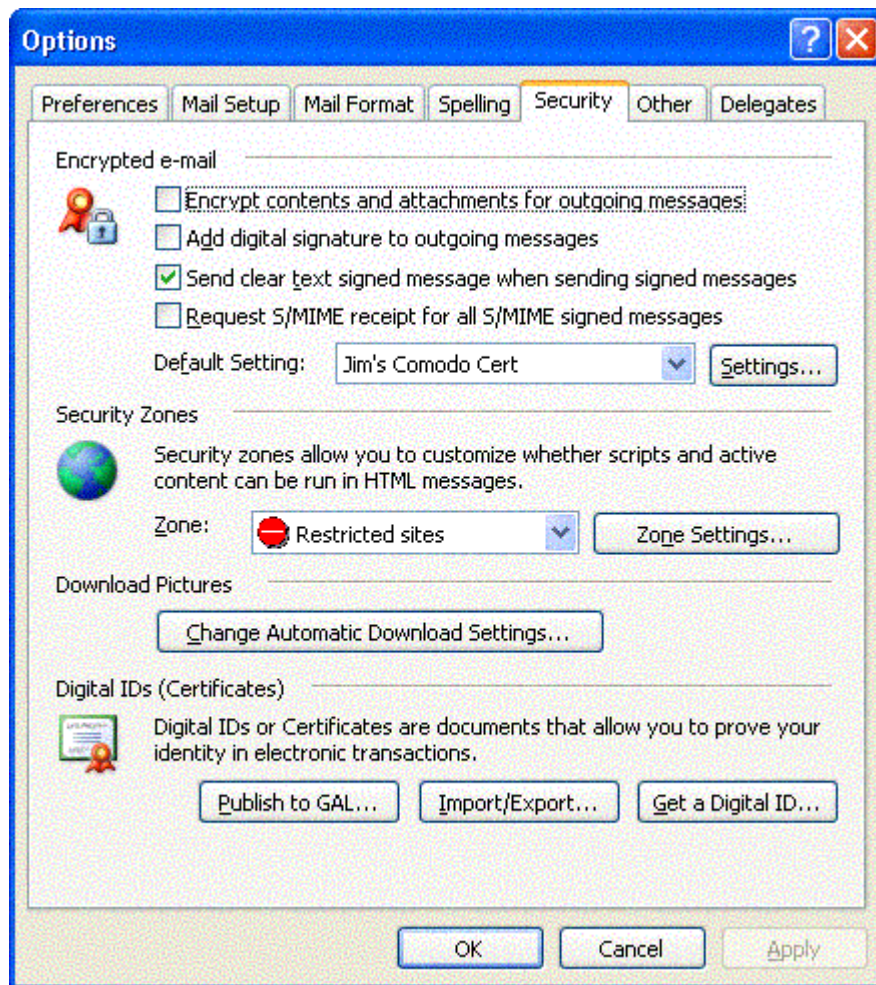


Figure 2.9: Outlook 2003 Security Options.

Outlook Express is configured similarly to Outlook in that you can designate whether email messages are treated as if they are in the Internet zone or the restricted sites zone. Figure 2.10 shows the Tools, Options, Security property options from Outlook Express. Outlook Express should always be configured in the restricted sites zone.

There are two extra check boxes in Outlook Express that you do not find in regular Outlook: the *Warn me when other applications try to send mail as me* check box and the *Do not allow attachments to be saved or opened that could potentially be a virus* check box. Users should be strongly discouraged from clearing these two check boxes because doing so will lower security on Outlook Express attachments.

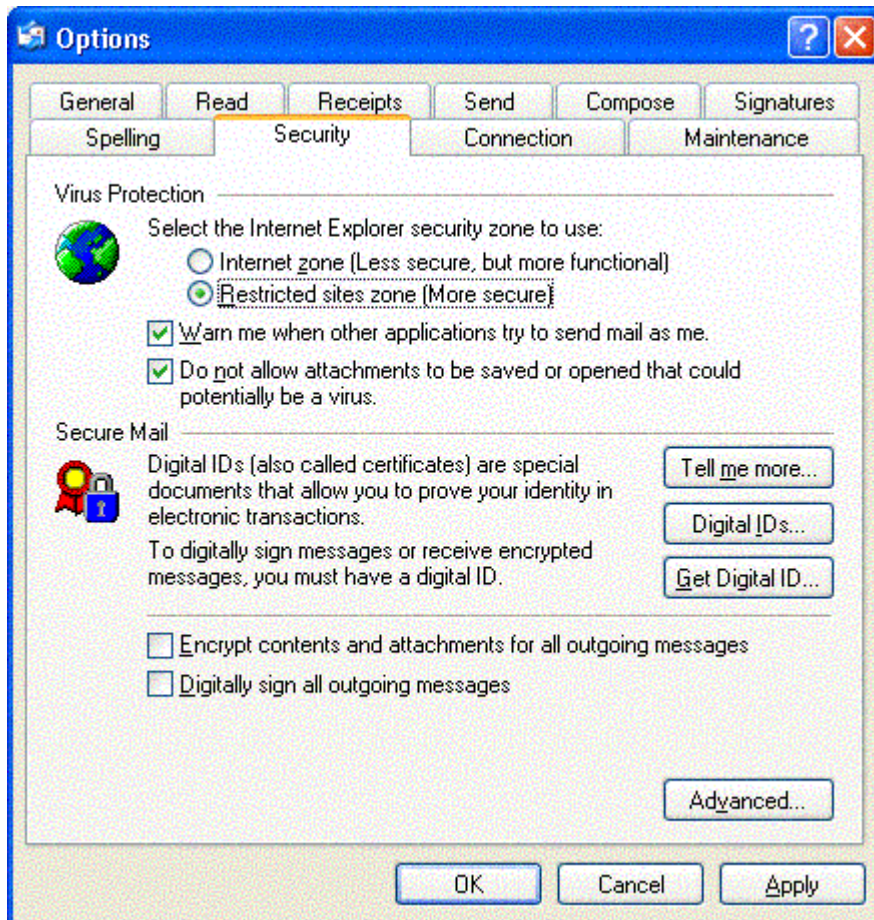


Figure 2.10: Outlook Express security options.

The zone settings can either be edited through Outlook or through IE. From within Outlook, select Tools, Options, Security, and click Zone Settings. From within IE, select Tools, Internet Options, Security. Figure 2.11 shows the Security Settings dialog box from the restricted sites security zone.

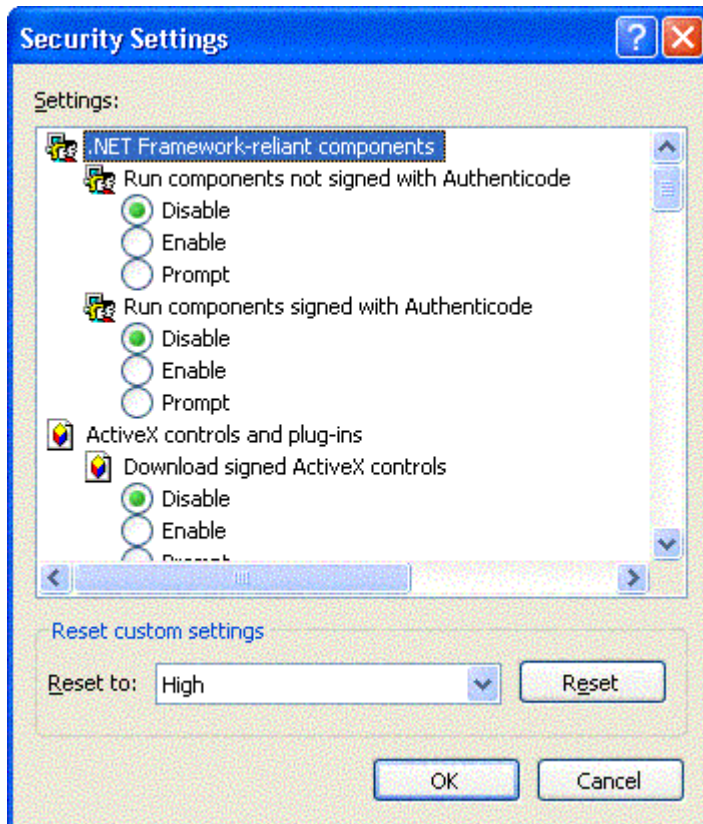


Figure 2.11: Changing the security settings for the restricted sites security zone.

You can configure the security zone settings centrally using an AD GPOs. The settings are found in the User Configuration of the policy in the Windows Settings, Internet Explorer Maintenance, Security section of the policy. You can also deploy custom security zone settings using the Internet Explorer Administration Kit (IEAK).

Determining which security settings are right for your restricted sites security zone will depend on your organization. Table 2.3 contains a list of custom settings that you can apply to your restricted sites zone to lock down the configuration as tightly as possible.

Option	Recommended Setting
Run components not signed with Authenticode (.NET Framework)	Disable
Run components signed with Authenticode (.NET Framework)	Disable
Download signed ActiveX controls	Disable
Download unsigned ActiveX controls	Disable
Initialize and script ActiveX controls not marked as safe	Disable
Run ActiveX controls and plug-ins	Disable
Script ActiveX controls marked as safe for scripting	Disable
Allow cookies that are stored on your computer	Disable
Allow per-session cookies (not stored)	Disable
File download	Disable
Font download	Disable

Java permissions	Disable Java
Access data sources across domains	Disable
Don't prompt for client certificate selection when no certificates or only one certificate exists	Disable
Drag and drop or copy and paste files	Disable
Installation of desktop items	Disable
Launching programs within an IFRAME	Disable
Navigate sub-frames across different domains	Disable
Software channel permissions	High safety
Submit non-encrypted form data	Disable
UserData persistence	Disable
Active Scripting	Disable
Allow past operations via script	Disable
Scripting of Java applets	Disable
Logon	Anonymous logon

Table 2.3: Restricted sites security zone settings for secure Outlook configuration.

The settings in this table apply to any site that is in the restricted sites zone, not just to Outlook. The restricted sites zone is just that—it is a zone that contains sites from which you can trust the content.

Reading Messages as Plain Text

I like Rich Text and HTML formatting in messages. Embedding text formatting, font changes, tables, and graphics into some messages can make them more effective. Unfortunately, as with many other useful technical features, HTML mail has been exploited to deliver malware—it seems like someone always comes along and spoils the party by misusing cool features. If you do not have a complete warm-and-fuzzy feeling after performing the other security steps in this chapter, another, and more extreme, step is to force Outlook to display all unsigned messages as plain text. This feature is only available with Outlook 2002 SP1 and later.

To enable this feature, you must make a registry change. Locate the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Office\10.0\Outlook\Options\Mail key. In this registry key, add a new REG_DWORD value called ReadAsPlain; set the data for this value to 1. Once enabled and the Outlook client is restarted, the following changes will be noticed for messages that do not have a digital signature:

- Images are now attachments
- Message content loses its formatting (font changes, colors, and so on)

This setting will affect messages both in the Preview Pane and when the message is opened in Outlook. The message is not converted to plain text in the mailbox store, but merely displayed in plain text; the message retains its full content in the mailbox store.

Centralizing Some Client Security Features

If you are the administrator of a few dozen client computers, you probably don't mind manually configuring the security settings and applying updates to these computers. The primary consideration in such situations are your time and making sure that you apply the settings consistently. However, as those few dozen machines turn in to a few hundred or more, manually applying anything is not going to cut it. In this scenario, learning some features of Windows, AD, and Exchange can make your life a lot easier.

AD to the Rescue!

If you are not already familiar with AD GPOs, there is no time like the present. There are a number of things you can do through AD GPOs that can help you to fight viruses and make your email clients more secure:

- The Software Settings GPO option will allow you to automatically deploy antivirus software and Outlook updates to client workstations. As long as the software you want to install is packaged as a Windows Installer (.MSI) file, you can install it to any AD site, domain, or organizational unit (OU).
- Windows startup and shutdown scripts will let you install software and updates that do not have MSI files.
- All versions of Outlook have ADM files (administrative templates) that you can load into a GPO (using the GPO's Administrative Templates component). These ADM files let you apply Outlook settings and prevent end users from changing them via GPO. Outlook 2000, 2002, and 2003 ADM files are found in the Office Resource Kit tools bundle (ORKTOOLS.EXE) for the version of Office that you're running; Outlook 97 and 98 can be found in the Outlook administration kit.
- In the User Configuration node of the GPO, under Windows Settings, Internet Explorer Maintenance, Security (see Figure 2.12), you can deploy custom IE security zones.

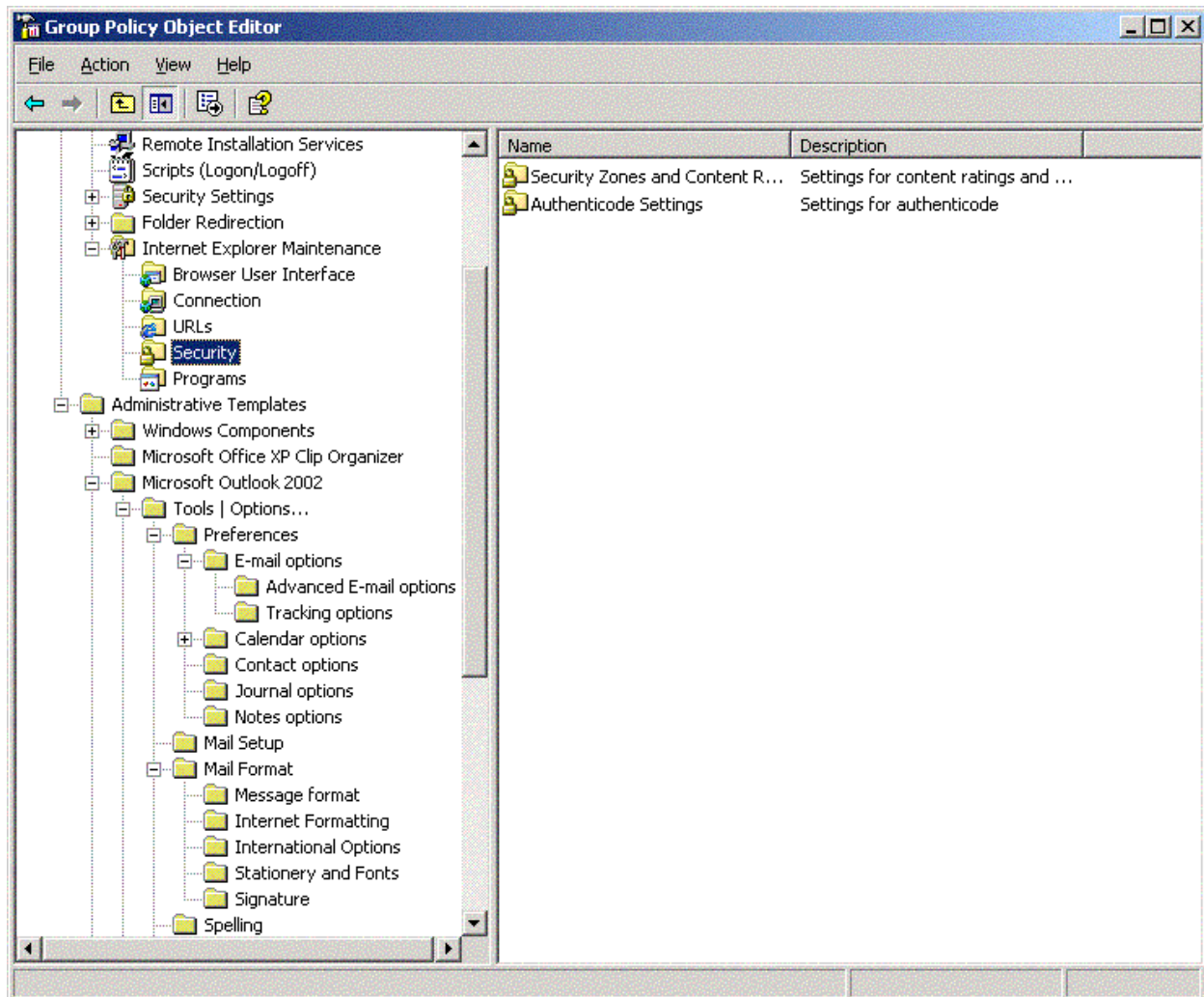



Figure 2.12: Setting security zone information through a GPO.

Using the Outlook Security Features Administrative Package

With the introduction of the Outlook Security Update, there came a need to control the Outlook security features centrally. The Outlook Security Features Administrative Package contains the tools necessary to control these security settings through a form in an Exchange public folder. This public folder can be on an Exchange 5.5, 2000, or 2003 server; Outlook 2000 SR2 and later are hardwired to look for this folder and honor settings found therein.

 The administrative package will only work if you are using Outlook as the email client and the Outlook mail delivery option is configured to deliver mail to the Exchange server's Inbox.

You can find this package on the Office 2002 resource kit CD-ROM in the `\ORK\Files\PFFiles\ORKTools\ORK10\Tools\Admpack` directory or you can download it from <http://www.microsoft.com/office/ork/xp/appndx/appa11.htm>.



Before you install the Outlook Security Features Administrative Package, be warned that many of the check boxes and items in the templates actually serve to **lower** security, not improve it. Use it with extreme caution.

Run the ADMPACK.EXE utility and specify a directory to which the files should be installed. Doing so should decompress four files:

- The readme.doc file contains documentation for how to use the administrative package.
- OutlookSecurity.oft is an Outlook item template that you will need to open and add to the Outlook Security Settings public folder in order to configure the security settings.
- Comdlg32.ocx is a program that is necessary to specify trusted COM add-ins. This file must be copied to the administrator's workstation and placed in the \windows\system32 or \winnt\system32 directory. This ocx file must be registered by typing
`regsvr32.exe comdlg32.ocx.`
- Hashctl.dll is a dll that is necessary to specify trusted COM add-ins. This file must be copied to the administrator's workstation and placed in the \windows\system32 or \winnt\system32 directory. This dll file must be registered by typing
`regsvr32.exe hashctl.dll.`

Next, create a folder in the root of the public folder hierarchy called either Outlook Security Settings (for Outlook 2000) or Outlook 10 Security Settings (for Outlook XP and Outlook 2003). This folder must be exactly one of these two names and it must be in the root of the public folder tree. You can have both folders, in which case you can apply separate settings to the two client families; however, doing so might end up causing confusion for your users and thus hassles for you. Change the folder's permissions so that the administrator has full control of the folder, Default has Reviewer permission, and Anonymous has None, as Figure 2.13 illustrates.

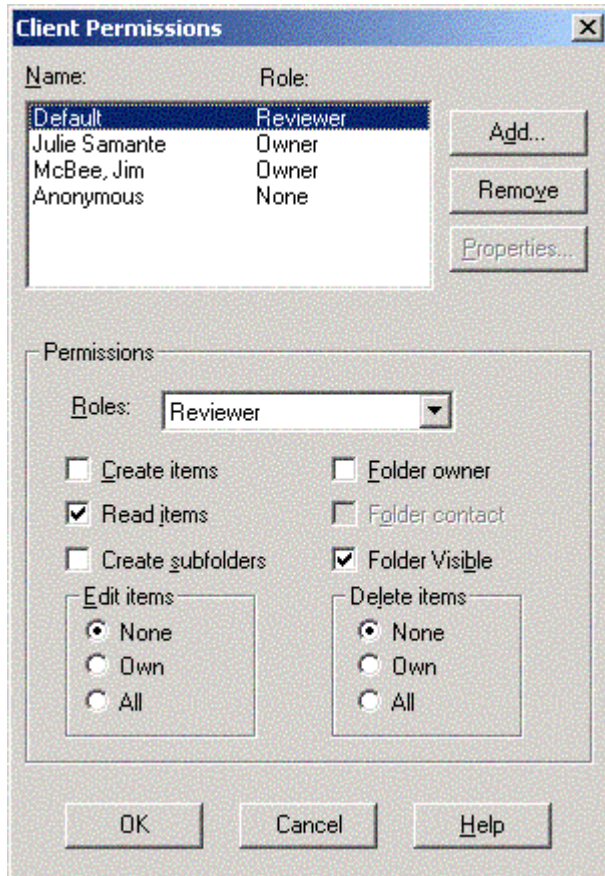


Figure 2.13: Client permissions for the Outlook Security Settings folder.

You now need to get the form published into the public folder's forms list. To do so, follow these steps:

1. On the client computer on which you registered the hashctl.dll and comdlg32.ocx files, double-click the OutlookSecurity.oft file.
2. Specify the folder name you created in the root of the public folder tree. You should now see the Default Security Settings form.
3. From the form's Tools menu, select Forms, Publish Form.
4. In the Look In drop-down list box, select the name of the public folder (Outlook Security Settings or Outlook 10 Security Settings)
5. In the Display Name enter
Outlook Security Form
the Form Name will automatically be filled in.
6. Click Publish.
7. Close the form without saving changes.

You are now ready to create custom Outlook security settings. To create a new form, highlight the Outlook Security Settings folder, and select Actions, New Outlook Security Form. Figure 2.14 shows the Outlook Security Settings property tab of this form. This page mostly controls access to attachments and which attachments are considered Level-1 or Level-2 attachments.

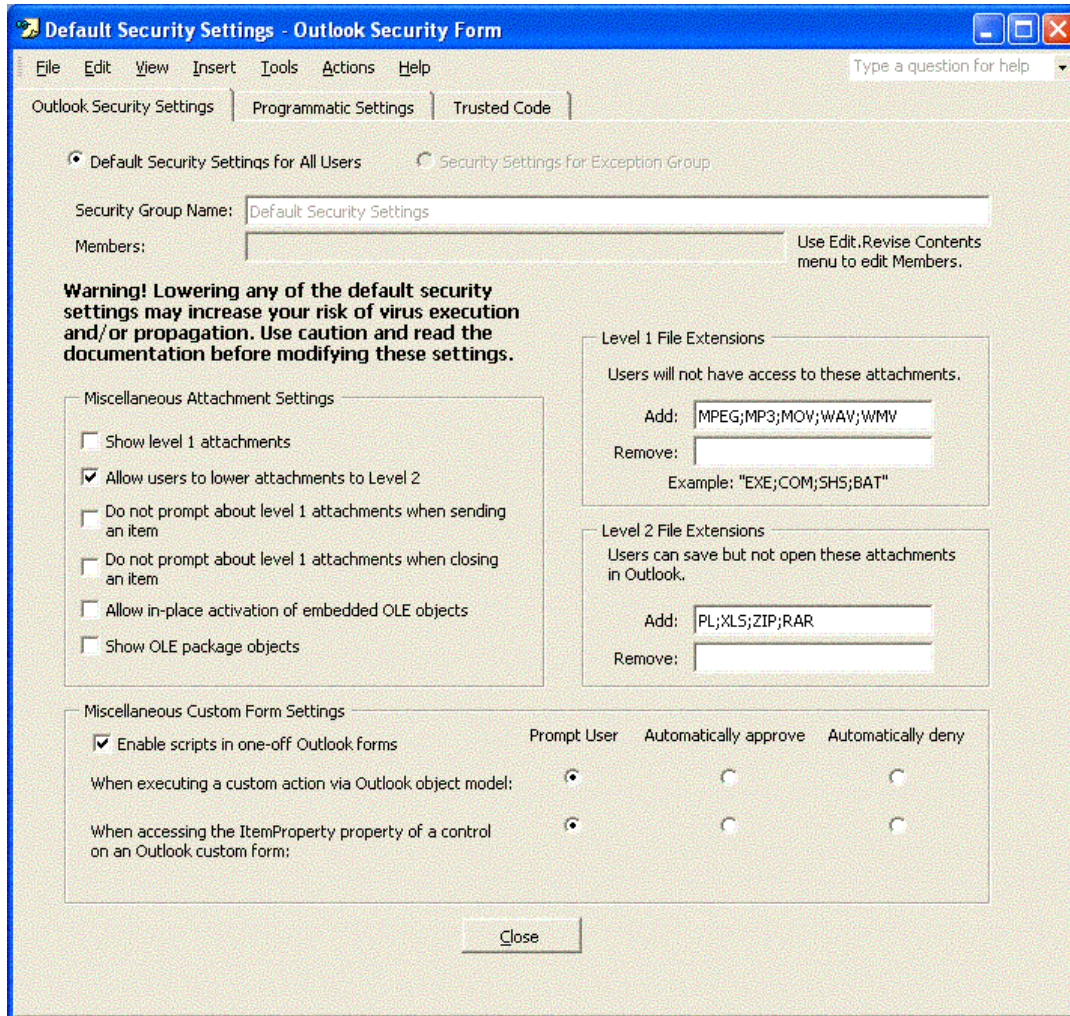



Figure 2.14: The Outlook Security Settings form.

As I previously mentioned, understanding these settings is important before you implement this template because most configuration options on this template reduce security rather than increasing it.

 Much of the Outlook Security Settings template refers to Level-1 attachments. See Table 2.2 earlier in this chapter for a partial listing of the attachments that are considered Level-1 by default.

When you create a form, you are either creating a form that will apply to everyone or a form that applies to a specific group of users. If the radio button *Default Security Settings for All Users* is selected, the form applies to everyone. However, if you create a form in which the *Security Settings for Exception Group* option is selected, you will be able to specify a list of email aliases in the Members list to which this template is specifying exceptions to the default. The Miscellaneous Attachment Settings section of this form controls the types of attachments to which the user has access. Table 2.4 lists these settings and provides a description of each.

Setting	Description
Show level 1 attachments	The setting name is deceiving. If selected, this setting allows users access to the attachment. Selecting this check box will <i>lower</i> attachment security!
Allow users to lower attachments to Level 2	Lets the user demote Level-1 attachments to Level-2 attachments by either changing the registry value that contains these attachments or by using a third-party tool (such as Slovak Technical Services' Attachment Options tool). Without this setting, users can still make the client-side changes, but Outlook will ignore them. Selecting this check box may allow users to <i>lower</i> attachment security that you did not want lowered.
Do not prompt about level 1 attachments when sending an item	Disables the warning users receive when they send a message that includes a Level-1 attachment. Selecting this check box stops the warning messages; users should always get warning messages to remind them before they send potentially unsafe attachments.
Do not prompt about level 1 attachments when closing an item	Disables the warning users get when they close a message that includes a Level-1 attachment. Selecting this check box stops the warning messages; users should always be reminded when they are storing a potentially unsafe message.
Allow in-place activation of embedded OLE objects	Enables the ability for OLE embedded attachments to be opened if the user double-clicks the attachment (such as an Excel spreadsheet or PowerPoint presentation). If Word is selected as the email editor, embedded Word documents will always open regardless of this setting. This setting can compromise attachment security if any malware has managed to propagate as an OLE attachment.
Show OLE package objects	Allows the OLE objects that have been packaged to show. This setting exists because it is easy to change the icon of an embedded package and thus disguise something that might really be dangerous.

Table 2.4: Miscellaneous attachment settings.

On the right side of the template are the Level-1 and Level-2 File Extensions sections. From these sections, you can add or remove extensions from the default Level-1 file list. As you will recall, Level-1 attachments cannot be opened or saved from within the Outlook client. The Level-2 list is for files that must be saved to the hard disk or shared storage before they can be opened. This setting is useful if you decide that document types that can have embedded macros (DOC, XLS, PPT, and so on) should be saved to hard disk and not retrieved directly from the email message.

At the bottom of the form is the Miscellaneous Custom Form Settings section. These settings control the behavior of Outlook when custom controls are added to an Outlook form. Table 2.5 lists the settings and describes what each setting does.

Setting	Description
Enable scripts in one-off Outlook forms	A one-off form is a form that is contained within the message rather than in a personal, folder, or organization forms library. This setting enables the use of one-off forms if your users receive messages that include the form itself. Ideally, forms should be stored in a trusted location (such as the Exchange organization forms library), but this setting may have to be enabled if your users receive messages that include forms from outside organizations. If this occurs regularly, take steps to get copies of the forms, validate them, and store them in your own organization forms library.
When executing a custom action via the Outlook object model	This setting specifies what happens when a custom form is using the Outlook object model: <ul style="list-style-type: none"> • Prompt User—The user receives a dialog box that states that the Outlook object model is being used by a program and asks whether this is okay. • Automatically approve—Custom forms can automatically access the Outlook object model. • Automatically deny—Custom forms will automatically be blocked from using the Outlook object model.
When accessing the ItemProperty property of a control on an Outlook custom form	This setting specifies what happens when a custom control is included in a form, and the control attempts to access the address field of the message: <ul style="list-style-type: none"> • Prompt User—The user receives a dialog box that states that a program is attempting to access properties which could give that program access to the address book. • Automatically approve—Custom forms can automatically access the Outlook ItemProperty. • Automatically deny—Custom forms will automatically be blocked from using the Outlook ItemProperty.

Table 2.5: Miscellaneous custom form settings.

The next dialog box on the Outlook Security Settings template is the Programmatic Settings, which Figure 2.15 shows. The default setting for each of these programmatic settings is to allow the user to designate whether the Outlook add-in or third-party application works.

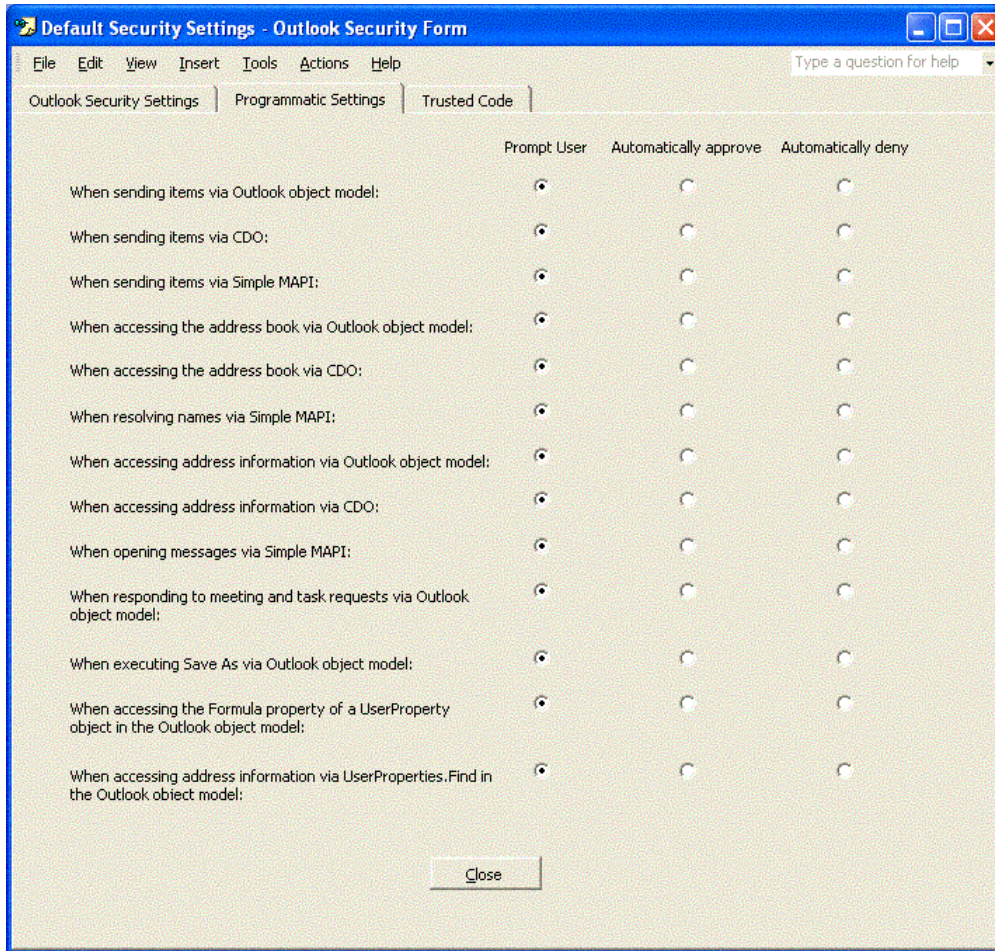


Figure 2.15: Programmatic Settings options.

The Programmatic Settings page lets the administrator specify the behavior of Outlook when third-party add-ins, forms, and external applications attempt to use CDO, the Outlook object model, or MAPI functions. The default configuration for each of these 13 settings is to prompt the user when one of these actions is taking place. How you configure these settings is determined by whether you trust your users to think before they allow a program to use the Outlook address book, and so on. For organizations that are connected to public email systems, these settings should never be configured to Automatically approve.

There are three options for each of these settings:

- Prompt user—The user receives a message stating that a particular application is attempting to use a programmatic feature of Outlook. The user can either say it is okay or deny the operation. Some of the prompts allow the user to specify how long the application can have access to Outlook (see Figure 2.16).
- Automatically approve—External applications will automatically be allowed programmatic access to Outlook.
- Automatically deny—External applications will automatically be blocked from programmatic access to Outlook.

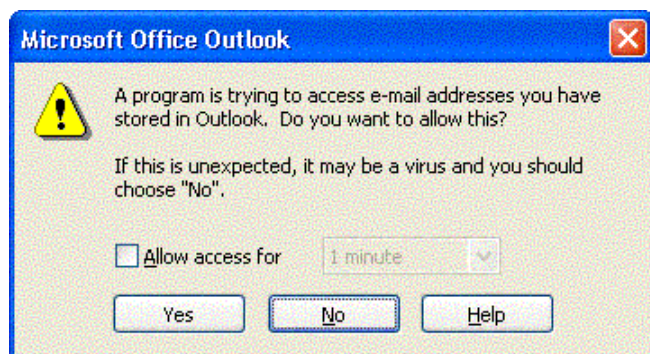


Figure 2.16: Outlook users are prompted when an external program programmatically attempts to access Outlook.

Table 2.6 provides information about each of the settings.

Setting	Description
When sending items via Outlook object model	Affects behavior of Outlook when an application attempts to programmatically use the Outlook object model to send a message
When sending items via CDO	Affects the behavior of Outlook when an application attempts to send mail programmatically using CDO
When sending items via Simple MAPI	Affects the behavior of Outlook when an application uses Simple MAPI function calls to send a message
When accessing the address book via Outlook object model	Affects the behavior of Outlook when an application attempts to gain access to the Outlook address book via the Outlook object model
When accessing the address book via CDO	Affects the behavior of Outlook when an application attempts to gain access to the Outlook address book via CDO
When resolving names via Simple MAPI	Affects the behavior of Outlook when an application attempts to gain access to the Outlook address book via Simple MAPI
When accessing address information via Outlook object model	Affects the behavior of Outlook when an application attempts to gain access to a message's addressing properties (To, From, CC, and so on) via the Outlook object model
When accessing address information via CDO	Affects the behavior of Outlook when an application attempts to gain access to a message's addressing properties (To, From, CC, and so on) via CDO
When opening messages via Simple MAPI	Affects the behavior of Outlook when an application attempts to gain access to a message's addressing properties (To, From, CC, and so on) via Simple MAPI
When responding to meeting and task requests via Outlook object model	Affects the behavior of Outlook when an application attempts to manipulate meeting and task requests via the Outlook object model
When executing Save As via the Outlook object model	Affects the behavior of Outlook when an application attempts to use the Save As feature using the Outlook object model
When accessing the Formula property of a UserProperty object in the Outlook object model	Affects the behavior of Outlook when an application attempts to access the UserProperty object's Formula property using the Outlook object model

Table 2.6: Programmatic settings options.

For more information about CDO, MAPI, and the Outlook object model, visit <http://msdn.microsoft.com/office>.

Finally, the Trusted Code property page is used to specify custom and third-party add-ins that should be trusted by Outlook. These trusted COM add-ins can run without hitting the restrictions placed on them through the Programmatic Settings Outlook object model restrictions. Figure 2.17 shows the Trusted Code list. Simply specify the names of the DLLs that are provided by the vendor or developer of the third-party COM add-in.

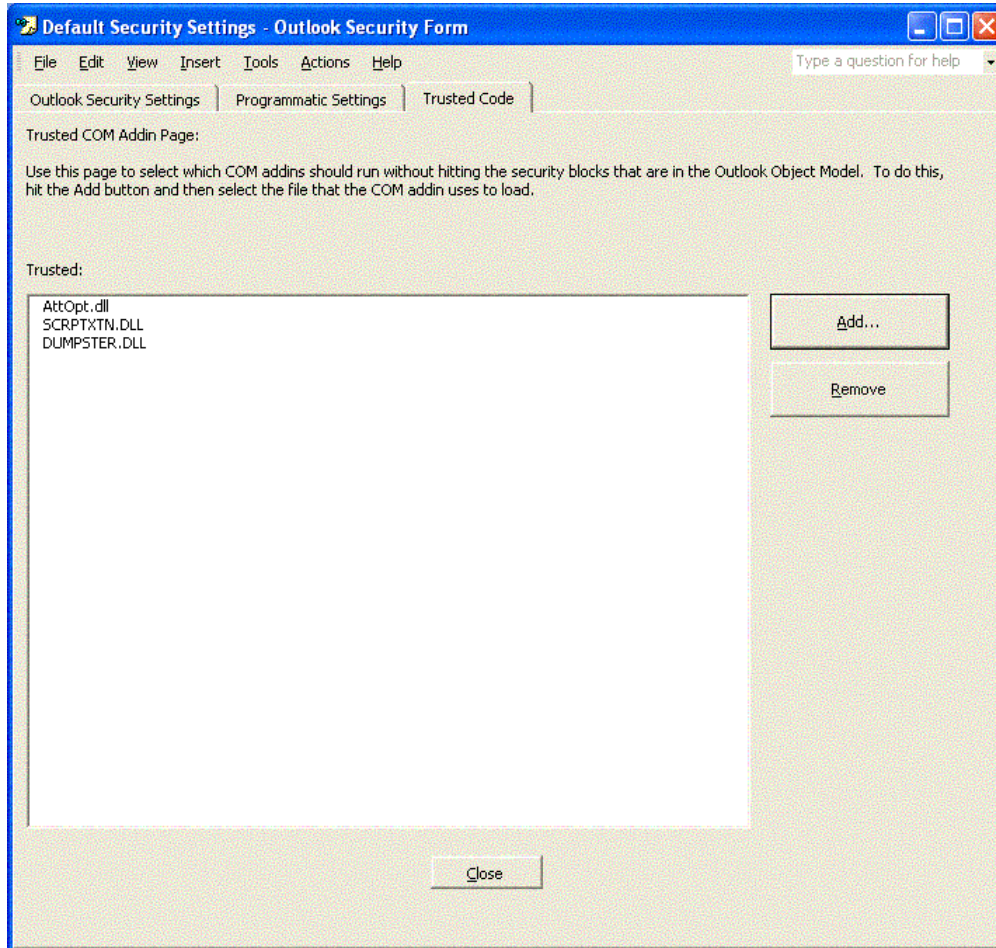


Figure 2.17: Trusted Code list.

Where Are My Custom Settings?

There is one final step that you need to take in order to make the Outlook Security Settings work for Outlook clients. You need a registry setting that instructs Outlook to check the public folder for the Outlook Security Settings templates. Locate the HKEY_CURRENT_USER\Software\Policies\Microsoft\Security registry key (you might need to create the Security subkey if it does not exist). In this key, create a REG_DWORD value called CheckAdminSettings. Set the data to one of three settings:

- A value of 0—Use the default settings, which means it does not consult the public folders
- A value of 1—Use the custom settings found in the Outlook Security Settings folder
- A value of 2—Use the custom settings found in the Outlook 10 Security Settings folder

Final Thoughts About Outlook Security Settings

A couple of parting thoughts about the Outlook Security Settings feature that I want you to keep in mind when you implement this feature:

- Many of these settings can easily undo any additional security settings that you may have gained with the Outlook Security Update or by installing later versions of Outlook
- The template should work fine for Outlook 2000 SP3 and later
- The public folder that holds this template should be replicated so that it is accessible in all Exchange sites or routing groups
- When you save a template, you are prompted for your credentials twice; enter your credentials both times
- Finally, test your features thoroughly on a test system before introducing it into production; make sure that all custom messaging applications work properly

Using Exchange Server to Control Outlook Client Versions

Exchange 2000 SP1 introduced a new feature that allows the administrator to restrict access to the Exchange Server system based on the MAPI version of the Outlook client. This feature is also supported on Exchange Server 2003. I find this feature extremely useful if I decide that I only want a specific set of client versions to be allowed to use the Exchange Server. For example, perhaps I only want Outlook clients that are running Outlook 2000 SP3 or later to be able to use Exchange; this configuration will help to guarantee that client-side precautions are being taken. You can even constrain the versions so that only a specific range of versions are allowed and nothing later than that version. This configuration can ensure that no one installs an Outlook client that has not yet been approved and tested (for example, if you want to prohibit anyone from installing a beta version of Outlook 2003).

The feature must be enabled in the registry of each Exchange Server, and you must know the exact MAPI versions of the client according to the Exchange Server. Simply looking in the Outlook client's Help, About screen will not give you the information you need. I found this information by connecting to the Exchange Server using different versions of the MAPI client, then recording the information found in the Exchange mailbox store's Logons container (see Figure 2.18).

User Name	Windows 2000 Account	Logon Time	Last Access Time	Client Version
_Spam1 Test	NT AUTHORITY\SYSTEM	7/25/2003 6:31 ...	7/26/2003 12:23 PM	HTTP
Administrator	VOLCANOSURF\Administrator	7/27/2003 6:05 ...	7/27/2003 6:05 PM	6.0.6944.1
Ayden Hutchinson	VOLCANOSURF\AHutchinson	7/27/2003 8:46 ...	7/27/2003 8:46 PM	HTTP
Ayden Hutchinson	VOLCANOSURF\AHutchinson	7/27/2003 8:46 ...	7/27/2003 8:57 PM	HTTP
Ayden Hutchinson	NT AUTHORITY\SYSTEM	7/27/2003 8:46 ...	7/27/2003 8:46 PM	HTTP
Clayton Kamiya	VOLCANOSURF\CKamiya	7/27/2003 8:56 ...	7/27/2003 8:57 PM	11.0.5329.6
Clayton Kamiya	VOLCANOSURF\CKamiya	7/27/2003 8:57 ...	7/27/2003 8:57 PM	11.0.5329.6
McBee, Jim	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/25/2003 4:20 PM	HTTP
McBee, Jim	VOLCANOSURF\JMcBee	7/27/2003 7:14 ...	7/27/2003 8:28 PM	10.0.0.4115
Paul Robichaux	VOLCANOSURF\Paul	7/27/2003 8:26 ...	7/27/2003 8:51 PM	5.0.3165.0
Paul Robichaux	VOLCANOSURF\Paul	7/27/2003 8:26 ...	7/27/2003 8:26 PM	5.0.3165.0
Paul Robichaux	VOLCANOSURF\Paul	7/27/2003 8:26 ...	7/27/2003 8:26 PM	5.0.3165.0
SMTP (KILAUEA-{B55C70A4-E...	NT AUTHORITY\SYSTEM	7/27/2003 8:53 ...	7/27/2003 8:53 PM	SMTP
SMTP (KILAUEA-{B55C70A4-E...	NT AUTHORITY\SYSTEM	7/27/2003 8:53 ...	7/27/2003 8:53 PM	SMTP
SMTP (KILAUEA-{B55C70A4-E...	NT AUTHORITY\SYSTEM	7/27/2003 8:53 ...	7/27/2003 8:53 PM	SMTP
SMTP (KILAUEA-{B55C70A4-E...	NT AUTHORITY\SYSTEM	7/27/2003 8:53 ...	7/27/2003 8:53 PM	SMTP
Sueko Miura	VOLCANOSURF\SMiura	7/27/2003 8:52 ...	7/27/2003 8:56 PM	HTTP
Sueko Miura	VOLCANOSURF\SMiura	7/27/2003 8:54 ...	7/27/2003 8:54 PM	HTTP
Sueko Miura	NT AUTHORITY\SYSTEM	7/27/2003 8:54 ...	7/27/2003 8:54 PM	HTTP
Suriya Supatanasakul	NT AUTHORITY\SYSTEM	7/27/2003 8:51 ...	7/27/2003 8:51 PM	HTTP
Suriya Supatanasakul	VOLCANOSURF\SSupatanasakul	7/27/2003 8:51 ...	7/27/2003 8:51 PM	HTTP
Suriya Supatanasakul	VOLCANOSURF\SSupatanasakul	7/27/2003 8:51 ...	7/27/2003 8:56 PM	HTTP
System Attendant	NT AUTHORITY\SYSTEM	7/25/2003 4:18 ...	7/27/2003 8:56 PM	6.0.6944.1
System Attendant	NT AUTHORITY\SYSTEM	7/26/2003 12:0...	7/27/2003 3:20 PM	6.0.6944.1
System Attendant	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/25/2003 4:20 PM	6.0.6944.1
System Attendant	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/25/2003 4:20 PM	6.0.6944.1
System Attendant	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/25/2003 4:20 PM	6.0.6944.1
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:18 ...	7/25/2003 4:18 PM	OLEDB
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:18 ...	7/25/2003 4:18 PM	OLEDB
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/27/2003 8:46 PM	HTTP
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/25/2003 4:20 PM	HTTP
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/27/2003 8:46 PM	HTTP
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 4:20 ...	7/27/2003 8:46 PM	HTTP
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/25/2003 6:31 ...	7/25/2003 6:31 PM	HTTP
SystemMailbox{B55C70A4-E0...	NT AUTHORITY\SYSTEM	7/27/2003 8:51 ...	7/27/2003 8:51 PM	HTTP
SystemMailbox{B55C70A4-F0...	NT AUTHORITY\SYSTEM	7/27/2003 8:52 ...	7/27/2003 8:52 PM	HTTP

Figure 2.18: Determining the MAPI client version.


The MAPI version number looks much like an IP address, except that each decimal place is a 16-bit number, so the range can be between 0 and 65,535. The number is in the form of *W.X.Y.Z*. Table 2.7 shows the MAPI client version. The value you actually need in the registry is in the Value Required to Restrict Logon column. When determining which version number to put in to the Disable MAPI Clients registry value, use only the *W.Y.Z* format (leave out the *X*).

Table 2.7 is by no means inclusive of all MAPI clients because each service release and security fix may update this version number. In addition, the version in Help, About does not always agree with the version the server sees.

Client	Help About Version	MAPI Version	Value Required to Restrict Logon
Exchange 4 Inbox Client	4.0.993.3	4.0.993.3	4.993.3
Exchange 5 Inbox Client	5.0.1457.3	5.0.1457.3	5.1457.3
Outlook 97 (from Office 97 CD-ROM)	8.02.4212	5.0.1457.3	5.1457.3
Outlook 97 8.03 (with Exchange 5.5)	8.03.4629	5.0.1960.0	5.1960.0
Outlook 98	8.5.5104.6	5.0.2178.0	5.2178.0
Outlook 2000 (with Office 2000)	9.0.0.2711	5.0.2819.0	5.2819.0
Outlook 2000 SR-1	9.0.0.3821	5.0.3121.0	5.3121.0
Outlook 2000 SR-1 (after Office 2000 SP2 applied)	9.0.0.4527	5.0.3144.0	5.3144.0
Outlook 2000 SR-1 with the Security Update	9.0.0.5414	5.0.3158.0	5.3158.0
Outlook 2000 SP3	9.0.0.6627	5.0.3165.0	5.3165.0
Outlook 2002	10.2627.2625	10.0.0.2627	10.0.2627
Outlook 2002 SP1	10.3513.3501	10.0.0.3416	10.0.3416
Outlook 2002 SP2	10.4219.4219	10.0.0.4115	10.0.4115
Outlook 2003 SP2 January 2003 Update	10.4712.4219	10.0.0.4115	10.0.4115
Outlook 2003 Beta 2	11.5329.5329	11.0.5329.6	11.5329.6
Exchange 2000 SP1 components	N/A	6.0.4712.0	6.4712.0
Exchange 2003 components	N/A	6.0.6944.1	6.6944.1

Table 2.7: MAPI client and Outlook versions.

To restrict access to MAPI clients you will need to locate the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeIS\Parameters` System registry key. In this key, create a valued named `Disable MAPI Clients` of type `REG_SZ`. You then populate this value with the list of MAPI client versions that are not allowed to access the server.

 The MAPI version 6 components must always be allowed to log on. These are Exchange 2000 and Exchange 2003 components such as the System Attendant.

You can mix and match values in the registry key to allow certain ranges of clients to access the information store. Additionally, you can put in multiple values by separating them with commas. Table 2.8 provides some examples of restrictions you can place on MAPI clients.

Registry Value	Effective Results on MAPI Clients
-6.0.0, 7.0.0-	Allows only the Exchange 2000 or Exchange 2003 components to access the information store by blocking all clients before version 6 and all clients after version 7.
10.0.2627	Prevents the original release of Outlook XP clients from accessing the store.
-5.3165.0	Prevents any clients prior to Outlook 2000 SP3 from accessing the store.
4.993.3-5.1457.3	Prevents Exchange 4, 5, and original Outlook 97 clients from accessing the store.
--5.3165.0, 10.0.4115 -	Allows only clients between Outlook 2003 SP3 and Outlook 2002 SP2.

Table 2.8: Examples that you can use in the *Disable MAPI Clients* registry value.

Once this registry value is in place and the information store has been stopped and restarted, clients will get a *The attempt to log on to the Microsoft Exchange Server computer has failed* message if they try to access the Exchange Server from a client whose MAPI version you are blocking. If you allow only very specific versions of MAPI clients, you will have to remember to always update this if you update the MAPI client version.

Summary

You can achieve and maintain client-side email security with a minimal loss of functionality as long as you have a good sense of the security components you need to put in place and as long as end users remain diligent to potential security threats. When designing a client-side email scheme, keep the following considerations in mind:

- All client computers should have up-to-date antivirus software installed—no exceptions!
- Desktop OSs should remain reasonably up-to-date on OSs and service packs. If you are running Windows NT Workstation 3.51 or Windows 98, it is time to consider an upgrade. If you have not applied OS and browser critical updates in the past 2 months, it is time to apply them.
- Email client software should be updated with the latest security patches available. If you are running Outlook 97 or Outlook Express 4.0, it is time to consider upgrading.

In the next chapter, we'll explore what you can do to help protect your server—and the messaging resources and services it offers to clients—from viruses, worms, and malware.

Chapter 3: Server-Side Antivirus Protection

Recent outbreaks of viruses, worms, and blended threats provide evidence that there is much more to virus protection than installing antivirus software on a client computer and hoping for the best. Some of the more notable *blended threats* such as Nimda, the Code Red worm, BugBear, SoBig, and Blaster have caught many network administrators by surprise. Blended threats are threats that combine characteristics of viruses, worms, and Trojan horses in one nasty package.

In most organizations, virus protection is not a single action that administrators perform but rather a combination of procedures and protection. Security experts call this layering *defense in depth*—the point is to harden your systems by putting up multiple barriers. The first two chapters of this guide discussed the basics of viruses and worms and the basics of protecting Microsoft clients against email-based threats. This chapter builds on the knowledge you have gained from the previous three chapters and focuses on protection of the Exchange server.

Protecting your Exchange servers from viruses, worms, Trojan horses, and blended threats is not simply a process of picking any virus protection software and assuming you are protected. Servers must be properly patched, Exchange must be properly configured, and the software that you pick for your Exchange servers must be able to accurately detect viruses and protect against malicious attachments.

Basics for Protecting Your Organization

Regardless of which virus protection mechanism you choose for your organization, true protection involves much more than simply picking the right antivirus software. Protection measures need to include properly configuring the Exchange server, protecting the network with a firewall, using more than a single virus protection system, keeping software updated, and blocking specific files.



Many of the recommendations in this section will affect the functionality of your messaging system; thus, you should thoroughly discuss potential changes within the IT department and properly publish them to the user community.

Properly Configuring the Firewall

Protecting your network against hostile email content is not often associated with proper firewall configuration. However, there are several firewall configuration measures you can take in order to better protect your organization against viruses and other hostile mail content. The following list highlights suggestions for firewall implementation and configuration:

- Only servers that absolutely *must* be exposed to the Internet should be, such as servers that accept inbound SMTP mail or that provide access to Internet protocols (HTTP, POP3, or IMAP4). If your firewall allows reverse proxying of ports, use that option instead of directly opening each port. Open only the ports that are necessary.

- Require SSL for inbound IMAP, POP, and OWA connections. Doing so will cause some users to complain, but if you don't, your users' credentials may be transmitted in plain text, visible to any attacker. Properly and thoroughly documenting the setup and use of SSL and properly notifying users before SSL is required can help to mitigate some of these complaints.
- Outbound SMTP mail should only be accepted from authorized servers, such as the Exchange servers that host an SMTP Connector or your SMTP content/virus inspection gateway. This configuration will prevent viruses such as SoBig that run their own SMTP engines from sending viruses to users outside of your organization.
- Outbound POP3 and IMAP4 requests to ISP servers outside of your organization should be restricted. Doing so will prevent users from retrieving mail to their desktops from other mail services. This restriction might be an annoyance for some users; you can recommend that they use Web mail clients to these ISPs.
- Direct, inbound RPC requests from the Internet should be blocked for MAPI clients; if a remote MAPI client wants to use Outlook as a MAPI client remotely, they should connect to your organization via a VPN, Microsoft's Internet Security and Acceleration (ISA) Server, or Exchange 2003's RPC-over-HTTP feature. In years past, some organizations chose to open RPCs directly to their Exchange servers; doing so is not a good practice because of the vulnerabilities in various implementations of RPC.
- Exchange front-end servers that reside in a perimeter or DMZ network should be restricted to only the connectivity on the internal network necessary. Be careful to open only the minimum set of ports. Better yet, move the front-end servers to the internal network so that you do not have to open as many ports between the DMZ and the internal network. The most secure method is to configure a solution that allows front-end servers to be accessible only via a reverse proxy.
- If your firewall software has plug-ins available that will perform virus or content inspection on inbound HTTP, SMTP, or other traffic, you should consider using these as an additional layer of defense.



Beware of Cisco's PIX MailGuard, which doesn't work all that well with Exchange. For more information, see the Microsoft article "XCON: Cannot Send or Receive E-mail Messages Behind a Cisco PIX Firewall."

Avoid Directly Publishing Exchange Server Resources

Though this topic might seem more inline with a good security strategy overall than for virus and worm protection, avoiding directly publishing Exchange Server resources is still relevant. The Nimda and Code Red worms managed to spread by connecting to unpatched NT and Win2K servers running IIS. In many organizations, this server was sitting in the DMZ. In cases with worms such as the Blaster worm, a single workstation or server could become infected on the internal network; in many cases, such occurred because someone plugged in an infected notebook to an internal network. Once these servers were infected, these worms gained access to the internal network and began to spread to the servers and the desktop computers on the internal network. Once one computer on the internal network is infected, you will see almost an exponential infection rate internally!

You can allow your remote users to use Exchange features such as OWA, POP3, or IMAP4 from the Internet without ever directly accessing a Windows IIS server. To do so, you employ some type of reverse proxy solution, which is the type of solution that Microsoft recommends as the most secure method of publishing OWA for Internet users.

Reverse proxy technology acts in the opposite way that a regular proxy server behaves. A regular proxy server (forward proxy) receives requests from a client (usually HTTP), then forwards those request on to the destination server. A reverse proxy examines inbound IP requests (HTTP, POP3, IMAP4, and so on), often performs some type of content inspection or URL inspection on the request, repackages the request, and directs it on to the intended Web server. The only host that has direct access to the back-end servers is the reverse proxy server.

The most secure method of implementing a reverse proxy solution is to put all of your Exchange servers (front-end and back-end) on the internal network and to put the reverse proxy solution in the perimeter network. Figure 3.1 shows an example implementation of a reverse proxy solution for OWA clients.

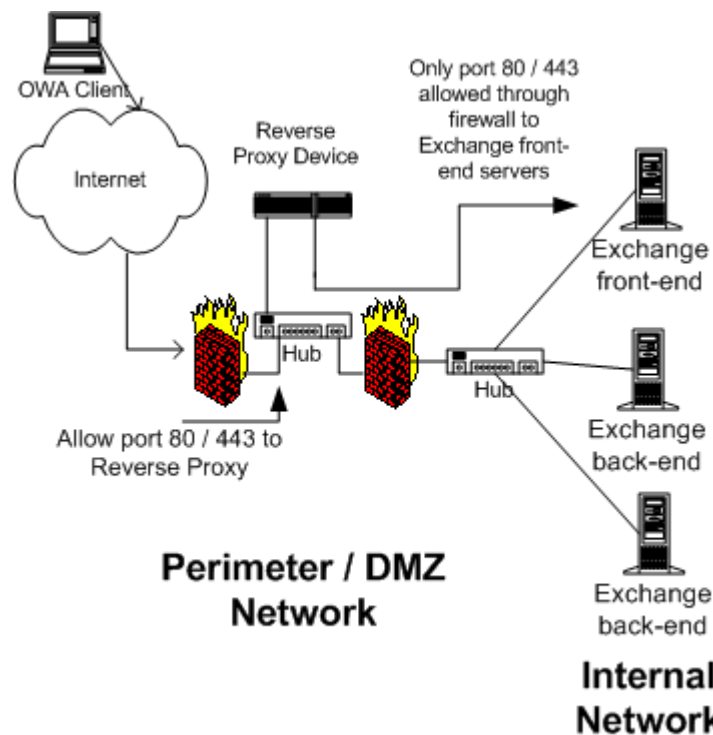


Figure 3.1: Example OWA solution using a reverse proxy device.

Employing a Multi-Layered Virus Protection Strategy

No single antivirus scanning engine catches 100 percent of viruses. Occasionally, even the most accurate scanning engine allows a virus or worm to sneak by. Viruses can sneak in to your organizations from many different directions. For these reasons, I recommend that you implement a multi-tiered approach to scanning for viruses. In a multi-tiered approach, you implement at least one additional method of scanning on your network. I recommend the three-layer approach that Figure 3.2 illustrates.

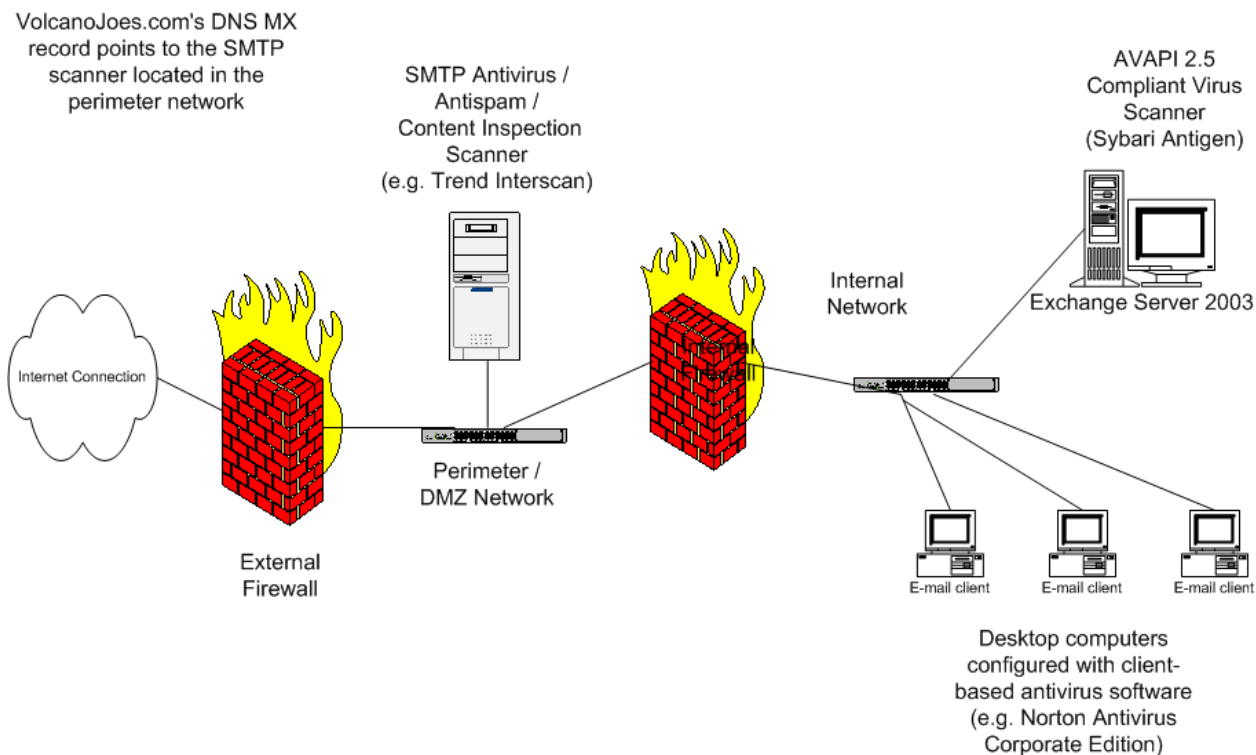


Figure 3.2: Multi-layer email virus scanning solution.

Figure 3.2's network has three *separate* antivirus systems at work, each from a different vendor and each using a different scanning engine and set of signatures. Inbound mail is delivered to an SMTP scanning system located in the DMZ; no inbound SMTP traffic ever goes directly to the Exchange server. The Exchange server runs a separate scanning engine specifically designed to scan the Exchange server information store. Finally, each desktop client on the network has client-based scanning software. This setup gives you better protection by ensuring that a single vendor's failure to update their signatures in time to catch a new threat won't leave your network entirely unsupported.

Differing Scanning Policies

You might want to consider implementing different scanning and attachment policies for email that arrives from the Internet as opposed to mail that is to be delivered internally. Implementing a multi-tier approach such as the one shown in Figure 3.2 allows you to do so.

For example, on the SMTP scanning system that handles mail that is arriving from the Internet, you could ban all types of attachments except for office automation applications (Word, Excel, PowerPoint, and so on). However, on the Exchange server, you could allow files that might be more dangerous, such as ZIP files, compiled Help (.CHM) files, and the like.



Some companies block ZIP attachments using the logic that the file might contain hostile content. Although this is certainly true, ZIP files remain a valid way to get other content into an organization and usually should not be blocked. Most virus scanning software has the ability to open and scan ZIP attachments anyway. A better policy towards ZIP files is to quarantine them when they cannot be scanned or if they contain an infected file.

Updating the Software

One of the common reasons that the most invasive worms have spread so quickly is that they have taken advantage of vulnerabilities in the Windows OS, IIS, or the Web browser. Keeping the OS and IIS up to date and properly patched is critical. One of the most important and difficult decisions that an administrator must make is which patches to apply and how quickly.

A couple of security fixes and updates are usually released each week for the various Windows platforms. The Nimda and Code Red worms exploited a weakness in IIS; unfortunately, a patch for this weakness had been released nearly 6 months before the authors of these worms unleashed their wrath on email users and administrators. The Blaster worm exploited a weakness in the entire NT family (NT, Win2K, Windows XP, and WS2K3) for which a patch had been offered a few weeks earlier.

So how do you make good decisions about deploying these updates? After all, most of these updates require reboots and, consequently, at least a few minutes of downtime for each server. Keep an eye on the patches that are released and the vulnerabilities that exist:

- Subscribe to Microsoft's Security Notification service at <http://www.microsoft.com/security>.
- Regularly visit a third-party notification site such as Carnegie Mellon's CERT Coordination Center at <http://www.cert.org> and the SANS Institute at <http://www.sans.org>.
- If a security threat makes it to national or local news, investigate it immediately. Major media organizations tend to publish sensationalistic, technically inaccurate reports, but sometimes these reports are the first notice you get of a new vulnerability.
- Check multiple sources for information—don't depend on any single source for your news about vulnerabilities.

When you receive notification of new updates and fixes, evaluate each one to determine whether the update or fix applies to your environment. Fixes related to Windows Media Player or DirectX are non-critical for servers and can be deferred until the next scheduled reboot. Fixes relating to Internet Explorer (IE) can be deferred to your next scheduled downtime; after all, you should not be surfing the Internet from the console of your Exchange servers. Patches that enhance functionality can be deferred indefinitely.

However, fixes that affect IIS, core Windows functionality, or RPCs (including fixes that affect Exchange Server security problems) should be applied at the next available opportunity. Even if your next scheduled downtime is not for a couple of weeks, you should take the time to reboot each server after you've informed users that you're going to do so—when users know about downtime, it is far less convenient than a virus outbreak that shuts out users from their email for hours!

☞ Don't sacrifice reliability or security for better availability. An hour of unscheduled downtime is far more inconvenient than even several hours of scheduled downtime.

Applying Exchange Restrictions

Most experienced Exchange administrators are firm believers in placing as many limits and restrictions on the user community as possible. These limits should be as unobtrusive as possible and be implemented while keeping in mind that users must be able to continue to do their jobs.

Dummy Entries Offer No Protection to Your Address Book

Tips and hints for protecting your organization from email replicating viruses have emerged since the first such virus hit in 1999. The Melissa virus sent itself to the first 50 entries in the Exchange Global Address List (GAL), so many administrators created 50 “dummy” mailboxes that would be sorted to top of the GAL. Later email-based viruses merely picked names at random or didn't even use the GAL—worms such as SoBig, BugBear, Nimda, and Blaster don't even need Exchange to replicate. These worms can scan emails in your Inbox, files on your hard drive, or local pages in your Web browser cache looking for email address to send messages to or from which to claim that the message was sent.

Another incorrect tip instructs users and administrators to put an entry into their address books that starts with !0000, 0000, AAAAAA, or other sequences of letters, numbers, and characters. These tactics do not work against modern viruses, so don't bother.

Protecting Mail-Enabled Groups

Mail-enabled groups (distribution lists) are the easiest way for a virus or worm to send itself to a large number of users with a single recipient. In Exchange 2000, Microsoft implemented a couple of features for protecting mail-enabled groups, and mail-enabled group protection was again extended in Exchange Server 2003. Figure 3.3 shows the Exchange general property page for a mail-enabled group in AD 2003 and Exchange 2003.

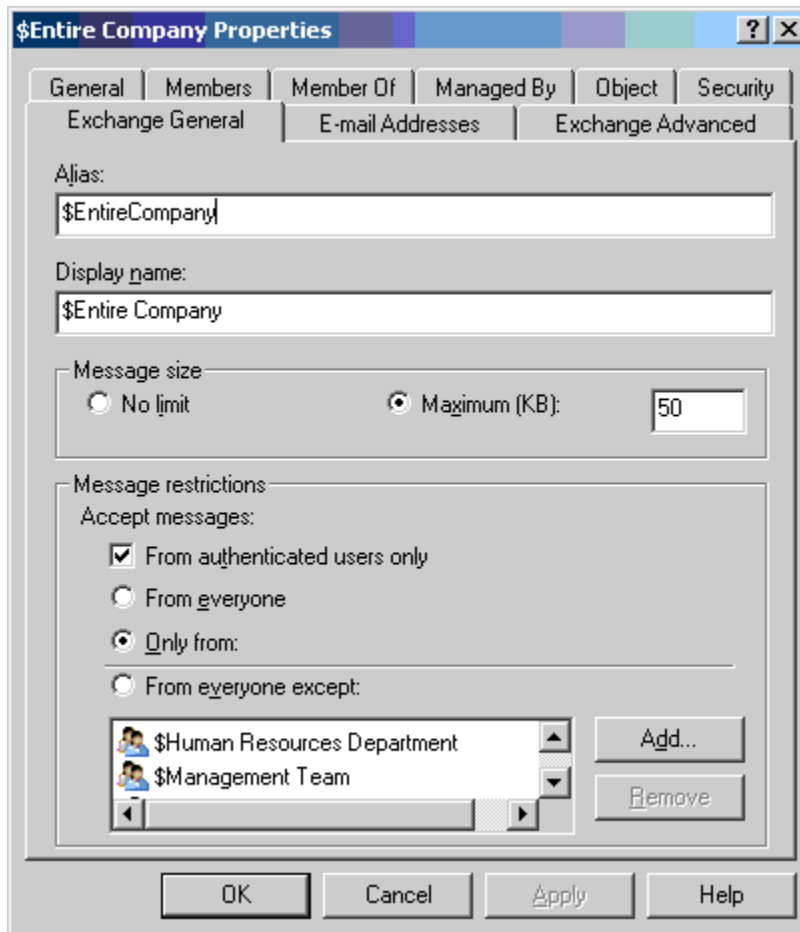


Figure 3.3: Exchange general properties for a mail-enabled group.

Some of the features of a mail-enabled group that can help prevent mail abuse or hostile content from spreading include:

- Configure the message size limit to prevent large messages from blasting your distribution lists. I advise setting this limit to a reasonably small amount, especially if the group membership is very large.
- Set a group's restrictions so that only authenticated users can send mail to the group (this option is available in Exchange 2003). Doing so will prevent an anonymous user from connecting to the server via SMTP and sending messages to a mail-enabled group.
- Restrict a group so that it will only accept messages from specific users or groups. Doing so is extremely useful for groups that have large memberships and company-wide groups.

Restricting Message Size and Recipient Count

Exchange 2000 and Exchange 2003 have another set of features that help reduce hostile email content from spreading too quickly and can help prevent users from abusing their email privileges—the Message Delivery Properties, which Figure 3.4 shows. You can find these settings in Exchange System Manager in the Global Settings container.

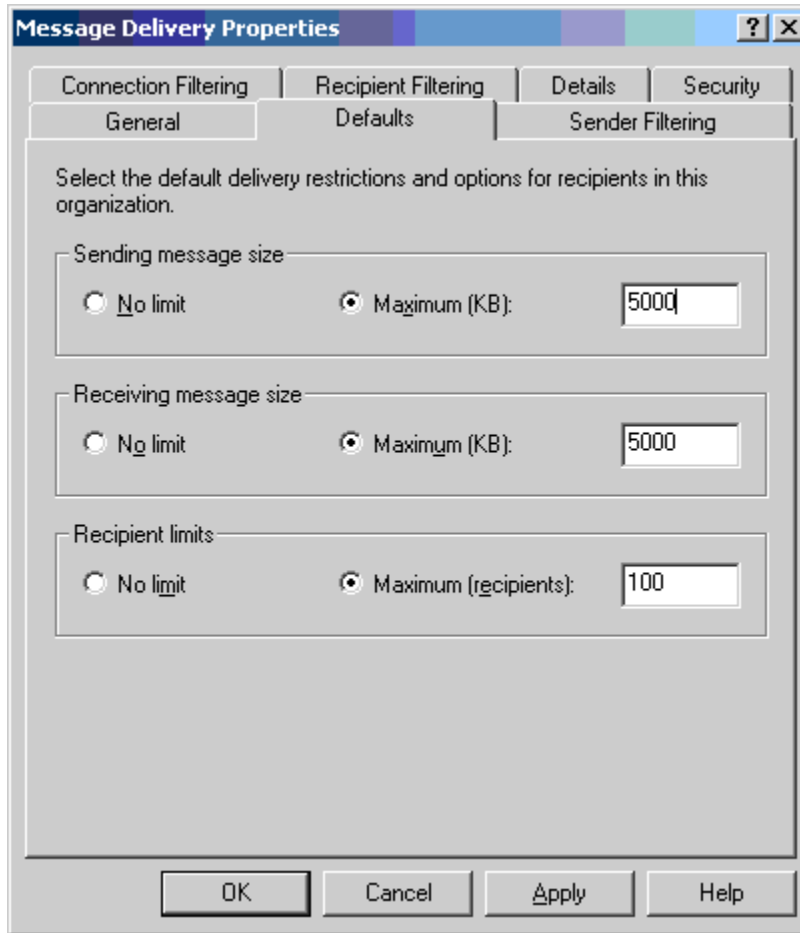


Figure 3.4: Global message delivery restrictions.

The restrictions that Figure 3.4 show allow the administrator to globally restrict the maximum incoming and outgoing message sizes as well as the maximum number of recipients per message. Each of these settings can be overridden on a user-by-user basis for VIPs or users that require larger messages or more recipients per message. The maximum number of recipients per messages is the maximum number of recipients in the To, Cc, and Bcc lines; the total number of recipients in a group is included in this count. If a restriction of 100 recipients per messages is enforced, then a virus or worm will not be able to send to a distribution list with 101 recipients. Of course, this offers no protection from worms that send to a single recipient at a time.

Limiting Mailbox Storage

Mailbox size does not necessarily relate directly to viruses spreading through an organization, but if your organization is hit by a worm or virus that is quickly spreading, it is possible that mailboxes may become extremely large. Excessive mailbox growth *will* lead to a server shutting down due to a lack of disk space.

To help thwart such attacks, place limits on all mailbox stores. When calculating these limits, be generous and allow users enough space to properly do their jobs. These limits should be published so that the users are aware of them. Even if you don't intend for your users to be subject to the limits, setting a 1GB per mailbox limit can help protect your server from running out of space during a virus outbreak. A Prohibit Send and Receive limit should take into consideration the maximum number of mailboxes on the mailbox store and the total amount of mail storage on the server.

Figure 3.5 shows the Limits tab of a mailbox store, which you can also set through Exchange Mailbox Store Policies.

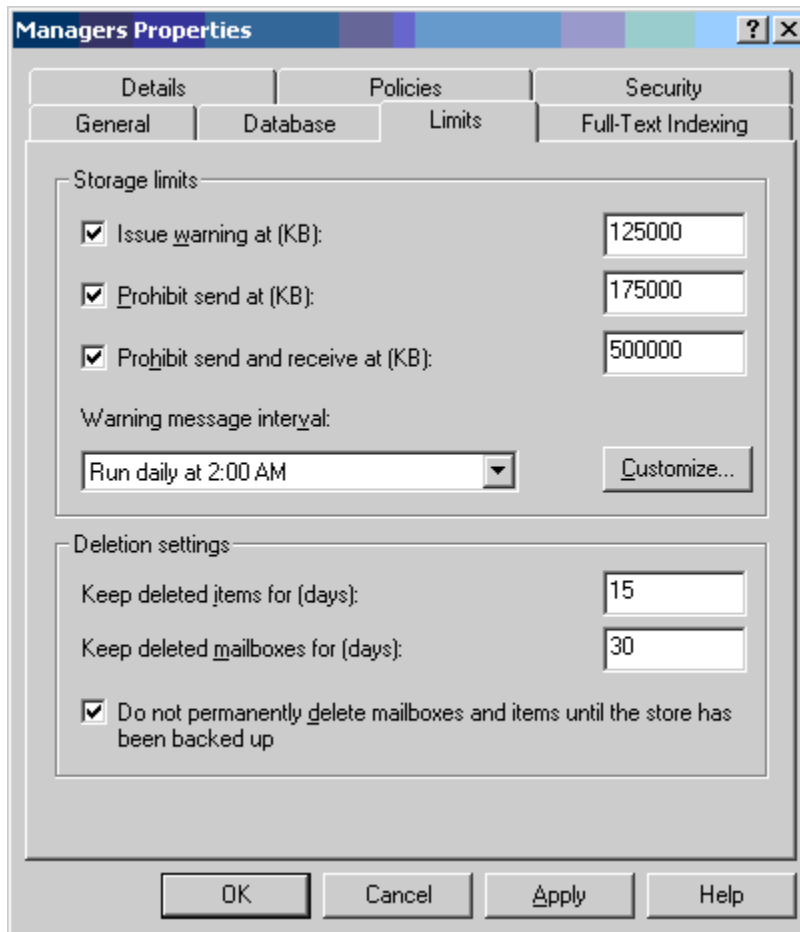


Figure 3.5: Mailbox store storage restrictions.

Scanning and Blocking File Attachments


Develop a list of forbidden file attachment types for your organization that is published to your users. In the early days of email replicating viruses, IT departments were continually adjusting this list to include the seemingly ever-increasing list of possibly dangerous file types. Blocking some of these files is politically difficult, as many users need access to certain file types.

☞ Some server-based antivirus software gives you the choice of scanning all file attachments or only specific attachment types. Scan all files.

Which file types should be included in this forbidden list of attachment types? The answer is hotly debated. Table 3.1 contains the standard list of attachment types that I ask my clients to block at their perimeter network scanner or on their Exchange server. I think this list still allows maximum functionality while protecting against most threats.

Attachment Extension	Description
asp	Active Server Pages scripts
bat	DOS batch files
chm	Compiled HTML Help files
cmd	Windows Command scripts
com	DOS program files
eml	Embedded email files
exe	Executables
htm/html	HTML files
Js	JavaScript files
pif	DOS/Windows 3.1 Program Information Files
pl	Perl script files
reg	Windows registry script files
scr	Screen saver files
shs	Shell Scrap files
vb	Visual Basic files
vbs	Visual Basic Script files
wsc	Windows Script Component
wsh	Windows Script Host scripts

Table 3.1: Significant file attachment types that should be blocked.

 In Chapter 2, we explored the file types that Microsoft considers Level-1 attachments (which we also discussed in Chapter 2).

If your antivirus software offers the functionality, configure the software to automatically send a message to the sender of a message that includes a file attachment that is not allowed. Instruct the sender about how to get the attachment to the user.

The Exchange Antivirus API

With Exchange 5.5 SP2 and later, the only way to scan for a virus in the Exchange information store is to use MAPI programming calls. Essentially, Exchange-aware antivirus software had to log on to each mailbox, then wait for a MAPI notification that a new message had arrived. Once the messages arrived, the antivirus scanning software could then scan the message, assuming the software was not busy scanning other messages. If a virus was detected, the message was opened, the virus was removed, and the message was re-saved. However, because this process took place mailbox-by-mailbox, messages that were sent to 50 people had to be scanned 50 times. Organizations often suffered as a result of the shortcomings of this method.

Another limitation of MAPI-based scanning is that MAPI can only send 64 new mail notifications to the antivirus software at any one time. Thus, if a virus outbreak occurs and the server is receiving hundreds of messages in a short period of time, many messages will hit the mailbox and the antivirus software will not have a chance to scan the message before a quick user opens the message and spreads the virus.

Microsoft had a painful demonstration of the shortcomings of the MAPI-scanning approach with the outbreaks of viruses such as Melissa in 1999. The Exchange team developed an API that would allow third-party vendors to hook messages as they arrived and before they are placed in the user's mailbox. This API is known as the Antivirus API or AVAPI 1.0; you will also see AVAPI referred to as VAPI.

As usual, new products generate more feature requests. As vendors developed solutions using this API, customers and vendors realized the limitations of this first API. AVAPI 1.0 only allowed for scanning of the attachments in the message and not the message body itself; embedded viruses such as BubbleBoy could still slip through. In addition, the message had to be delivered to the information store; AVAPI had no method of inspecting the message in the SMTP or MTA queues.

In 2001, Microsoft released a new virus scanning API for use with Exchange 2000 SP1 and later called AVAPI 2.0. This version addressed many of the issues surrounding the capabilities of AVAPI 1.0-based products by allowing the scanner access to recipient information and the ability scan for embedded viruses. AVAPI 2.0 also allowed scanning of not only the EDB database file but also the STM database file.

With the release of Exchange 2003, Microsoft has published yet another antivirus API, AVAPI 2.5, which further extends the abilities of earlier versions. This version allows the scanning software to scan messages not only in the information store but also in the SMTP queues on SMTP bridgehead servers or SMTP front-end servers.

Learning More About the Exchange AVAPI

For more information about the Exchange AVAPI and virus scanning on Exchange servers, see the Microsoft articles "Overview of Exchange Server 2003 and Antivirus Software" and "XADM: Understanding Virus Scanning API 2.0 in Exchange 2000 Server SP1."

Designing a Server-Based Protection Scheme

All Exchange servers need an Exchange-aware antivirus software package installed and running. Exchange server-based antivirus scanning software used to be considered a luxury, but now it is essential for the security and reliability of your messaging system. Pricing varies from vendor to vendor and can depend on a variety of things such as the number of users, software subscription service (free updates), add-on packages such as anti-spam/junk mail scanning, and the number of scanning engines, if applicable. Retail pricing for Exchange AVAPI products range from less than \$25US per seat to \$35US per seat when purchasing 100 seats. You should include these estimated costs in any budget you put together for Exchange services. The following list highlights characteristics and features that you should consider when evaluating Exchange AVAPI software:

- Automatic updates of antivirus software
- Ability to configure blocked attachment list
- Customizable notifications
- Multiple scanning engines
- The ability to scan zip and other compressed files
- Anti-spam features or plug-in that enables anti-spam features
- Ability to kick off a manual scan of existing messages
- Quarantine where files that could not be scanned or viruses can be stored in case they need to be examined or released later
- Remote management of the software
- Manually initiated scans of the existing data in the mailbox stores
- For Exchange 2000, the product should support AVAPI 2.0, and for Exchange 2003, the product should support AVAPI 2.5
- Ability to handle encrypted and password-protected files
- Adequate notification of the appropriate people in your organization when viruses are detected
- Comprehensive reporting features
- Outbreak monitoring features—will the product proactively notify you if it detects a potential outbreak and do the configurable notification features run scripts?

Multiple Scanning Engines

No single antivirus scanning engine and set of signatures is 100 percent effective all the time. Some Exchange AVAPI-based scanners have the capability to use more than one scanning engine; Sybari's Antigen, for example, allows as many as five different scanning engines to be enabled. This functionality greatly increases the likelihood that all viruses will be detected and removed.

Exchange-Aware Antivirus Vendors

There are many Exchange-aware antivirus products available. To get a better idea of their strengths and weaknesses, search the Exchange Usenet newsgroups and mailing lists for discussions about virus protection and recommended products. The following list provides vendor names and Web sites of products that were designed for Exchange:

- Sybari at <http://www.sybari.com>
- FRISK Software at <http://www.f-prot.com>
- GeCAD Software at <http://www.ravantivirus.com>
- Kaspersky Lab at <http://www.kaspersky.com>
- McAfee Security at <http://www.mcafeeb2b.com>
- Norman Data Defense Systems at <http://www.norman.com>
- Panda Software at <http://www.pandasoftware.com>
- Softwin at <http://www.bitdefender.com>
- Symantec at <http://www.symantec.com>
- Trend Micro at <http://www.trendmicro.com>

File-Based Virus Scanners

Many administrators implement a file-based antivirus scanning system on their Exchange servers. If you choose to do so, keep the following considerations in mind:

- The file-based scanner *must* be excluded from scanning any directories that contain Exchange data and transaction logs, including the following directories
 - \exchsrvr\mtadata
 - \exchsrvr\mdbdata
 - \exchsrvr\ServerName.log (message tracking log files)
 - \exchsrvr\mailroot
 - \exchsrvr\imcdata
 - \exchsrvr\srsdata
 - Wherever your SMTP (and X.400 MTA) queues are located
- If the Exchange Installable File System (ExIFS) drive (the M drive) is enabled, the file-based virus scanner must *never* scan this drive.
- File-based virus scanners are not a substitute for an Exchange AVAPI-based scanner.
- Test the interaction of the two A/V scanners before you put them into production.
- Be prepared to disable the file scanner if you are performing an upgrade, disaster recovery, or database maintenance operation.
- Don't be surprised if Microsoft Product Support Services asks you to completely remove the product (not just disable it) if you are having problems.

Monitoring Exchange Server Virus Protection

Most Exchange server antivirus products have reporting features that allow you to run reports on the number of viruses received, blocked attachments, and quarantined messages. I recommend running these reports between weekly and monthly intervals and keeping a record of past reports. These reports are useful and provide management with proof that the antivirus software is of value to the organization.

Most vendors provide canned reports and easy-to-read screens that give you the most recent virus statistics at a glance. Figure 3.6 shows the Sybari Antigen product's Incidents screen that details the most recent viruses detected and where they were detected. As with most products, this screen can be exported.

The screenshot shows the 'ANTIGEN 30-DAY EVALUATION' window. The 'INCIDENTS' section contains the following table:

Time	State	Name	Folder	Message	File
9/3/2003 10:58:18 PM	Removed	First Sto...	First Storage Group\Robichaux, Paul (S...	RE: Operations rep...	RequestedFile...
9/3/2003 10:58:14 PM	Removed	First Sto...	First Storage Group\Robichaux, Paul (S...	RE: Operations rep...	annualreport.z...
9/3/2003 10:57:32 PM	Removed	First Sto...	First Storage Group\Hoffmann, Aran (L...	RE: Operations rep...	annualreport.z...
9/3/2003 10:56:57 PM	Removed	First Sto...	First Storage Group\Supatanasakul, Su...	FW: New 9' boards	RequestedFile...
9/3/2003 10:55:12 PM	Removed	First Sto...	First Storage Group\McBee, Jim (HNL...	More information	annualreport.z...
9/3/2003 10:46:20 PM	Removed	First Sto...	First Storage Group\McBee, Jim (HNL...		RequestedFile...

The 'STATISTICS' section shows the following data:

	Internet		Realtime		Manual		MTA	
	Logical	Physical	Logical	Physical	Logical	Physical	Logical	Physical
Attachments Scanned	0	0	10	10	0	0	0	0
Attachments Detected	0	0	0	0	0	0	0	0
Attachments Cleaned	0	0	0	0	0	0	0	0
Attachments Removed	0	0	6	6	0	0	0	0
Messages Purged	0	0	0	0	0	0	0	0
Total Attachments Scanned	0	0	10	10	0	0	0	0
Total Attachments Detected	0	0	0	0	0	0	0	0
Total Attachments Cleaned	0	0	0	0	0	0	0	0
Total Attachments Removed	0	0	6	6	0	0	0	0
Total Messages Purged	0	0	0	0	0	0	0	0

Figure 3.6: The incidents report screen from Antigen.

Monitoring Windows Performance Counters

The Exchange AVAPI also includes additional performance monitor counters that are most useful when put into System Monitor's report view, as Figure 3.7 shows. This type of report is useful to provide evidence to management that although viruses aren't reaching users, viruses are still arriving in the mail system. These counters reset each time the store is stopped and restarted.

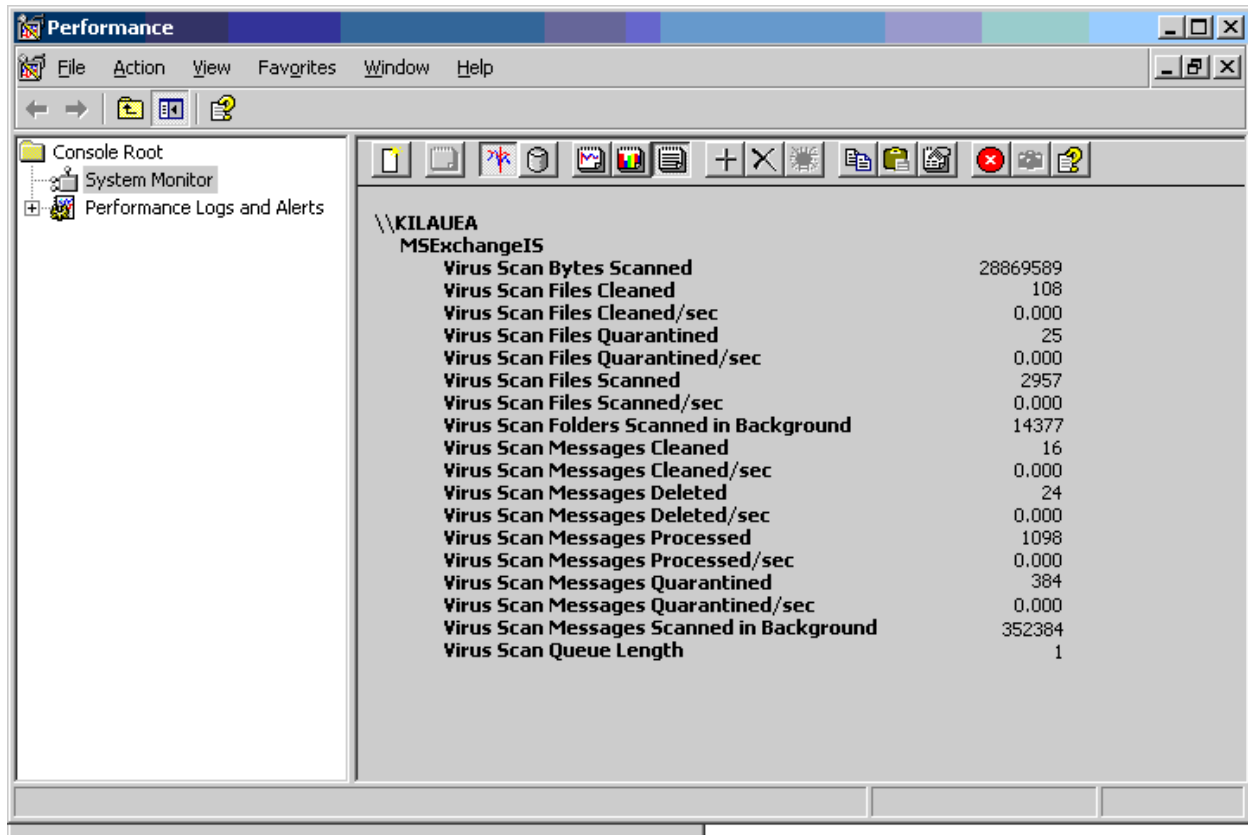



Figure 3.7: AVAPI performance monitor statistics.

Most of the statistics you can track from System Monitor are reasonably self-explanatory. Table 3.2 provides descriptions of these statistics; all of these counters are found under the MExchangeIS object.

Counter	Description
Bytes Scanned	Total size of all messages and attachments that the AVAPI has scanned for viruses
Files Cleaned	Total number of file attachments that had viruses but were successfully cleaned and released to the intended recipient
Files Cleaned/sec	Current scanning activity for file attachments
Files Quarantined	Total number of file attachments moved to quarantine
Files Quarantined/sec	Number of file attachments quarantined per second
Files Scanned	Total number of file attachments scanned
Files Scanned/sec	Number of file attachments scanned per second
Folders Scanned in Background	Total number of folders scanned during a background or scheduled antivirus scan
Messages Cleaned	Total number of messages that have been cleaned
Messages Cleaned/sec	Number of messages cleaned per second
Messages Deleted	Total number of messages deleted as the result of a rule such as a forbidden attachment rule (Exchange 2003 only)
Messages Deleted/sec	Number of messages deleted per second (Exchange 2003 only)
Messages Processed	Total number of messages that have been processed
Messages Processed/sec	Number of messages processed per second
Messages Quarantined	Total number of message that have been put in quarantine
Messages Quarantined/sec	Number of messages that are put in to the quarantine per second
Messages Scanned in Background	Number of messages scanned during a background scan of the information store; these scans can be kicked off manually or on a schedule
Queue Length	Current number of messages waiting to be scanned; if the queue always has more than one or two messages waiting, you have performance problems on your server

Table 3.2: AVAPI System Monitor counters.

 If you want to be notified of a potential outbreak of a known virus, you could set a threshold of viruses detected per second by using the Performance Logs and Alerts console and the Messages Cleaned/sec or Messages Quarantined/sec counters.

Examining the Windows Application Event Log

The Exchange AVAPI provides useful diagnostics logging capabilities for virus scanning and detection operations. The information you will receive from the event logs might not be as comprehensive as the information you will be able to access directly from the antivirus software's reporting tools, but the information in the Application event log is available to any event log monitoring tools.

To enable this logging level, you must turn on the Virus Scanning diagnostic logging for each Exchange Server. Figure 3.8 shows the Diagnostics Logging property page for an Exchange Server 2003 system called KILAUEA. Locate the Virus Scanning category under MExchangeIS, System, and set the value to a medium logging level.

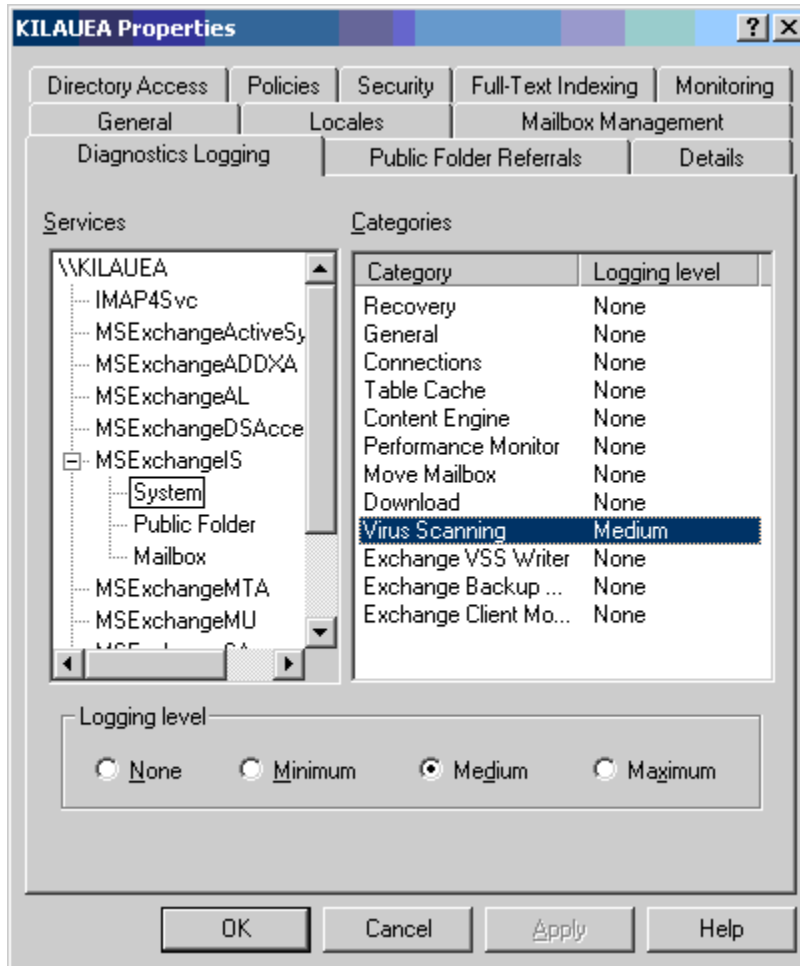


Figure 3.8: Enabling Virus Scanning diagnostics logging.

Once diagnostics logging is enabled, you will see events in the Application event log that report on virus detection. Table 3.3 shows some of the more common events you may see for AVAPI 2.0 and later scanning software. All of these events are from the source MExchangeIS and category Virus Scanning; however, some of the events in this table will only be available if you set the logging level to maximum. Not all AVAPI antivirus software packages will generate all of these error messages.

Event ID	Severity	Explanation
9572	Warning	A virus has been discovered. Message was cleaned successfully. Details can be found in the event description.
9565 and 9566	Error	An error occurred during the initialization of the virus scanner due to an invalid configuration parameter. Review the event details for more information.
9568	Error	An error occurred in one of the virus scanning components. Review the event details for more information.
9569 and 9570	Error	An error occurred when scanning a message. This error could be the result of the message being corrupted or it could be a problem with the virus scanning software. Confirm that you have the latest scanning engine and updates. This error might also result from the configuration of the antivirus software.
9571	Error	An error occurred while scanning a message and the scanning software is unloading. This error is fairly serious. Confirm software and scanning engine updates. Consult the event details for more information.
9573	Warning	A virus has been discovered. The message could not be cleaned so the message was quarantined. Details can be found in the event description.
9574	Informational	AVAPI virus scanning software has been loaded.
9575	Informational	AVAPI virus scanning software has been stopped.
9575	Informational	AVAPI virus scanning software has been restarted.
9578	Informational	Scanning software is starting a background or scheduled task to scan the database specified in the event details.
9580	Informational	If diagnostics logging is turned off, you will see this message when the scanning software is started. This message simply tells you that the scanning software has started but that you will see no additional diagnostics logging information.
9581	Error	Virus scanner failed to start during initialization routine, see event description for more information.

Table 3.3: Events generated in the Application event log by AVAPI-aware antivirus products.

Errors indicating that the antivirus software is not working properly or failing to scan messages can be serious; especially if the scanning engine has stopped altogether, which means that a virus could sneak through. If you are seeing these problems, try the following:

- Restart the antivirus software
- Update the scanning engine
- Update the antivirus signatures
- Reboot the server
- Determine whether the problem is with one specific message or with all messages




Though not as common as it once was, virus signatures can be corrupted when they are downloaded from the vendor. A corrupted signature database will cause the virus scanner software to fail.

Containing a Virus Outbreak

You are driving to work and the morning news mentions a new and dangerous virus. You wonder if the story is hype or if the virus is real threat. You arrive at work and find your boss and 30 new voicemail messages waiting for you. Your Inbox has 2500 copies of this new virus. A virus has managed to get into your organization through mechanism you had not previously thought of, and your organization is too large to go from user to user advising the users on what to do next.

Your first and most immediate task must be to stop further exposure. First, stop all SMTP services on your Exchange Server, perimeter SMTP gateways, and stop the Exchange MTAs.

 By stopping SMTP, you have stopped *all* mail flow. Evaluate whether this step is right for the situation—especially if you need to send a message to the user community about what to do next.

Once SMTP and the MTAs are stopped, it is time to assess the situation quickly. By stopping SMTP and the MTAs, you have stopped the virus from spreading, but you have also stopped email from flowing.

Next, before you even get back in touch with your boss, force virus signature updates from the vendor's update site. If the virus has already been announced on the news, your vendor should know about it and have something on the Web site. If the virus is only a few hours old, signatures may not be available for a few more hours, so you have more work to do. In this case, consider the following factors:

- How bad is the outbreak? Is it all servers and all locations?
- How long can you remain offline (without email flowing)?
- Does this situation warrant a shutdown?
- Are the WAN links clogged?
- Do the SMTP queue and MTADData directories have many copies of this message?
- Is there something recognizable or unique about the virus such as a subject, attachment name, or attachment type?

If you are in a larger organization, you are probably going to have to make a recommendation to your boss about what to do next. A complete shutdown is not a very popular option. Upper management and your users do not appreciate or understand how the virus managed to hinder email flow, just the fact that email is down. The pressure is now on and you need to get mail flowing again as soon as possible.

Locking Users Out of the Exchange Server

Locking your users out of the Exchange server is not going to be advisable under most circumstances, but it might be the only way to subdue the virus or worm that is taking advantage of your disk storage and bandwidth. For POP3, IMAP4, HTTP, and NNTP clients, doing so is a simple matter of disabling the appropriate services:


- Microsoft Exchange POP3
- Microsoft Exchange IMAP4
- NNTP
- World Wide Web Publishing Service

For MAPI clients, it is a little different; you could simply disable the information store service, but doing so will prevent you from extracting any messages using ExMerge. On Exchange 2000 SP1 and later, you can create a registry key that will disable certain versions of MAPI clients; in this case, you are interested in disabling all versions except the Exchange server components themselves. Open the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeIS\ParametersSystem` key, and create a value of type `REG_SZ` called `Disable MAPI Clients`. In this value, enter in the data field

```
-6.0.0, 7.0.0-
```

Then stop and restart the Microsoft Exchange Information Store service. Doing so prohibits any version of the MAPI client except 6.0.0 through 7.0.0 from connecting to the information store; this will be the Exchange components (and ExMerge, if running from the Exchange server's console).

 For more information, see the Microsoft article "XADM: Feature to Disable MAPI Clients."

 Once you are ready to allow users back on to the servers on which you have disabled the MAPI clients, delete the `Disable MAPI Clients` registry key and restart the information store.

Cleaning Up Queue Directories

If you examine the `\Exchsrvr\Mailroot\vs1\1\queue` directory, you will find all the inbound messages that had not been processed when you shut down the SMTP service. It would be easy to delete every file in this directory, but you don't know viruses from valid messages merely by looking at the file names. If you have truly been in the middle of a major outbreak, the Advanced Queuing Engine will probably have been overwhelmed and you might have thousands (or tens of thousands) of files in this directory. The same holds true for the `\Exchsrvr\MTAData` directory, if you are using the Exchange MTA.

Figure 3.9 shows the queue directory of an Exchange 2003 SMTP virtual server; this directory contains a few hundred messages that were the result of a minor virus outbreak. However, not all of these messages are viruses, so I'm going to use Windows' built-in `FINDSTR` to remove only the viruses.

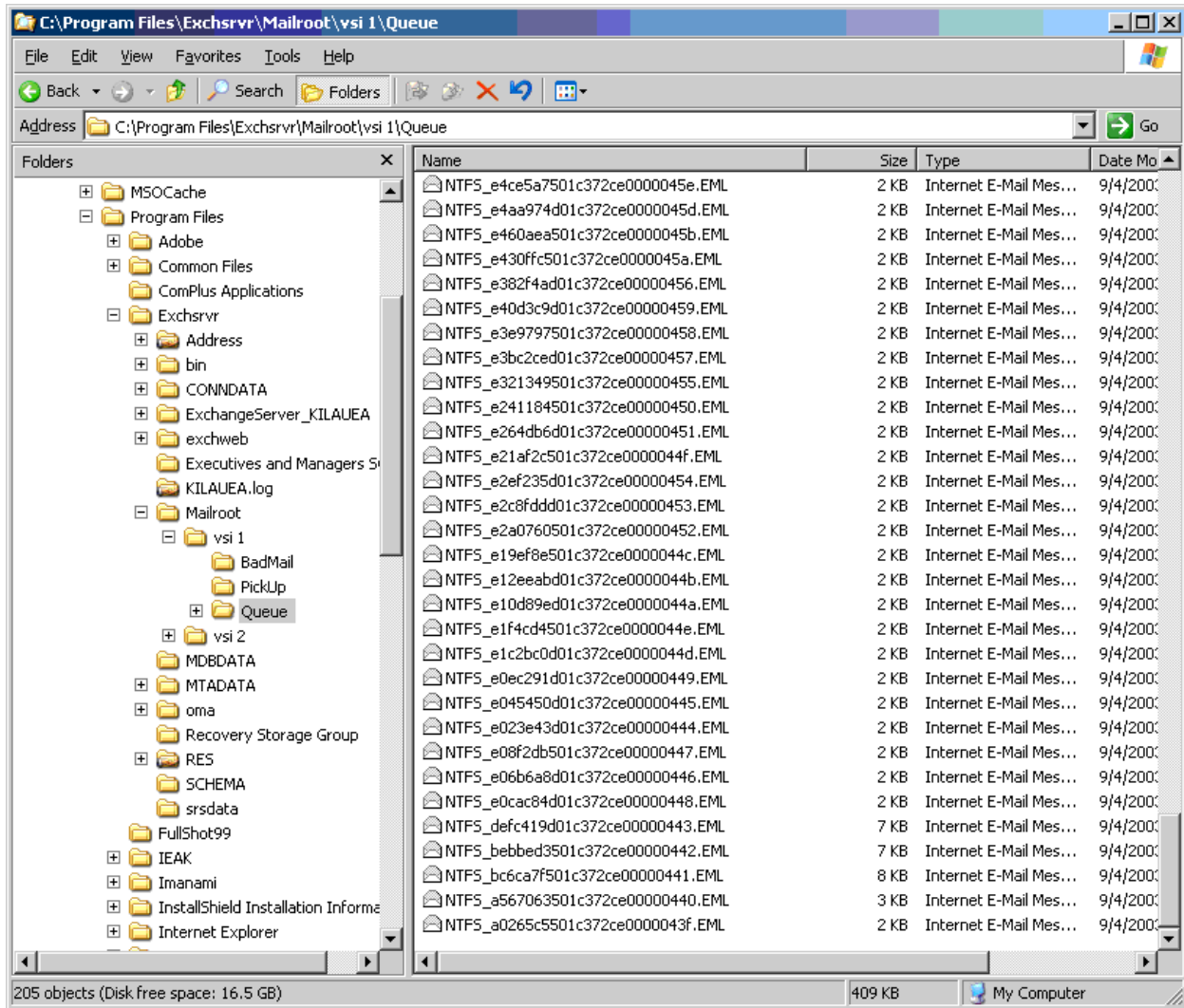


Figure 3.9: Virus explosion in the queue directory!


The first task is to find something unique about the virus; hopefully so unique that it will not occur in normal messages in the queue directory. In this example, I'm going to search for the Love Bug virus; Love Bug included in the subject line ILOVEYOU. Thus, the simple command

```
Findstr /c:ILOVEYOU c:\program files\exchsrvr\mailroot\vs1
1\queue\*.eml
```


will search for files containing the target string and list those that it finds. At that point, you can inspect the files to confirm that they're infected, and move them to another directory or remove them altogether. (You can also use Windows' search command to search the files; this method is slower than FINDSTR, but the search tool shows all the results at once so you can simply hit Ctrl+A, followed by Shift+Del, to remove all the suspect files.)

Cleaning the MTADData directory

A few years ago, Microsoft wrote a utility called `findbin.exe` for the specific purpose of finding viruses in queue directories. It was released by Microsoft's support services initially, then later, when the Love Bug virus hit, it was included in the ILOVEYOU.ZIP cleanup kit.

 You can find a link to `findbin.exe` at in <http://www.somorita.com/webcasts>.

`Findbin.exe` is an extremely simple program. You supply it with a hexadecimal string of data that represents something unique about a virus along with a list of files to look through. If `findbin` finds that string in any of the specified files, it will move those files to a directory specified in the command. Using the ILOVEYOU example again, I'll walk through the process of cleaning the MTADData directory with `findbin`. Because the MTA queue files are in binary format, I need the hexadecimal equivalent of my unique string; thus, I need to convert ILOVEYOU to hexadecimal, which is 494C4F5645594F.

 Don't convert to hexadecimal often? Visit <http://www.asciitable.com>.

If the MTA is not already stopped, you will need to stop it before you can clean up the messages in this directory, so make sure the Microsoft Exchange MTA Stacks service is stopped. Next, I'm going to put `findbin.exe` into the path on the Exchange server (probably the `\WINNT` directory), and create a subdirectory in the queue folder called `VirusMsg` into which `findbin.exe` will move any viruses messages it finds. I'm going to make my current directory the MTADData directory so that I don't have to type long paths in the command-line options, then, I will type

```
FINDBIN 494C4F5645594F DB* .DAT VIRUSMSG
```

Doing so should move all the infected MTA data files into the `VirusMsg` directory. Once this is done and before I restart the MTA, I will run the `MTACHECK` program to make sure that the MTA database is in good shape.

Getting Rid of the Virus from the Stores

Regardless of how many times you tell people to delete a message that has a virus, a few of your users are going to open the message anyway. Part of an effective removal strategy is to make sure that the virus is completely eradicated from the information store. Theoretically, once your virus signatures are finally updated, they will catch any viruses that anyone tries to open, but it is a good backup to at least attempt to remove the virus from users' mailboxes. To do so, you can use the `ExMerge` utility.

 You can download `ExMerge` from <http://www.microsoft.com/exchange/tools/2003.asp>, then place it in the `\Exchsrvr\bin` directory.

The first thing you will need is a user account that has permissions to open all the users' mailboxes. I won't explore the political or security dangers of creating this user, just keep in mind that this user must be protected from unauthorized use because it has the permissions to access anyone's email. This user cannot be a member of the Domain Admins or Enterprise Admins groups because these groups are automatically blocked from being able to open users' mailboxes.

First, create a global security group, for example, Exchange Demi-God Admins, then create a user—ExchSuperAdmin—that has a strong password. This user should be a member of the global security group because you will assign permission to the group rather to an individual user.

Open Exchange System Manager while logged on as an Administrator that has Full Exchange Admin permissions, and display the properties of the Organization or an Administrative group. If the Security property page is not visible in ESM, you will need to enable it in the registry by opening the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Exchange\ExAdmin registry key. In this key, create a REG_DWORD value called ShowSecurityPage, and set it to 1. When you close and reopen Exchange System Manager, the Security property page should be visible. On the Security property page, add the Exchange Demi-God Admins group and give it Full Control (as Figure 3.10 shows). If you use the Delegation Wizard, the Receive As and Send As permissions are automatically denied and the user account you're creating will not be able to open other users' mailboxes.

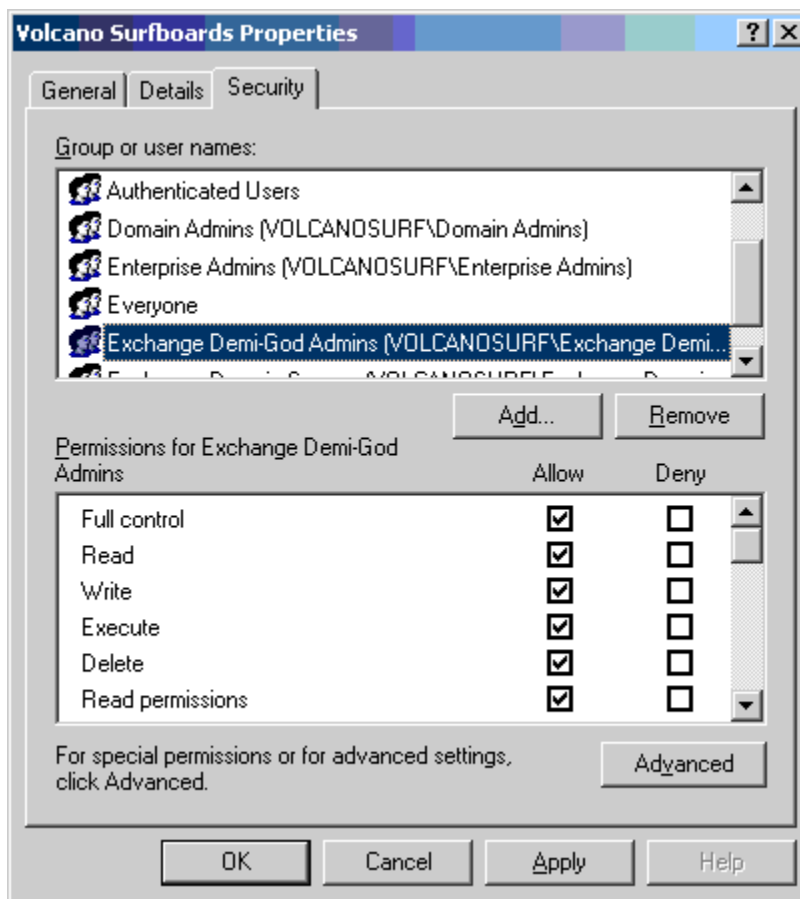


Figure 3.10: Granting a group Full Control permissions.

If your Exchange server is running on a domain controller, you should also make your ExchSuperAdmin user a member of an operators group such as Server Operators; otherwise it will not be able to log on to the console of the Exchange server. Once your user has sufficient permissions to access everyone's mail messages, it is time to use the ExMerge utility. Before you start, ensure that you have isolated a unique characteristic about the virus-infected messages, such as a subject or attachment name.

I use the archive feature to archive mail that meets a certain criteria to PST files. That way, if I accidentally removed an important message, I have a copy of the PST file. Also, I recommend performing the cleaning with ExMerge in two steps. First, look through only the Inbox, Outbox, Deleted Items, and Sent Items folders to help ensure speed when removing messages. I perform a second pass later, once the server is back online and users are working, that extracts the virus from any other folder to which it might have been moved. The following steps walk you through the process of extracting virus-infected messages that have a subject of ILOVEYOU from the Inbox, Outbox, Sent Items, and Deleted Items folders:

8. Run the ExMerge Wizard, and click Next.
9. Select the Extract or Import (two-step procedure) radio button, and click Next.
10. Select the Step 1: Extract data from an Exchange Server Mailbox radio button, and click Next.
11. In the Source Server dialog box, enter the name of the Exchange server, the name of domain controller, and click Options.
12. On the Import Procedure property page, click the Archive Data to Target Store radio button.
13. On the Folders tab, select the *Process only these folders* radio button, then click Modify. Add the folders you want to process to this list (see Figure 3.11).

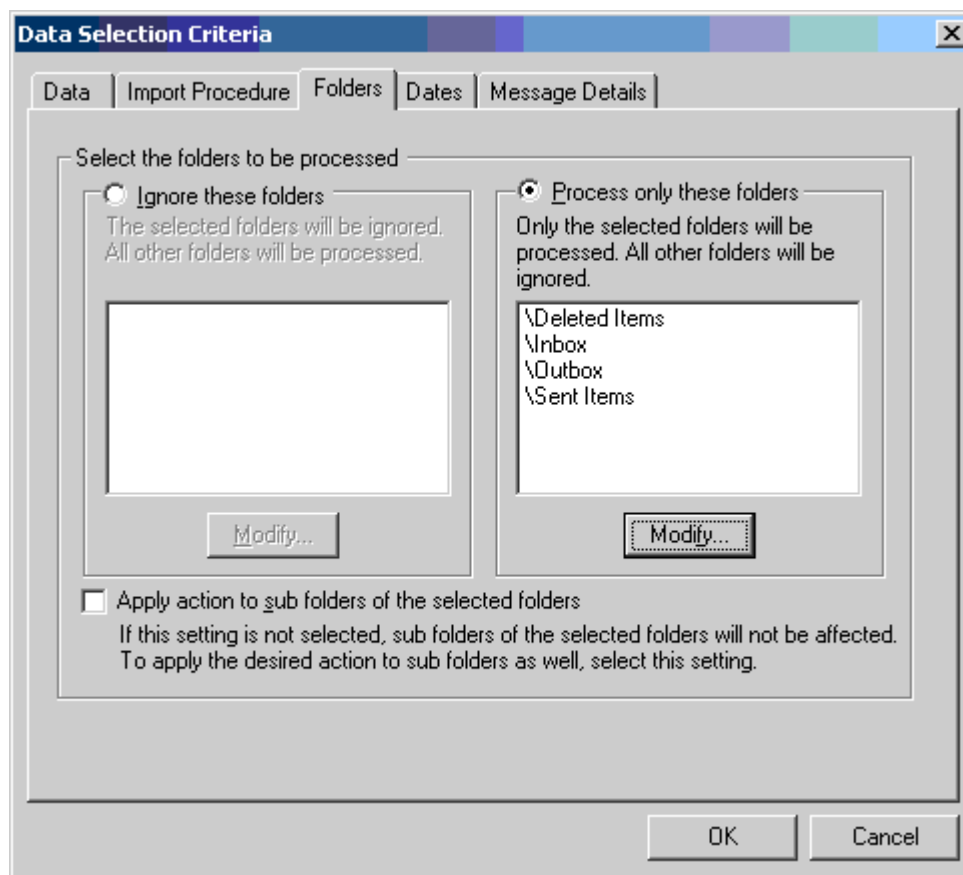


Figure 3.11: ExMerge's Folder's tab lets you specify which folders you will process.

14. Optionally, you can specify a date range on the Dates tab if the virus has just hit. Doing so will reduce the likelihood that you will accidentally remove valid messages.
15. On the Message Details tab (see Figure 3.12), specify the unique characteristic of the message that you want to remove from the folders. If you leave this blank, ExMerge will remove *all* messages from the specified folders. In this example, I have added ILOVEYOU to the list of selected subjects.

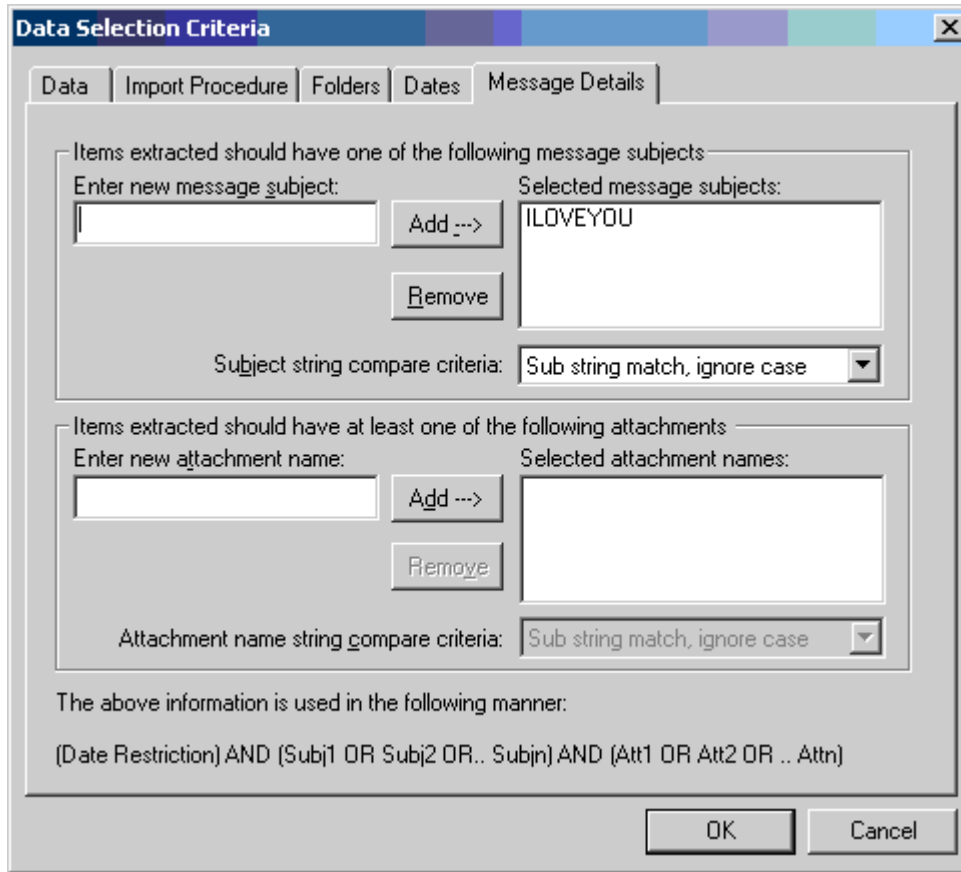


Figure 3.12: Specifying a message subject or attachment name.

16. When finished with the Options, click OK and Next.
17. In the Database Selection dialog box, select the mailbox stores you want to process, then click Next.
18. On the Mailbox Selection listing (see Figure 3.13), select the mailboxes you want to process. If you are extracting a virus, you will probably want to click Select All. When finished selecting mailboxes, click Next.

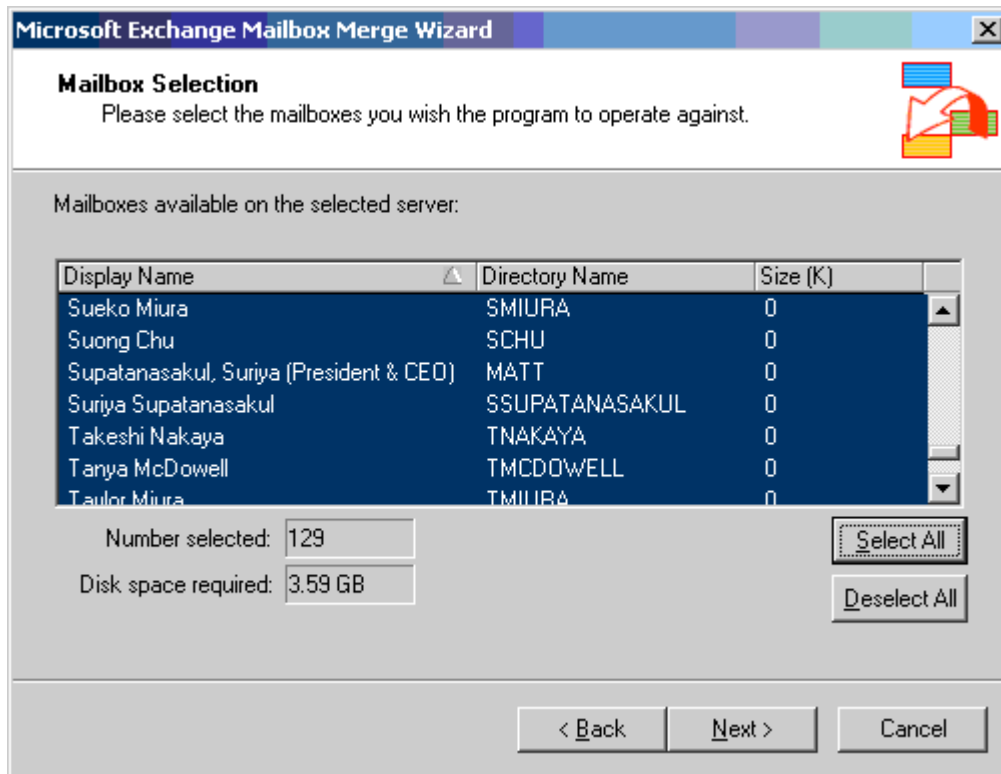



Figure 3.13: Selecting mailboxes to process.

19. Specify the default locale; if you don't know what this means, then leave the selection as English (US), then click Next.
20. Specify a path for the location to the PST files you are about to create. The location should be a drive with enough space to accommodate all of the PSTs. Don't panic if the estimated amount of space required seems excessive; ExMerge tends to exaggerate about how much space it thinks it will need. And, remember, you are only extracting a few messages. When finished, click Next twice to commence with the extraction.

This procedure should extract the messages you specified from the mailbox stores. A log file will be recorded in the directory from which you ran ExMerge called ExMerge.log. You should scan this log for any errors.

 ExMerge works great as long as the information store service is running and the mailbox store is mounted. Ontrack's PowerControls tool allows you to mount a mailbox store and access mailbox data even when the mailbox store is dismounted or the information store service is stopped. You can find more information about PowerControls at <http://www.ontrack.com/powercontrols>.

Best Practices for Virus Protection

If you follow good procedures and have a well-thought through antivirus protection scheme in place, you should be safe from most outbreaks. The following list provides best practices for preventing viruses from attacking your organization from the Exchange server perspective:

- Keep your antivirus scanning engine and virus signatures updated daily.
- Implement a multi-tier protection scheme in which at least all inbound mail is scanned by at least two separate virus scanning engines.
- Publish and implement on your perimeter SMTP gateway and Exchange servers a list of forbidden attachments.
- Do not assign administrative or operator account mailboxes. All IT users should have a non-privileged account for regular office automation tasks and a separate account with elevated privileges for administrative tasks; the administrative accounts should never have mailboxes.
- *Quickly* evaluate each security fix and critical update from Microsoft to determine whether it applies to your organization and, if so, how soon should it be applied. *Do not* procrastinate. Although it is true that Microsoft hotfixes sometimes cause new problems, you have to decide whether that small risk is better than the much larger risk imposed by failing to quickly apply critical fixes.
- Develop an escalation procedure that helps you to deal quickly and efficiently with a virus that sneaks in due to out-of-date signatures or previously unknown vulnerabilities.
- Create a separate mailbox that is used for antivirus reports. Monitor this mailbox at least weekly and don't forget to archive it.
- Run weekly reports of viruses detected and messages processed.

Summary

It is no secret that computer viruses, worms, and Trojan horses have become the scourge of every computer user and administrator. Countless hours of downtime and dollars have been lost due to viruses; billions of dollars are spent in the fight against viruses. In Chapter 1, we covered some of the basics of viruses, worms, and Trojan horses and explored a little history. The evolution of this malware clearly shows that the authors of these afflictions are becoming more creative and their spawn is more dangerous than ever.

Virus protection must start with the client computer. An end user or administrator can easily introduce viruses to a corporate network in a variety of ways including floppy, CD-ROM, external POP3 or IMAP4 mail servers, USENET newsgroups, instant messaging clients, Web mail, or email from an Exchange server. Viruses and worms that can enter a network through mobile devices or PDAs have already been discovered and future malware of this class will only be worse.

In Chapter 2, we explored virus and worm protection from the perspective of the client. Virus protection on the client-side must include not only up-to-date antivirus software but also an up-to-date OS and email client. Exchange Server can also be used to further secure clients by restricting MAPI client versions to only approved versions and centrally configuring Outlook 2000 SP3 and later mail security features.

In Chapter 3, we have explored the challenge associated with protecting email on the server. When possible, a virus perimeter should be built around the server in the form of correctly configured firewalls, SMTP virus and content scanning, and properly configured email clients. The Exchange server should also have antivirus software, preferably Exchange-aware antivirus software that uses AVAPI for Exchange for accessing data in the Exchange server.

The key to effective email protection is continued vigilance. You can never get complacent in the battle against malware—just when you thought you had considered everything, someone will prove you wrong.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.