

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 1.2: What is a multi-tier messaging security system?.....1
 Why Have More than One Messaging Security Solution?3
 Early Scanning Systems.....4
 Recommendations for Multi-Tier Messaging Security.....5
 Q 2.2: How do I go about developing an Acceptable Use Policy for email?12
 Starting Point13
 Who Should Be Involved?14
 What Should an Acceptable Use Policy contain?.....14
 Technical Measures15
 Behavior Restrictions and Information Security.....16
 Truth or Consequences17
 Finding More Information17
 Q 3.2: What is the best protection method for Internet clients?17
 Outlook Clients and Remote Procedure Calls19
 SSL Clients20
 Certificates and a Trusted Issuing Authority20
 Messaging Data Protection Moving Forward.....23
 Q 4.2: How are spam, viruses, and worms detected?23
 Virus, Worm, and Trojan Horse Detection.....23
 Signature or Pattern Detection of Known Viruses.....24
 Generic Detection24
 Heuristic Filters.....24

Traffic Analysis	25
Behavioral Analysis	25
Spam Detection.....	25
Real-Time Block Lists	26
Keyword Analysis.....	26
Bayesian Logic.....	26
Sender Authentication.....	27
Other Factors Involved in Spam Detection.....	27
Q 5.2: What are some of the special considerations when Outlook clients are separated from Exchange Server by a firewall?	28
Configuring Exchange Servers and Active Directory for Firewall Support.....	30
Exchange Server Ports	30
Domain Controller Ports	31
Alternatives to Opening RPC Ports	32
Q 6.2: What is Enterprise Rights Management?.....	32
Components of an ERM System.....	35
RMS Enabled Applications.....	35
RMS Client Software	35
RMS Server Software	35
Keys and Certificates	36
Protected Content.....	37

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 1.2: What is a multi-tier messaging security system?

A: Quite simply, a multi-tier messaging security system provides the back-end messaging system with two or more layers of messaging hygiene, including virus scanning, spam protection, and content inspection. Where the original “threat matrix” contained only simple viruses in attachments, now worms, Trojan horses, and phishing schemes threaten email systems and our user community. Not only can unwanted content arrive on your network via your mail server but users can introduce hostile content by doing something as simple as checking their personal Web mail clients.

As message-based threats and potential security breaches via email have evolved beyond simple, hostile attachments, the need for more levels of protection and more evolved protection has emerged. There are a number of places that messaging security can be implemented; Figure 1.3 shows several approaches to implementing messaging security.

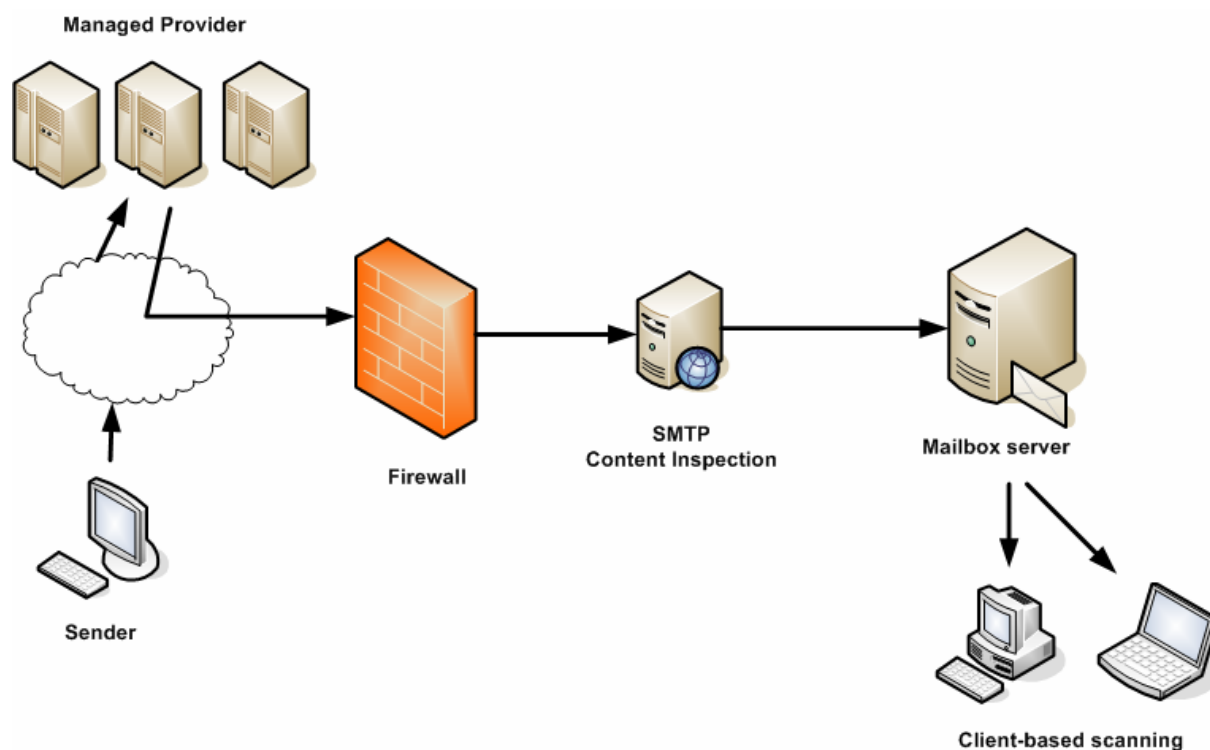


Figure 1.3: Possible components of a multi-tier messaging security system.

As you can see, there are many places where messaging security can be put in place. The whole premise of the multi-tier messaging security system is that you have more than one security mechanism in place.

The **sender's mail system** is the first place where mail security can be put in place. However, any sort of notion that senders on the Internet are implementing good message hygiene is a poor assumption.

A relatively new component of messaging security is the **managed provider**. If an organization uses a managed provider, the organization's inbound mail exchanger (MX) records point to the managed provider's mail servers. Inbound mail is then inspected by the provider's software and passed on to a host within your organization. Managed providers can handle all aspects of inbound message hygiene such as real-time block list (RBL—also known as real-time black-hole lists) lookups and spam, virus, inappropriate content, and day-zero threat detection. In addition, they can do so in an environment that is much more scalable and reactive than even larger organizations can do themselves.

For some organizations, an application-layer capable **firewall** can provide an additional component of messaging security. Some firewalls are capable of performing virus and spam inspection of inbound email as well as performing lookups such as RBL and Sender ID verification of inbound SMTP traffic.

A component that has gained more of the security market is **SMTP content inspection** systems. One of the reasons these are appealing is that regardless of what type of system hosts your mailboxes (Exchange, Notes, UNIX, and so on) all inbound and outbound mail can be inspected by a single platform. These systems can handle all aspects of message hygiene.

Mailbox server message inspection is probably one of the oldest approaches to mail-based virus protection. This solution requires software that is designed to work with the mail system that hosts the mailboxes, as the inspection software must be able to open the messages stores and detect and clean viruses without damaging the message system's databases.

Client-based virus and content scanning is the oldest method of virus protection—some antivirus scanners have been on the market for nearly 20 years. No messaging security system is complete without protection at each client on the network.

Each of these layers of protection serves as a barrier against unwanted messaging content. Unwanted email content—including spam, viruses, worms, and phishing schemes—serve as a disruption to users and to IT resources, and thus cost a business money.



The tangible and intangible costs of a single virus outbreak within a midsized organization's email system can justify the cost of an additional layer of protection.


Why Have More than One Messaging Security Solution?

When most organizations are presented with the idea that they should have more than one mechanism for protecting against viruses and malicious mail content (and certainly mail-based viruses), their initial reaction is to wonder why it is necessary or to argue that their existing systems are sufficient. However, worms such as SoBig, Sober, and Blackmal may have helped to change people's minds about the necessity of multiple layers of security. These worms (often incorrectly classified as viruses) have multiple mechanisms, including an SMTP engine, with which they can propagate. Even if the message did not arrive via the mail server, a user's computer can become infected; these worms will find email addresses on the user's computer and send messages outbound directly from the user's computer (thus bypassing server-based email security).

The lesson is that hostile content can be brought into an organization not only via email destined for an organization's email server but also through other mechanisms. A number of common mechanisms that viruses and content have used to make their way into an organization's network include:

- Users accessing personal Web-based or POP3 email from their work computers
- Hostile content being downloaded from Web pages
- Laptops that have become infected while being used remotely are then returned to the corporate network
- Home computers and remote computers infect the corporate network via virtual private network (VPN) connections
- Extranet connections from business partners can introduce hostile content

It is important to realize that no single form of protection is 100 percent bulletproof. Although most scanning systems today are far more accurate at detecting virus, worms, and other hostile content, relying on any single mechanism may allow something to squeak by. Implementing multiple layers of protection will also help if you use different scanning engines, signature sets, and detection methods. For some types of scanning, such as behavioral analysis, it is better to offload the overhead of virus detection from the mail server or the client platform.

 A reliable multi-tier messaging security system should include client-side scanners as well as some form of perimeter mail protection and mail server scanning.

Finally, for the sake of reducing the overall load of processing on a mailbox server, preventing as much unwanted or dangerous content from arriving on the server itself is valuable. This reduces the resource requirements and the potential for hostile content to actually be accessed by a user.

Early Scanning Systems

The message hygiene industry has seen great convergence over the years. Just 3 or 4 years ago, in order to perform virus scanning, content inspection, and anti-spam activities, you had to purchase two or three separate pieces of software. Figure 1.4 illustrates an example of a healthcare organization's email security system.

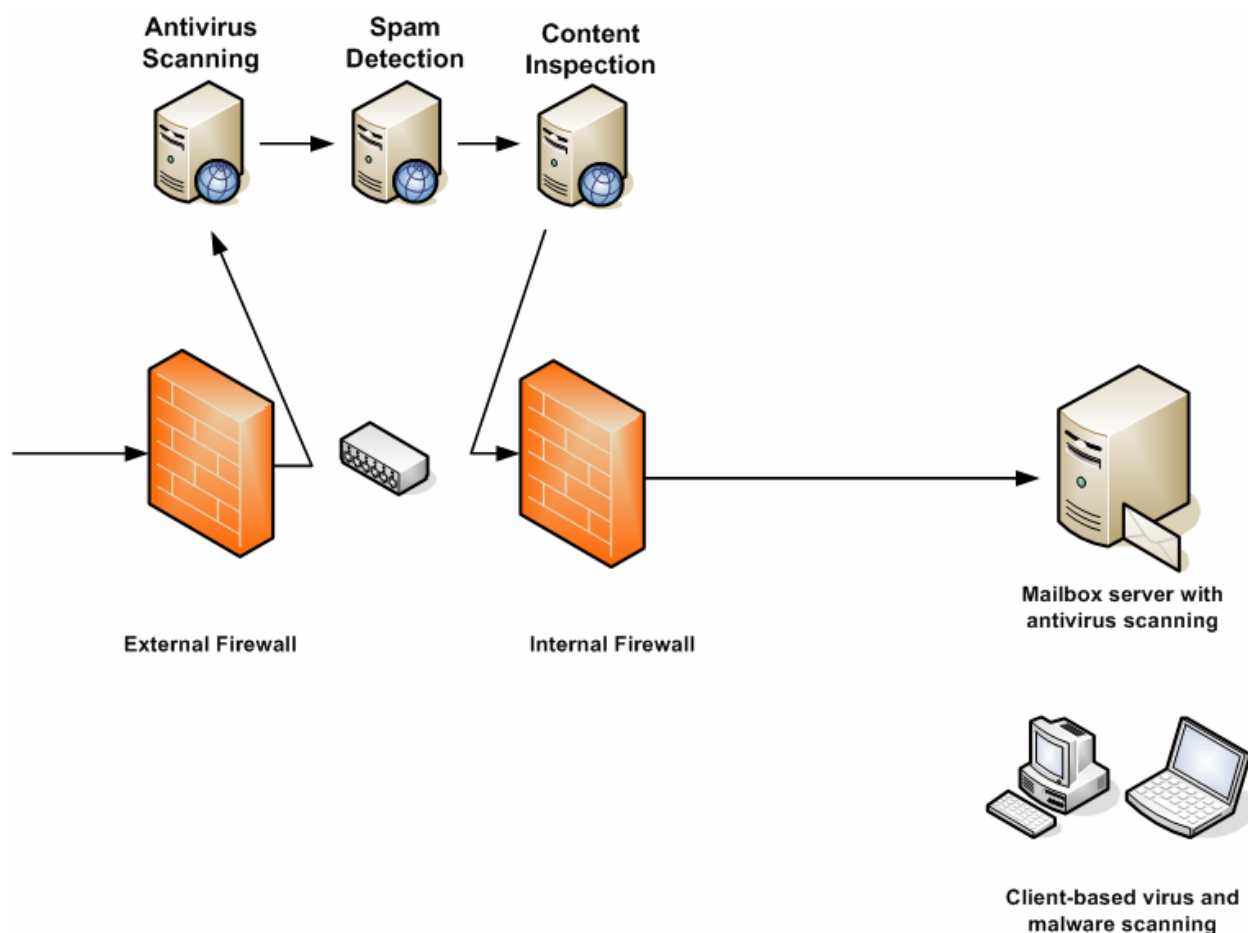


Figure 1.4: Going to extremes with message hygiene.

In the example in Figure 1.4, the healthcare organization built additional layers of protection in to their network one component at a time starting with the email scanning system on their Exchange 2000 mailbox server, and, of course, including virus scanning on the client side. Soon they realized the value of an additional layer of virus scanning, and added an SMTP virus scanning system that performs just virus scanning.

Later, as spam became more of a problem for their users, an additional SMTP gateway was installed whose function was exclusively detection of spam. In the case of this organization, not only was the software from different vendors, but it was installed on separate hardware. The anti-spam system is monitored continually by a Help desk staff member to ensure no false positives are tagged in the spam filter's quarantine. The IT director of this 375-mailbox organization recently noted that monitoring the company's quarantine consumes approximately 4 hours per day of the Help desk's time.

Next came the requirement to perform content inspection in order to meet Health Insurance Portability and Accountability Act (HIPAA) regulations. As neither the spam detection gateway nor the antivirus scanning system were capable of meeting the requirements for scanning inbound and outbound SMTP messages to detect whether messages were meeting HIPAA compliance, a third SMTP gateway was installed. The third scanning system was designed to inspect the content, compare the text of messages against a lexicon of HIPAA terms, and, if necessary, quarantine a message until it could be inspected by an administrator.


All inbound messages that the organization received were sent through three separate SMTP scanning systems prior to the messages being delivered to the organization's Exchange server. This setup introduced complexity, management overhead, and potential single points of failure into the organization's messaging infrastructure.

A current trend is that a product that does a single function (such as just virus detection or only spam prevention) is now the exception rather than the rule. Vendors are now providing products that have multiple capabilities, or vendors are teaming with other vendors in order to provide products that have a single management interface and require a single server platform and perform all necessary message hygiene functions.

Recommendations for Multi-Tier Messaging Security

If you currently have only a single layer of messaging or content security for your organization's email and are looking for a recommendation for how to proceed, you will be relieved to know that implementing multi-tier messaging security is not as complex as it may appear. A solid email protection system really only needs three levels of protection, unlike the organization shown in Figure 1.4 that has three separate SMTP scanning systems or organizations that also implement scanning systems on their firewall.

In the first example, shown in Figure 1.5, the organization has implemented three levels of protection. This organization has chosen to manage their entire mail security system internally. The first layer of defense is an SMTP scanning system located in the perimeter network or DMZ that is capable of not only detecting viruses but also filtering spam. In this case, the organization purchased and manages their own SMTP message hygiene gateway. The organization's MX records should point to this server in the DMZ.

 Firewalls should not only restrict inbound SMTP to authorized hosts but also allow outbound SMTP only from authorized mail servers.

The internal firewall should be configured so that inbound and outbound SMTP traffic (TCP port 25) is restricted to only the mailbox server(s) and the SMTP scanning system in the DMZ. This setup will prevent infected clients from sending infected email directly to the Internet, thereby bypassing the SMTP scanning system and the mailbox server.

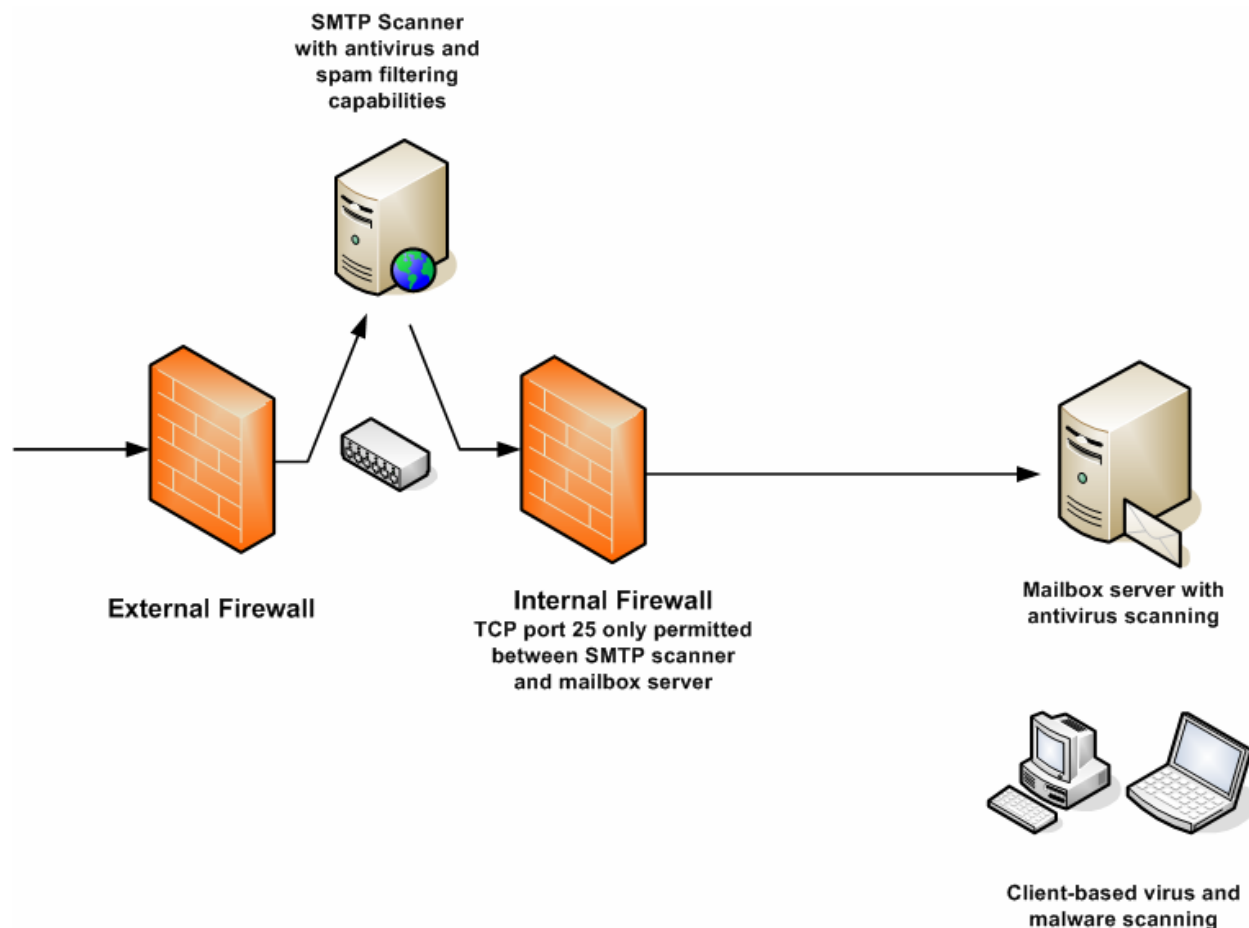


Figure 1.5: Implementing an internally managed SMTP inspection system in the DMZ.

The second layer of defense is the software on the mailbox server that can scan messages in the users' mailboxes as well as scan incoming mail. In the case of Microsoft Exchange, this software would use the Antivirus Application Programming Interface (AVAPI) to access the mail store and examine messages in the queues. Ideally, the software that is running the SMTP scanning system and the software that handles virus detection on the mail server should be from different vendors or use different scanning engines and signatures.

☞ To improve the possibility of detecting all hostile content, the mailbox server's virus scanning software should be from a different vendor or use a different scanning engine than the one handling the SMTP scanning on the perimeter network.

The final layer of security is once again at the client. If you have properly implemented a multi-layer approach to detecting viruses that arrive on your email servers, the chance that a virus will arrive on your clients from your organization's mail servers is very slim. Even if a message containing a virus manages to make it all the way to a user's inbox, the Exchange AVAPI will not allow a client to open a message that has not been scanned. However, in some organizations, users can be exceptionally determined when it comes to finding ways to bring hostile content into an organization; for example, users might configure their mail client to download POP3 or IMAP4 email from external mail servers or open infected email that is on external Web-based mail systems. In either case, the user is bringing content into your organization that has not been scanned by your message hygiene system.

The client computer should have an antivirus client that can not only protect the file system in real-time but also scan mail messages and attachments as they are being accessed by the mail client. Although scanning mail that is being opened from a well-protected Exchange Server is a duplication of functionality, this can serve as an additional layer of protection against viruses or worms that find other ways to enter your organization. Client-side antivirus software should have features such as Symantec Antivirus Corporate Edition's Microsoft Exchange Auto-Protect feature (shown in Figure 1.6). This feature allows the user or administrator to specify how email-based viruses or threats are treated when they are detected. When an email message is opened, any message attachments are immediately scanned even if the attachment has not yet been accessed. Client-side scanning solutions such as Symantec Antivirus can scan not only messages being opened by a MAPI client but also POP3, IMAP4, and SMTP message data streams.

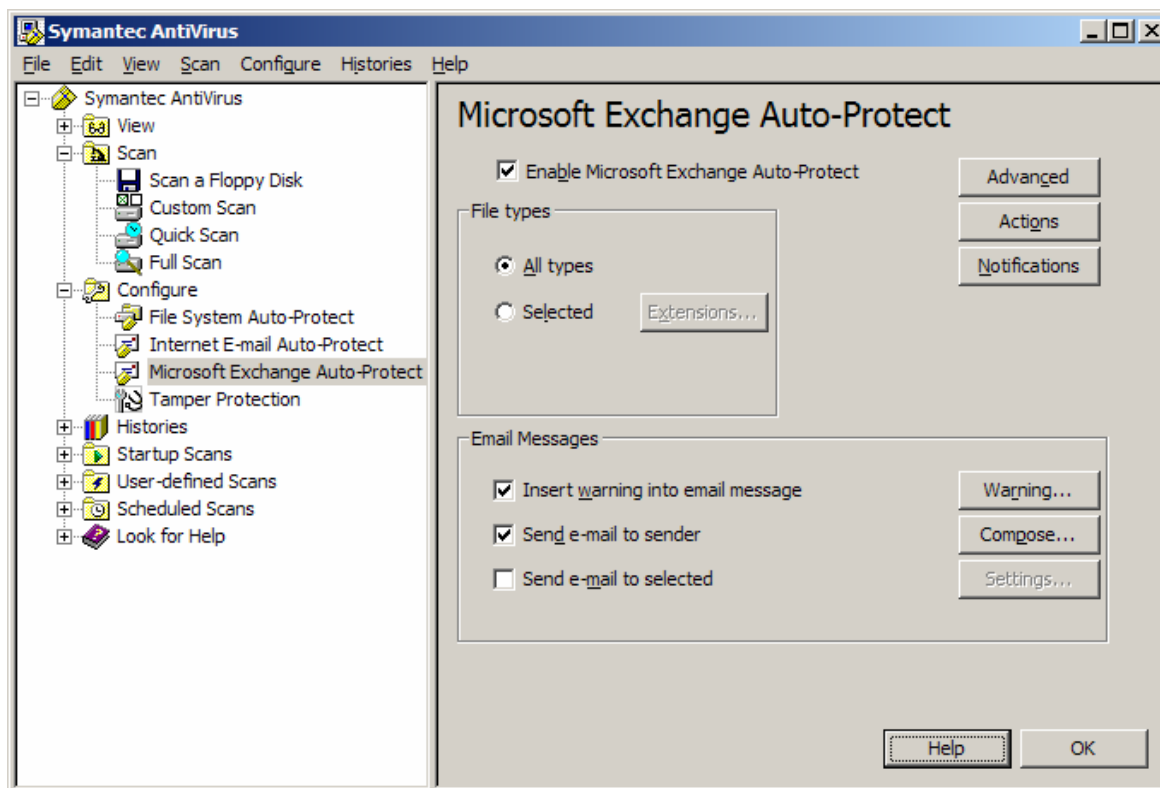


Figure 1.6: Email client protection.

Software that is intended for desktop client MAPI, POP3, IMAP4, or SMTP virus scanning should never be configured to do this type of scanning on an email server.

When implementing a mail hygiene system in the DMZ network, implement and enforce as much as possible your organization's corporate email policy. The perimeter scanning system should be configured to perform operations to protect the mailbox servers including:

- Performing an initial virus scan and blocking inbound messages that contain worms, viruses, and Trojan horses
- Blocking hostile file content (such as programs and scripts)
- Blocking content types that are not allowed (such as MP3, WAV, or MPG files)
- Detecting and eliminating spam and phishing schemes
- Performing Sender ID filtering and lookups against RBLs and known spammer lists
- Deleting or quarantining messages that are infected rather than passing them through to the mail server

The second example of a multi-tier messaging security system is a little simpler than the first but involves many of the same components. Figure 1.7 shows an example configuration in which the organization is using a managed provider as their first line of defense.

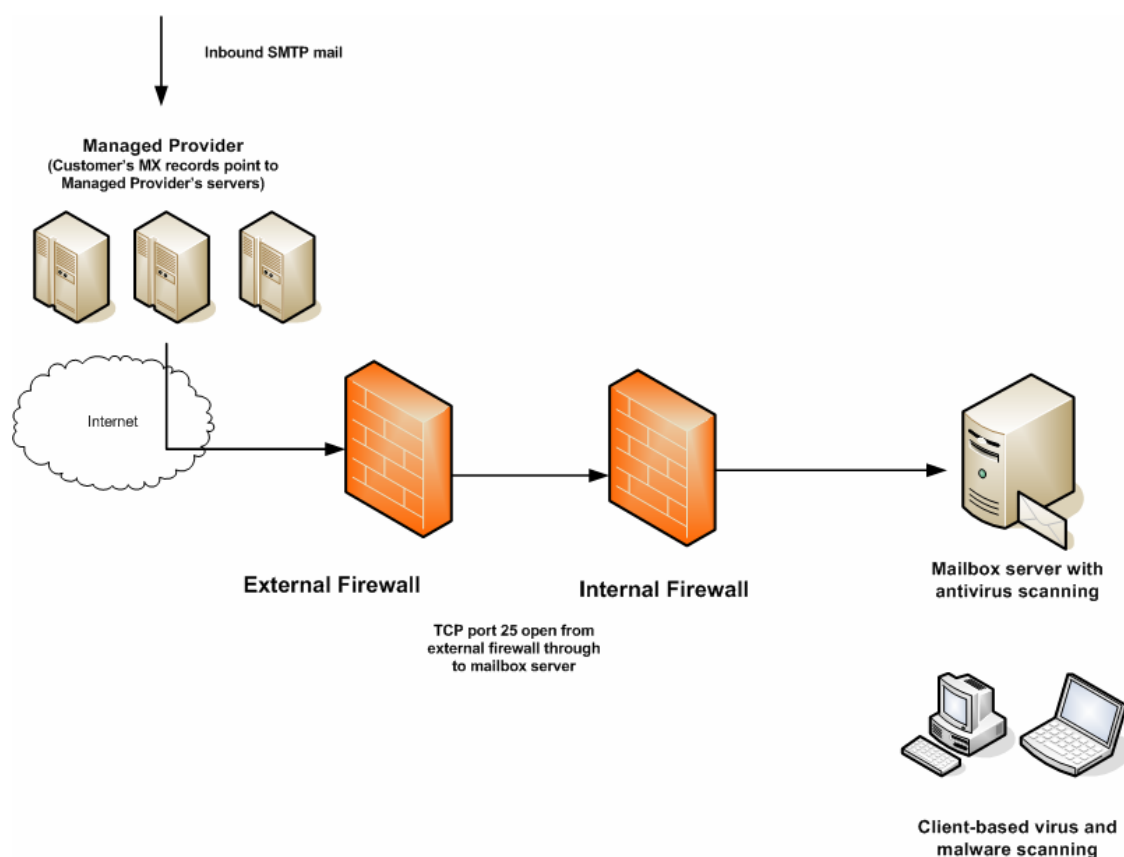


Figure 1.7: Multi-tier security with a managed provider.

The organization's MX records point to the managed provider's SMTP servers, not their own. All inbound mail goes through the managed provider's message hygiene system. The organization can lock down their SMTP ports so that their firewall will only allow inbound SMTP from the managed provider's IP addresses.

This configuration eliminates the need for a perimeter SMTP scanning solution. A managed provider can typically offer much greater reliability and scalability than even a midsized or large business can, and can cost the organization only a few dollars per mailbox. Further, managed providers are staffed and managed to a higher level of availability and can provide immediate reaction and protection against day-zero virus/worm attacks and adapt more quickly to changes in spam and phishing schemes.

☞ When deploying perimeter SMTP scanning solutions, the mail servers should be configured to accept inbound SMTP mail only from the authorized scanning systems.

Although managed providers offer an excellent front-line defense against unwanted mail content, they should by no means be considered an organization's sole line of defense. In Figure 1.7, the organization shown still employs antivirus protection on the mailbox server and the clients are still required to have antivirus software. However, the managed provider will help to reduce the total load on the organization's mail servers.

☞ Managed providers can reduce the unwanted content that makes its way onto your mail servers and reduce resource overhead on the mail servers.

Some analysts attribute 70 percent or more of an organization's email traffic to spam. In addition to placing a tremendous burden on the user community, this amount of spam burdens an organization's messaging system due to extra disk space being consumed, Internet bandwidth being used, and excessive server connections being employed. Excessive server connections due to spam have been exacerbated by the fact that many viruses are simply massive, distributed spam systems. Figure 1.8 shows the inbound SMTP sessions for a small organization (less than 50 mailboxes) that receives a lot of spam daily.

































Current Sessions			
User	From	Connected Time	
 alicedsl.de	222.137.137.56	3349 seconds	
 angelfire.com	80.171.1.10	93 seconds	
 221.212.249.187		61 seconds	
 charlottenc.nefn.com	80.34.39.237	42 seconds	
 furmanselz.com	210.221.57.10	41 seconds	
 sbcglobal.net	220.75.52.226	40 seconds	
 cox.net	88.7.169.173	37 seconds	
 lycos.com	59.16.174.56	37 seconds	
 angelfire.com	221.163.128.184	36 seconds	
 mojozone.co.nz	65.245.103.77	35 seconds	
 moen.com	220.73.2.40	32 seconds	
 biddeford.com	58.234.95.25	32 seconds	
 finh.com	68.148.211.72	31 seconds	
 mitra.com	222.121.145.195	30 seconds	
 maine.rr.com	218.64.70.226	30 seconds	
 visuallink.com	220.169.56.69	24 seconds	
 alltel.net	219.250.200.136	22 seconds	
 netvip.com	222.121.35.130	21 seconds	
 bellsouth.net	200.104.245.14	18 seconds	
 execpc.com	82.38.89.202	16 seconds	
 mindspring.com	222.115.168.103	15 seconds	
 higherreturnmedia.com	64.239.70.32	15 seconds	
 winn-cache-4.server.n...	82.26.30.138	14 seconds	
 lightlink.com	221.150.82.44	13 seconds	
 69.142.58.121		13 seconds	
 ix.netcom.com	219.250.200.136	13 seconds	
 knittingbasket.com	200.77.198.47	10 seconds	
 artware.com	220.81.233.150	10 seconds	
 modeteam-lorette.de	66.108.230.193	6 seconds	
 rrm.co.uk	84.165.125.98	3 seconds	
 adelphia.com	147.47.247.61	3 seconds	
 bellsouth.net	69.141.164.181	3 seconds	

Figure 1.8: Inbound SMTP sessions delivering spam.

Upon first glance, the sessions in Figure 1.8 might appear to be valid sessions. In fact, the only reason the administrator called this into question was that one of the sessions had remained connected for more than an hour. Upon further investigation in the SMTP protocol logs, all these sessions were attempting delivery of random names for that organization. The log showed that there were multiple SMTP conversations from a single IP address. Listing 1.1 shows code that filters out the SMTP conversations from a single IP address.

```

07:22:22 222.65.236.137 emailhut.net HELO - +emailhut.net 250
07:22:22 222.65.236.137 peacemail.com HELO - +peacemail.com 250
07:22:22 222.65.236.137 zalau.ro HELO - +zalau.ro 250
07:22:22 222.65.236.137 vegemail.com HELO - +vegemail.com 250
07:22:22 222.65.236.137 emailhut.net MAIL -
+FROM:+<oralienmell@emailhut.net> 250
07:22:22 222.65.236.137 peacemail.com MAIL -
+FROM:+<christenmae@peacemail.com> 250
07:22:22 222.65.236.137 zalau.ro MAIL - +FROM:+<resther@zalau.ro> 250
07:22:25 222.65.236.137 goowy.com HELO - +goowy.com 250
07:22:25 222.65.236.137 goowy.com HELO - +goowy.com 250
07:22:25 222.65.236.137 vegemail.com MAIL -
+FROM:+<reighnerz@vegemail.com> 250
07:22:25 222.65.236.137 chocofan.com HELO - +chocofan.com 250
07:22:25 222.65.236.137 goowy.com MAIL - +FROM:+<maureen@goowy.com>
250
07:22:25 222.65.236.137 goowy.com MAIL - +FROM:+<marlanas@goowy.com>
250
07:22:26 222.65.236.137 chocofan.com MAIL -
+FROM:+<corine@chocofan.com> 250
07:22:31 222.65.236.137 wildmail.com HELO - +wildmail.com 250
07:22:31 222.65.236.137 wildmail.com MAIL -
+FROM:+<heathlyn@wildmail.com> 250
07:22:38 222.65.236.137 peacemail.com RCPT - +TO:+<yasso@somorita.com>
550
07:22:38 222.65.236.137 zalau.ro RCPT - +TO:+<crump@somorita.com> 550
07:22:40 222.65.236.137 emailhut.net RCPT - +TO:+<roper@somorita.com>
550
07:22:40 222.65.236.137 vegemail.com RCPT - +TO:+<keen@somorita.com>
550
07:22:40 222.65.236.137 goowy.com RCPT - +TO:+<sinclair@somorita.com>
550
07:22:40 222.65.236.137 chocofan.com RCPT - +TO:+<purvis@somorita.com>
550
07:22:42 222.65.236.137 chocofan.com QUIT - chocofan.com 240
07:22:47 222.65.236.137 wildmail.com RCPT - +TO:+<hsgrb@somorita.com>
550
07:22:47 222.65.236.137 goowy.com RCPT - +TO:+<sorensen@somorita.com>
550
07:22:53 222.65.236.137 peacemail.com RCPT - +TO:+<neff@somorita.com>
550
07:22:53 222.65.236.137 zalau.ro RCPT - +TO:+<vogel@somorita.com> 550
07:22:56 222.65.236.137 emailhut.net RCPT - +TO:+<seakmg@somorita.com>
550
07:22:56 222.65.236.137 vegemail.com RCPT -
+TO:+<mpedersen@somorita.com> 550
07:22:59 222.65.236.137 goowy.com RCPT - +TO:+<fish@somorita.com> 550
07:23:02 222.65.236.137 wildmail.com RCPT - +TO:+<mims@somorita.com>
550
07:23:02 222.65.236.137 goowy.com RCPT - +TO:+<swartz@somorita.com>
550
07:23:09 222.65.236.137 peacemail.com RCPT -
+TO:+<worley@somorita.com> 550
07:23:09 222.65.236.137 zalau.ro RCPT - +TO:+<rusba@somorita.com> 550

```

Listing 1.1: Code that filters out the SMTP conversations from a single IP address.

Notice that in each of the RCPT commands, the result was an error code 550 indicating that this address does not exist at this organization (and, in fact, never has). To slow the progress of these “attacks,” an SMTP tar pit was put in place to slow the return of the error codes, but at any given time, this server still has between 10 and 50 inbound SMTP sessions consuming bandwidth and resources. The tar pit can help to slow the amount of data that is sent to the organization’s SMTP server but does not totally eliminate the bandwidth consumed. Managed providers reduce the risks associated with this type of spamming by eliminating worthless traffic from your network connection.


Topic 2: Policies and Procedures

Q 2.2: How do I go about developing an Acceptable Use Policy for email?


A: An Acceptable Use Policy for your organization can span many different technologies and functions within an information system. This tip will focus on email systems and email communications.

Short of lawyers, few people appreciate the power of the written word. Email has evolved in most organizations from being an informal mechanism for communication in a business to a critical tool for doing business. Few people would pass around a tasteless joke printed in an official memorandum stationary or share their personal music via company courier. However, the casual approach to the use of email has resulted in legal problems for both individuals and entire companies as a result of the endless stream of file sharing, inappropriate jokes, and other objectionable material that now passes through many mail systems.

An Acceptable Use Policy must be part of the core foundation that you build for your messaging system’s operations. Defining acceptable and unacceptable use of your messaging system helps to define the expected behavior of the user community, sets the user’s expectations, helps with capacity planning, protects your organization from legal problems, and defines consequences if a user does not follow the usage guidelines.

 Get legal advice prior to publishing an Acceptable Use Policy.


Ultimately, the Acceptable Use Policy should be documented, put into place, and be acknowledged by the user community in writing once a system is implemented. Certainly, an Acceptable Use Policy should be in place before restrictions are placed on the users that were previously not in affect (such as message size limits or banned attachment types). One surefire way to alienate your user community is to take away capabilities that they once had without first warning them.

 Users do not like to have functionality taken away from them. Unless handled well and thoroughly explained, doing so will alienate your user community and possibly create a rift between IT and the user community.

Starting Point

Sometimes the most difficult part of developing an Acceptable Use Policy is just getting started or realizing that your organization needs to define guidelines for your user community. A good starting point is to define an outline of what types of technical restrictions you need to place on the user community, what measures you need to put into place to improve security, and what types of behavior or actions on the system are considered unacceptable.


Next, pick a team of people to review the policy, provide feedback, and ultimately approve its use. The team (including Human Resources and a legal advisor) should measure the enforceability of the policy.

 An Acceptable Use Policy should cover and apply to everyone in an organization. It should not provide exceptions for management or exclude departments or divisions.

Finally, and probably most difficult, is getting the policy in front of the user so that they review it and acknowledge that they understand it. This step can be accomplished in a number of ways:

- During an employee's annual performance review
- After training for use of the system
- When receiving a user name and password to access the system

Keep in mind that if your organization spans multiple states, provinces, or countries, laws governing employee privacy or work regulations may be different. In addition, there may be cultural and social differences that you need to take into consideration.

 An Acceptable Use Policy must be "acceptable" to your user community while still protecting your organization from liability and system misuse.

Keeping the policy clearly written and self explanatory is important so that users have an understanding as to why the policy is essential to the organization's stability and security.

Who Should Be Involved?

Picking a team of people to help define the contents and enforceability of an Acceptable Use Policy will help to define the success of the policy and whether it must be revised shortly after it is put in place or stands the test of time (at least a year or two). The team should include users from across departments and specializations in your organization:

- The Acceptable Use Policy must have an executive-level sponsor; this is someone from the senior management team that helps to approve the policy and agrees that it speaks for “the company” and not just the IT group. Otherwise, the policy may not be enforceable.
- The Human Resources department’s input is vital for the development of this policy. HR will more than likely be responsible for collection of signed documents saying that users have reviewed the policy, and HR will ultimately be involved in any type of action against users that do not follow the policy.
- The IT group should provide at least one team member, and this team member should be familiar with what types of technical enforcement are possible given current technologies. Other considerations that the IT representative should be able to provide for the group include potential technical enforcement mechanisms that can be implemented if necessary.
- The team should include someone from the organization’s legal team or from an organization that represents the company’s legal interests. This individual must help guide the development of the Acceptable Use Policy in a direction that will keep it within the boundaries of the law, ensure that the organization is not opening itself up to lawsuits, and that the policy provides guidance for any regulatory requirements.
- Representatives from the end user community and each major department or division of the company should be involved. The policy should be clear enough that all departments understand the importance of following the policy’s guidelines. Guidance from the user community can help make the policy understandable and hopefully help the users to understand why it is important.

What Should an Acceptable Use Policy contain?

Deciding on the text for your Acceptable Use Policy is an interesting balancing act. On one side of the equation is the need to keep the policy readable and understandable; on the other side is the need to ensure that the user is adequately educated about the restrictions that need to be put in place. Although the actual policy may not be divided into behavioral, information security, and technical measure restrictions, it is easier to examine them in that regard.

Technical Measures

The technical portion of the Acceptable Use Policy is the portion that defines the system restrictions placed on the user by either the mail system, perimeter filtering, or other mechanisms. Users do not have much control over technical measures; nonetheless, it is important that the user understand why these measures are in place. Technical measures include:

- Mailbox size limits (including the size at which the server will start rejecting mail for the mailbox)
- Message size limits internally or to and from the Internet
- Forbidden message attachment types including attachment types that are blocked; if you block compressed files (such as ZIP or CAB files), the user should know what to do if they must send or receive a file of that type; if an attachment is blocked, state whether it is deleted or quarantined
- Anti-spam technologies in place and the disposition of detected spam (whether it is passed to the user's junk email folder, quarantined, or deleted entirely)
- Antivirus technologies in place and whether messages with viruses are cleaned and passed to the user, quarantined, or deleted
- Use of system-enforced disclaimers for outbound messages
- Automated mailbox cleanup processes such as moving old messages to an alternative folder or purging deleted items that are older than a specified age
- Message archival technologies in place and what criteria (age, size, content, job function) are used to archive messages as well as how to retrieve them
- Message formats (HTML, rich text, plain text) that are permitted for inbound and outbound messages
- Content scanning systems that are in place and the type of content that may be blocked or quarantined for further examination, including jokes or inappropriate content, as well as information such as protected health information, customer records, financial data, or company proprietary information

Behavior Restrictions and Information Security

The behavior portion of the policy defines the factors that the user actually has some control over. This portion the policy is also the part that some users will violate. Coverage should include:

- Outlining your organization's policy on written communication with respect to sexual harassment, discrimination, inappropriate remarks, threatening remarks, pornographic material, and other forms of media and ensure that the policy clearly defines that email communication is a form of written communication.
- Defining the types of messages that a user is allowed to send, including whether users are allowed to use their work email system for personal messages, and covers sending chain letters, jokes, urban legends, and other personal or non-work related content. Many organizations take an official stance that "personal email" is not allowed but ignore personal email use as long as it is not excessive.
- Defining levels of confidentiality and exactly what types of information a user is allowed to send internally and externally. If there are regulatory restrictions on content (such as healthcare information or non-public customer information), define how information may be sent (such as via encrypted emails).
- Outlining acceptable distribution of work email addresses. For example, work email addresses should never be used to register for online contests, to access news sites, for magazine subscriptions, and for greeting cards, as these are an invitation to receive spam.
- Defining expectations for what a user must do when they receive suspicious email, such as messages with attachments when they don't normally receive attachments from the sender.
- Outlining whether mailboxes may be inspected by the IT department; if so, define under what conditions it can happen, such as by order of the Human Resources department.


Mailbox Surfing

Over the years, I have heard about or helped investigate situations in which an organization's mail administrator has decided to go mailbox surfing. In these cases, the administrator (without authority or knowledge of the user) reads through people's email and sent items.

This is a serious breach of ethics and an organization's management should treat this very seriously. Although most users understand that their email can be read by their IT department, they generally understand that it will happen only if necessary or authorized. If your IT department has not had at least a briefing on computing and ethics, this is something you should look into. The Association of Information Technology Professionals has a sample code of ethics at <http://www.aitp.org/organization/about/ethics/ethics.jsp>.

Truth or Consequences

Finally, the Acceptable Use Policy must have teeth—there must be realistic consequences when the policy is violated and those consequences must be enforceable. In order for the policy to be enforceable, you must have executive-level sponsorship when the policy is put into place. When the first official “punishment” occurs, management must have the intestinal fortitude to stand by that punishment.

 The Acceptable Use Policy your organization uses must be realistic and enforceable.

In addition, the punishment must fit the crime; if a user is found to have been sending inappropriate pictures, the policy must define a realistic punishment. A realistic punishment might be that person gets a written reprimand in their HR file.


Finding More Information

Each organization is slightly different with a different corporate culture, security requirements, and management perspective. Thus, a policy that works well for one organization may alienate or anger the users in another. The SANS Institute has good information on security policies as well as sample policies and templates. You can find these at <http://www.sans.org/resources/policies>. Another resource is the Electronic Frontier Foundation’s Academic Computing Policy Statements archive at http://www.eff.org/Censorship/Academic_edu/CAF/policies. The Connecticut State Government publishes their Electronic Mail Acceptable Use Policy on the Internet; it can serve as a good guideline and can be found at <http://www.cmac.state.ct.us/policies/emailcon.htm>.

Topic 3: Architecture and Deployment Considerations

Q 3.2: What is the best protection method for Internet clients?

A: It is a popular misconception that email or Web traffic can be easily intercepted when traveling from mail server to mail server or from a browser client to a Web server. For someone interested in compromising your messaging data during data transmission, they have to be sitting in the path of the IP datagrams. Although this does occur, the intruder must have network monitoring equipment either on the client’s network, the server’s network, your corporate network infrastructure, or at an ISP that provides the Internet connectivity. Consequently, this does not happen very frequently, and people that want to intercept sensitive data such as credit card data or other information have to resort to schemes such as phishing or social engineering.

 Encrypting the data stream between a client and a server does not mean that the data is also encrypted when it is stored on the server or at the client.

Protecting your messaging data during transmission is still important in order to provide an additional layer of security. This is certainly true if your corporate information crosses public networks or networks to which you do not have control of the physical access to the network media. For some organizations, such as those affected by regulations such as the Health Insurance Portability and Accountability Act (HIPAA), an organization may be bound by law to encrypt some types of data.

☞ Data encryption on your network may be required if you must meet certain regulatory requirements.

As an experienced network or messaging administrator can tell you, if someone manages to access the network infrastructure through which your messaging data passes, it is a very simple matter to intercept clear-text data that is transmitted by protocols such as HTTP, POP3, IMAP4, and SMTP. Even intercepting user names and passwords can be deceptively simple. On first glance, the Outlook Web Access (OWA) logon session that Figure 3.4 shows appears to be secure.

The screenshot shows the Microsoft Network Monitor interface with a capture of two frames. The first frame (Frame 40) is an HTTP GET request from the client. The second frame (Frame 41) is the response from the server. The response includes an HTTP: Authorization header with basic authentication credentials.

Frame	Time	Src MAC Addr	Dst MAC Addr	Protocol	Description	Src Other Addr	Dst O
38	30.265625	LOCAL	HIIAKA	HTTP	Response to Client; HTTP/1.1; Status Code =...	KILAUEA	192.1
39	30.402343	HIIAKA	LOCAL	TCP	Control Bits: .A...., len: 0, seq:252272...	192.168.254.54	KILAU
40	48.511718	HIIAKA	LOCAL	HTTP	GET Request from Client	192.168.254.54	KILAU
41	48.608398	LOCAL	HIIAKA	HTTP	Response to Client; HTTP/1.1; Status Code =...	KILAUEA	192.1
42	48.760742	HIIAKA	LOCAL	TCP	Control Bits: .A...., len: 0, seq:252272...	192.168.254.54	KILAU
43	52.271484	HIIAKA	LOCAL	TCP	Control Bits: .AP..., len: 56, seq:190069...	192.168.254.54	KILAU
44	52.358398	LOCAL	HIIAKA	TCP	Control Bits: .AP..., len: 2712, seq:255597...	KILAUEA	192.1
45	52.358398	LOCAL	HIIAKA	TCP	Control Bits: .AP..., len: 1847, seq:255597...	KILAUEA	192.1

Expanded view of the HTTP response (Frame 41):

```

TCP: Control Bits: .AP..., len: 487, seq:2522723534-2522724021, ack:3441444663, win:17200, src: 3887 dst: 80
HTTP: GET Request from Client
  ...HTTP: Request Method =GET
  ...HTTP: Uniform Resource Identifier =/exchange/
  ...HTTP: Protocol Version =HTTP/1.1
  ...HTTP: Accept = image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel
  ...HTTP: Accept-Language =en-us
  ...HTTP: Accept-Encoding =gzip, deflate
  ...HTTP: User-Agent =Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1
  ...HTTP: Host =kilauea
  ...HTTP: Connection =Keep-Alive
  ...HTTP: Authorization =Basic dm9sY2Fub3NlcmZiXGNrYU1peWE6JHVwZXIkdWMyZXQxMjM=
  
```

Expanded view of the raw data (Frame 41):

```

00000140 20 4D 6F 7A 69 6C 6C 61 2F 34 2E 30 20 28 63 6F Mozilla/4.0 (co
00000150 6D 70 61 74 69 62 6C 65 3B 20 4D 53 49 45 20 36 mpatible; MSIE 6
00000160 2E 30 3E 20 57 69 6E 64 6F 77 73 20 4E 54 20 35 .0; Windows NT 5
00000170 2E 31 3E 20 53 56 31 3B 20 2E 4E 45 54 20 43 4C .1; SV1; .NET CL
00000180 52 20 31 2E 31 2E 3A 33 32 32 3B 20 49 6E 66 6F R 1.1.4322; Info
00000190 50 61 74 68 2E 31 3B 20 2E 4E 45 54 20 43 4C 52 Path.1; .NET CLR
000001A0 20 32 2E 30 2E 35 30 37 32 37 29 0D 0A 48 6F 73 2.0.50727)
000001B0 74 3A 20 6B 69 6C 61 75 65 61 0D 0A 43 6F 6E 6E t: kilauea
000001C0 65 63 74 69 6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 ection: Keep-Alive
000001D0 76 65 0D 0A 41 75 74 68 6F 72 69 7A 61 74 69 6F ve
000001E0 6E 3A 20 42 61 73 69 63 20 64 6D 39 73 59 32 46 n: Basic dm9sY2F
000001F0 75 62 33 4E 31 63 6D 5A 69 58 47 4E 72 59 57 31 ub3NlcmZiXGNrYU1
00000200 70 65 57 45 36 4A 48 56 77 5A 58 49 6B 5A 57 4E peWE6JHVwZXIkdW
00000210 79 5A 58 51 78 4D 6A 4D 3D 0D 0A 0D 0A yZXQxMjM=
  
```

Figure 3.4: Capturing basic authentication credentials

The HTTP header authorization identifies the authentication method as basic authentication. The authorization string is `dm9sY2Fub3N1cmZiXGNrYW1peWE6JHVwZXIkdWVwZXQxMjM=`. This is not an encrypted authentication string, but simply encoded using Base64 encoding. Decoding this string shows the string: `volcanosurfb\ckamiya:Super$ecret123`. The user's domain name is `volcanosurfb`, the account is `ckamiya`, and the password is `Super$ecret123`. A good password compromised by basic authentication.

Outlook Clients and Remote Procedure Calls

Connecting directly to an Exchange Server system using an Outlook client over the Internet is not the most common configuration. Remote Outlook clients usually use either a VPN connection or Remote Procedure Calls (RPCs) over HTTP; however, many organizations still use a standard RPC connection to connect to an Exchange Server remotely. Although RPCs are not easily readable using network or protocol monitoring tools, the data can still be captured and decoded by a skilled person.

Outlook and Exchange Server can take advantage of the built-in encryption capabilities provided by the Windows RPC functions. For client-server communication, this functionality is not enabled by default, but it can easily be turned on via the Outlook profile. On the Security property page (see Figure 3.5) of the Microsoft Exchange Server connection properties, simply select the *Encrypt data between Microsoft Office Outlook and Microsoft Exchange Server* check box. When Outlook is restarted, all RPC traffic between Outlook and the Exchange Server will be encrypted using 128-bit streaming encryption.

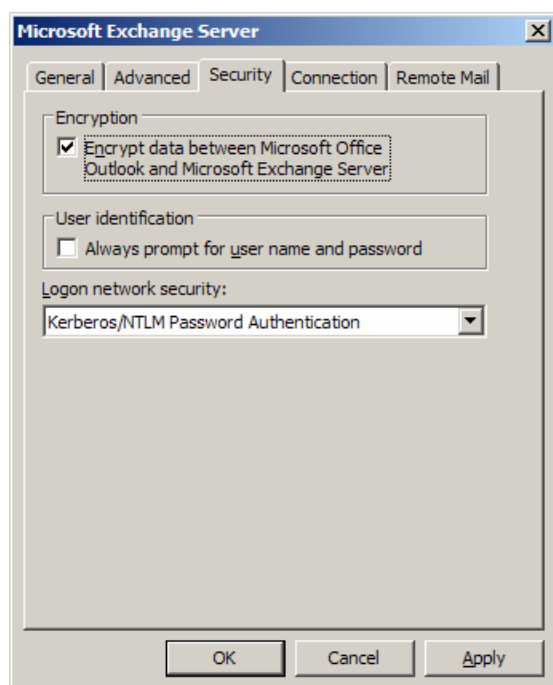


Figure 3.5: Enabling Outlook-to-Exchange encryption.

Enabling Outlook RPC encryption is a good idea if you are concerned about messaging data being captured and viewed within your organization's network.

SSL Clients

For POP3, IMAP4, OWA, ActiveSync, and RPC over HTTP clients, using Secure Sockets Layer (SSL) is the best approach to protecting the messaging data being transmitted over the network. SSL is an application-layer technology that provides encryption for client-server applications that do not handle encryption themselves. SSL was originally developed by Netscape, but it has become a de facto standard for client-to-server security for Web applications.

The most common application of SSL is found with Web browsers communicating with a Web server and can be identified when the URL is preceded by an HTTPS rather than just HTTP. However, this technology can be extended to other TCP/IP applications including POP3 or IMAP4 clients.



For more information about SSL and the evolution of its sibling Transport Layer Security (TLS), see http://en.wikipedia.org/wiki/Secure_Sockets_Layer.

Enabling SSL requires a certificate be installed on the server. For HTTP-based applications, this certificate is installed using the Internet Information Services administrator program. Installing the certificate through IIS Admin allows OWA, ActiveSync, and RPC over HTTP client traffic to use SSL because all three of these functions use the Web server. To protect POP3 or IMAP4 clients, the same certificate can be used; simply export a copy of the certificate using IIS Admin, then import it for the POP3 or IMAP4 virtual server using Exchange System Manager. As long as the fully qualified domain name (FQDN) being used for the HTTP certificate does not change for POP3 or IMAP4 clients, the certificate will remain valid.

Certificates and a Trusted Issuing Authority


Generating a certificate is deceptively easy. For an experienced Windows administrator, this task is as simple as installing the Microsoft Certificate Server and then requesting and issuing as many certificates as you please. Most small and midsized organizations have neither the resources nor the time to plan and correctly implement a certificate authority (CA) infrastructure; thus, at least for limited certificate issuance, they should consider using a trusted authority.

Browser clients will not “trust” certificates issued by an unknown CA because the browser does not have the issuing server’s certificate. A Web browser client will receive a pop-up similar to the one in Figure 3.6 indicating that the certificate was issued by a company you have chosen not to trust.



Figure 3.6: When a certificate is issued by an unknown certificate authority, an alert pops up.

If you simply click Yes in the Security Alert dialog box, the SSL connection will be established anyway. At least this works for Web applications; for Outlook using RPC over HTTP, the connection will not be established and there will be no warning.

 See Microsoft Knowledge Base article 555261 “To use Outlook 2003 RPC over HTTPS your client PCs must trust the root certificate” for more information.

This problem can be remedied by installing the issuing certificate server’s certificate as a trusted issuing authority on any client that may establish an SSL session using a certificate issued by that server. This is not difficult to do internally because trusted certificates can be deployed to all computers through a Group Policy Object (GPO) or other automated means. However, if the certificate is used by external clients, they must install the certificate on the computer on which they are currently working.

Of course, Web browser clients can be told simply to ignore the Security Alert, click Yes, and continue. This practice is not a good idea—asking your users to get into the habit of ignoring security warnings will sooner or later lead them to ignore a valid warning.

There are several CAs on the Internet that are trusted by most browser clients. In Internet Explorer (IE), you can view a list of the Trusted Root Certification Authorities and the Intermediate Certification Authorities by selecting Tools, Internet Options, Content, Certificates. In the Certificates dialog box, which Figure 3.7 shows, you can see a list of the Trusted Root Certification Authorities and the Intermediate Certification Authorities. For Windows power users, you can see the same list using the Certificates management console.

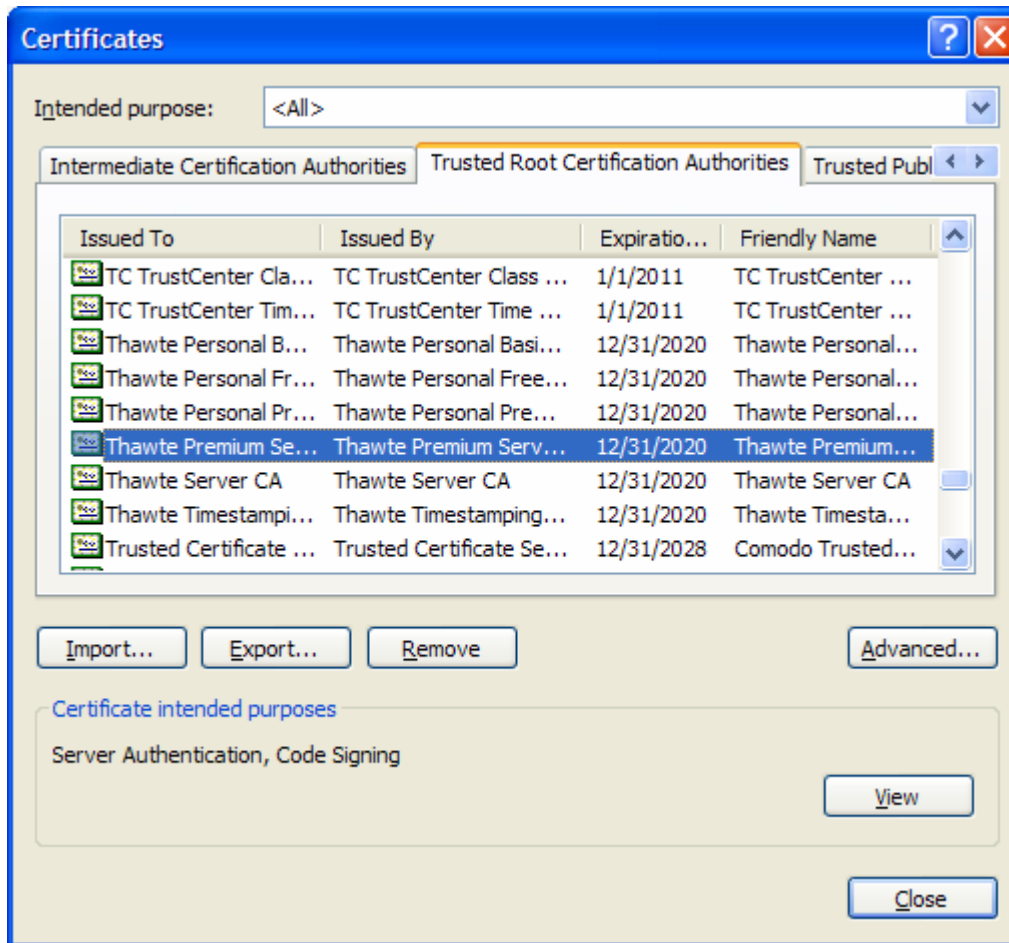


Figure 3.7: Trusted Root Certification Authorities in IE.

Getting a certificate that is issued by a trusted authority is pretty simple. Simply generate a certificate signing request (CSR) using IIS Admin and send it to the CA of your choice. For certificates that are widely trusted and provide better liability protection (such as for e-commerce sites), authorities such as VeriSign and Thawte are a good choice. If you simply need to provide SSL for data protection, though, there are a number of lower-cost alternatives including InstantSSL/Comodo and GoDaddy.

Messaging Data Protection Moving Forward

As mentioned earlier, just because the data stream of an email message is encrypted on the network using technologies such as SSL/TLS, IPsec, or RPC encryption, it does not mean that the data is completely secure or that the person that currently has possession of the data cannot forward it to unauthorized users. Regulatory requirements for data protection are often vague on exactly what is required to protect information. HIPAA, for example, states that “reasonable and appropriate” measures be taken to address privacy, security and business policies, and requirements of protected health information (PHI).

However, HIPAA does not provide technical guidance on what those “reasonable and appropriate” measures are. In order to take “reasonable and appropriate” measures, some organizations simply filter any outbound messages containing PHI that might leave their organization. Others are implementing SSL/TLS connections between SMTP servers. Ultimately, though, to ensure that PHI is completely protected and to do due diligence, technologies such as S/MIME (PKI) and Enterprise Rights Management (ERM) should be investigated.

Topic 4: Antivirus and Anti-Spam Strategies and Best Practices

Q 4.2: How are spam, viruses, and worms detected?

A: Just as methods for distributing viruses, worms, Trojan horses, spam, and phishing scams have evolved, so have the mechanisms for detecting them. At first glance, the tools to detect both spam (including phishing schemes) and malicious content (including viruses, worms, and Trojan horses) in an email message seem to be identical. Simply open the message and perform a scan; in reality, the mechanisms are somewhat different and much more sophisticated than even the most advanced virus scanner was just a few years ago.

Virus, Worm, and Trojan Horse Detection

To keep the descriptions reasonably brief, I’m going to bundle viruses, worms, and Trojan horse content into simply the “virus” or malware category. The typical virus that affects many computers on the Internet today is vastly more complex than mail-based viruses such as the Love Bug or Melissa.

Signature or Pattern Detection of Known Viruses

Originally, viruses were detected exclusively through pattern recognition: a file or email message was opened, and the text of the file or message was compared against a list of strings of known viruses. Although this tactic might have been practical when there was a few thousand known viruses, as of early 2006, Symantec is tracking just over 72,000 total threats. Many of these are not detectable via simple signature scans and pattern detection, as the malware authors have developed methods to change the virus code so that it cannot easily be detected. Viruses that can change their behavior and appearance fall into two categories: polymorphic or metamorphic viruses.

Polymorphic Viruses

With each new infection, polymorphic viruses attempt to change their appearance by swapping their code and inserting padding—such as programming code that does not do anything—into the virus so that it appears to be a different program.

Metamorphic Viruses

Metamorphic viruses mutate in a fashion similar to a polymorphic virus. However, metamorphic viruses can not only switch around blocks of code within the virus but also change program sequences and use different instructions.

Generic Detection

Generic detection is similar to signature detection. If suspected content is not identified using signature detection, the scanning software can look for certain sequences within a possible virus that usually do not change even if the virus is a polymorphic or metamorphic virus.

Heuristic Filters

Heuristic filters are clever in that they can help to find and isolate malware that does not yet have a published signature. They do so by starting with a basic set of rules about the behavior of email-based viruses and use artificial intelligence techniques to make decisions about whether a message contains a virus. Although heuristic filters may catch viruses that have yet to be defined, they are not foolproof and are subject to false positives.

These filters work well with malware detection systems that include a *dynamic quarantine*. A dynamic quarantine allows scanning systems to conditionally quarantine messages until either the message is manually released by an administrator or is rescanned with new scanning rules or updated virus signatures.

Traffic Analysis

Malware detection using traffic analysis works on the assumption that virus outbreaks generate unusual traffic patterns, such as the same message being sent over and over. Detecting these outbreaks usually involves a combination of artificial intelligence techniques and experienced operator intervention. Also, accurate detection using traffic analysis requires a very large pool of message traffic to analyze. Thus, traffic analysis detection is more accurately performed by a service provider or a managed provider that has access to information about mail being received.

Behavioral Analysis

When using behavioral analysis, a malware detection system creates a “sandbox” for each suspected file or message that it must scan. This sandbox is essentially a virtual computer environment in which the file can be loaded or executed and the behavior monitored. Although this can be a good way to determine whether an unknown attachment or program is carrying malicious code, it requires quite a lot of computing resources and thus does not scale well to large messaging environments on centralized scanning systems. Behavioral analysis scanning systems work best in an environment (such as a managed provider) designed to support large-scale scanning operations.

Spam Detection

Detecting spam requires a slightly different approach than detecting malware. Unfortunately, there is much spam and no two spam messages are the same, so signatures are difficult to create. Further, spam can come from many different sources. An alarming trend that has recently emerged is that some organizations that send spam are now in collusion with the authors of viruses and worms. Some virus writers are now writing spam distribution viruses for profit instead of just notoriety.



Virus writers and spam operators are now joining forces to send out spam more quickly and in an attempt to avoid filters.

A virus or worm can deposit a small amount of code that operates in much the same way as viruses such as Sober and SoBig (which have their own SMTP engines.)—except instead of sending out viruses and attempting further infection, this malware sits idle until the spammer is ready to start sending out their next batch of spam mail. When the spammer starts sending mail, they have at their disposal hundreds, thousands, or tens of thousands of computers on every corner of the Internet that can assist in sending messages. These distributed systems can also mount a directory harvesting attack against many mail servers much more efficiently because it takes much longer to recognize what is happening and block the activity.

Real-Time Block Lists

Real-time block lists (RBLs—also known as a real-time black hole lists) are probably the oldest method of spam prevention. RBLs don't assist in any sort of detection, per se. RBLs are lists of known spammers, open relays, dial-up IP addresses, and DHCP addresses. SMTP servers use DNS to query the RBL list to determine whether an IP address is on that list. If the IP address in question is on the RBL list, the connection from that server is rejected.

The accuracy and value of RBLs is widely debated among messaging experts; some messaging administrators consider them to be very effective tools and others consider them less valuable. The primary objection to an RBL is that a mail server's IP address can be erroneously reported to the provider, a mail server can take an IP address that once occupied an open relay, or an entire block of IP addresses may be blocked when only one or two offenders exist in an entire range. Depending on the RBL provider, getting your IP address off the list can be a difficult process.

Keyword Analysis

The earliest anti-spam products that reached the market performed keyword analysis against the content of the message. The earliest generations of these products would simply quarantine, delete, or reject a message if it found one word or phrase that was on the blocked list. This method is very prone to false positives.


Currently, keyword analysis software has taken to creating a rating for the message that indicates the probability that the message is spam—commonly known as the spam confidence level (SCL). Multiple keywords and their proximity to one another are scanned. If the word “free” and “Viagra” are within a few words of one another, the message will have a higher SCL rating than if the word “Viagra” was seen by itself. Of course, again, this can still generate false positives, but at a much lower rate than previous methods.

Bayesian Logic

Bayesian logic is a branch of logic and probability theory named for English mathematician Thomas Bayes. A filter that uses Bayesian logic must be provided with a representative list of the spam that an organization receives (such as asking your users to send all their spam to a certain mailbox). The more spam that you provide to a spam detection system using Bayesian logic, the better it can be trained to detect spam. Some vendors are reporting 99 percent detection rates with less than 1 percent false positive rates when using Bayesian methods. See <http://www.paulgraham.com/spam.html> for more information about the use of Bayesian logic to detect spam.

Sender Authentication

Recently, efforts by a number of industry leaders have produced mechanisms for verifying that a sending SMTP server is really authorized to send mail on behalf of a domain. Among these are the Sender Policy Framework (known as SPF, Caller ID, or Sender ID) and DomainKeys. Sender ID seems to have produced the most common mechanism of domain verification. SPF requires that all mail senders on the Internet register the names and IP addresses of their SMTP servers that would send email to the Internet. A receiving system accepts a message and can examine the SMTP headers and the source IP address to determine whether the message originated on an authorized server from the supposed sending domain.

 Microsoft has produced a Web-based wizard that can help you to determine whether you have valid Sender ID/SPF records in DNS; if not, the wizard provides the information necessary to create them. See <http://www.anti-spamtools.org> for more information.

Unfortunately, this early in the life of Sender ID, administrators of many valid mail servers have not yet created the necessary SPF records in DNS, and it remains an inaccurate way of validating the sending server. See <http://mostlyexchange.blogspot.com/2005/07/sender-id-is-coming-get-your-txt.html> for more information.

Other Factors Involved in Spam Detection

Modern anti-spam filters are now using a number of additional techniques to detect spam. For the most part, none of these are used by themselves to determine whether a message is spam, but rather each is used to increase or decrease the SCL of a particular message. These are often used in conjunction with keyword filters, sender authentication, and RBLs. Factors that can raise the SCL for a message include:

- Messages with a blank subject line
- Messages with a missing sender address
- Subject line or a significant portion of the message in all capital letters
- Message contains phrases common to spam messages such as “To no longer receive our offers,” “This is an advertisement,” “To be terminated from future deployments,” and other such phrases
- Messages that are predominantly links and images with very little text in the message body
- Messages with lots of colored text
- Time of day or the day that the message arrives (on the assumption that non-official mail arrives on the weekend or late at night)

Topic 5: Firewall Strategies and Best Practices

Q 5.2: What are some of the special considerations when Outlook clients are separated from Exchange Server by a firewall?

A: If you have ever tried to put an Outlook MAPI client on the other side of a firewall from the Exchange Server, you have probably experienced some of the same frustrations as your fellow Exchange administrators. Opening ports through the firewall for a POP3, IMAP4, or Web client is much simpler than opening ports for a MAPI client due to the complexity involved with remote procedure calls (RPCs).

For a long time, an acceptable practice was to allow remote Outlook clients to access Exchange Server by opening all the necessary RPC ports. Due to a number of RPC-based worms, this is no longer an acceptable practice. However, in midsized and larger networks, a concept that is gaining popularity is the concept of the data center firewall. Figure 5.4 shows an example of how this might be deployed.

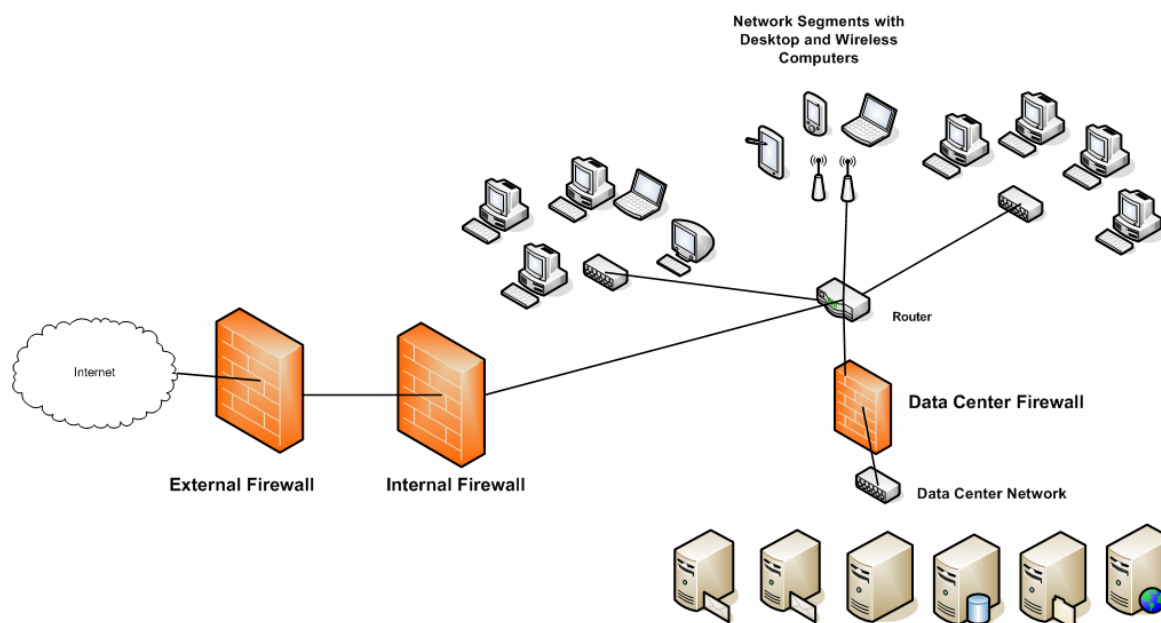


Figure 5.4: Implementing a data center firewall.

In addition to the firewall that serves to protect the entire network, an additional firewall (or a filtering router) is placed between the data center network and the networks that host the users. This provides an additional layer of protection from viruses, worms, or other types of security compromises that can occur on the end-user network. Figure 5.4 may seem like a lot of routers, but in a larger network and academic networks where it is difficult to control the configuration of the desktop computers, this additional layer of protection between the users and the servers can give you peace of mind knowing that the critical components of your network are better protected.

To view the RPC ports that are opened and ascertain which ports RPC services are using, you can use the Port Query utility and query the RPC end-point mapper. Figure 5.5 shows the PortQueryUI.exe utility; you can download the Port Query graphical utility at <http://tinyurl.com/cbk9n>. Microsoft Knowledge Base article 832919 “New features and functionality in PortQry version 2.0” has more information about the PortQryv2 command-line utility.

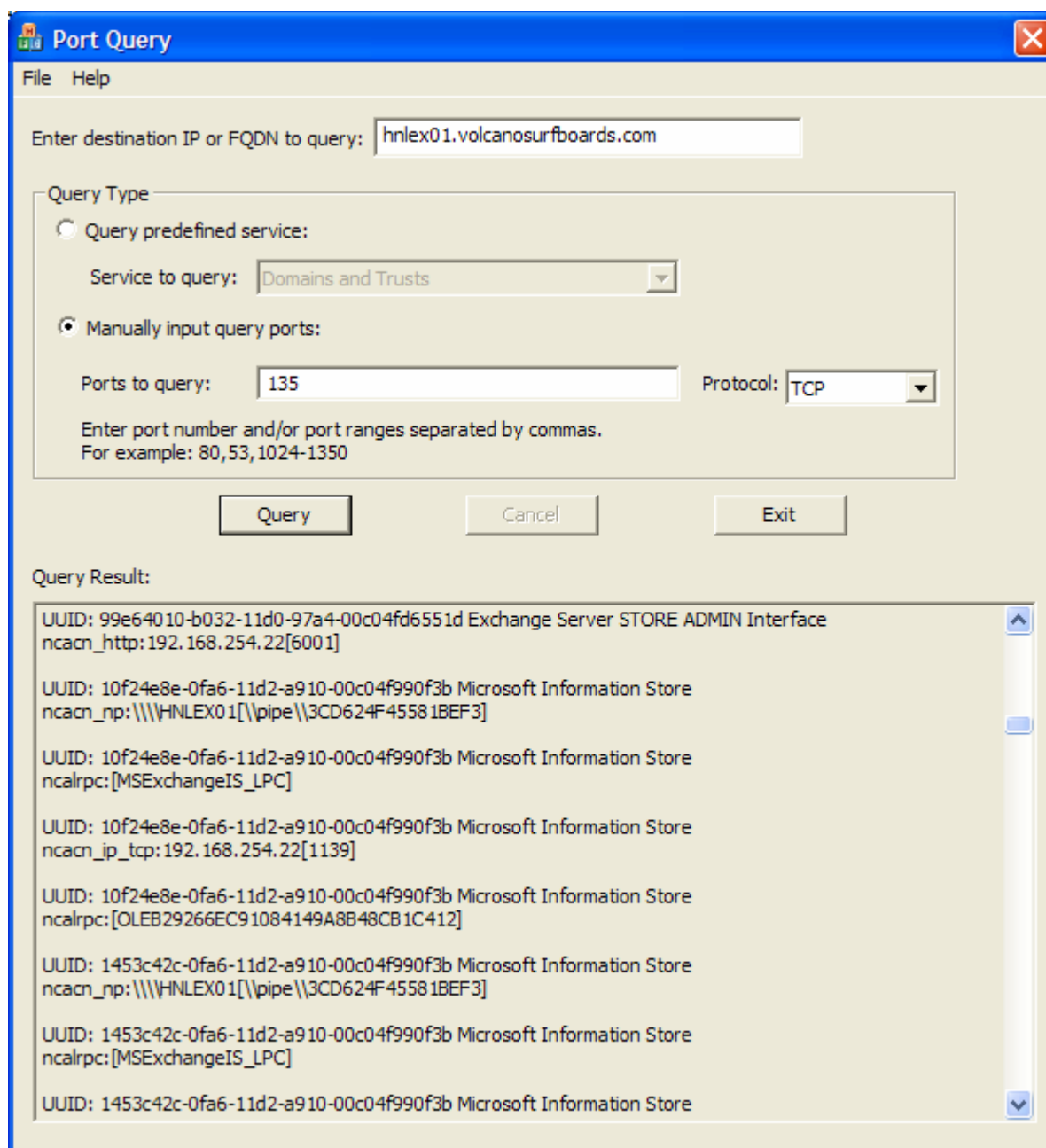


Figure 5.5: Port Query utility.

Configuring Exchange Servers and Active Directory for Firewall Support

Outlook MAPI clients that use RPCs to communicate with the Exchange Server require access to the Exchange Server's port 135, the RPC end-point mapper, plus the directory service referral port, and the information store. In addition, Outlook requires RPC access to the Global Catalog (GC) servers that are referred to Outlook by the Exchange Server to which they connect. Unfortunately, by default, these RPC ports are not fixed to the same port number; instead, they are dynamic and may be different on different servers or when the server reboots. However, the ports can be statically mapped so that they use the same port number each time.


Exchange Server Ports

To configure an Exchange Server so that the RPC ports that are used by Outlook clients can be accessed through a firewall, there are three ports that must be statically mapped. The first of these is the System Attendant's referral interface. This is the component that Outlook 2000 and later clients use to receive referrals to GC servers. Create a registry value of type REG_DWORD called TCP/IP Port in the HKLM\SYSTEM\CurrentControlServices\MSEExchangeSA\Parameters registry key. Set this newly created value to a decimal number above 5000. In this example, I'll use port 5001.

Next, the System Attendant's Name Service Proxy Interface (NSPI) needs to be statically mapped. The NSPI proxy interface is used by Outlook 98 and earlier clients that cannot be referred to a GC. The NSPI proxy interface performs directory lookups on behalf of the Outlook client. Create a registry value of type REG_DWORD called TCP/IP NSPI Port in the HKLM\SYSTEM\CurrentControlServices\MSEExchangeSA\Parameters registry key. Set this value to a decimal number above 5000. In this example, I'll use port 5002.

Finally, the TCP port that the information store uses has to be statically mapped. Create a registry value of type REG_DWORD called TCP/IP Port in the HKLM\SYSTEM\CurrentControlServices\MSEExchangeIS\ParametersSystem registry key. Set this value to a decimal number above 5000. In this example, I'll use port 5003. Once the ports have been set, the System Attendant and the Information Store services will need to be restarted.

For MAPI clients to communicate with Exchange Server, the firewall will have to permit ports 135, 5001, 5002, and 5003 be opened to all mailbox and public folder servers. However, there will also need to be ports opened for clients to communicate with Active Directory (AD) GC servers.

 For information about other ports that Exchange Server uses, see Microsoft Knowledge Base article 278339 "TCP/UDP ports used by Exchange 2000 Server."


Domain Controller Ports

Outlook 2000 and later clients are referred directly to a GC server. By default, the Exchange Server will refer the Outlook clients to GC servers that are in the same AD site as the Exchange Server. For this reason, Outlook clients may need to access any of the GC servers in the site with the Exchange Server.

Outlook can be configured to use a specific GC server or to use a closer GC instead of the one to which it is referred. For more information, see Microsoft Knowledge Base article 319206 “How to configure Outlook to a specific global catalog server or to the closest global catalog server.”

Like the information store service and directory service referral ports on an Exchange Server, the AD RPC/MAPI port is dynamic and may change each time the domain controller is restarted. To statically map the RPC port for a domain controller, create a registry value of type REG_DWORD called TCP/IP Port in the HKLM\SYSTEM\CurrentControlSet\Services\NTDS\Parameters registry key. Set this value to a decimal value between 1024 and 5000; in this example, I am using port number 5000. Microsoft recommends the port range 1204 to 5000; see Microsoft Knowledge Base article 270836 “Exchange Server static port mappings” for more information. Once the domain controller is restarted, it will use port 5000. The firewall will need to have ports 135 and port 5000 opened between the clients and all GC servers that they may use.

If you don't want to open the RPC ports for the GC servers, you can disable the referral interface and Outlook clients will use the NSPI proxy interface. This will slightly increase the load on the Exchange Server, though. To disable the referral interface, create a registry value of type REG_DWORD called No RFR Service in the HKLM\SYSTEM\CurrentControlSet\Services\MSExchangeSA\Parameters registry key. Set this value to 1 to disable the referral service or 0 to enable the referral service. Doing so will not take effect until the System Attendant has been restarted. After the restart, Outlook clients will not need direct access to GC servers.

 For more information about how Outlook clients use the NSPI proxy or the directory service referral interface, see Microsoft Knowledge Base article 302914 “How Outlook 2000 Accesses Active Directory”.

Alternatives to Opening RPC Ports

If opening up the RPC ports through your firewall to your Exchange Server is not a viable option, you have a few alternatives. For Outlook 2002 and earlier, the most common method of allowing Outlook access to an Exchange Server was to require the user to open a VPN connection. Outlook 2003 allows RPC requests to be encapsulated in HTTP or HTTPS packets and passed through a firewall using port 80 or port 443.

Some firewalls provide an RPC filter that will dynamically open required ports on the external interface of the firewall and pass the correct port requests for Exchange services such as the directory interface or the information store into the internal network. Firewalls such as the Checkpoint Firewall-1 and Microsoft ISA Server provide an RPC publishing feature that supports Outlook. The ISA Server's RPC filter performs application-layer inspection of inbound RPC connections prior to the connection being passed to the Exchange Server, thus ensuring that only legitimate RPC traffic is passed to the internal network and only to Exchange Servers.

Topic 6: Protecting and Controlling Sensitive Information in Email

Q 6.2: What is Enterprise Rights Management?

A: Rights management broadly describes protection technologies for all types of digital media, whereas Enterprise Rights Management (ERM) can be considered a business application of rights management. ERM is considered a subset of the rights management umbrella, but is more focused towards corporate or government applications. ERM is closely related to Digital Rights Management (DRM), but the term DRM is more closely identified with the protection of commercial media content such as music and movies.

In one form or another, rights management technologies have been around for a long time and have other sibling technologies such as copy protection and technical protection measures. Content protection of digital media such as DVD movies, CD music, and even software has been available for quite some time. For years, some specialized software packages have required a “dongle” or USB key be attached to a computer before the software can be used. As copying technologies have become more prevalent and the ability to distribute content quickly to many people has emerged, protection of digital media has become even more important.

The popularity of the Internet and the emergence of digital media have given new cause for concern about copying and distribution of protected media because a copy of digital media is a perfect copy, unlike a copy of a video tape (the quality of a video tape is not as good as the source). A DVD or CD without protection can quickly and easily be copied and shared with thousands of Internet users through peer-to-peer file-sharing programs. Software that was licensed for one computer can be installed and used on other computers if it is not adequately protected.

ERM has evolved from the concept of DRM because sensitive documents, presentations, spreadsheets, and other organizational information can easily be accessed by unauthorized persons, accidentally or intentionally passed on to others, or modified without authorization. ERM technologies focus on business applications and allow the creator of digital content (specifically documents, spreadsheets, presentations, and email) to encrypt the content they create and specify access privileges that the other users have to this content. Implementations, features, benefits, and file formats/applications supported vary from one vendor to the next. Common features that are found in ERM solutions include:

- Encrypting content so that only authorized persons can consume that content
- Controlling the types of access the content consumer may have, including read, copy, modify, print, or forward
- Specifying expiration dates after which the content is no longer accessible by the consumer or after which new content supersedes the existing content
- Allowing persistence of the digital rights whether the file or email message remains within an organization's boundaries, if it is burned to optical media, emailed to users in another organization, or stored on portable media (USB drives)
- Revoking or modifying access rights if the content owner deems this necessary to prevent further access
- Providing access and audit trails

The benefits of these features are clearly obvious. Content creators can control the content they create both within their corporate boundaries as well as outside. Even if USB media is lost, a laptop is stolen, content is accidentally forwarded to unauthorized persons, or content is intentionally passed on to someone the content creator did not intend, the content remains protected.



ERM solutions provide an organization with a persistent mechanism of policy enforcement regardless of where the data is accessed.

When an author creates protected media, an ERM system enables them to specify restrictions that are placed on the users of that content. Some ERM systems include the ability to revoke rights later after the content has been distributed and enable auditing usage of protected content. However, the application (and possibly the operating system—OS) must support the ERM system being used.

Applications must be enabled to perform and enforce rights management functions. Information Rights Management (IRM) is functionality that is built-in to an application to allow creation of or viewing of ERM-protected content. Figure 6.1 shows an example of permissions that can be assigned to a document by the rights management components of Microsoft Word 2003 and Windows XP Pro.

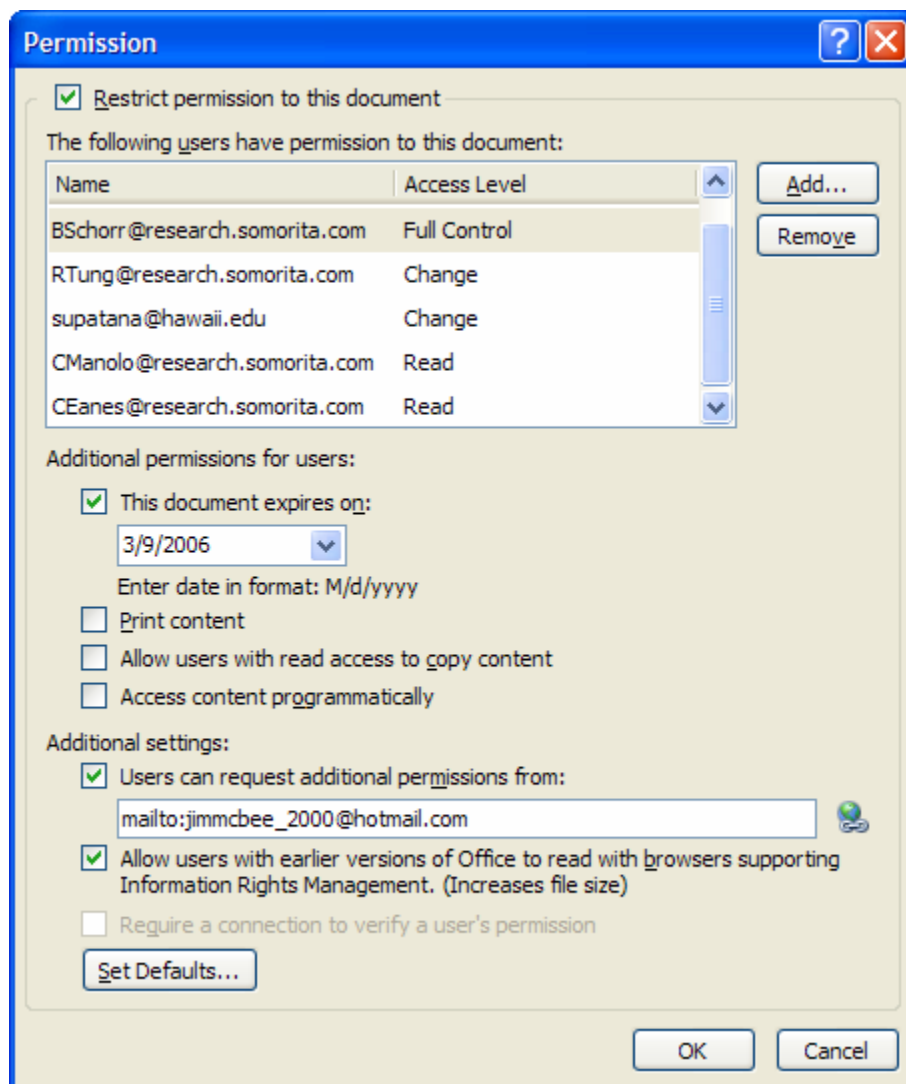



Figure 6.1: Defining permissions on a document in Microsoft Word.

Once these permissions are assigned, the document is encrypted and the access restrictions that have been placed on the document follow the document wherever it may be sent; this is true even if the document leaves an organization's internal servers.

 S/MIME messaging protects email message content from authorized viewing in transit and while stored. ERM solutions protect additional content types and provide limitations as to what the recipient can do with that content. S/MIME and ERM can be used together when it is necessary for attestation of message content due to regulatory requirements such as Sarbanes-Oxley.

Components of an ERM System

Understanding the various components and requirements of an ERM system can better help with the planning and deployment of an ERM solution for your organization. There are both client- and server-side applications as well as components such as digital certificates and, of course, the protected content.

RMS Enabled Applications

For users to protect the content they create, the application they are using must be rights management enabled. Applications such as Internet Explorer (IE—through an add-on) and Microsoft Office Professional 2003 are automatically capable of using Microsoft's Windows Rights Management Services (RMS) ERM solution. Liquid Machines extends Microsoft's baseline support with extensions for earlier versions of Microsoft Office, Adobe Acrobat, and other file types and provides RMS support for RIM Blackberry devices. The Windows Vista operating system (OS) will include features that allow any document type to be converted to an XPS document and to be protected with RMS. Vendors such as Authentica provide plug-ins for Microsoft Office and Adobe Acrobat applications that work with the Authentica Policy Server.

RMS Client Software

Windows client OSs prior to Windows Vista that are going to participate in an RMS-trusted environment must have the client software installed. RMS-enabled applications must interact with this software when creating protected content or when the recipient of protected content consumes that content. The client software provides the necessary APIs for the RMS-enabled application, handles communication with the RMS server, stores certificates, and handles client authentication. The person receiving the RMS-protected content is referred to as the recipient in some ERM solutions, while other solutions may refer to the RMS client software as the recipient.

RMS Server Software

The RMS server software handles the certification of trusted entities, licensing of rights-protected content, and authentication of credentials as well as validates rights account certificates. In some implementations of ERM solutions, RMS servers may have multiple roles, such as the user provisioning and content licensing server. The server that issues usage licenses is called the *License Server*.

The RMS server creates the rights and policies that can be applied to content and pushes these to the client; in addition, the server provides a repository for the encryption keys that are used to encrypt the protected content, a method of identifying and authenticating users such as or an Active Directory (AD), and a license generator. The Windows Rights Management Services support only AD authentication, but some other solutions support generic LDAP authentication.

Keys and Certificates

The RMS client software works in a similar fashion to PKI applications in that it uses public and private keys for client/server authentication and for protecting content encryption keys. There are a number of certificates that are used by the RMS client software to protect the encryption keys, authenticate to the rights management server, and to send information to the rights management server. In a Windows Rights Management Services environment, these include:

- The *Machine Certificate* is used to identify a trusted machine. There is one machine certificate issued to a computer for each user that uses that computer. This setup allows the computer to participate in the RMS environment.
- The *Rights Account Certificate (aka Rights Management Account Certificate)* is issued to a trusted user. The RAC allows a user to consume protected content if the user is so authorized.
- A *Client Licensor Certificate (CLC)* is issued to publish content even when the client is not connected to a rights management server; this is the default method of publishing for Microsoft Office 2003. The CLC is issued to a user to allow that user to publish content on behalf of the rights management server. This certificate includes the CLC public key, the CLC private key (encrypted with the user's RAC public key), and a copy of the rights management server's Server Licensor Certificate.
- A *Publishing License (PL)* is created by the creator of protected content and includes rights and conditions under which the content may be consumed. The PL contains the policy information, the content encryption key, and the URL of the licensing server. The content encryption key is protected by encrypting it with the public key of the RMS server.
- The *Use License (UL)* is issued to a user by a rights management server to allow the user to consume content based on the conditions in the publishing license. The UL contains the symmetric AES encryption key used to encrypt protected content. The AES encryption key is encrypted using the user's RAC public key.
- *Advanced Encryption Standard (AES)* keys are the 128-bit symmetric keys that are used for encryption and decryption of protected content. These keys are either protected with the user's public key in ULs or to the RMS server's public key and/or the CLC public key in publishing licenses.
- *XrML certificates* are the format that Microsoft uses to store rights management system policy information. XrML is a rights expression language that allows the definition of rights and restrictions that can be placed on protected content.
- *Rights Expression Language (REL)* was developed by the Motion Pictures Experts Group (MPEG) for expressing rights that can be granted to digital content.

Protected Content

Protected content is generated by RMS-enabled applications. Although the exact steps that ERM vendors follow may be somewhat different, there are some similarities: for example, a protected file includes AES encrypted content, the rights information, and the URL of a rights management server.

Figure 6.2 shows an example of a file's contents once it has been protected with rights management software using Microsoft Office applications. All the components necessary to protect the content are embedded in a single file including the encrypted content. There are variances on this concept with other implementations; even Microsoft Outlook varies slightly in that UL are not appended to the document but are cached in the user's profile directory instead.

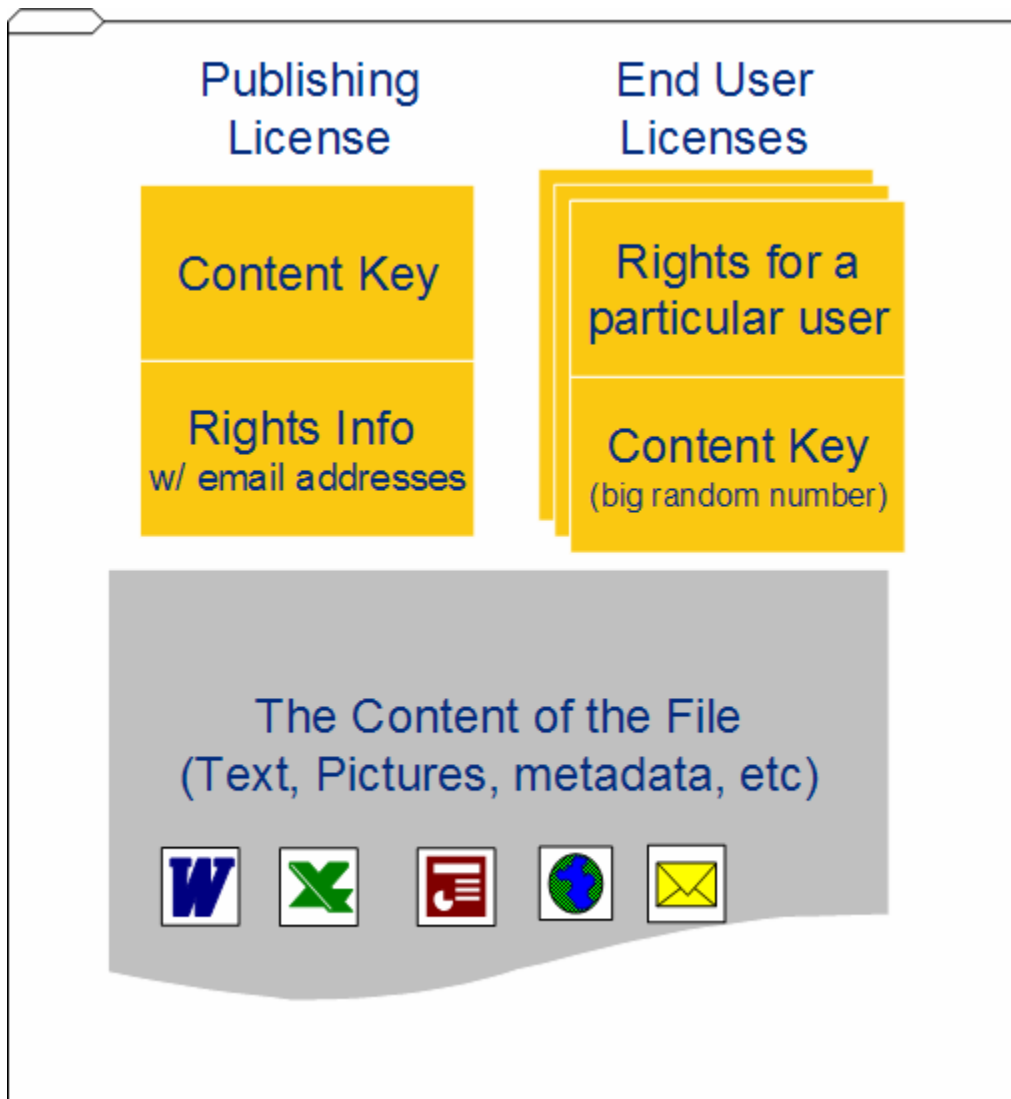


Figure 6.2: File protected with rights management software.

The publishing license is created when the file content is protected. This includes the encryption key that is used to encrypt and decrypt the protected content. The content key is encrypted with the rights management server's public key. The rights information contains a list of individual email addresses of users or groups that are allowed to access the content. This is also protected with the rights management server's public key.

For each of the users that have licensed the content, there is an end user license in the file (unless the rights management policy on the document prohibits caching of ULs). The ULs include the rights for a particular user and the encryption key. The user's individual rights and the content key are encrypted with the user's public key by the rights management server.

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>.