

realtimepublishers.comtm

Tips and Tricks
Guidetm To

**Secure
Messaging**

Jim McBee

Note to Reader: This book presents tips and tricks for six email security topics. For ease of use, the questions and their solutions are divided into topics, and each question is numbered based on the topic, including

- Topic 1: Strategies for Defending Email Infrastructure
- Topic 2: Policies and Procedures
- Topic 3: Architecture and Deployment Considerations
- Topic 4: Antivirus and Anti-Spam Strategies and Best Practices
- Topic 5: Firewall Strategies and Best Practices
- Topic 6: Protecting and Controlling Sensitive Information in Email

Q 1.3: Should I block file attachments?.....1
 Attachments that Should Be Blocked2
 Absolute Blocking List3
 Microsoft Client Blocking3
 Policy-Blocked Attachments10
 Compressed Files11
 Intelligent Attachment Inspection.....12
 Q 2.3: Are there “best practices” for the IT department with respect to messaging security?12
 Mailbox Surfing12
 Detecting Improper Mailbox Access13
 Procedures for Opening Mailboxes14
 Administrative Accounts and Mailboxes.....15
 Email Client Software on Server Consoles.....15
 Q 3.3: Is there a way to guarantee a sender’s identity?16
 SMTP Authentication17
 Whitelisting.....20
 Sender ID21
 DNS and SPF Records21
 Determining the Purported Responsible Address26
 Validating the Sender’s SMTP Server27
 Digital Signatures.....30
 The Mechanics of Message Signing31
 Q 4.3: How do I choose an antivirus software package for Exchange?34

Exchange Server-Aware Virus Scanning Software35

 Features and Decision Points36

Topic 5: Firewall Strategies and Best Practices.....37

Q 5.3: Is Outlook using RPC over HTTP the right solution for my remote users?37

 Requirements38

 Deployment Scenarios39

Topic 6: Protecting and Controlling Sensitive Information in Email41

Q 6.3: What can Enterprise Rights Management do for my company?41

 ERM and Compartmentalized Information42

 Confidential Reports to Customers42

 Preventing Accidental Disclosures of Sales Information43

Copyright Statement

© 2006 Microsoft Corporation. All rights reserved.

[**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>.]

Topic 1: Strategies for Defending Email Infrastructure

Q 1.3: Should I block file attachments?

A: Much of the hostile and unwanted message content that enters a messaging system today is hostile because of the attachments that the message carries—including scripts, executables, screen savers, and even compressed files. For this reason, most messaging systems administrators have adopted a strategy of blocking any message that originates outside of their network that might potentially be carrying a virus, worm, or Trojan horse. An antivirus scanning system residing directly on the mail server can handle this task quite easily; this setup merely requires a message scanning system that is capable of scanning the message stores or the message transport directly on your Exchange Server system. A better approach is to keep the unwanted attachments from ever reaching your mail server. Figure 1.9 shows possible solutions that can be quickly implemented by any organization.

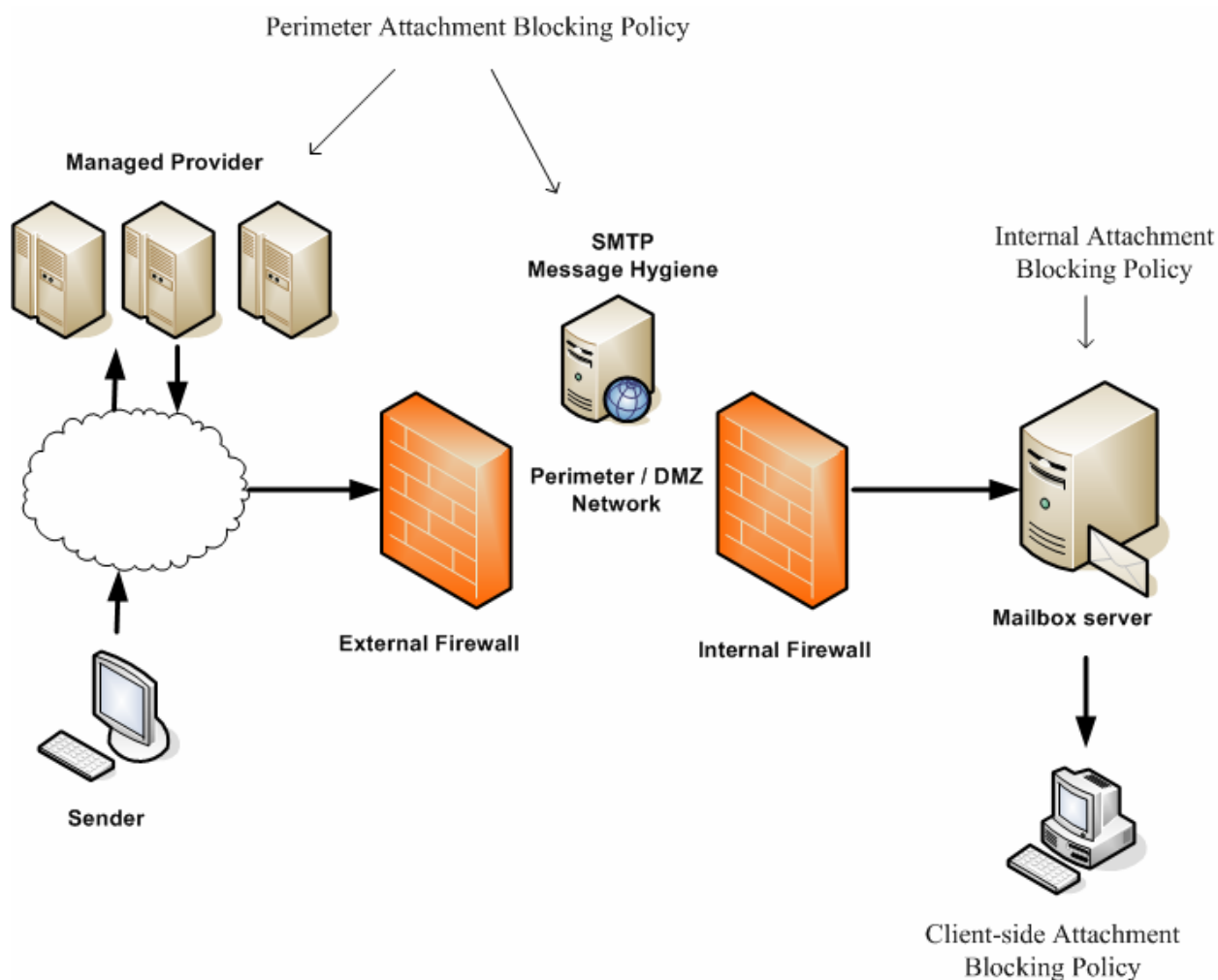


Figure 1.9: Implementing attachment-blocking policies.

With the setup that Figure 1.9 shows, this organization can either block the inbound message content using a managed provider or implement a Simple Mail Transfer Protocol (SMTP) scanning system in the DMZ or perimeter of the network. The SMTP scanning system or the managed provider is then configured to block the list of file attachments that the organization wants blocked. The scanning system or managed provider can then take one or more actions on the blocked attachments:

- Quarantine the message and the attachment entirely
- Quarantine just the attachment
- Submit the message to a dynamic quarantine whereby the message is rescanned for some period of time before being released to the user
- Delete the message and attachment entirely
- Do not notify the user or the sender
- Notify the sender of the blocked message attachment
- Notify the intended recipient that an entire message—or just an attachment—was blocked
- Pass the message (without the attachment) to the user's mailbox

The actions taken will depend on the capabilities of the scanning software being used and the organization's policies on attachment blocking. One advantage of using a managed provider is that the hostile content never arrives within the boundaries of any of your servers, therefore minimizing the risk of infecting internal systems and reducing Internet bandwidth usage.

If an organization applies attachment blocking only at the perimeter of the network, however, a virus or worm might still infect internal messaging components. For this reason, organizations should develop two attachment blocking policies. The first policy dictates the types of attachments that are blocked at the perimeter of the network. The second policy defines the attachments that are not allowed to be sent internally.

In Figure 1.9, the external blocked attachment lists includes attachment types that might not be considered acceptable when receiving messages from the Internet, such as multimedia files as well as dangerous attachments. The internal blocked attachments list includes the list of files that are permitted internally. For some organizations, this list will be identical, while distinctly different for others. Further, clients such as Outlook 2000 and later as well as Outlook Web Access (OWA) 2003 allow for certain types of attachments to be blocked so that they cannot be saved and/or opened.

Attachments that Should Be Blocked

The list of attachment types that must be blocked is going to vary widely from one organization to another. This list will end up being based on the politics and needs of an organization's users. One organization might insist on allowing renamed .EXE files, while another may insist on blocking all compressed files.

Absolute Blocking List

The following recommended absolute blocking list consists of the common attachments types that have been used as attack vectors for most of the viruses, worms, and Trojan horses that have appeared in the wild over the past few years. Table 1.1 shows this list including a description of what the attachment is and the typical attachment extension. For the most part, the files defined in this list are generally not files that should be passed via email messages.

Attachment Extension	Attachment Description
BAT	DOS/Windows batch files
CMD	Windows command files
COM	DOS command file
EXE	Executable programs
JS	JavaScript
MSI	Windows installer files
PIF	Program information file for 16-bit applications
SCR	Screen saver executables
SHS	Shell scrap objects
VB	VBScript file
VBS	VBScript files
WSC	Windows script component

Table 1.1: Generic dangerous attachment list.

Microsoft Client Blocking

Microsoft introduced a comprehensive list of potentially harmful attachment types with the Outlook Email Security Update for Outlook 98 and Outlook 2000. Outlook 2002, Outlook 2003, and OWA 2003 include this list of attachments. Collectively, the attachment lists are called the Level-1 and Level-2 attachments. The intention of this update was to categorize potentially harmful attachments into the Level-1 attachment list, which was meant to prevent a user from opening or saving the attachment. The Level-2 attachments can be opened, but only after the user has saved the attachment to the file system.

Level-1 Attachments

Unless a user works in the IT department or is a developer, the user should not be receiving Level-1 attachments; thus, these attachments should be viewed with much suspicion. Table 1.2 shows the Level-1 attachment list as of Outlook 2002 SP2 and later.

Attachment Extension	Attachment Description
ADE	Microsoft Access project extension
ADP	Microsoft Access project
APP	Microsoft Visual FoxPro application
ASP	Active server page
ASX	Windows media audio or video shortcut
BAS	Visual Basic class module or a BASIC program
BAT	DOS/Windows batch files
CER	Security certificate
CHM	Compiled HTML help file
CMD	Windows command files
COM	DOS command file
CPL	Control Panel extension
CRT	Security certificate
CSH	KornShell script file
EXE	Executable programs
FXP	Microsoft Visual FoxPro compiled program
HLP	Windows help file
HTA	HTML program
INF	Windows setup information file
INS	Internet naming service
ISP	Internet communication settings
JS	JavaScript
JS	JScript Script file
JSE	Jscript encoded script file
KSH	KornShell script file
LNK	Link or shortcut file
MDA	Microsoft Access add-in program
MDB	Microsoft Access program
MDE	Microsoft Access MDE database
MDT	Microsoft Access workgroup
MDW	Microsoft Access workgroup
MDZ	Microsoft Access wizard program
MSC	Windows console definition
MSI	Windows installer files
MSI	Windows installer package

MSP	Windows installer patch
MST	Windows installer transform file
OPS	Office preferences file
PCD	Photo CD image
PIF	Program information file for 16-bit applications
PIF	Shortcut to MS-DOS program
PRF	Microsoft Outlook profile settings
PRG	Microsoft Visual FoxPro program
PST	Microsoft Outlook Personal Folders file
REG	Registration entries
SCF	Windows Explorer command
SCR	Screen saver executables
SCR	Screen saver
SCT	Windows Script Component
SHB	Shell Scrap Object
SHS	Shell scrap objects
SHS	Shell Scrap Object
TMP	Temporary file
URL	Internet shortcut
VB	VBScript file
VB	VBScript file
VBE	VBScript encoded script file
VBS	VBScript files
VBS	Visual Basic Script file
VSMACROS	Visual Studio .NET macro project file
VSS	Visio shapes and Visio stencils file
VST	Visio template file
VSW	Visio workspace
WS	Windows script file
WSC	Windows script component
WSC	Windows Script Component
WSF	Windows Script file
WSH	Windows Script Host Settings file

Table 1.2: Microsoft Level-1 attachments.

If a user tries to send a message that contains an attachment that is on the Level-1 attachment list, the user will see a warning indicating that he or she is attempting to send an attachment that might potentially be harmful; however, Outlook will allow the user to send the message.

You might have noticed that Table 1.2 is missing an entry for compressed files. Compressed files such as ZIP files are commonly used as an attack vector for email-based worms and viruses. The message carrying the ZIP file tries to trick the user into opening the attached file. ZIP files and other attachments can be added to the Level-1 attachment list via different approaches. The simplest of these is to add the additional attachments to the registry. To do so, locate the HKEY_CURRENT_USER\Software\Microsoft\Office\X\Outlook\Security registry key. Replace the X value with the version of Outlook that you are using. The following are the valid version numbers:

Outlook 2000 SP3 and later	9.0
Outlook 2002	10.0
Outlook 2003	11.0

Next, open the Level1Add value and set the attachment types that you want to define as additional Level-1 attachments. If the Level1Add value does not exist, create a new REG_SZ type value called Level1Add. The format for entering additional file types is shown in Figure 1.10.

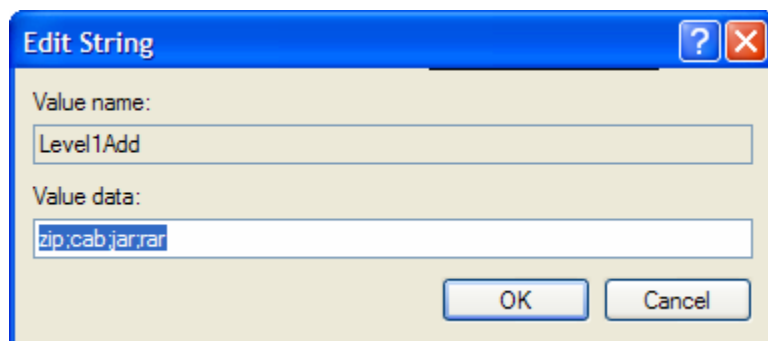


Figure 1.10: Manually adding Level-1 attachments.

Level-1 and Level-2 attachment types for OWA 2003 clients are defined on a per-server basis and include definitions not only for file extensions but also for MIME types. Figure 1.11 shows the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeWeb\OWA registry key, which holds these values.

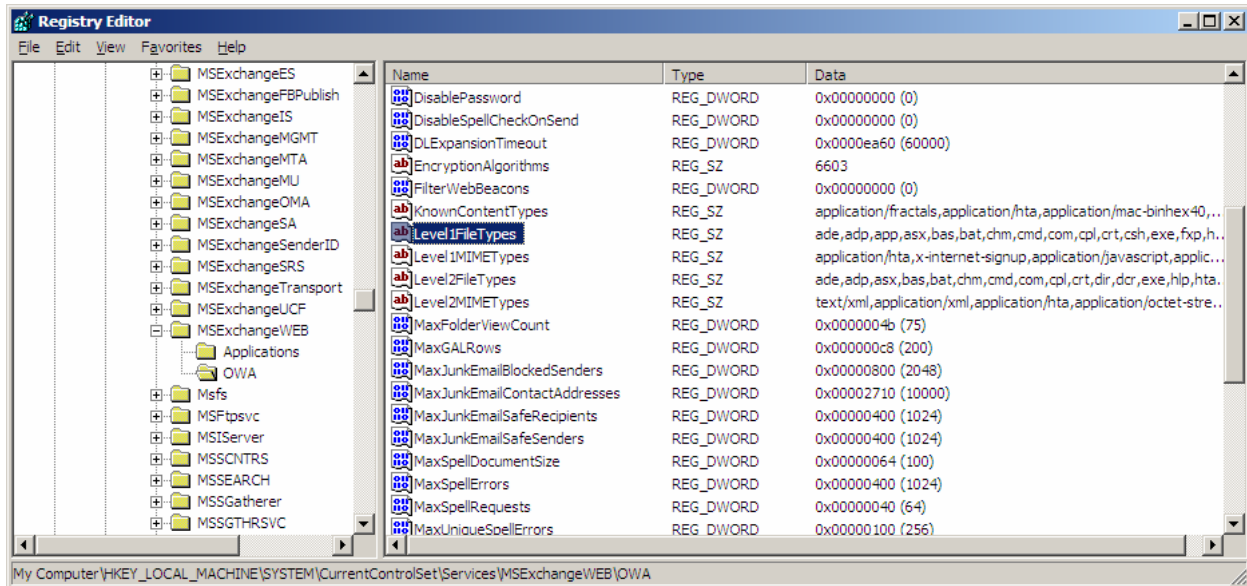


Figure 1.11 OWA attachment-blocking registry values.

Although these settings can be edited via the registry editor, a simpler way to manipulate the OWA attachment handling rules is to use the OWA Web Administration tool that can be downloaded from the Exchange 2003 tools page at <http://tinyurl.com/9cpt6>.

Level 2-Attachments

Level-2 attachments are attachment types that are considered possibly unsafe and therefore the user should go through the extra step of saving the attachment to the file system before opening it. By default, the Level-2 attachments list is empty, but a Level-1 attachment can be demoted to a Level-2 attachment via a registry key edit. Figure 1.12 shows two messages that are presented if you open an attachment using Outlook 2003. The warning dialog box on the left is the default message that users see when they open any attachment that is not on the Level-1 or Level-2 attachment list. The warning on the right is the warning that users receive if they try to open a Level-2 attachment.

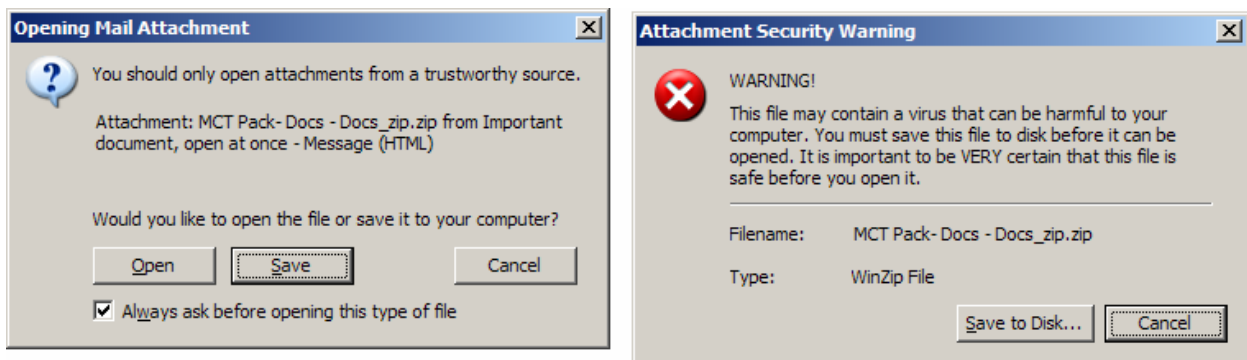


Figure 1.12: Warning messages when opening attachments in Outlook 2003.

Using the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeWeb\OWA registry key, you can define Level-2 attachment types by creating a REG_SZ value called Level1Remove and adding the attachments that you want to remove from the Level-1 attachment list. You can also define attachments that are not on the Level-1 list, for example, defining restrictions for compressed files.

Bypassing or Managing Client Attachment Blocking

Editing the registry of each user that is going to need to have Level-1 file types demoted is not a productive use of time. There are a couple of options that can help you with this management task. Outlook MVP Ken Slovak wrote a very useful COM add-in for Outlook 2000 SP3 and later that allows the user to manage the attachments that are in the Level-1 attachment list through a graphical user interface (GUI). You can download the Attachment Security & Options add-in for Outlook from Ken's Web site at <http://www.slovaktech.com/attachmentoptions.htm>. Figure 1.13 shows the Attachment Security & Options property page.

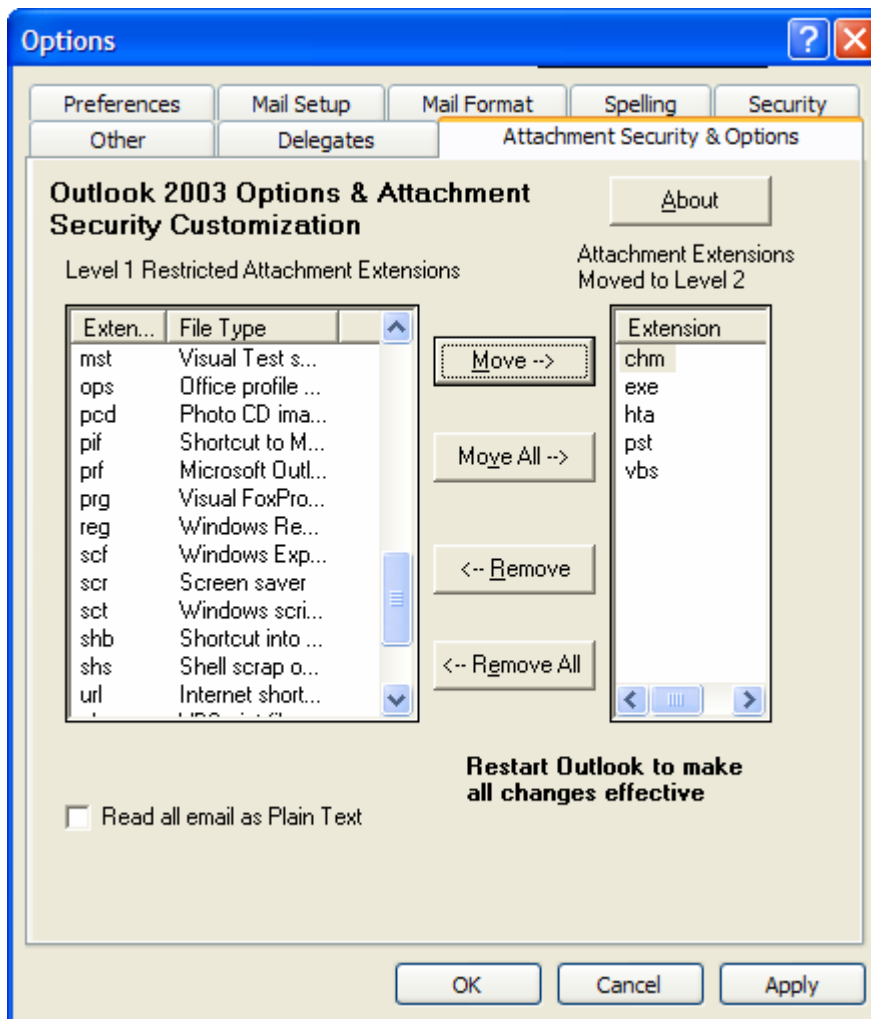


Figure 1.13: Attachment Security & Options COM add-in for Outlook.

If you are running Outlook 2003, you can configure the Level-2 attachment list using a Group Policy Object (GPO). The Office 2003 Resource Kit includes administrative templates for all Microsoft Office applications, including Outlook. The *Allow access to e-mail attachments* policy setting (shown in Figure 1.14) allows the administrator to define attachment types that should be moved from the Level-1 to the Level-2 list as well as attachments that should be added to the Level-2 attachment list.

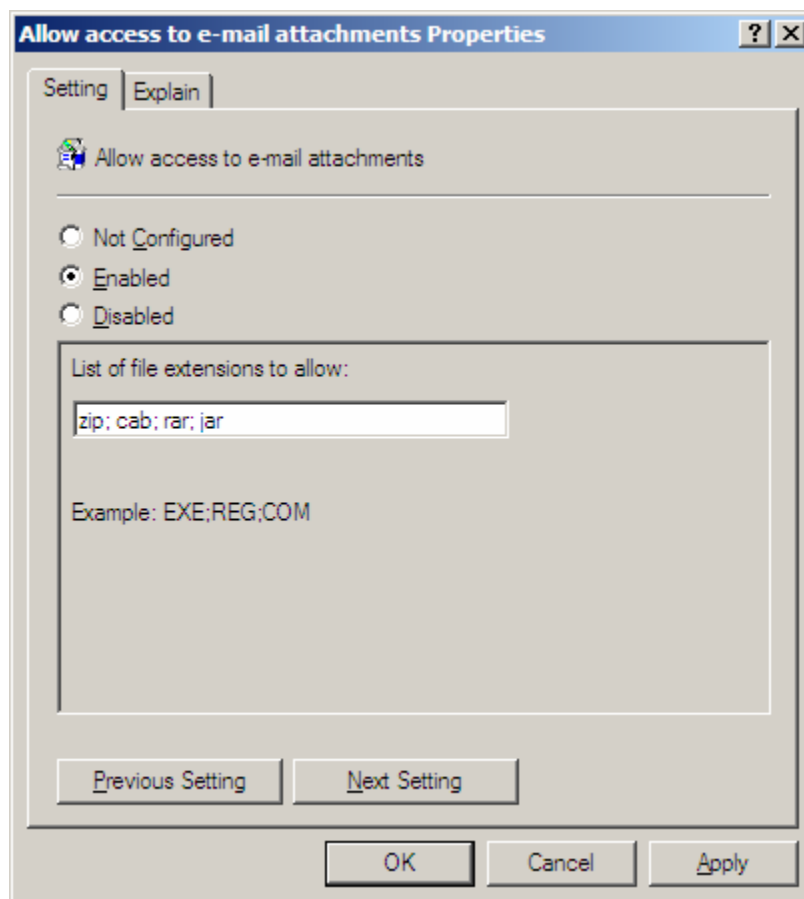


Figure 1.14: Configuring Level-2 attachments through a GPO.

The administrative templates for the Office Resource Kit can be downloaded from <http://www.microsoft.com/office/ork>. Once the Outlook 2003 administrative template has been loaded into a policy, this policy setting is found in the policy under Administrative Templates, Microsoft Office Outlook 2003, Tools, Options, Security.

Still another way that an Exchange Server administrator can control the restricted attachments is to use the Outlook Security Settings form (shown in Figure 1.15) to define attachment security as well as other Outlook security components.

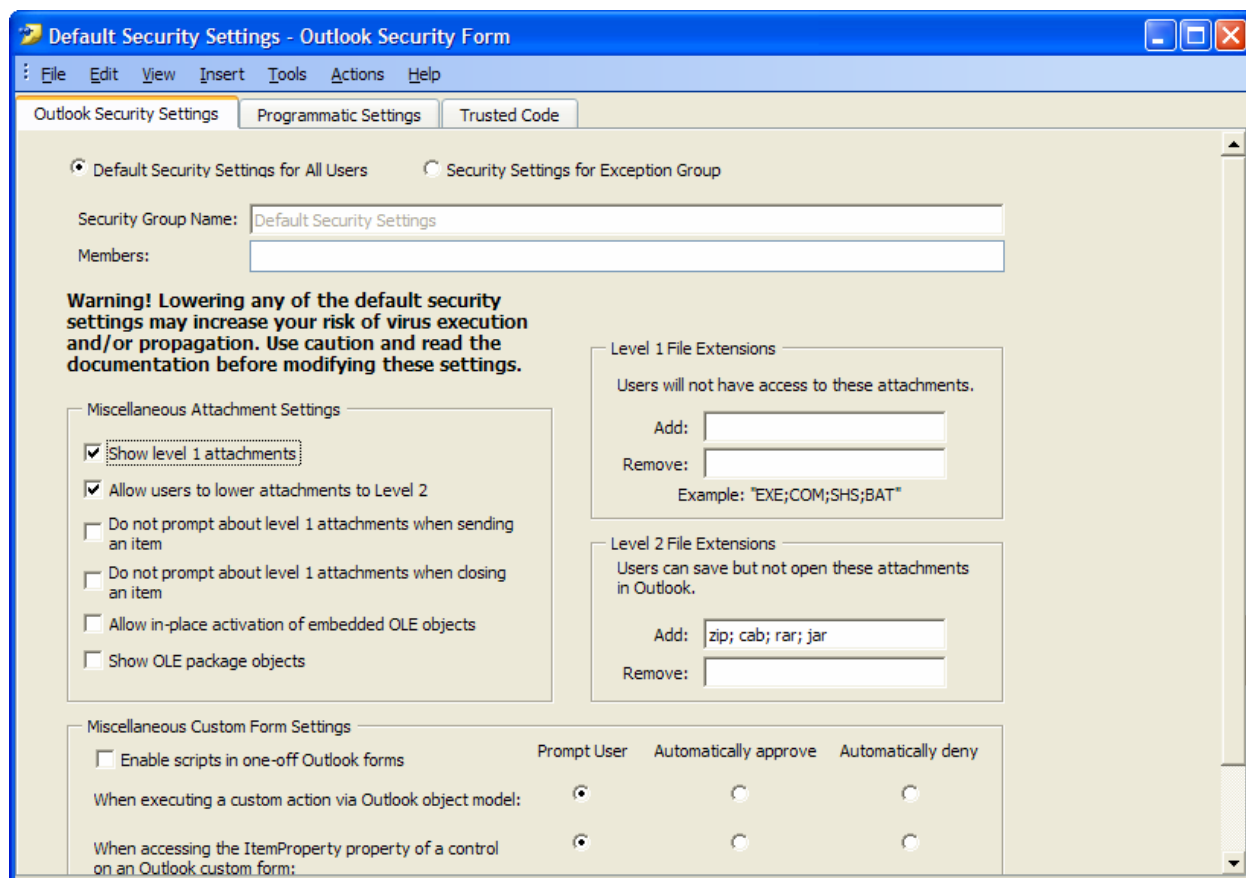


Figure 1.15: The Outlook Security Settings form.

The Outlook Security Settings form can be used with all versions of Outlook later than Outlook 2000 SP3. More information about using this form can be found in the Microsoft article “Administrator Information About the Outlook E-mail Security Update: June 7, 2000.”

Policy-Blocked Attachments

For some organizations, it might be acceptable to use the email system to pass around multimedia content that is business related; in other organizations, such content is not appropriate. Worse still is that some content such as a user-copied WMA or MP3 file might violate copyright laws if the user is illegally distributing the file. In addition, if the content is offensive, it might be violating an organization’s acceptable use policy. Depending on your organization’s policies, blocking this type of content might be wise. Table 1.3 shows a list of common multimedia file types that you might consider blocking.

Attachment Extension	Attachment Description
AVI	QuickTime audio/video interleave file associated with video or audio files
GIF	Graphic Interchange Format files can contain animations
MID/MIDI	Musical Instrument Digital Interface files
MOV/QT	QuickTime video clip
MP3/MPEG3	MPEG audio stream usually associated with audio or music files
MPG/MPEG/MPEG2	MPEG animation usually associated with video
PPS	PowerPoint show files
RAM/RM	RealMedia streaming media
WAV	Waveform audio associated with audio/music files
WMA	Windows Media audio file
WMV	Windows Media file

Table 1.3: Multimedia files that might need to be blocked.

Compressed Files

In some organizations, users might not be able to do without compressed files (for example, if they receive ZIP files that compress large documents, spreadsheets, and presentations). This is convenient because it reduces the space that users' mailboxes consume and allows the attachments to be transmitted quicker. Some organizations even use tools such as C2C's MaX Compression, MAPI Lab's Attachments Zip Compressor, or WinZip's Companion For Outlook so that all outbound attachments in files are automatically compressed into ZIP files.

However, several common viruses such as variants of the Beagle and Spester worms have used compressed files such as ZIP files to get past antivirus scanning systems and scanning systems that block scripts or executables. For this reason, the messaging administrator and IT decision makers must balance usability with the potential that a user may open a compressed file that contains hostile content. Table 1.4 shows a list of common compressed file types.

Attachment Extension	Attachment Description
ARJ	Compressed archive file
BIN	Compressed file format usually used by the Macintosh OS
CAB	Microsoft cabinet file frequently used as an installation archive
JAR	May be an archive file or a Java archive file
RAR	WinRAR compressed archive file
TAR	Tape archive file frequently used with UNIX systems
ZIP	Most common compressed or archive file format

Table 1.4: Compressed attachments that might need to be blocked.

Intelligent Attachment Inspection

Some antivirus systems include a feature that allows the antivirus system to intelligently inspect attachments to determine the attachment type. Quite simply, the attachment is examined to determine what it is. Thus, a user could rename an EXE file to a TXT file, but the intelligent inspection system would still ascertain that the file was really an EXE and block or quarantine it as appropriate. The danger of such systems is that if you really need to receive ZIP files, you will not be able to instruct remote senders to rename the extension for you.

Topic 2: Policies and Procedures

Q 2.3: Are there “best practices” for the IT department with respect to messaging security?

A: Unfortunately, very few organizations provide their IT employees with an IT ethics briefing. Although most IT staff simply use common sense when dealing with IT systems and private information, guidelines are appropriate to ensure that everyone follows consistent procedures. Specifically, guidelines for IT staff should be created when handling potentially sensitive information in a messaging system. Message content may be found in several places within a messaging system infrastructure:

- Users’ mailboxes
- Public folders
- Personal mailbox storage files (such as PST files)
- Message system queues
- Message hygiene/content inspection quarantines
- Email archival systems

Mailbox Surfing

In the past 15 years of being a messaging systems consultant, I have been asked on numerous occasions to help audit inappropriate message access by messaging administrators. In one situation, the mail administrator was accessing a user’s mailbox simply so he could play a practical joke on the user by sending out a message saying that he was supporting a different sports team from his alma mater. Although this seems like a minor offense, it is nonetheless a breach of privacy and the administrator was fired.

In another case, the mail administrator responsible for reviewing the anti-spam and antivirus quarantines was occasionally seeing sensitive messages. The administrator was then sharing this information with other coworkers. Although the mail administrator had never been advised on the appropriate handling of this type of information, one would think that common sense would prevail and the information would be held in confidence. This administrator was counseled by the organization’s human resources department, received an official reprimand, and had her responsibilities changed.

A simple policy that encompasses email access, information disclosure, and IT workers and that covers the disclosure of this type of information might state something like this: “In the course of working with message systems, queues, and quarantines, IT workers will be exposed to private or confidential data. This data must not be disclosed except as part of the worker’s official duties for the organization.”

Detecting Improper Mailbox Access

In most of these cases, catching the culprit is easy as long as auditing is enabled. On an Exchange Server, the Diagnostics Logging category MExchangeIS Mailbox, Logons needs to be configured for at least minimum logging. Then monitor the Exchange Server’s Application event log for Event ID 1016, indicating that a user account that is not the owner of the mailbox has tried to access the mailbox. This event is shown in Figure 2.1.

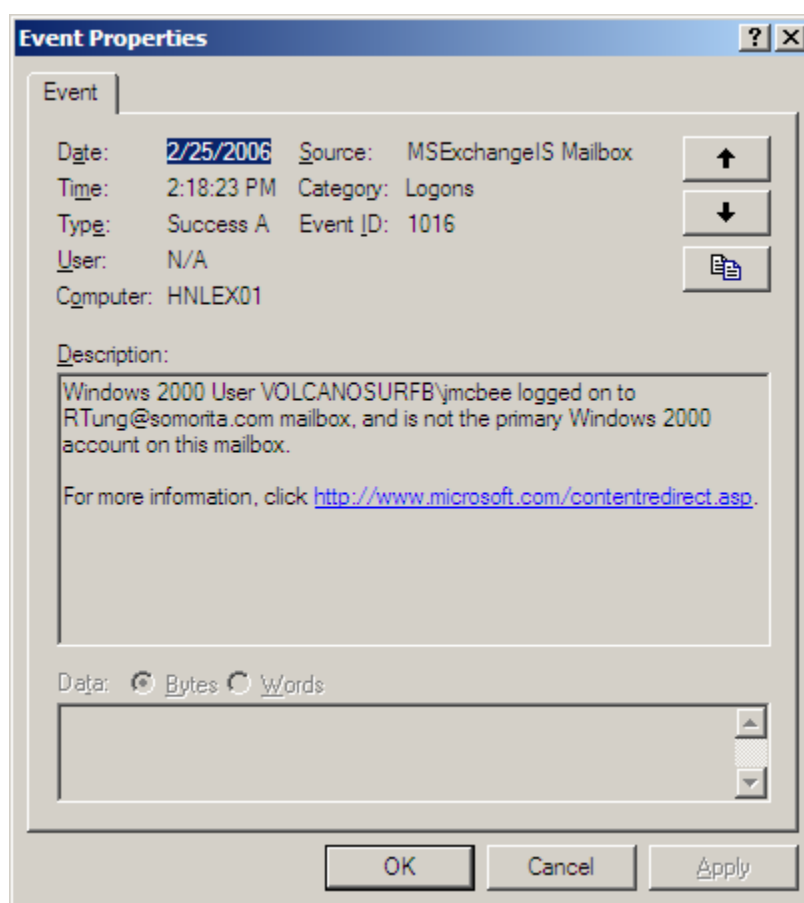


Figure 2.1: Event properties indicating that a user is accessing another user's mailbox.

Although event ID 1016 may not indicate an actual intrusion (since the user may have been given permissions to this user’s mailbox for official reasons), it does indicate non-owner access of mailboxes that might need to be investigated.

Procedures for Opening Mailboxes

If your organization does not have a policy regarding access to users' mailboxes, develop—at the very least—an informal policy indicating under what circumstances mailboxes may need to be accessed. If you export mail data to personal message stores (such as PST files) or mail is archived to a message archival system, these storage systems should be included in the policy.

Your policy might state something as simple as: “No user mailbox other than administrative or system mailboxes will be accessed by anyone other than the owner of that mailbox or their delegates unless specifically requested by the Human Resources department, Information Assurance/Security department, or the Directory of Information Technology.”

A good practice is to create one security group that has been delegated full administrative control to the entire message system, including the ability to open mailboxes. For an Exchange-based mail system, the group must be delegated the Full Control permission to the organization and have the Receive As permission changed from Deny to Allow. The Deny permission is explicitly set for an administrator that has been delegated permissions to the Exchange organization or administrative group using the delegation wizard. You can remove these permissions only on the Security property page (shown in Figure 2.2).

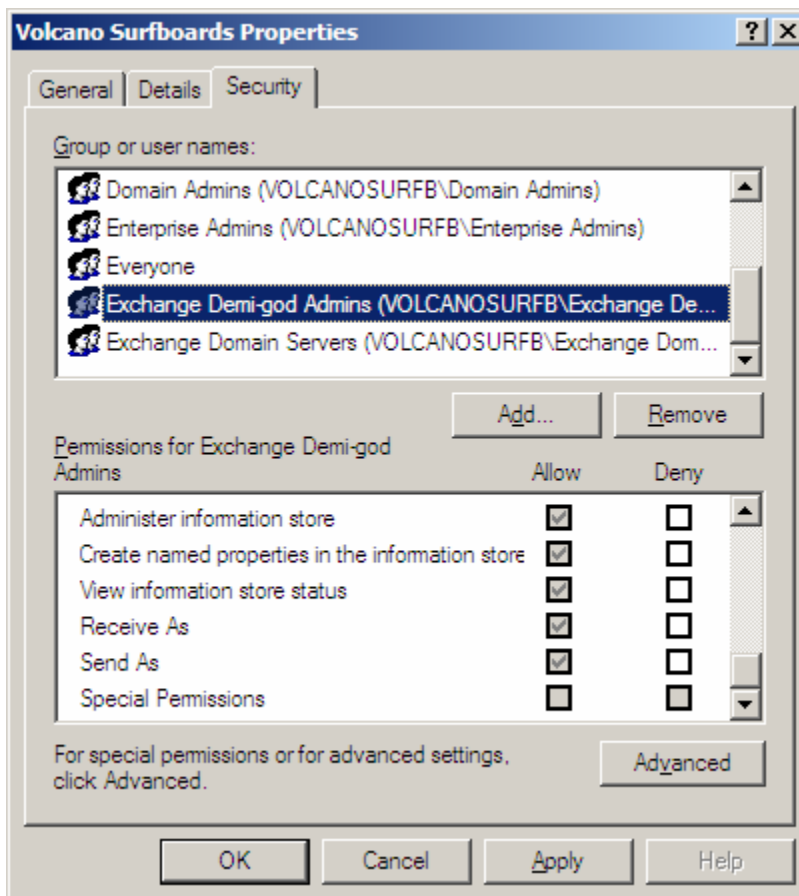


Figure 2.2: Giving a group the ability to open mailboxes.

By default, the Security property page does not appear on the organization and administrative group objects. See the Microsoft article [Security Tab Not Available on All Objects in System Manager](#) for more information about how to enable the Security property page.

Often, the user account that is a member of the security group delegated to have full mailbox access has “two person” integrity on the account, meaning two people are required to logon as this user. For example, one person has the smart card for the account and the other person has the PIN, or each person has one half of the password.


Administrative Accounts and Mailboxes

A few simple steps can improve security for organizations by segmenting permissions and responsibilities. When creating user accounts and mailboxes, keep these points in mind:

- User accounts should have no elevated permissions
- Create separate accounts for administrators that have their necessary administrative permissions
- Typical office automation/knowledge worker tasks such as word processing, spreadsheet manipulation, email communication, and Web surfing should only be done when using non-administrative accounts
- Administrative accounts should not have mailboxes
- Practice the principal of least permission where administrative accounts are assigned only the permissions they require to do their jobs
- Do not create administrative accounts with mailboxes; doing so might make the spread of viruses or other malware easier

Email Client Software on Server Consoles

Another common mistake that administrators make is that they install email client software on their servers. With few exceptions, this is a bad practice for a couple of reasons. The first reason is that the mail client software might interfere with the operation of the mail server software, such as in the case of installing Microsoft Outlook on an Exchange Server.

 Do not install email client software on a server. Email client software running on a server may allow servers to be infected with viruses or impeded the functions of a mail server.

The second reason this is a bad practice is that an administrator might access his or her mailbox while working on a server console and infect the server with a virus or worm. In cases in which an email client is required on a server to perform tasks such as automated message processing or to work with a specific application, ensure that the server has adequately configured antivirus software to prevent the server from becoming infected with some type of malware. Except in the case of a lab environment, email client software should not be installed on a mail server.

Topic 3: Architecture and Deployment Considerations

Q 3.3: Is there a way to guarantee a sender's identity?

A: The Simple Mail Transfer Protocol (SMTP) was designed to be open and very simple. The original standard for the protocol provided for no security or authenticity mechanisms. Even as the SMTP standard has evolved to meet modern messaging needs, mechanisms for ensuring the identity of senders have remained elusive.

Businesses increasingly rely on email to share business-critical information, yet that very information is easily spoofed and falsified. Figure 3.8 shows a message that was easily spoofed using a POP3 mail client. There are few clues in this message that a typical end user can use to determine whether the message is authentic and came from the purported sender.

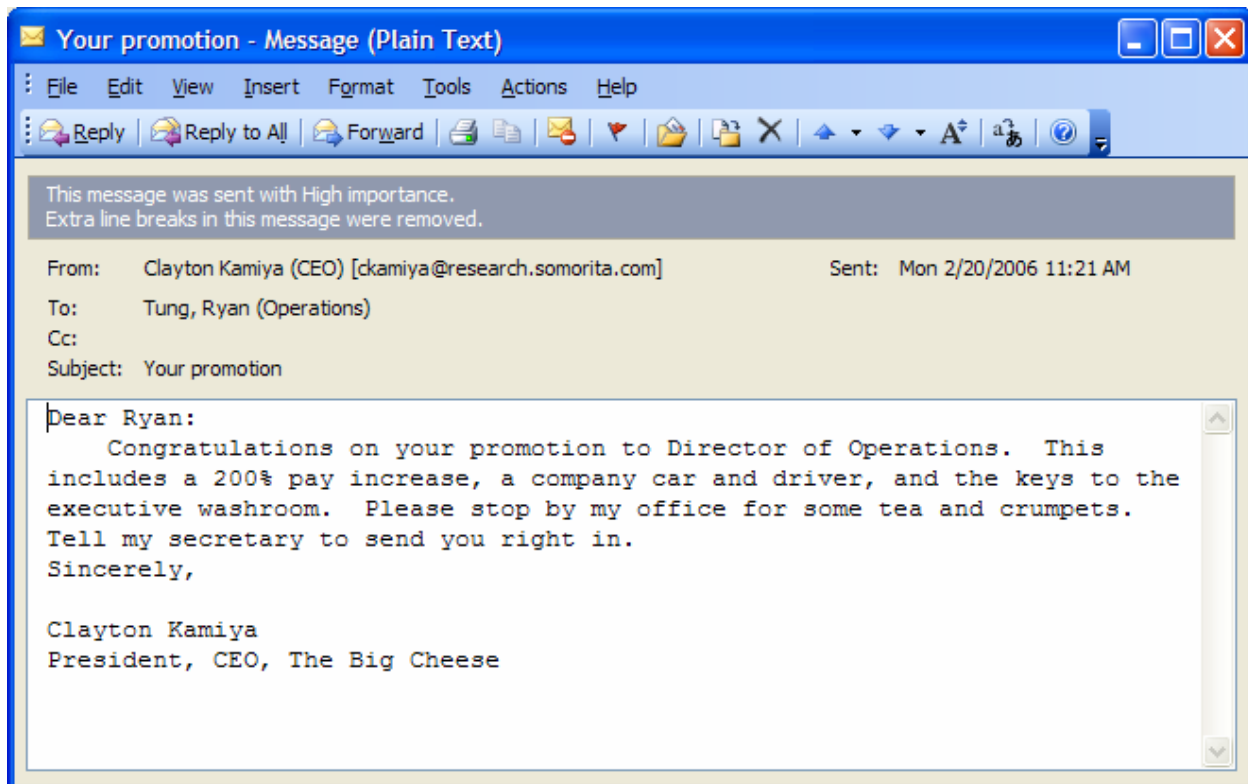



Figure 3.8: A spoofed message.


Phishing schemes, easily spoofed messages, and massive amounts of spam that arrive in users' mailboxes have only contributed to the erosion of faith in the authenticity of the source of many of the messages. For example, to reduce the likelihood that any customers succumb to a phishing scheme, many banks have announced on their Web sites that they will never send announcements or news email messages to customers and thus customers should not respond to any message that might appear to be from them. By doing so, the banks have denied themselves a very efficient mechanism for communicating with their customers.

 All SMTP email between two Exchange Server systems in the same Exchange organization is automatically authenticated.

There are a couple of mechanisms that can be used to provide varying levels of authentication for messages that arrive within your organization. These include requiring SMTP authentication, employing whitelisting systems, using Sender ID, and implementing S/MIME digital signatures.

SMTP Authentication

The most modern of SMTP systems include an authentication mechanism—the enhanced SMTP (ESMTP) verb AUTH. The AUTH verb allows an SMTP client system (the sender) to authenticate to the SMTP server. However, this setup requires that credentials be created for each organization that is going to send you mail, and their servers would have to be configured to use these credentials. Although configuring SMTP authentication is simple to do, it is not practical except in situations in which you are communicating with the same SMTP servers all the time (such as business partners or other SMTP systems within your organization).

 SMTP authentication is not practical for large numbers of domains to which you must send authenticated mail, as it requires a lot of administrative attention to configure and maintain (such as changing passwords that must be reset).

Configuring SMTP authentication is simple using a mail system such as Exchange 2000 or Exchange 2003. To illustrate, let's review a simple example of how this would work. Suppose an organization needs to send mail to a remote organization, and the connection is authenticated. The sending organization is Volcano Surfboards (volcanosurfboards.com) and the receiving organization's domain is Somorita Surfboards (somorita.com); the administrators at somorita.com have created a username and password for the Volcano Surfboards mail server.

The mail administrator at Volcano Surfboards needs to create an SMTP Connector that will be used by the mail destined for somorita.com. Most administrators are used to creating an SMTP Connector whose scope is *, which means that it will deliver outbound mail to all domains. To limit the scope of this connector so that it delivers mail only for somorita.com, a custom SMTP address space is defined on the SMTP Connector. This is shown in Figure 3.9.

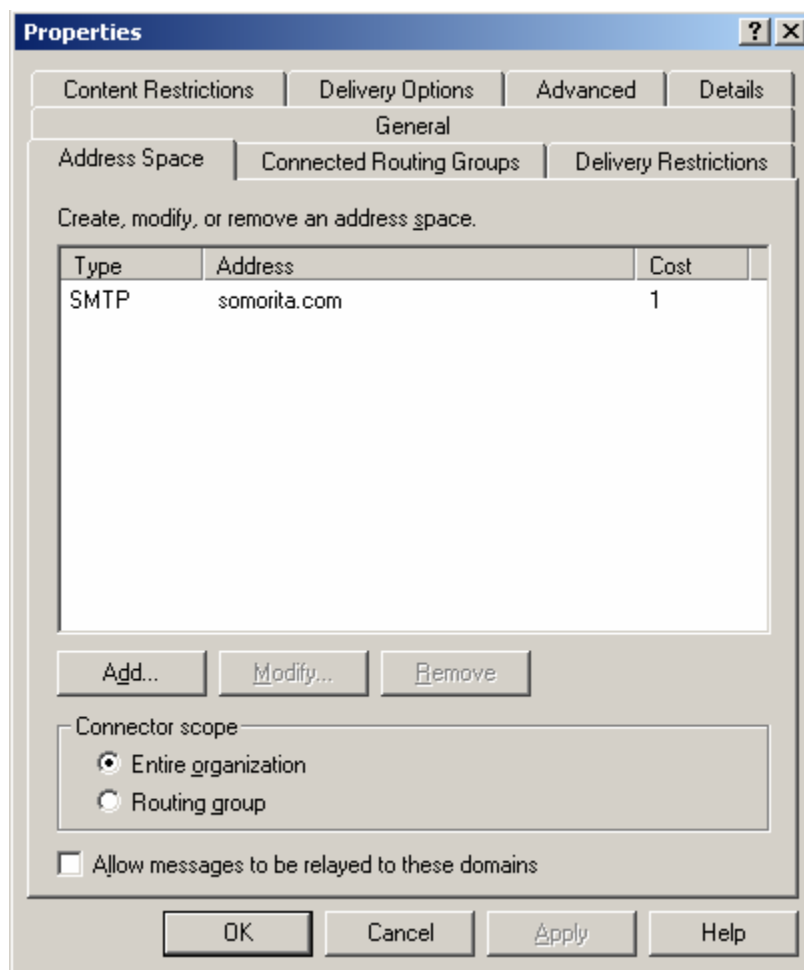


Figure 3.9: Configuring a specific address space.

Once the scope of the SMTP Connector is limited to send mail only to the somorita.com domain, the authentication options can be configured. On the Advanced property page of the SMTP Connector, the Outbound Security button lets you configure the authentication mechanism. The Outbound Security property page is shown in Figure 3.10. The default setting for outbound security is anonymous access. Basic authentication will be the most common option because this setting will work with most any SMTP server that supports authentication including other Exchange Servers.

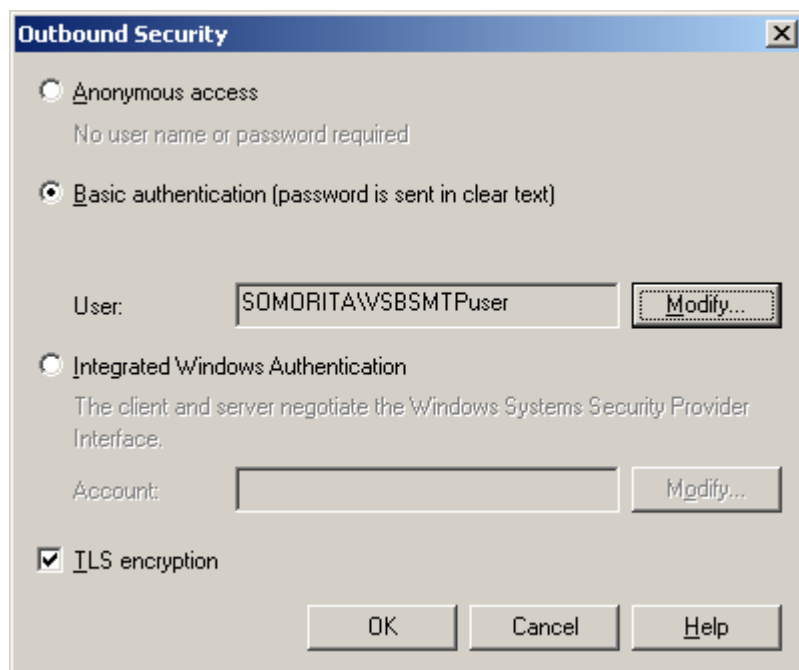


Figure 3.10: Defining SMTP authentication.

If you are using basic authentication, you should select the TLS Encryption check box because basic authentication sends the user name and password in clear text. Using TLS Encryption will require that the remote SMTP server have a certificate installed and configured for its SMTP server. The Integrated Windows Authentication option will only work when selecting user accounts that are located in your Active Directory (AD) forest or with an AD forest to which you have a trust relationship. Unfortunately, users have no simple way to verify that messages were received from an SMTP connection that is authenticated.

Whitelisting

Whitelisting is a technology that is normally used to fight spam but allows you to create a list of authorized recipients and thus can also help ensure that the threat from bulk phishing schemes is reduced. Figure 3.11 shows how a whitelisting system might function either when provided by a managed provider or by some type of SMTP-based system in the organization's DMZ.

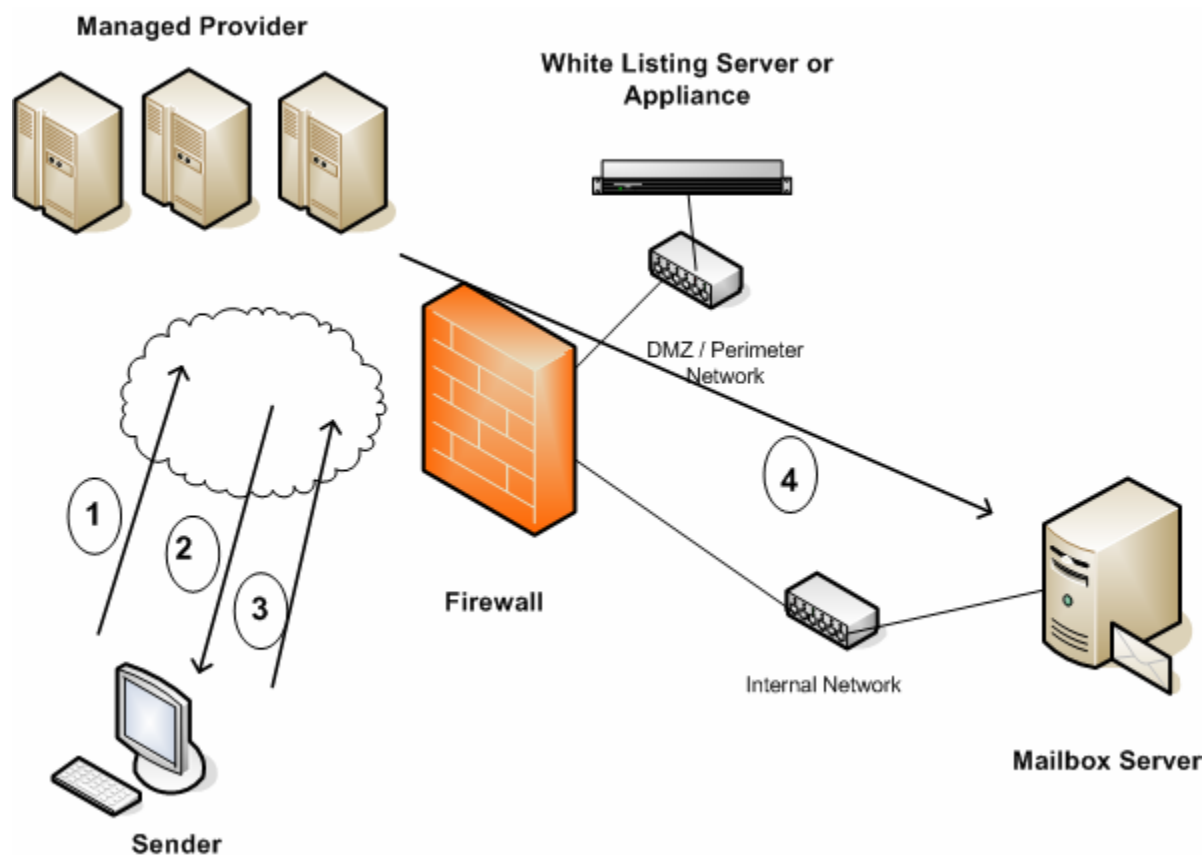


Figure 3.11: Using a whitelisting service.

In the example in Figure 3.11, a sender on the Internet sends a message to a recipient that is protected by a whitelisting service. The whitelisting service could either be in the form of a managed provider such as FrontBridge or Spam Arrest or it could be an appliance or SMTP system in the organization's DMZ such as a Sendio appliance. The whitelisting service has a database of authorized senders; this database can be pre-populated by the users or IT department of the organization being protected.

If the sender is not on the whitelist, the service sends back a challenge to the sender. The sender must do something to confirm their identity such as connect to a Web page, reply to the message, or enter an authentication code. Once the sender has authenticated, some services and software packages will still require the recipient to validate the sender as authentic. When the user is authenticated, the whitelisting service adds that sender's email address to its database of authenticated users, then passes messages through to the recipient's mailbox server. If you place a whitelisting service in front of your mailbox servers, you decrease the likelihood that bulk spamming or phishing attacks will affect your user community, but a practical joke or an intentional deception such as the one shown earlier is still possible.

Sender ID


Sender ID is one of the email industry's attempts at reducing spam and validating that a message has indeed been received by its intended recipient. Specifically, Sender ID is part of the Coordinated Spam Reduction Initiative, which is intended to reduce spam and to make spoofing a message more difficult. An SMTP server that has Sender ID enabled essentially evaluates the sender of the message, the SMTP server from which the message was received, and a list of SMTP servers that are authorized to send messages for the sender's domain. The message is evaluated and ranked into one of three categories:

- The sending SMTP server is on the authorized list of SMTP servers
- The sending SMTP server is not on the list of authorized SMTP servers
- The SMTP domain from which the message originated does not have a list of authorized servers

Based on these categories, the receiving SMTP system can reject the message or pass the category on to the end user or an anti-spam system for further processing. Although this process might sound simple, there is quite a bit more to it than you would first think. And even if you don't implement Sender ID on your own SMTP servers, you should still identify your authorized SMTP servers so that other organizations implementing Sender ID won't incorrectly reject or quarantine mail from your servers.

DNS and SPF Records

To identify a list of authorized servers for a specific domain, a receiving SMTP server must check somewhere for a list of authorized SMTP servers for that domain. The list of authorized servers is published for that domain using DNS in the form of Sender Policy Framework (SPF) resource records. The SPF record contains a list of the IP addresses from which your organization will transmit email to the Internet.

 Creating SPF records requires knowledge of exactly which servers send mail on behalf of your domain. This includes smart hosts and managed provider SMTP servers.

Microsoft has created a Web-based wizard that will verify the existence of an SPF resource record, lookup your organization's MX records and A records for your domain, and take you through the necessary steps to create an SPF record for any domain and any type of mail system. You can find this wizard on the Internet at <http://www.anti-spamtools.org>.

Creating SPF resource records is simple with the Sender ID Framework SPF Record Wizard found at <http://www.anti-spamtools.org>.

The following scenario outlines a very simple example of how to do so for a domain called somorita.com. In this example, somorita.com has only a single outbound SMTP server. This SMTP server is the same server that is used for inbound and outbound mail. In the third step of the Sender ID Framework SPF Record Wizard, you simply need to select the *Domain's inbound servers may send mail* check box and verify that the host name that was read from the MX record is correct (see Figure 3.12).

Sender ID Framework SPF Record Wizard - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address <http://www.microsoft.com/mscorp/safety/content/technologies/senderid/wizard/default.aspx>

Microsoft.com Home | Site Map

Search Microsoft.com for: Go

Safety

Wizard Home | About Sender ID

Sender ID Framework SPF Record Wizard

Step 3 of 4: Create SPF Record

Use the form below to create or modify your SPF record. Some parts of the form have already been filled in with information the wizard found in DNS for **somorita.com**.

Domain Not Used for Sending E-Mail

Please check this option if this domain is not used for sending outbound e-mail. Domains which do not send out e-mail will have no outbound mail servers. ([What's this?](#))

No mail is sent from domain

Inbound Mail Servers Send Outbound Mail

If your inbound mail servers are also used to send outbound mail, you should add this option to your new SPF record. If you are not sure, we recommend you add this option, since most inbound mail servers can at least send outbound non-delivery reports (NDRs). ([What's this?](#))

Domain's **inbound** servers may **send** mail

These addresses are currently listed in MX records for **somorita.com**. Check each MX address that is a valid outbound e-mail server for this domain.

serenity.dnsalias.com

Enter any additional domain names whose MX records refer to valid outbound e-mail servers for **somorita.com** (one domain name per line).

Internet

Figure 3.12: Creating a simple SPF resource record.

On the fourth page of the wizard, it creates a simple text string shown here:

```
v=spf1 mx mx:serenity.dnsalias.com ~all
```

The DNS administrator for this domain must then create a DNS TXT record for the somorita.com domain. In this example, the same SMTP server is used for sending mail as is used for receiving mail.

In a more complex example, an organization called volcanosurfboards.com needs to create an SMTP record. Figure 3.13 shows this slightly more complex environment; this organization has two servers internally that can originate SMTP mail to the Internet, and they use a managed provider through which outbound mail is normally delivered. Their SMTP servers are configured to send mail directly to the recipient if the managed provider is down.

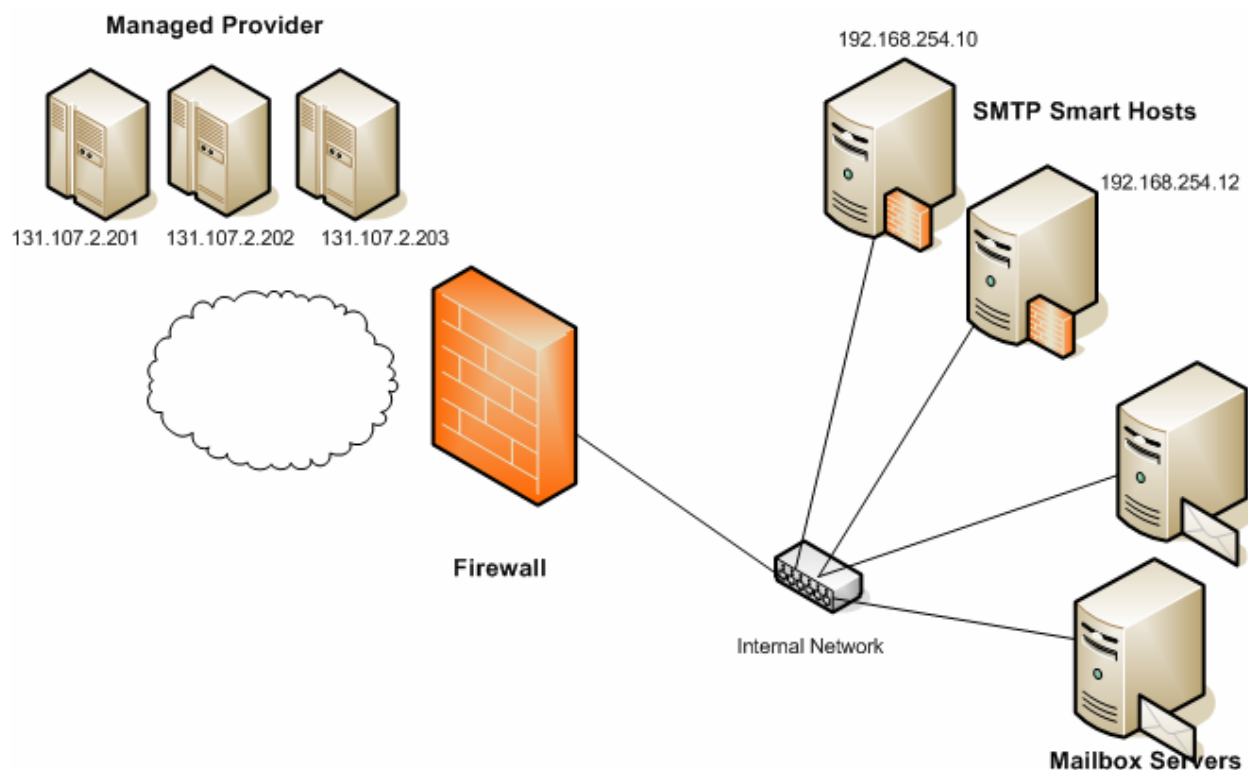


Figure 3.13: Creating SPF records for a slightly more complex organization.

The two servers on the inside of the network that are allowed to send internal mail have IP addresses of 192.168.254.10 and 192.168.254.12. These are different than the IP addresses that are in the organization's MX records.

The managed provider provides the IP addresses of their outbound mail servers; those are 131.107.2.201, 131.107.2.202, and 131.107.2.203. Using the Sender ID Framework SPF Record Wizard, the IP addresses of the individual outbound mail servers are defined in step 3 of the wizard. This is shown in Figure 3.14.

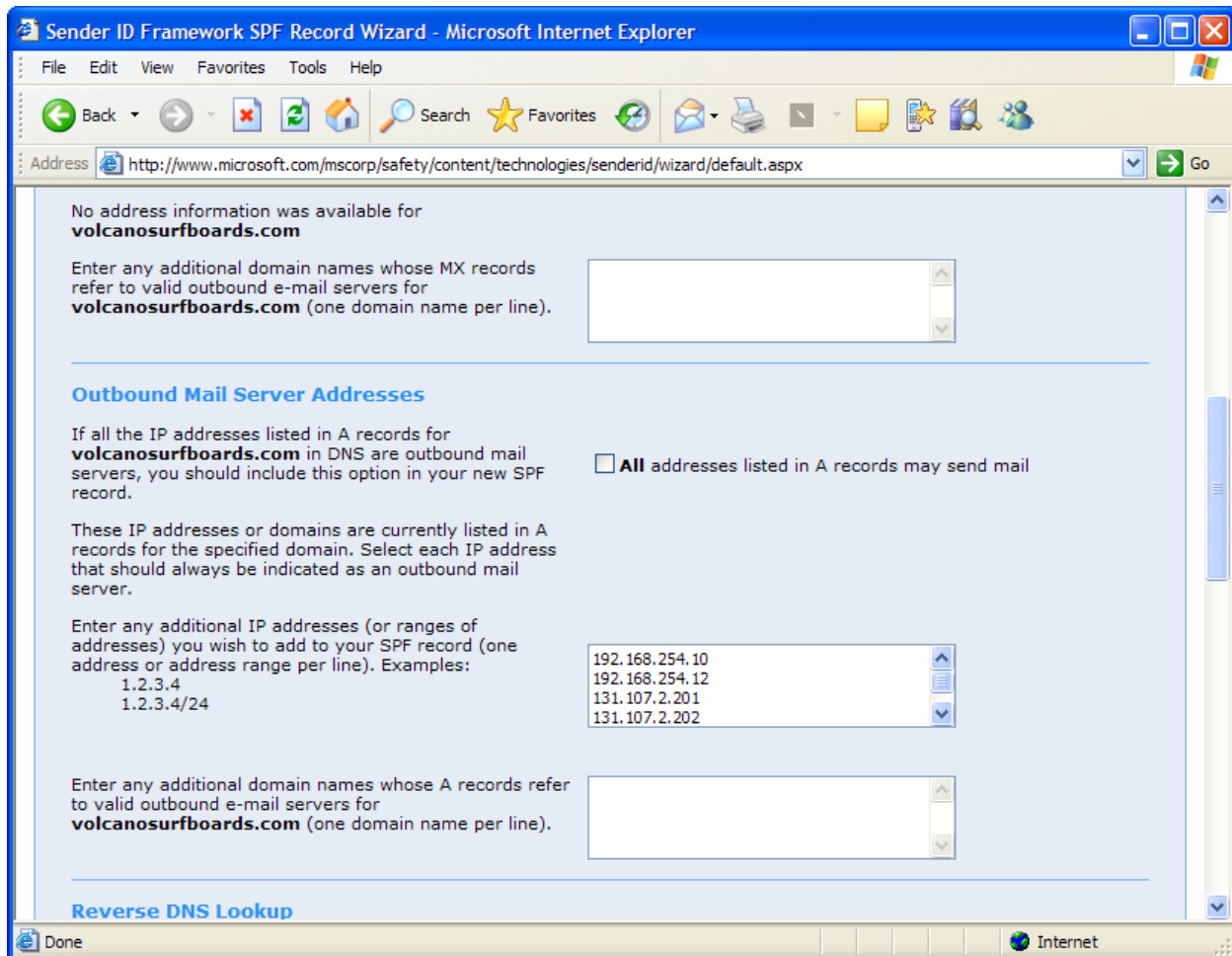


Figure 3.14: Assigning specific IP addresses for outbound SMTP mail servers.

The Sender ID Framework SPF Record Wizard creates a slightly more complex SPF record for volcanosurfboards.com. This record looks like this:

```
v=spf1 ip4:192.168.254.10 ip4:192.168.254.12 ip4:131.107.2.201
ip4:131.107.2.202 ip4:131.107.2.203 ~all
```

Defining the SPF records, even if the organization is not using Sender ID, will help other organizations to properly identify and authenticate the email coming from the organization's SMTP servers.

You can also check another organization's SPF records by simply using NSLOOKUP.EXE. The previous examples are still fairly simple and reflect what a smaller organization might have. For example, if I want to check the SPF records for aol.com, I would type

```
C:\>nslookup -q=txt aol.com
Server:   kilauea1.volcanosurfboards.com
Address:  192.168.254.15

Non-authoritative answer:
aol.com text =

                "v=spf1 ip4:152.163.225.0/24 ip4:205.188.139.0/24
ip4:205.188.144.0/24 ip4:205.188.156.0/23
ip4:205.188.159.0/24 ip4:64.12.136.0/23 ip4:64.12.138.0/24
ptr:mx.aol.com ?all"
aol.com text =

                "spf2.0/pra ip4:152.163.225.0/24 ip4:205.188.139.0/24
ip4:205.188.144.0/24 ip4:205.188.156.0
/23 ip4:205.188.159.0/24 ip4:64.12.136.0/23 ip4:64.12.138.0/24
ptr:mx.aol.com ?all"
```

In this example, you can see many IP addresses that have been entered in the form of IP subnets. An organization can also consolidate these records by using include statements such as the case of microsoft.com.

```
C:\>nslookup -q=txt microsoft.com
Server:   kilauea1.volcanosurfboards.com
Address:  192.168.254.15

Non-authoritative answer:
microsoft.com text =

                "v=spf1 mx include:_spf-a.microsoft.com include:_spf-
b.microsoft.com ~all"
```

Determining the Purported Responsible Address

If you are implementing Sender ID on your SMTP servers, the first thing that your SMTP server that receives a message from the outside world has to do is to determine the responsible sender. The originating email address is known as the Purported Responsible Address (PRA); a more precise definition of the PRA is the email address that is most recently responsible for injecting the email message into the messaging system. The PRA must be determined before you can evaluate whether the message was sent from an SMTP server that is responsible for that domain.

If you already know something about SMTP, your first guess might be that the receiving SMTP server simply examines the inbound SMTP data stream and evaluates the RFC 2821 SMTP conversation. In this portion of the SMTP conversation, the sending SMTP system sends the MAIL FROM and RCPT TO verbs. A simple assumption is that the MAIL FROM verb would contain the valid identity of the sender and, in most cases, that assumption would be correct. However, for organizations that use smart hosts, mail forwarders, mailing lists, or alternate delivery addresses, this assumption does not hold true. Thus, the entire message must be first accepted so that the RFC 2822 portion of the message can be examined. Listing 3.1 shows an example of an SMTP header for a piece that ended up in the spam quarantine.

```
x-sender: bounce+OM_1641909368@mail.classmates.com
x-receiver: lbullock@somorita.com
Received: from relay1.somorita.com ([131.107.2.200]) by
mailserver.somorita.com with Microsoft SMTPSVC(6.0.3790.1830);
    Mon, 20 Feb 2006 15:04:53 -1000
Received: from 65-243-133-26.classmates.com ([65.243.133.26]) by
relay1.somorita.com with Microsoft SMTPSVC(6.0.3790.1830);
    Mon, 20 Feb 2006 15:04:47 -1000
Received: from uurain01.sea2.cmates.com (unknown [10.10.142.201])
    by 65-243-133-26.classmates.com (Postfix) with SMTP id 8ABFE30424
    for <lbullock@somorita.com>; Mon, 20 Feb 2006 17:04:43 -0800
(PST)
From: Classmates <ClassmatesEmail@classmates.com>
To: lbullock@somorita.com
Subject: Lyle, you have 19 visits to your profile
Mime-Version: 1.0
Content-Type: multipart/alternative;
boundary=100839671140483883572.CmatesMail.classmates.com
Envelope-Sender: MAIL
Message-Id: <20060221010443.8ABFE30424@65-243-133-26.classmates.com>
Date: Mon, 20 Feb 2006 17:04:43 -0800 (PST)
Return-Path: bounce+OM_1641909368@mail.classmates.com
X-OriginalArrivalTime: 21 Feb 2006 01:04:48.0214 (UTC)
FILETIME=[CF88DF60:01C63682]
X-SCL: 7 89.02%
```

Listing 3.1: An example of an SMTP header for a piece that ended up in the spam quarantine.

The receiving SMTP system must examine the header of the RFC 2822 message and evaluate the different fields that may indicate the sender's email address. The following RFC 2822 headers are used in this order to determine the PRA:

- Resent-Sender
- Resent-From
- Sender
- From

If these fields are not found in the RFC 2822 portion of the message, the Sender ID system will revert back to the original RFC 2822 MAIL FROM verb that was found in the original message transmission. In the RFC 2822 header that Listing 3.1 shows, the PRA of the message is ClassmatesEmail@classmates.com as determined by the From field in the header.

Validating the Sender's SMTP Server

Now that the purported recipient address of the message is determined, the Sender ID system can determine the SMTP server that was responsible for sending the message to your messaging system. The SMTP header contains all the SMTP hosts that have processed or relayed the message through the sender's organization, the Internet, and your internal network. From the previous example, the SMTP hops looks like this:

```
Received: from relay1.somorita.com ([131.107.2.200]) by
mailserver.somorita.com with Microsoft SMTPSVC(6.0.3790.1830);
    Mon, 20 Feb 2006 15:04:53 -1000
```

```
Received: from 65-243-133-26.classmates.com ([65.243.133.26]) by
relay1.somorita.com with (Mirapoint Messaging Server MOS 3.3.2-
CR)
```

```
with ESMTP id AGC33650;
```

```
    Mon, 20 Feb 2006 15:04:47 -1000
```

```
Received: from uurain01.sea2.cmates.com (unknown [10.10.142.201])
    by 65-243-133-26.classmates.com (Postfix) with SMTP id
8ABFE30424 for <lbullock@somorita.com>; Mon, 20 Feb 2006 17:04:43
-0800
```

For this organization's Sender ID setup, they must configure all the IP addresses of their mail servers. This would include the IP addresses of any server that handles mail on their behalf, such as a mail relay host, SMTP bridgeheads, or managed providers. For Exchange 2003 SP2, this is configured on the General property page of the Message Delivery properties. Figure 3.15 shows the Sender ID and Connection Filter Configuration Settings for somorita.com.

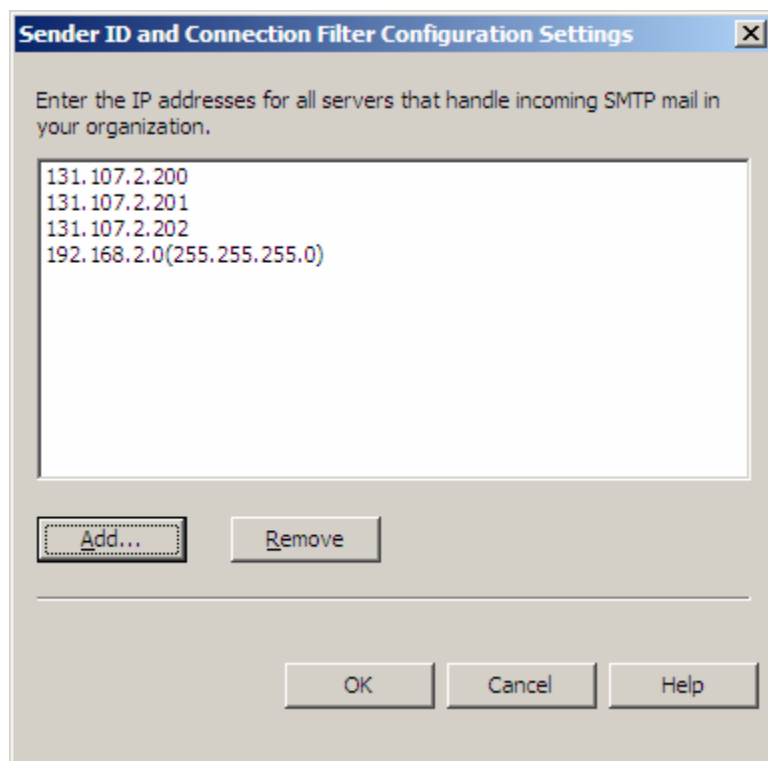


Figure 3.15: Defining IP addresses that may relay SMTP mail internally.

Any SMTP header that contains one of the SMTP addresses shown as a source host will be considered internal and thus not an authorized sender for the message that is being analyzed. Thus, in this case, the first line SMTP header shown in the following example is considered internal because it was received by IP address 131.107.2.200 and that IP address is defined as one of the servers that handles incoming SMTP mail.

```
Received: from relay1.somorita.com ([131.107.2.200]) by
mailserver.somorita.com with Microsoft SMTPSVC(6.0.3790.1830);
    Mon, 20 Feb 2006 15:04:53 -1000
```

The next piece of host information is the following header:

```
Received: from 65-243-133-26.classmates.com ([65.243.133.26]) by
relay1.somorita.com with (Mirapoint Messaging Server MOS 3.3.2-
CR)
    with ESMTPE id AGC33650;
```

This was received from IP address 65.243.133.26. This IP address is not part of the internal mail delivery infrastructure and thus is considered the source address for this message. A quick NSLOOKUP of the sender's domain (classmates.com) shows that they do indeed have SPF records for their hosts. Thus, the Sender ID system will rank this message as being from a validated sender. Ironically, the message was spam, but many organizations that consider themselves "reputable Internet marketers" do have SPF records.

There are several Sender ID status results that might come of the examination of the sending SMTP server and the PRA. Table 3.1 shows the possible results and what this may mean.

Sender ID status result	Meaning
Validation failed - Malformed domain	The domain name for the PRA is not complete or is in an incorrect format.
Validation failed - Non-existent domain	The domain name for the PRA is not a valid domain name in DNS.
Validation failed - Not permitted result	The DNS data that is retrieved does not contain valid information, is not in a valid format, or contains information that cannot be interpreted by this Sender ID system.
Validation with a neutral result	An SPF record exists, but the published Sender ID data is explicitly inconclusive.
Validation with a none result	No SPF record for the sender's domain was found in DNS.
Validation with a pass result	The sender's IP address for the PRA is found in the SPF record.
Validation with a PermError result	A permanent error has occurred such as a malformed or corrupted DNS SPF record.
Validation with a SoftFail result	A "weaker" failure error where the Sender ID system could not determine conclusively whether the sender's IP address was in the SPF record.
Validation with a TempError result	A transient error has occurred such as unable to contact the DNS server.
Message has no PRA	The PRA cannot be located and thus the sending domain cannot be resolved.

Table 3.1: Sender ID status results.

Successful validation of the PRA's Sender ID information will result in the message being ranked as less likely to be spam and failures will contribute to the ranking of the message as being a higher likelihood of being spam. Differing SMTP systems will let you analyze this information differently. Exchange Server does not add any information about the PRA or the Sender ID validation to the SMTP header, but that information is passed on to the Exchange Server and can be viewed with a custom view in Outlook. See <http://tinyurl.com/ga6o8> for more information. Similarly, the SCL of messages can be viewed in the same way; see <http://tinyurl.com/b2p5n> for more information.

On an Exchange Server 2003 SP2 system, you can monitor the summary statistics in the Performance console by looking at the counters found in the MExchange Sender ID object. Figure 3.16 provides a sampling of these statistics.

	SenderId
Total Messages Missing Originating IP	0
Total Messages Validated by Sender ID	8628
Total Messages Validated with a Fail - Malformed Domain Result	0
Total Messages Validated with a Fail - Non-existent Domain Result	142
Total Messages Validated with a Fail - Not Permitted Result	232
Total Messages Validated with a Neutral Result	437
Total Messages Validated with a None Result	6813
Total Messages Validated with a Pass Result	263
Total Messages Validated with a PermError Result	41
Total Messages Validated with a SoftFail Result	553
Total Messages Validated with a TempError Result	147
Total Messages With No PRA	16

Figure 3.16: Monitoring Sender ID validations.

Take note of the most common counter—Total Messages Validated with a None Result. This counter indicates that an SPF record was not found. This is due, in part, to the slow adoption of Sender ID and the fact that many mail systems administrators are simply not creating the necessary SPF records in DNS. In Figure 3.16 almost 80 percent of all the SPF record looks resulted in a record not being found. Clearly, rejecting or deleting inbound mail based on the lack of an SPF record would be a bad idea. However, even if you do not plan to implement Sender ID lookups on inbound mail, you should get your SPF DNS records published.

Digital Signatures

The most reliable and simplest way for a user to verify that a message sender is valid is to request that anyone sending your users' email that requires a higher degree of authenticity use S/MIME digital signatures. Implementing an S/MIME infrastructure for an organization allows users to encrypt sensitive message content and let the recipients of their messages know that the message content has not been altered and comes from the stated sender. The secure messaging system that allows sender verification should have the following characteristics:

- Message origin must be verifiable.
- Message integrity must be verifiable. If the message body or attachments were altered in transit, the user must be notified.
- Message sender must not be able to repudiate the message. A sender cannot later claim that they did not send the message.

Additionally, messages may be required to be private, meaning that only the intended recipient and the sender can view the message content. However, a secure messaging system based on S/MIME does not allow control of the content once an authorized recipient has the content in their possession. To keep the scope of this discussion focused, I will only be discussing the use of S/MIME for digital signatures.

S/MIME digital signatures provide the most reliable and simplest way for end users to verify the authenticity of SMTP mail. Most mail clients intended for business use support S/MIME. This does not, however, extend to Web mail services, which may simply strip out the digital signature.

For a small organization (less than a few hundred mailboxes), deploying an S/MIME solution is fairly simple because you can go to an organization such as Thawte to request a free Thawte Personal Mail certificate. For larger organizations, this will require deploying a public key infrastructure (PKI) that will be sufficient to meet your needs and at the same time be trusted by all your business partners that need to authenticate certificates that you have issued to your users, such as in the case of S/MIME signed email messages.

The Mechanics of Message Signing

On first impression, S/MIME digital signatures are almost indistinguishable from magic. The sender of the message merely clicks an icon or chooses a message option that creates a digitally signed message. The recipient opens a digitally signed message and has some indication on the message form that the message has not been altered and did indeed come from the sender. Figure 3.17 shows an example of a digitally signed message.

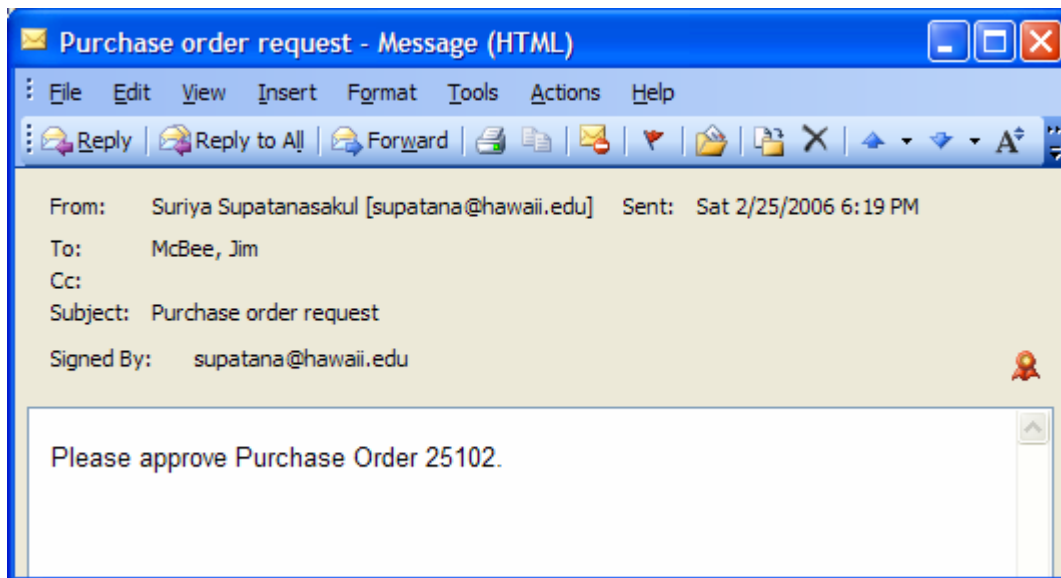


Figure 3.17: A digitally signed message.

The digitally signed message (when viewed from Outlook 2003) shows the SMTP address of the signer on the left side of the message above the message body, and on the right side shows a small red and yellow certificate icon. If the user clicks on the red and yellow certificate icon, the user can verify the digital signature.

If there is a problem with the digital signature (such as untrusted certification authority or an altered message), the red and white certificate icon is replaced with a warning that looks like a red exclamation mark on a yellow background. If the user clicks the warning icon, the user can view the Message Security properties (shown in Figure 3.18).

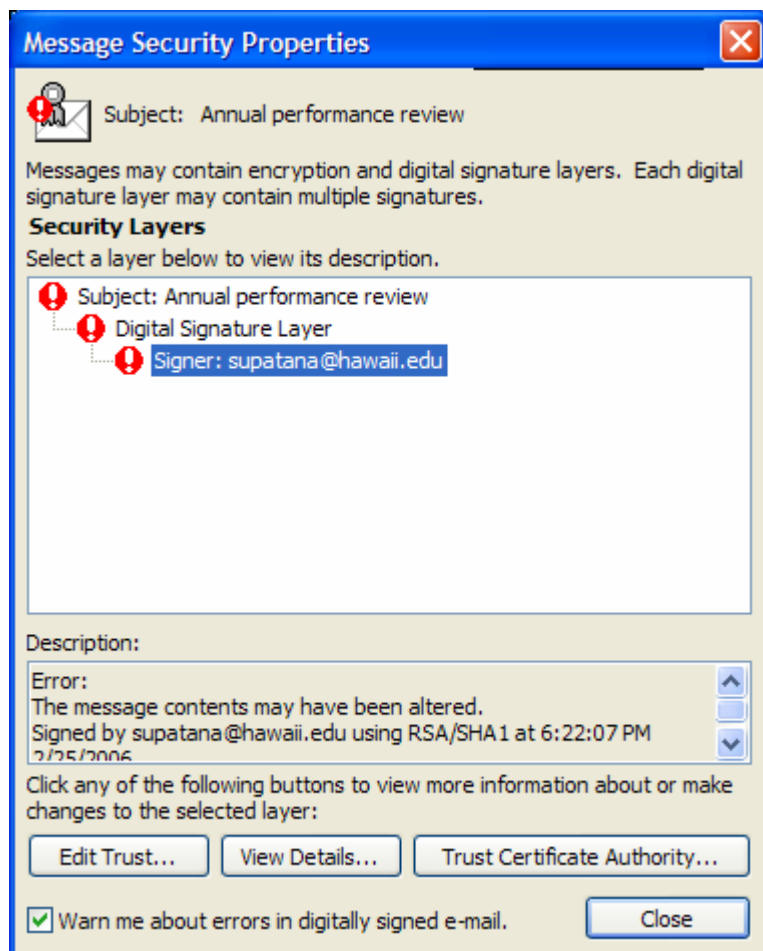


Figure 3.18: Displaying a broken message signature.

Notice in the description field of the Message Security properties in Figure 3.18, the signature indicates the message was signed by the correct SMTP address, but the error indicates that the message may have been altered in transit. The message alteration could be due to someone maliciously changing the message body or attachments after the sender clicked Send or might be due to an SMTP gateway or server process doing something such as appending a text disclaimer to the message after the message was signed.

So that you can appreciate the digital signature process, let's review how a digital signature is calculated and appended to a message. The sender must have an S/MIME certificate that contains a public key and must have a private key associated with the public key. In order for the certificate to be trusted by recipients outside of your organization, the certificate should be signed by a trusted certificate authority.

When the sender prepares a new message, if they have an S/MIME certificate installed on the machine and associated with their mail client, they should see a toolbar button or a message option that creates a digital signature, as Figure 3.19 shows.

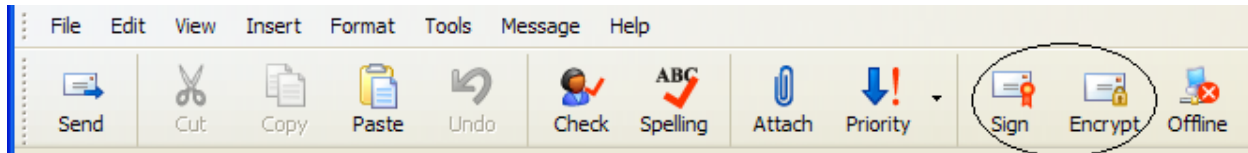


Figure 2.19: Message toolbar that includes message signing and encrypting options.

The user merely clicks the Sign button or icon to assign a digital signature to the message. Depending on the configuration of the machine, the user may be asked to confirm that they are accessing the cryptography API or they may even be required to provide the password that is used to protect their private keys. After that is complete, the message is signed. So what happened when the button was clicked? The following list outlines the process:

- The mail client verifies that the sender has a digital certificate that allows message signing
- The mail client examines the message body and the attachments and calculates either a 160-bit SHA-1 (secure hashing algorithm) hash or a 128-bit MD5 (message digest) hash. The hash of the message cannot be reversed and if even a single bit of the original message is changed, the hash will be different.
- The mail client accesses the sender's private key and encrypts the message hash with the private key.
- The mail client attaches to the message the digital signature, which consists of the encrypted hash and the user's signing certificate (containing the public key).
- The message leaves the client computer and is sent through the message transport system.

This entire process (except for clicking Sign) is entirely transparent to the user. Once the hash is calculated for the message and is encrypted with the sender's public key, the hash cannot be recalculated because the only person that would be able to re-encrypt the hash is the owner of the private key.

Digital signatures are difficult to fake, but if the sender's private key (or the entire computer) is compromised, this is possible. When the recipient receives the signed message and opens it, the mail client automatically verifies the signature. If the message signature is not valid or the sender's certificate is not trusted, the recipient will receive some type of warning either in the message or via a pop-up warning. The following list offers a high-level view of what happens when the recipient opens a signed message:

- The mail client examines the message and determines that the message has been digitally signed. Based on the signing algorithm that was used when the message was signed (either SHA-1 or MD5), the message body and attachments are examined and the message hash is calculated.
- The mail client opens the sender's certificate and decrypts the public key with the hash that was calculated when the message was transmitted. Because the hash was encrypted with the sender's private key, the only key that can decrypt it is the public key.
- The mail client examines the sender's certificate to ensure that the sender of the message is the same as the certificate owner's SMTP address and that the certificate was signed by a trusted certificate authority.
- The mail client compares the hash of the message that it received with the hash of the message that was calculated when the message was transmitted.
- If the message hash is invalid, the sender's SMTP address is not correct, or if the certificate was issued by an untrusted authority, the user is warned.

 Digital signatures apply to more than just email. There are some excellent tutorials on S/MIME and digital signatures. See <http://tinyurl.com/zo5sl>, <http://tinyurl.com/fvwsm> and <http://tinyurl.com/h2fff> for more information.


Topic 4: Protecting and Controlling Sensitive Information in Email

Q 4.3: How do I choose an antivirus software package for Exchange?

A: The process of choosing an antivirus software package is almost universal for all mail systems regardless of whether the mail system is based on Exchange. Almost immediately, I can unequivocally state that no matter how good the antivirus software is that runs on your mail servers or within your perimeter network, this does not exclude the need for antivirus software on all client computers on your network. Thus, if you were thinking that you might not need client-side software, put that thought out of your head.


Exchange Server-Aware Virus Scanning Software

If you are choosing software that will run on the perimeter of your network (such as a Simple Mail Transfer Protocol—SMTP—scanning system in your DMZ), the decision points are going to be almost identical to the decision points for choosing an Exchange-based antivirus software package. The only difference is that the software that runs on your Exchange Server must be Microsoft Exchange AVAPI-aware (antivirus application programming interface). For Exchange 2000, the software must support AVAPI 2.0 and for Exchange 2003, the software must support AVAPI 2.5.

 You should never load more than one Exchange AVAPI-aware antivirus package on an Exchange Server.

Software that uses the Exchange AVAPI is written so that it can open individual mailboxes and scan the messages and attachments found in the mailbox. Further, AVAPI can prevent the user from even accessing a message until it has been scanned. AVAPI 2.5-aware software on Exchange 2003 has been improved so that it can scan messages not only in the information store but also as the messages traverse the SMTP queues. This is useful on bridgehead servers.

The only way to scan messages in an Exchange database is to use an AVAPI-based software package. File system-based virus scanning solutions such as Symantec Antivirus Corporate Edition, Kaspersky Lab's Anti-Virus, F-Prot Antivirus, and so on do not have the necessary data structures defined that would allow them to scan data in the mailbox stores or to clean a virus that they might find in the mailbox store. Thus, if a file system-based virus scanner were to make changes to an Exchange mailbox store, the file would be permanently damaged and require restoration from backup.

 File system-based antivirus scanning software must never be used to scan an Exchange mailbox store, public folder store, or transaction logs. Corruption is almost guaranteed.

Features and Decision Points

When choosing a virus scanning software package, I tend to become a “feature creep.” I have worked in so many environments and have had a number of different requirements for email antivirus scanning software packages both on Exchange Servers and systems that work within the perimeter of your network. Some of the features are more important than others. I have tried to rank my considerations for evaluating an antivirus software package based on what has been most important to the organizations in which I have helped evaluate or implement these packages:

- Configure signature and scanning engine updates hourly or some customizable time interval.
- The ability to use more than one scanning engine and signature set in a single software package.
- The ability to apply file attachment restrictions so that messages carrying certain types of file attachments can be quarantined or deleted. The file attachment list must be customizable based on an organization’s policies.
- Override features so that messages with specific characteristics, such as a specific subject, can be stopped in case of a zero-day attack.
- The ability to control actions for different types of threats, such as the ability to clean certain types of threats but delete mass-mailing viruses, Trojan horses, worms, and so on rather than passing the cleaned message on to the user.
- The use of real-time block list (RBL) functionality.
- Centralized reporting and quarantine management for organizations with more than one mail server or SMTP gateway.
- The ability to control and schedule background scans of mailbox stores.
- Intelligent attachment scanning to determine whether a file has been renamed. This is useful, for example, in the case where you do not allow EXE files but a user or a virus/worm has renamed the EXE to TXT but has instructed the user to save it and rename back to EXE.
- Versatility of notification features such as email notifications and event log notifications. Custom notifications are also a compelling feature if you can customize messages such as notification of rejected attachments.
- Automatic management of quarantine folders (purging quarantined items after so many days) and virus logs.
- Versatility of reporting and monitoring features.
- Allow multiple layers of compressed attachment scanning such as scanning a ZIP file within a ZIP file.

Over the past few years, there has been a convergence of antivirus software packages with other message hygiene solutions such as anti-spam and content inspection systems. Although this convergence certainly reduces the amount of hardware and software that you have to deploy, consider this convergence a convenience but don’t sacrifice the features and functions you need from your antivirus system merely so that you don’t have to deploy as many pieces of software.

Topic 5: Firewall Strategies and Best Practices

Q 5.3: Is Outlook using RPC over HTTP the right solution for my remote users?

A: One of the most compelling features of Exchange 2003 when combined with Outlook 2003 is the ability to use the RPC over HTTP feature. RPC over HTTP enables an Outlook client to encapsulate RPC data inside HTTP frames. The only port that is required to be opened on a firewall is the HTTP or HTTPS ports. That very description helps to clearly delineate exactly who should use RPC over HTTP. RPC over HTTP is intended for Outlook 2003 users that are separated from their Exchange 2003 servers by a firewall such as are shown in Figure 5.6. Opening the necessary RPC ports between the client and the server is considered a security risk.

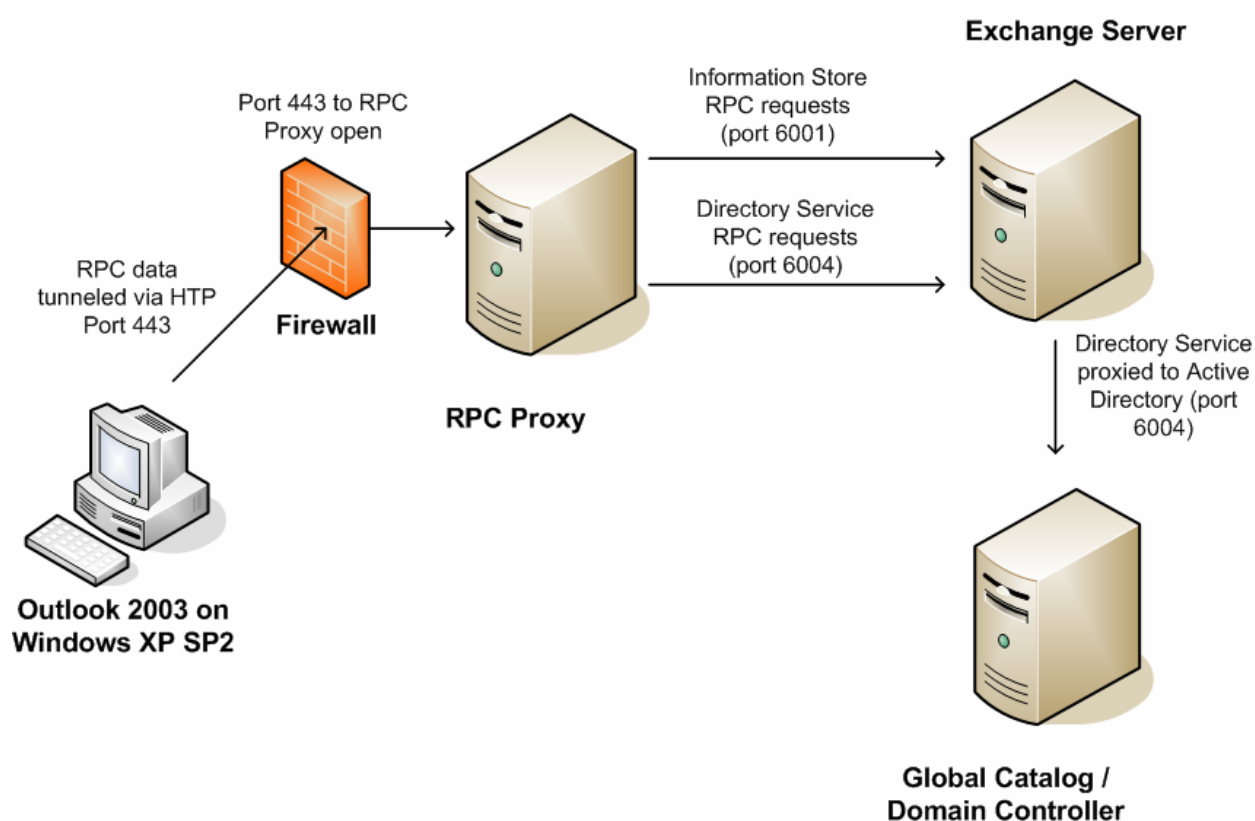


Figure 5.6: Typical RPC over HTTP implementation.

☞ RPC over HTTP can be deployed for Outlook 2003 clients on a local area network (LAN), but it is best suited for clients connecting to an Exchange 2003 Server when separated by a firewall.

RPC over HTTP can greatly reduce the support necessary for virtual private network (VPN) users because a VPN is no longer required for Outlook 2003 users. Once my own organization had implemented RPC over HTTP, this cut my use of the company VPN by 95 percent; I used a full-time VPN connection almost exclusively for Outlook access to the Exchange Server.

Switching users from a VPN connection on their home or laptop computer to an RPC over HTTP connection only to Exchange not only provides users the functionality they require from their mail system but also helps improve security on the internal network. The RPC over HTTP session is only established to an RPC over HTTP proxy and limits the remote client's potential access to other hosts on the internal network. This configuration for remote Outlook 2003 clients potentially also reduces the number of ports that need to be opened on a firewall and thus improves security.

However, a common misconception is that there is less overhead associated with an RPC over HTTP connection than there is a traditional RPC over TCP/IP connection. The amount of data transferred between the client and the server is almost the same.

A possible downside is that configuring RPC over HTTP is slightly more complex than configuring Outlook 2003 to connect directly to the Exchange Server. If the option is either to use RPC over HTTP or a VPN connection, the benefits and reduced complexity will outweigh the additional configuration. To decide whether RPC over HTTP can be implemented for your organization, a couple of factors need to be considered. They include:

- Does your server infrastructure have the minimum software versions required?
- Do the clients have the minimum software versions required?
- Can you deploy reverse proxy solutions for RPC over HTTP to provide an additional layer of protection?

Requirements

Prior to implementing RPC over HTTP, you need to make sure that all the components within your network meet the minimum requirements. The following is a checklist of the minimum requirements for the servers on your network:

- All Exchange Servers must be running Exchange 2003, but I strongly recommend Exchange 2003 SP1 or later due to improvements in configuration. Exchange 2003 must be running on Windows 2003.
- All domain controllers/Global Catalog (GC) servers must be running Windows 2003.
- A server on the inside of your network must be designated as the RPC Proxy server. In a small, single Exchange Server environment, the RPC Proxy service will run on the same server as the Exchange Server. In a large environment, you should use the same server or servers on which the Exchange 2003 front-end servers run.
- The RPC Proxy server must be issued an SSL certificate, and the certificate authority that issues the certificate must be trusted by all client computers.

On the client side, there is also a minimum configuration. The following are requirements for the client:

- Outlook 2003 (recommend Outlook 2003 SP1 or later)
- Windows XP Pro SP1 with the hotfix in KB 331320, though that hotfix is applied in Windows XP SP2.
- The certifying authority that issued the SSL certificate for the RPC Proxy must be trusted by the client.

Deployment Scenarios

If you start putting together the potential permutations of configuration possibilities for RPC over HTTP, you will end up with almost as many combinations as you do organizations that want to deploy it. Figure 5.7 shows a typical deployment of RPC over HTTP for Internet-based clients; this environment can easily be scaled up for larger environments or scaled back for single-server organizations.

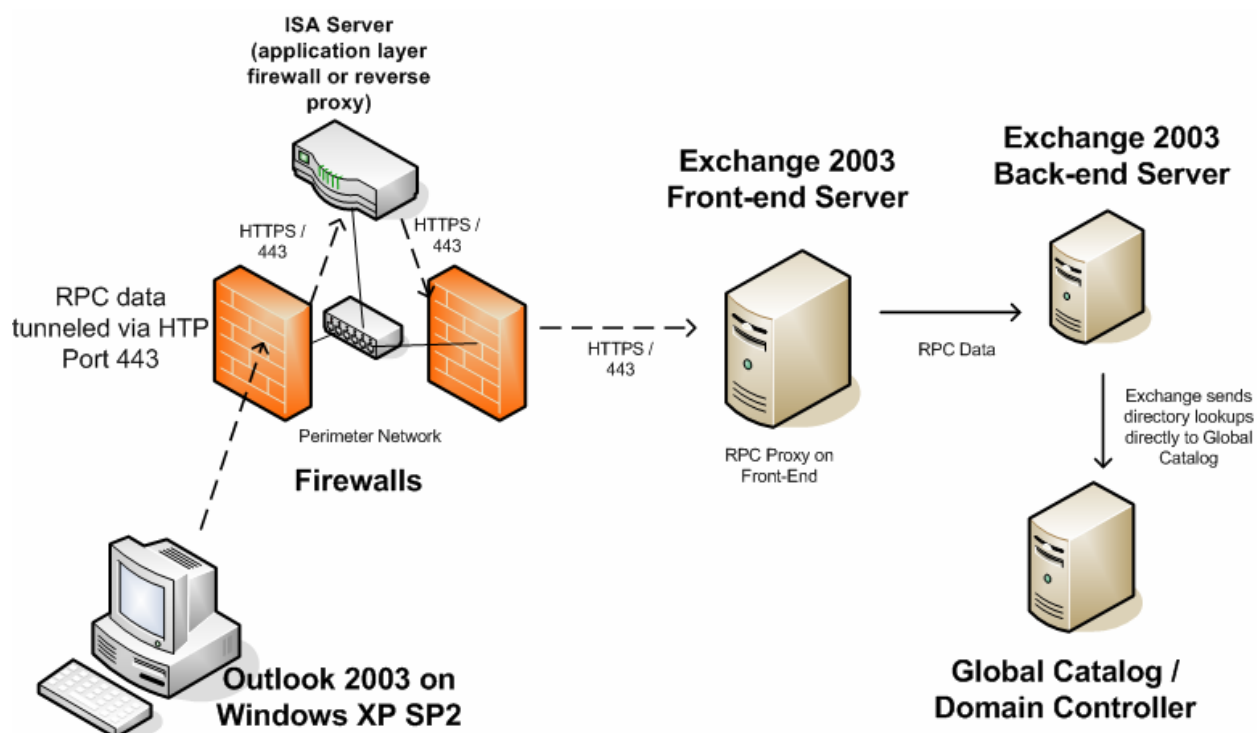


Figure 5.7: Typical deployment of RPC over HTTP with front-end server and ISA Server.

One of the most important security precautions that is in place for the organization in Figure 5.7 is that a reverse proxy server such as ISA Server is used to publish the RPC over HTTP resource to the Internet. Internet clients do not connect directly to the RPC Proxy server running on the Exchange 2003 front-end server; instead, they connect to the ISA Server. The ISA Server authenticates the connection, offloads the SSL overhead, and validates URLs before passing the RPC over HTTP data on to the RPC Proxy server.

Additional security and protection for your internal Exchange resources can be provided by using a reverse proxy server to publish RPC resources to the Internet.

Figure 5.7 can easily be scaled up to larger organizations by merely adding additional ISA Server or reverse proxy servers in the DMZ and configuring load balancing and then adding Exchange 2003 front-end servers on the internal network.

Some organizations don't have the level of complexity on the firewall side of things. Figure 5.8 shows an internal firewall, an external firewall, and an ISA Server inside the perimeter network. There is also a front-end server on the internal network. In a single-server environment, the organization may have a single firewall and Exchange Server. RPC over HTTP can easily be scaled back to this point and still provide reverse proxy security. Figure 5.8 shows how a small organization can use RPC over HTTP just as securely as a large organization.

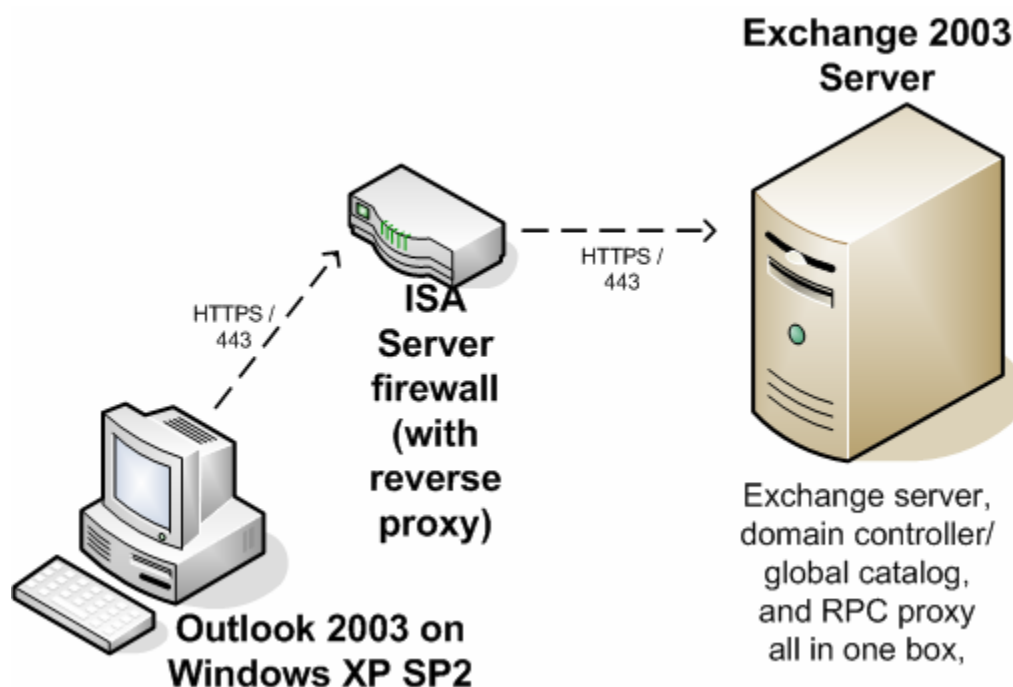


Figure 5.8: Single server and single firewall scenario.


In this scenario, the Exchange 2003 Server also has the RPC Proxy component installed, so only a single server is required.

There are a number of possible deployment scenarios for RPC over HTTP depending on the size of your organization. For more information, read "Exchange Server 2003 RPC over HTTP Deployment Scenarios"; this document can be found at <http://tinyurl.com/26ac7>.


Topic 6: Protecting and Controlling Sensitive Information in Email

Q 6.3: What can Enterprise Rights Management do for my company?

A: Deploying an Enterprise Rights Management (ERM) system offers many benefits for your organization. In the simplest description, an ERM system allows a more comprehensive enforcement of your organization's information security policy. Although there is far more to an ERM than that, this description gives you a good starting point for understanding the benefits of ERM.

 ERM systems help a company enforce their information security policies.

As you may already be aware, email allows for a quick and easy information leak out of your organization whether the leak is intentional or accidental. An ERM solution significantly reduces the likelihood of accidental disclosure and makes intentional disclosure more difficult by limiting what the information consumer can do with the information. Unlike an S/MIME solution that can protect email content, an ERM solution can protect any type of binary content to which an application can be enabled—such as documents, spreadsheets, Web pages, presentations, and PDF files.

 ERM solutions prevent accidental disclosure or sharing of sensitive information. A user must consciously act to subvert information security policy.

For organizations that are affected by laws governing private information or information security, ERM can help ensure compliance with these laws. In the United States, these laws include the Health Insurance Portability and Accountability Act (HIPAA), the Sarbanes-Oxley Act (SOX), the Gramm-Leach-Bliley Act (GLBA), and other regulations that are enforced by the Federal Trade Commission (FTC).

Governments are now enacting “disclosure laws” that cover when sensitive or private information about a company's customers is disclosed. Recently enacted laws such as the California Security Breach Information Act (SB 1386) states that companies must alert customers whenever “unencrypted personal information was or is reasonably believed to have been acquired by an unauthorized person.” Similar laws are now in effect in 23 states and will likely be adopted at the federal level. Best practices and regulations such as NASD 2711 stipulate that investment banking be run separately from research and trading to ensure trust in the public markets. However, unless physical separation is maintained, technologies that improve communications, such as email or portal services, can serve as a conduit of improper communication. This is often referred to as the “Chinese or Ethical Wall” scenario.

ERM solutions are not just for organizations that are concerned with regulatory compliance, though. Any organization that handles sensitive information, proprietary information, or intellectual property can benefit from ERM solutions. Unlike traditional means of securing data—such as firewalls, encryption, and file system permissions—enforcement of an organization's information security policies does not stop at the organization's boundaries. Information can be securely shared with business partners, customers, and vendors; the use of the information can be audited, expired at the end of a project, or superseded by more recent content. To get a better idea of how some organizations have benefited from ERM solutions and how ERM has helped them overcome some of their information security problems, the following sections explore a few ERM scenarios.

ERM and Compartmentalized Information

A government agency operates a classified network that is physically separated from its other networks. However, not all users that use this classified network are cleared to access all the information that is stored, processed, or shared. In the past, email containing compartmentalized information has frequently been sent to distribution lists that included users that are not cleared for that particular classification of information. In some cases, protecting the information is important enough that the email administrator has to rebuild the server and restore data from a backup that was taken prior to the accidental disclosure. Classified information is occasionally (and accidentally) placed on removable media (floppies, CD-ROM, USB drives) and moved to another network. These types of spillage of information are costly in terms of time, resources, and often lost information.

ERM solutions allow this organization's content authors to prepare content that is specific to a particular compartment and apply an ERM template to that content when it is created. This template automatically defines the users that are allowed to consume the content and it defines what their rights to the content consist of, such as print, copy, modify, or forward. Users from other compartments can be temporarily cleared to view specific information without having to give them access to an entire public folder, file share, or Web server.

Confidential Reports to Customers

A consulting company provides confidential analysis of risk assessments, security vulnerabilities, and mitigation recommendations to their customers. The reports contain large amounts of sensitive infrastructure information about the customer's network and computer systems as well as current vulnerabilities and potential threats to the customer. Quite naturally, customers are very concerned that this information does not fall into the wrong hands. The reports also contain proprietary analysis information and methodologies that the consulting company uses to gather and analyze the information. In the past, these reports have been passed along to their competitors despite non-disclosure agreements (NDAs).

Clearly document watermarks and threat-of-legal-menace NDAs were not enough to protect this information. In order to better protect both their customers' sensitive information as well as their own intellectual properties, the consulting company had to implement some mechanism that provides tighter controls for the information they are providing to their customers.

Implementing an ERM system allows the consulting company to publish protected content that can be released to the customer but the consulting company still retains a great degree of control over the content. A consulting report is valid for 60 days from date of issue after which the content expires. All access to each report is audited by the rights management server. Specific recipients at each customer are allowed to read the document, but only one user (usually the person responsible for information security or the Director of Information Technology) is assigned the rights to make printed copies.

The consultancy has found that an organization is much more likely to treat printed copies with greater care when only a single person is responsible for dissemination of the reports. The NDA now includes statements defining the responsibility and disposition of the printed copies of any reports issued to customers.

Preventing Accidental Disclosures of Sales Information

A large reseller of computer equipment has had a couple of embarrassing incidents in which customer pricing and profit reports have accidentally been released to customers by careless email users. In some of these cases, this has resulted in lost sales and lost customers. In one case, the recipient forwarded a customer's pricing data on to a competitor. The company had to implement some type of situation that would protect their sales data and statistics from accidental disclosure.

All "internal only" sales-related content must now be protected at creation. The rights associated with the content are applied via a rights management template so that the rights are applied consistently every time. The rights allow anyone in the sales organization and company executives to review the information, but only the content owners can modify the information or print it out. Content expiration for all proposals are automatically assigned a lifetime of 90 days so that out-of-date information is not used.

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>.