CHAPTER 5

Solving Secure Email's "Grand Challenge" with Signature-Only Email

In 1999 Carnegie Mellon University graduate student Alma Whitten and her advisor J. D. Tygar published "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0."[WT99] The paper reports on a user study in which Whitten asked 12 subjects to create keys and send messages that were digitally signed and sealed using the PGP 5.0 and Eudora.

What made the *Johnny* paper popular—it remains one of the most heavily cited on the topic of usability and security—was not the fact that it presented research findings that were novel or surprising, but that it provided scientific justification for a common observation: Email encryption programs are hard to use. This was true in 1999 when the paper was published, and it is still true, more or less, today.

Secure email has effectively become a "grand challenge" of current research into the interaction of security and usability. This is because any system that enables its users to reliably send and receive mail that is both digitally signed and sealed with encryption requires that many other problems be solved first. For example, today's secure email systems use symmetric and asymmetric encryption, hash functions, and third-party certificates. They require key distribution and revocation systems, because the users may be communicating asynchronously without ever both being online at the same time. They must also have message formats that must pass through multiple untrusted system and be able to handle multiple character sets and attached content. Unlocking the user's private key requires solving the authentication problem and probably the trusted path problem. Protecting that key requires host security and sanitization. Finally, allowing users to make sense of the identities behind the digital signature requires sensible solutions to the phishing problem.

This chapter takes an alternative approach and argues that sensible progress can be made on the email encryption problem through the incremental adoption of a half-way solution—email that is signed but not sealed. Through an analysis of history, standards, and currently deployed software, it argues that there are few if any usability barriers to the receipt of email that is signed with an S/MIME signature. Presenting data based on a survey of Amazon.com merchants, it argues that today's e-commerce participants believe that email should be digitally signed. It then presents specific recommendations for improving the usability and security of mail clients and webmail systems.

5.1 Background: Three Decades in Pursuit of Secure Messaging

In their seminal 1976 paper disclosing the invention of public key cryptography, Diffie and Hellman wrote somewhat optimistically that their invention "enables any user of the system to send a message to any other user enciphered in such a way that only the intended receiver is able to decipher it." [DH76]

(In fact, the invention allowed a message to be enciphered so that anyone possessing a specific private key could decipher it. The potential disconnect between an intended human recipient and the holder of a private key has haunted public key cryptography ever since.)

Diffie and Hellman proposed that public keys would be placed in "a public directory." The following year (1977), Rivest, Shamir and Adelman introduced what has come to be known as the *RSA Cryptosystem*, an algorithm that provided a practical realization of the kind of public key system that Diffie and Hellman foresaw. In 1978 Loren Kohnfelder proposed in his MIT undergraduate thesis [Koh78] that certificates could be used as an efficient and scalable system for distributing public keys.

With these three inventions—public key cryptography, the RSA algorithm, and certificates—the basic building blocks for a global secure messaging system were in place. Yet nearly 30 years later, after the deployment of a global Internet and the creation of low-cost computers that can perform hundreds of RSA operations in the blink of an eye, the vast majority of the legitimate mail sent over the Internet lacks any form of cryptographic protection.

Although this is a problem that lends itself to incremental solutions, many of the solutions that have been proposed have attempted to simultaneously solve all of the requirements outlined in the previous paragraph. It is quite possible that the heavy emphasis on technical correctness and complete functionality has prevented the deployment of incremental solutions that would have given us an email infrastructure significantly more secure than the one we have today.

5.1.1 Early work on secure messaging

Speaking at the 1984 ACM Annual Conference, Charles Wood from Bank of America presented a visionary paper describing the so-called "fifth generation computers" of the 1990s and the computational infrastructure that they would enable. In his talk, Wood described how public key encryption technology would be applied to solve security issues in computer networks. Such systems, Wood predicted, would use message authentication codes and digital signatures to protect the contents of messages from modification, and would have sophisticated key management systems for "changing keys, procedures for backing-up and archiving encrypted keys, recovery procedures, and the like" which would be chosen by the user.

> "Ideally, all this will be entirely transparent to the end user. He will of course, through application system or local operating system facilities, have the ability to specify what part(s) of his data he wishes to encrypt/decrypt, apply a MAC to, or sign with a digital signature. And he will additionally have some responsibility for maintaining the secrecy of his personal keys, perhaps via his own memory or that in a small plastic card."[Woo84]

Despite Wood's apparent equal emphasis on privacy, integrity and authentication, there was in fact little perceived need for signature-only systems during the first decade following the discovery of public key cryptography. Spam and email sent with forged `From:` addresses were not significant problems in the 1980s. On the other hand, there was considerable interest in techniques for adding "privacy"[1] to email moving over the network—probably a result of the military's priorities influencing academic computer science research.[CW87]

Cryptographic systems that provide signatures alone have the advantage that signatures can be placed on documents and ignored by the recipients without a decrease in message fidelity. As a result, such systems can be incrementally deployed. Deploying a system that mandates both signatures and message privacy is much harder because it is not possible to "ignore" the encryption and still understand the contents of the message that is sealed. As a result, many different tasks must be accomplished before the first message can be enciphered, sent, deciphered, and sensibly understood by the intended recipient:

1. Formats for representing cryptographic keys and email messages need to be created. In the case of messages, these formats need to be carefully designed so that the messages will survive transit over the existing email infrastructure.
2. Software that implements these formats needs to be deployed.
3. Keys need to be created for email correspondents—either individuals need to create their own, or else the software needs to create keys automatically.
4. Keys need to be distributed.
5. Individuals who would use the security systems need to be given sufficient incentive to use the new email systems, or existing systems need to be shut down so that only secure systems can be used. (As was the case in the migration from unencrypted HTTP to encrypted SSL communications for sending credit card numbers over the Internet.)

Further complicating matters, it is necessary for all participants to use mutually compatible security systems.

5.1.2 Standards and support for secure mail

On the Internet in the 1980s, the traditional procedure by which compatibility was achieved was for the protocol and a working implementation—"running code"[Cla92, p.543]—to be iteratively designed, with the protocol eventually being standardized through the Internet Engineering Task

[1] Lampson explains that computer security professionals really should use the word *secrecy* to describe technologies that assist in disclosure control, but that "the NSA hijacked the word *secrecy* in the 1960s to mean something else, so computer scientists have had to use other words ever since."[Lam05]

Force's Request For Comments process. Concurrent with this standardization process other implementations would be created.

The standardization process for developing a secure email standard was one of the most complicated tasks that the IETF had ever embarked upon:

- By the 1980s there were many pre-existing email systems, all with their own notions of email addresses, message envelopes, allowable character sets, and so on. All of these systems worked well enough when sending raw ASCII over SMTP, where messages could receive minor modifications *en route* but nevertheless be intelligible by the recipient. On the other hand, when a message that was enciphered or contained a digital signature was modified, the resulting message would be unreadable. Thus, some system for reliably enveloping messages that were being sent through the existing mail infrastructure needed to be developed.
- Because of the confusion surrounding export controls, it was not entirely clear whether or not the work could proceed in an international forum. At the time it was believed that reference implementations of cryptographic software could not be exported from the US in source code form over the Internet. This significantly complicated the development process.
- At the time, it was widely believe that public key cryptography required the use of a certification hierarchy to protect against man-in-the-middle attacks. Thus, any workable protocol to provide for *either* privacy or authentication needed to solve the global authentication problem as well.

The following sections discuss the three techniques for secure message authentication which successfully made their way through the IETF standardization process: Privacy Enhanced Mail, S/MIME, and OpenPGP.

Privacy Enhanced Mail (PEM)

The Internet Activities Board's Privacy Task Force started working on email encryption standards in the mid 1980s. These standards became known as Privacy Enhanced Mail (PEM), embodied in RFC 989 [Lin87] issued in 1987. The PEM standards were revised twice, with the final set of RFCs [Lin93, Ken93, Bal93] published in 1993. These documents defined a signature and encryption standard for ASCII email messages based on public key cryptography using the RSA algorithm.

PEM defined two main protection features: (1) Signed Messages and (2) Signed and Encrypted Messages. It is interesting to note that PEM made no provision for messages that were encrypted but not signed. Although this option was discussed, those directing the PEM project thought that such messages could be used to spoof end-users: it was conjectured that a user receiving an encrypted-only message might become confused and assume that the purported sender really did send the message. That is, the recipient might assume that error free processing by the PEM software meant that the message had been signed, when in fact it was not.[Sch04a]

But the PEM standards were complicated by the magnitude of their task. Not only did they have to describe how messages could be signed and sealed—they also had to describe how keys were created, signed and distributed. Furthermore, the standards had to invent the base64 encoding for sending binary objects through existing mail systems—techniques later adopted by the MIME standards.

Rather than inventing a new certificate format, PEM's creators adopted the digital signature standard defined by the CCITT X.509 Standard. These certificates were signed using the private RSA key of a Certifying Authority (CA). The public key of the Certifying Authority was placed in another certificate, which itself could be signed by another CA, and so on, composing a Certificate chain that led back to a single trusted *root*. Although not necessary, the root of the chain was also stored in a certificate—a so-called "self-signed" certificate that was signed with the root's own private key.

Because there was no centralized online public key directory in 1989, PEM was designed to operate without one. This was accomplished by including all of the certificates in the chain needed to verify the signature of a signed message. Once received, PEM implementations were supposed to store the accompanying certificates on the recipient's computer. The recipient could then reply to messages with a response that was both signed with the sender's own key and encrypted with the public key of the intended recipient.[Sch04a]

With the exception of the US Securities and Exchange Commission, which continues to use PEM signatures for its EDGAR electronic records filing system (Figure 5-1), the PEM standard has been largely abandoned. Schiller attributes three factors to the demise of PEM:

1. The lack of available software to implement PEM.
2. The requirement that end-users obtain certificates, a process that was never well documented and cumbersome at best.
3. Public apathy, there wasn't much market demand.

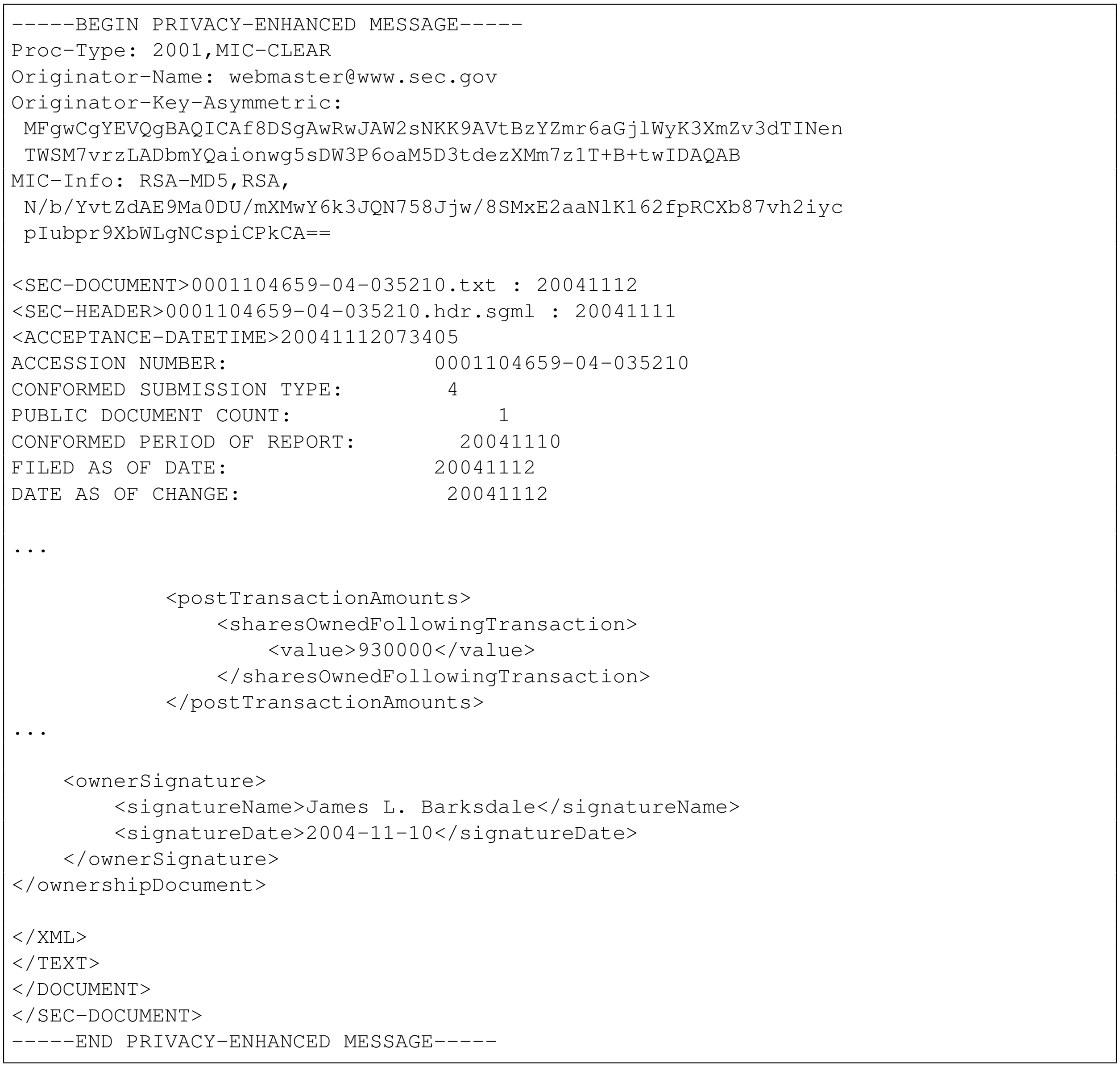
Secure Multipurpose Internet Mail Extensions (S/MIME)

When work on PEM stalled shortly after the publication of the PEM standards, RSA Data Security began a new project to re-implement the PEM concept on top of the new MIME mail standards. Called S/MIME, this work was eventually migrated to the Internet Engineering Task Force (IETF) and standardized through RFC2311 and follow-ons. [DHR$^+$98, Ram04b] Figures 5-2 and 5-3 show the MIME parts of a signed and sealed S/MIME message, respectively. A message that is to be both signed and sealed is simply signed first, then the entire message body is sealed.

Because management of a single root with a single certification policy proved to be problematical, S/MIME implementations do not implement a strict hierarchy of certificates, but instead accommodates any number of trusted Certificate Authorities. In practice, they ship with a relatively large number of CA keys that are pre-trusted by the authors of the software. Although some organizations audit the certificate list and remove the CA keys, most do not.

Microsoft became an early adopter of S/MIME in 1996, when the company announced support for the standard, claiming that support would be present "in a 1997 release of Microsoft Exchange client, Microsoft Outlook, and Microsoft Internet Mail."[Cor96] Netscape responded by adding support for S/MIME into its Communicator email client.[Net97]

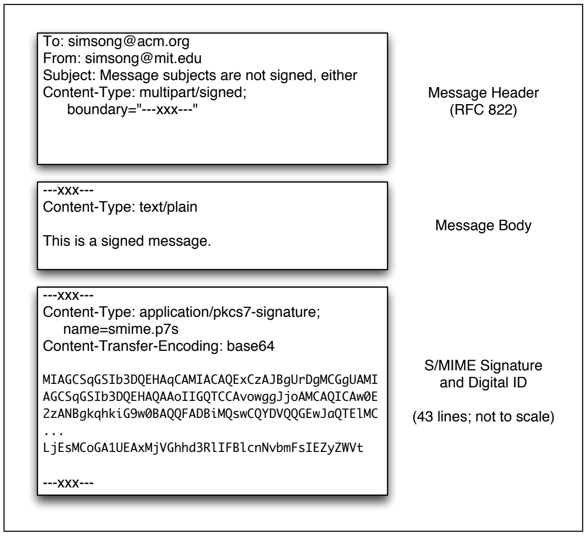
Today support for S/MIME is integrated into many email clients, including Microsoft Outlook and Outlook Express, Netscape Communicator, Lotus Notes, and others. Support for S/MIME is scheduled to be added to Eudora sometime in 2005.[Don05] But support for S/MIME is notably missing from AOL's client software as well as from many web-based mail systems (*e.g.,* Yahoo, Google's



```
-----BEGIN PRIVACY-ENHANCED MESSAGE-----
Proc-Type: 2001,MIC-CLEAR
Originator-Name: webmaster@www.sec.gov
Originator-Key-Asymmetric:
 MFgwCgYEVQgBAQICAf8DSgAwRwJAW2sNKK9AVtBzYZmr6aGjlWyK3XmZv3dTINen
 TWSM7vrzLADbmYQaionwg5sDW3P6oaM5D3tdezXMm7z1T+B+twIDAQAB
MIC-Info: RSA-MD5,RSA,
 N/b/YvtZdAE9Ma0DU/mXMwY6k3JQN758Jjw/8SMxE2aaNlK162fpRCXb87vh2iyc
 pIubpr9XbWLgNCspiCPkCA==

<SEC-DOCUMENT>0001104659-04-035210.txt : 20041112
<SEC-HEADER>0001104659-04-035210.hdr.sgml : 20041111
<ACCEPTANCE-DATETIME>20041112073405
ACCESSION NUMBER:                0001104659-04-035210
CONFORMED SUBMISSION TYPE:        4
PUBLIC DOCUMENT COUNT:                1
CONFORMED PERIOD OF REPORT:        20041110
FILED AS OF DATE:                20041112
DATE AS OF CHANGE:                20041112

...

            <postTransactionAmounts>
                <sharesOwnedFollowingTransaction>
                    <value>930000</value>
                </sharesOwnedFollowingTransaction>
            </postTransactionAmounts>
...

    <ownerSignature>
        <signatureName>James L. Barksdale</signatureName>
        <signatureDate>2004-11-10</signatureDate>
    </ownerSignature>
</ownershipDocument>

</XML>
</TEXT>
</DOCUMENT>
</SEC-DOCUMENT>
-----END PRIVACY-ENHANCED MESSAGE-----
```

Figure 5-1: An excerpt of SEC form 4, filed electronically with the United States Securities and Exchange Commission, shows that the PEM format is still used today to sign XML-encoded filings. Complete form available online at `http://www.sec.gov/Archives/edgar/data/1008699/000110465904035210/0001104659-04-035210.txt`

GMail, Hotmail). On these systems, digitally signed S/MIME messages appear as ordinary messages with an additional attachment typically named `smime.p7s`. (S/MIME messages that are sealed with encryption are naturally indecipherable on systems that do not support S/MIME.)



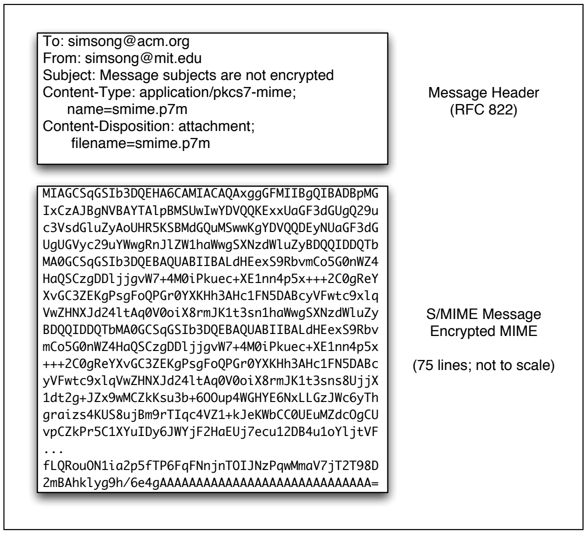
```
To: simsong@acm.org
From: simsong@mit.edu
Subject: Message subjects are not signed, either
Content-Type: multipart/signed;
     boundary="---xxx---"
```

```
---xxx---
Content-Type: text/plain

This is a signed message.
```

```
---xxx---
Content-Type: application/pkcs7-signature;
    name=smime.p7s
Content-Transfer-Encoding: base64
```

```
MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMI
AGCSqGSIb3DQEHAQAAoIIGQTCCAvowggJjoAMCAQICAw0E
2zANBgkqhkiG9w0BAQQFADBiMQswCQYDVQQGEwJaQTElMC
...
LjEsMCoGA1UEAxMjVGhhd3RlIFBlcnNvbmFsIEZyZWVt

---xxx---
```

Figure 5-2: A sample S/MIME-signed message



```
To: simsong@acm.org
From: simsong@mit.edu
Subject: Message subjects are not encrypted
Content-Type: application/pkcs7-mime;
     name=smime.p7m
Content-Disposition: attachment;
      filename=smime.p7m
```

```
MIAGCSqGSIb3DQEHA6CAMIACAQAxggGFMIIBgQIBADBpMG
IxCzAJBgNVBAYTAlpBMSUwIwYDVQQKExxUaGF3dGUgQ29u
c3VsdGluZyAoUHR5KSBMdGQuMSwwKgYDVQQDEyNUaGF3dG
UgUGVyc29uYWwgRnJlZW1haWwgSXNzdWluZyBDQQIDDQTb
MA0GCSqGSIb3DQEBAQUABIIBALdHEexS9RbvmCo5G0nWZ4
HaQSCzgDDljjgvW7+4M0iPkuec+XE1nn4p5x+++2C0gReY
XvGC3ZEKgPsgFoQPGr0YXKHh3AHc1FN5DABcyVFwtc9xlq
VwZHNXJd24ltAq0V0oiX8rmJK1t3sn1haWwgSXNzdWluZy
BDQQIDDQTbMA0GCSqGSIb3DQEBAQUABIIBALdHEexS9Rbv
mCo5G0nWZ4HaQSCzgDDljjgvW7+4M0iPkuec+XE1nn4p5x
+++2C0gReYXvGC3ZEKgPsgFoQPGr0YXKHh3AHc1FN5DABc
yVFwtc9xlqVwZHNXJd24ltAq0V0oiX8rmJK1t3sns8UjjX
1dt2g+JZx9wMCZkKsu3b+6OOup4WGHYE6NxLLGzJWc6yTh
graizs4KUS8ujBm9rTIqc4VZ1+kJeKWbCC0UEuMZdcOgCU
vpCZkPr5C1XYuIDy6JWYjF2HaEUj7ecu12DB4u1oYljtVF
...
fLQRouON1ia2p5fTP6FqFNnjnTOIJNzPqwMmaV7jT2T98D
2mBAhklyg9h/6e4gAAAAAAAAAAAAAAAAAAAAAAAAAAAAA=
```

Figure 5-3: A sample S/MIME-sealed message

Pretty Good Privacy (PGP)
In 1991 a programmer in Colorado named Phil Zimmermann released PGP, a program that implemented the basics of public key cryptography and key management.[Zim91b, Zim91c]

Although PGP was technically a proprietary encryption system, the fact that it was distributed in source-code form made it possible for others to experiment with the system's algorithms, formats, and underlying design as they would with a traditional reference implementation for a proposed standard. The result of this experimentation was PGP 2, a workable encryption system that became quite popular in some technical and academic communities.

Compared with S/MIME, PGP had the advantage that people could use it immediately: the freely downloadable software contained a complete key management system that could be used to create encryption keys, have keys verified by third-parties, and both sign and seal messages. What's more, PGP worked equally well with keys that *weren't* certified: the program simply printed a warning message. (In principle S/MIME can also be used with keys that are not certified, but this mode of operation was never encouraged by the makers of S/MIME software. We shall return to this issue in Chapter 6.)

Despite its initial appeal, PGP 2 did not gain widespread acceptance. Commonly cited reasons at the time were that PGP was difficult to centrally manage, PGP did not come with licenses for the patented public key technology that it employed, and PGP was a separate program that did not transparently interoperate with existing email systems. Some of these objections were overcome with the introduction of commercial PGP version in 1997 that included all necessary patent licenses and plug-ins that let PGP interoperate with popular email systems such as Microsoft Outlook and Eudora. PGP message formats were eventually standardized by RFCs 1991, 2015 and

2440. [ASZ96, Elk96, CDFT98] Nevertheless, by all accounts PGP has failed to gain widespread penetration.

5.1.3 S/MIME usability today

Modern S/MIME clients address many of the usability errors that Whitten and Tygar identified in PGP 5.0:

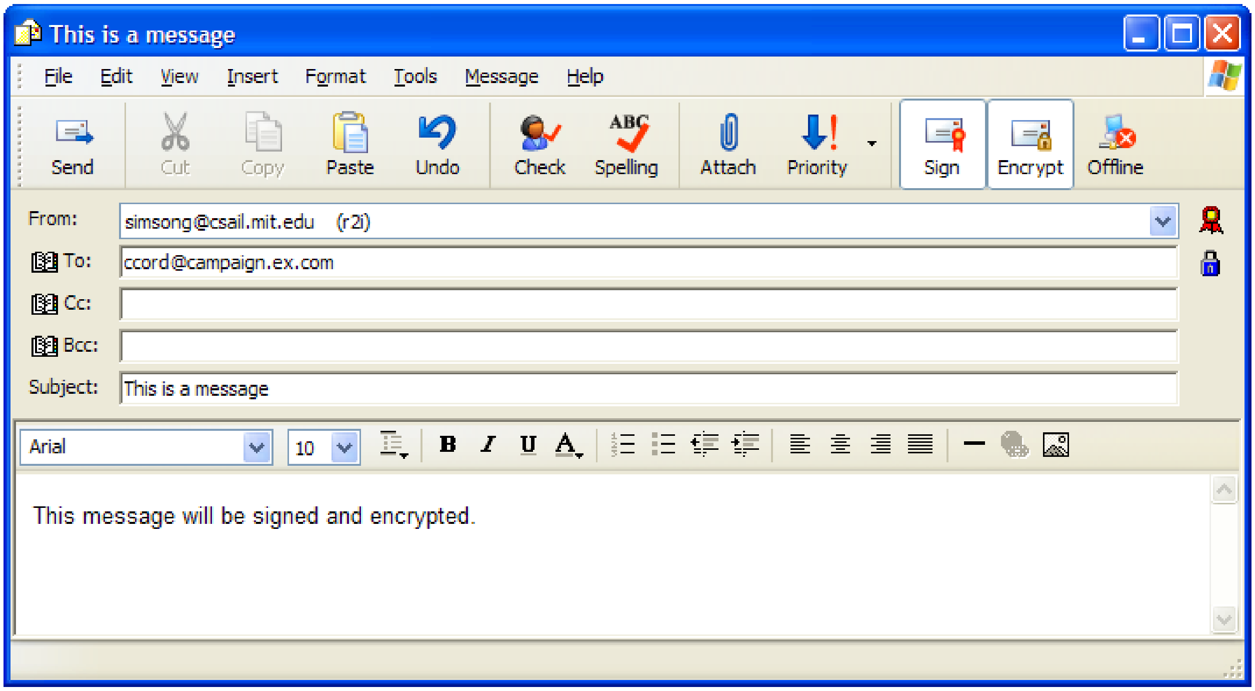
- Whereas PGP 5.0 supported two incompatible key types, forcing users to manually determine which kind of key to use for which kind of recipient, S/MIME supports but one key type and has a mandatory set of required encryption algorithms.
- Whereas message unsealing with PGP 5.0 was manual, unsealing with Outlook Express and similar programs is automatic: if the mail client receives a sealed message and the client possesses the matching private key, the message is unsealed.
- Many modern programs have buttons labeled "Encrypt" and "Sign" clearly indicated in the window that is used to compose and send new messages. (Figure 5-4). To digitally sign a message, the user only needs to click the button labeled "sign." Likewise, to seal a message for a recipient, only the "encrypt" button need be clicked.
- The S/MIME standard even automates a rudimentary form of key distribution: when a digitally signed message is sent, that message comes with a copy of public key certificate that can be used to verify the message. This certificate is automatically copied from the message into the user's address book, allowing the recipient of a signed message to respond with a sealed one simply by hitting the "reply" button, should the recipient wish to do so.

To make use of these features, it is necessary for either the S/MIME sender, the recipient, or both to create a public/private key pair and then to obtain an X.509v3 certificate for the public key that has the appropriate S/MIME extensions. Such a certificate is commonly called a *Digital ID*.[2]

For example, if an Outlook Express user wishes to send a piece of digitally signed mail and simply clicks the "Sign" button, then tries to send the message, a pop-up window appears informing the user that she must first obtain a Digital ID before a signed message can be sent (Figure 5-5). Trying to send a message that is sealed with encryption to a recipient for whom there is no Digital ID on file in the sender's OE6 Address Book generates a similar warning, this time giving the user a choice between aborting the send or sending the message without encryption (Figure 5-6).

Thus, it seems that modern S/MIME systems have simply replaced the difficulty in using the software (identified by Whitten and Tygar [WT99]) with the difficult of obtaining a Digital ID. Issues surrounding the difficulty of obtaining S/MIME certificates, and possible solutions, are discussed in Chapter 6.

[2] John C. Brezina applied for the service mark *Digital ID* on September 30, 1991 and abandoned on July 15, 1992. [Joh91]; VeriSign applied for Digital ID as a service mark on September 3, 1996 but abandoned the application on September 23, 1997. [Ver96] It thus appears that the term *Digital ID* can be used without risk of trademark infringement, at least in the United States.



From: simsong@csail.mit.edu (r2i)

To: ccord@campaign.ex.com

Cc:

Bcc:

Subject: This is a message

This message will be signed and encrypted.

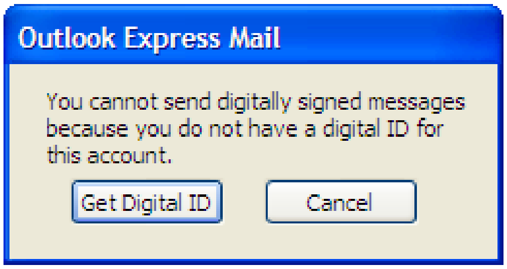
Figure 5-4: The toolbar of Outlook Express 6 allows messages to be signed or sealed ("Encrypted") simply by clicking a button. The little certificate icon to the right of the "From:" field indicates that the message will be signed, while the little padlock icon next to the "To:" field indicates that the message will be sealed for the recipient. Lotus Notes, Mozilla Thunderbird, and Apple Mail have similar provisions for sending mail that is signed and/or sealed

5.1.4 Closed systems: high usability in small communities

It is important to note that a variety of systems have been created and deployed that allow even relatively unsophisticated users to send and receive email with many cryptographic protections. These systems are typically integrated solutions in which keys are automatically created and distributed whenever new accounts are added by the system's manager.

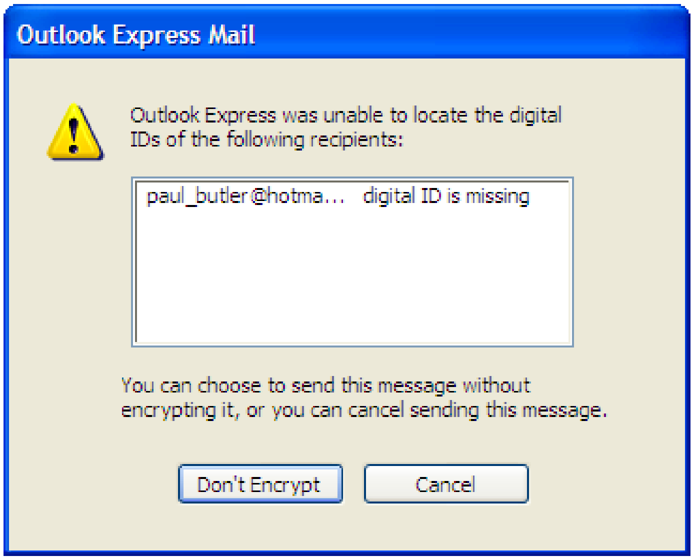
Examples of the such systems include Notes [Zur05b], Groove,[MBA05] and HushMail[Hus05]. Zurko states that there are more than 100 million Lotus Notes users, indicating that such systems can be used by very large user populations—although these users exist in separate certification hierarchies. Another factor simplifying Notes deployment is that the users of Notes systems generally have pre-existing relationships with the organizations using Notes—most often they are employees and have already had their identities certified. This is a prime example of the LEVERAGE EXISTING IDENTIFICATION pattern.

HushMail uses the OpenPGP standard RFC 2240, demonstrating that the IETF standards can be used in a manner that is both secure and usable in webmail systems. Alternatively, existing standards can be implemented with a proxy between the user's mail client and the mail server that automatically and transparently encrypts mail as it is sent and decrypts mail as it is received. [BS99, Gar03b, Per03, Rot05] Appendix D on page 413 describes one such proxy, Stream. Some of these systems use existing keys and certificates, while others generate and distribute keys and certificates as needed. But despite the technical appeal of such solutions, their existence has not made secure email commonplace.



You cannot send digitally signed messages because you do not have a digital ID for this account.

Figure 5-5: This warning appears if an OE6 user attempts to send a signed message and there is no Digital ID on file for the sender.



Outlook Express was unable to locate the digital IDs of the following recipients:

paul_butler@hotma... digital ID is missing

You can choose to send this message without encrypting it, or you can cancel sending this message.

Figure 5-6: This warning appears if an OE6 user attempts to send a sealed message and there is no Digital ID for the recipient in the sender's Address Book

5.2 A Survey of Secure Email Capabilities and Attitudes

Section 5.1 argued that two decades of effort has resulted in the widespread deployment of email encryption software and the use of that software in closed communities. It also showed that digitally signed but unsealed messages have a lower hurdle to adoption than mail that is both signed and sealed. But is the software really on people's desktops? And is signing enough?

This section discusses some results of a survey conducted in August 2004 of merchants in the US and Europe who were selling items on the Amazon.com web site. The survey appears to be the first reported in the open literature to examine the impact of receiving digitally signed messages on knowledge of and attitudes towards secure email. Results of the survey have been previously reported in [GSN$^+$05] and [GNM$^+$05]; only the results of the survey that are critical to the justifying this dissertation's arguments are presented here. Additional survey results appear in Appendix B.

5.2.1 Prior Work on Security Attitudes and Email Usage

There are few published studies that directly discuss popular attitudes towards encryption or other security technologies for achieving security or privacy. One is the 10th GVU WWW User Survey[GVU99], which found that a majority of respondents described themselves very (52.8%) or somewhat (26.7%) concerned about security. When asked what is "the most important issue facing the Internet," the answer most frequently selected by GVU's respondents was "privacy" (19.1%); "security of e-commerce" ranked 8th, garnering just 5% of the votes. That study was conducted six years ago and attitudes have probably changed in the intervening time.

There are also remarkably few publicly available studies that track the adoption rates and relative market share of email clients. One source cited by Garrett is Jupitermedia's Clickz Stats. [Gar04c] The percentages from Clickz Stats reported in Garrett's article are reprinted in Figure 5-7; neither Clickz Stats nor Jupitermedia responded to repeated requests for additional information.

		S/MIME
Email Client	Percentage	Enabled?
Microsoft Outlook	39.14 %	✔
Hotmail	25.82 %	
Microsoft Outlook Express	25.20 %	✔
Yahoo! Mail	19.67 %	
Other	19.06 %	?
Lotus Notes	6.35 %	✔
Netscape	5.33 %	✔*
AOL 7.0	4.92 %	
Eudora	4.30 %	
Unix Command-Line Based	1.43 %	
AOL 6.0	0.61 %	
AOL 5.0	0.61 %	
Juno	0.61 %	
AOL 4.0 or lower	0 %	

Figure 5-7: According to the market research firm Clickz Stats, part of Jupitermedia, more than half of the users that they queried have the ability to receive S/MIME-signed mail. (Users were asked "Which of these email clients do you use at work?" and were allowed to select more than one client from the list.) Because multiple selection was permitted and Clickz Stats has not provided access to the raw data, the overall percentage of users who had S/MIME-enabled clients cannot be determined. *Note that the answer "Netscape" is ambiguous, since Netscape Communicator supports S/MIME, but Netscape's webmail service does not. In all probability, the respondents were indicating that they were using Netscape Communicator on their desktop.[Gar04c]

5.2.2 Genesis of the survey

EU Directive 99/93/EU calls for the use of advanced electronic signatures for certain kinds of electronic messages. "Advanced electronic signatures" are generally taken to mean digital signatures, signed with a private key, that permits the recipient to determine whether or not the contents of the document were modified after the document was sent.[3]

Amazon Services Europe S.à r.l. started sending signed electronic Value Added Tax (VAT) invoices to each of its Amazon Marketplace, Auctions, and zShops sellers in June 2003. Amazon's signatures were S/MIME digital signatures certified by a VeriSign Class 1 Digital ID. At the time, Amazon did not send digitally signed messages to its sellers operating in America, Asia, and other geographic regions.

Because a substantial number of Amazon's sellers had been receiving digitally signed messages, the decision was made to survey them to determine if the sellers had been able to verify the signatures. By comparing the merchants who had received the digitally signed messages with those who had not, we also hoped to see if the act of receiving the messages had any discernible on the sellers' attitudes, or knowledge of cryptographic.

[3] Bohm *et al.* argue that Directive 1999/93/EC's requirements on "advanced electronic signatures" cannot be fulfilled because requirement 2(c) is for a signature that "is created using means that the signatory can maintain under his sole control." "We have concluded that neither PCs nor smartcards nor biometrics nor any methods currently available or likely to be available in the near future can enable a user to keep a signature key secure; and it follows in our view that condition 2(c) cannot be fulfilled, and that no advanced electronic signatures can be made."[BBG00]

"What's your highest level of education?"	ALL	Europe		US		Savvy		Green	
Some high school	2%	4%		1%		**4%**	*	**1%**	*
Completed high school	7%	**16%**	**	**5%**	**	8%		7%	
Some college	30%	27%		31%		31%		29%	
College degree	35%	30%		36%		**27%**	*	**39%**	*
Advanced degree	26%	23%		27%		29%		25%	
Total Respondents	410	74		336		137		273	
No Response	(7)	(1)		(6)		(1)		(6)	

$^{*}p < .05$; $^{**}p < .01$;

Table 5.1: Respondents were asked "What's your highest level of education:"

Digital signatures ensure the *integrity* of email, but did the recipients of the signed email think that such messages were more *trustworthy* or more likely to be *truthful* than messages that were not digitally signed? Did the sellers even know what a digital-signature was? How did receiving these signatures change the seller's opinion of Amazon? And to what other purposes did the sellers think digital certification should be applied? These were the questions that the mail security survey sought to answer.

5.2.3 Survey methodology

The survey consisted of 40 questions on 5 web pages. Respondents were recruited through a set of notices placed by Amazon employees in a variety of Amazon Seller's Forums. Participation was voluntary and all respondents were anonymous. Respondents from Europe and The United States were distinguished through the use of different URLs.[4] A cookie deposited on the respondent's web browser prevented the same respondent from easily filling out the survey multiple times.

A total of 1083 respondents clicked on the link that was posted in the Amazon forums in August 2004. Of these, 469 submitted the first web page, and 417 completed all five pages.

Respondent demographics

The average age of respondents was 41.5. Of the 411 who answered the question, 53.5% identified themselves as female, 42.6% as male, and 3.9% chose "Declined to answer." The sample was highly educated, with more than half claiming to have an advanced degree (26.1%) or a college degree (34.9%), and another 30.0% claiming some college education. More than three quarters described themselves as "very sophisticated" (18.0%) or "comfortable" (63.7%) at using computers and the Internet. Roughly half of the respondents had obtained their first email account in the 1990s, with one quarter getting their accounts before 1990 and one quarter getting their accounts after 1999.

When asked to rate their "understanding of encryption and digital signatures" on a 5 point scale, where 1 was "very good" and 5 was "none," the average response was 3.6, but the spread was large, indicating that respondents had a wide range of familiarity with the topic. (Table 5.2)

[4] This recruitment strategy may represent a methodological flaw in the survey: we should have explicitly asked respondents which country they were in. From reading the comments, however, it appears that the select based on source URL was accurate in distinguishing those from Europe and Great Britain from those in the US.

Very				
Good				None
“1”	“2”	“3”	“4”	“5”
5.1%	11.6%	24.6%	31.4%	27.3%
(23)	(53)	(112)	(143)	(124)
$N = 455$				

Table 5.2: When asked “On a scale of 1 to 5, where 1 is ”very good” and 5 is ”none,” please rate your understanding of encryption and digital signatures,” respondents indicated that they had a broad range of familiarity with the topic.

5.2.4 Awareness of cryptographic capabilities

It is important to know both how many of email recipients can verify digitally signed mail and also how many recipients are aware that they posses this capability. Our theory was that most had this capability but were not aware of it—thus, any survey of mail respondents asking them if they could receive signed mail would likely yield incorrect results. The survey confirmed this hypothesis.

Overall, the majority of survey respondents were either not aware of the cryptographic capabilities of their email programs (59%) or unaware what was meant by the phrase “encryption” (9%). (Table 5.3) By asking the respondents “Which computer programs do you use to read your email? Check all that apply,” we were able to determine that approximately 81% of the respondents were reading their email with programs that supported the S/MIME encryption standard. (Table 5.4)

Performing a cross-tabulation analysis between these two questions, we found that users of S/MIME-enabled programs were generally more aware of the cryptographic capabilities of their software that users who were not ($p < .001$). Those results are also presented in Table 5.3.

Awareness of digitally signed mail

Not surprisingly, the respondent’s lack of familiarity with the cryptographic capabilities of their software was matched by their unawareness as to whether the capabilities had been used or not.

To perform this analysis, we divided our sample according to whether they accessed the survey from the URL that was posted to the Amazon forums frequented by European sellers or those accessed by American sellers. We call these groups *Europe*, with 93 respondents, and *US*, with 376 respondents.

Recall that Amazon had been sending sellers in the *Europe* group digitally signed email since June 2003, while those in the *US* group have never been sent digitally signed email from Amazon. Reportedly a few recipients of digitally signed messages had sent messages back to Amazon exclaiming “what is this `smime.p7s` attachment? I can’t read it!” But the vast majority of them did not comment at all with regards to the digitally signed messages.

As shown in Table 5.5, only a third of the *Europe* merchants who had received a digitally signed message from Amazon were aware of the fact. As expected, the number is higher than the 20% of those in the US group who said that they had received mail that was signed—what’s surprising here is that the US number is so high. An interesting follow-up that we neglected to ask would have been a free-response question asking the respondents to describe the digitally signed message that they had received. This is an opportunity for further research.

		S/MIME-enabled	
ALL		yes	no
Yes	27%	**34%*****	**14%*****
No	5%	5%	5%
I don't know	59%	**54%***	**66%***
What's encryption?	9%	**7%****	**14%****
Total Respondents	446	291	155
No Response	(8)	(1)	(7)

$^{*}p < .05$; $^{**}p < .01$; $^{***}p < .001$;

Table 5.3: Despite the fact that merchants had the ability to handle S/MIME-signed or sealed mail, most were not aware of this fact. (Answers to the question "Does your email client handle encryption?"[GNM+05])

Mail Client		S/MIME Enabled ?
Outlook Express	41.8 %	✔
Outlook	30.6 %	✔
AOL	17.9 %	
Netscape	10.1 %	✔
Eudora	6.9 %	
Mozilla Mail	3.2 %	✔
Apple Mail	2.5 %	✔
Lotus Notes	2.1 %	✔
Evolution	0.9 %	✔
Any S/MIME capable program	81.1%	✔
Total Respondents	435	
No Response	(19)	

Table 5.4: According to the Amazon.com mail security survey, more than three-quarters of respondents have the ability to verify S/MIME-signed mail. (Amazon.com merchant responses to the question "Which computer programs do you use to read your email? Check all that apply."[GNM+05])

More curious is that 16% of those in Europe said that they had received mail that had been "sealed with encryption." What encryption system were these merchants using to receive the encrypted mail? Was it webmail over an SSL-enabled web site, or had they received password-protected Adobe Acrobat files, or did these merchants think that the *signed* mail from Amazon was in fact *sealed*? We neglected to ask. This is also an opportunity for further research.

Clues for answering these questions can be found in the free-format comments that our respondents were invited to write at the bottom of every page. Respondent 30130 appeared to believe that by "encrypted" we were in fact asking if they had used email or messaging at a secure site: "I believe encrypted means a secure site?" (30130, Europe)[5]

But some respondents clearly had some kind of experience or knowledge of cryptography:

> *Your survey did not address the fact that any email containing credit card information should be encrypted. We get emails from customers almost every day with card numbers with orders, rather than using our secure systems on our sales sites. It is more common than I would ever have believed. (30142, US)*

> *I use TurnPike, which is supplied with PGP preconfigured for signing and encryption.... But in the several years since I have installed it, I have never used it for encrypting email, or*

[5] When specific comments from respondents are quoted, the values in the parenthesis indicates the subject's unique identifier—a five-digit number beginning with a "3"—and the word "Europe" or "US" to indicate if the respnodent entered the survey through the URL posted to the European Seller's forum or the US Seller's forum.

"What kinds of email have you received? Please check all that apply:"	ALL	Europe	US
Email that was digitally signed	22%	**33%****	**20%****
Email that was sealed with encryption so that only I could read it.	9%	**16%***	**7%***
Email that was both signed and sealed.	7%	10%	6%
I do not think that I have received messages that were signed or sealed.	37%	30%	39%
I have not received messages that were signed or sealed.	21%	23%	20%
I'm sorry, I don't understand what you mean by "signed," "sealed" and "encrypted".	26%	**17%***	**28%***
Total Respondents	455	88	367
No Response	(15)	(5)	(9)

$^{*}p < .05$; $^{**}p < .01$;

Table 5.5: Asked what kinds of email they had received, many respondents in the survey thought that they had received mail that was signed, sealed, or both.[GSN+05]

> *sending signed email. I have received and verified signed email from my ISP. I have never received signed email from any other source. (30468, Europe)*
>
> *use dig. signature + encryption at work only (30498, Europe)*
>
> *I liked PGP a lot, but hardly anybody seems to be using it... (30504, Europe)*
>
> *I would use encryption more if more of my friends did. Normally I think it's secure etc but I bet the government somehow has a back door (30649, US)*
>
> *Encryption is only as useful as the ability of the sender and receiver being able to access, use, and decipher it. PGP is great unless you have users that are unable to use it without more hassles or inconvience. Secruity is an issue best left to the receiver's needs in my opinion, not the sender in 99% of internet situations.(30899, US)*
>
> *I played around w/Pretty Good Privacy program a long time ago, but no one I knew used it. I would love to be able to keep snoops out. I am also concerned with privacy issues due to "Homeland Security", and feel that the government has misused it's power in the past, and is likely to do so in the future. (30909, US)*
>
> *Would love to, but had trouble quickly understanding PGP - too busy to learn at length. (30938, US)*

5.2.5 Segmenting the respondents

In the previous section we examined the impact that having previously received digitally signed mail might have had on our respondents. In the process, we saw that respondents have considerable breadth of background when it came to self-reported experience with cryptography.

To see if background might impact views, we decided to examine a second partitioning of respondents into two new groups: *Savvy*, those who indicated that they had some familiarity with cryptography, and *Green*, those who did not.

A respondent was put into the *Savvy* group if any of the following conditions were true:

- The respondent answered 1 ("very good") or 2 when asked to rate their "understanding of encryption and digital signatures" on a 5-point scale (with 5 listed as "none")—23 and 53 respondents, respectively;[6]
- The respondent indicated that he or she had received a digitally signed message (104 respondents);
- The respondent indicated that he or she had received a message that was sealed with encryption (39 respondents);
- The respondent said they "always," or "sometimes," send digitally signed messages (29 respondents);

We did not include the 4 respondents who said that they "always" send email that is sealed for the recipient in the *Savvy* group, assuming that these individuals had misunderstood the question.

A total of 148 respondents met one or more of the *Savvy* criteria. Those 321 respondents not in the *Savvy* group were put in a second group called *Green*.

Thus, the *Europe/US* division measures the impact on attitudes given the actual experience in receiving digitally signed mail from Amazon, while the *Savvy/Green* division measures the impact of people's stated knowledge of or experience with both digital signatures and message sealing.

As before, the results of partitioning the respondents into two groups was deemed to be statistically significant if a logistic regression based on a Chi-Square test yielded a confidence level of $p = 0.05$ for the particular response in question.

We performed analysis in terms of education for both partitionings. Overall, both the *Europe* and *Savvy* groups were younger ($\overline{\text{age}} = 36.2$ vs. 42.7 years) and less educated (see Table 5.1) than their *US* and *Green* counterparts—differences that were statistically significant, although perhaps not very relevant.

5.2.6 Appropriate uses of signing and sealing

Some cryptography enthusiasts have argued that encryption should be easy-to-use and ubiquitous—and that virtually all digital messages should be sealed, at least, and probably signed with anonymous or self-signed keys.[Hug93]

Our respondents felt otherwise. In a series of questions aimed at determining what kinds of email messages they thought should receive protection, respondents indicated that matters involving money or government were worthy of protection, while personal email messages generally were not.[7]

[6] We asked our segmenting questions before defining terms such as *encryption* and *digital signature*. Although this decision resulted in some criticism from respondents, we wanted to select those in the *Savvy* based on their familiarity with the terminology of public key cryptography (e.g. "digitally sign," "encrypt"), rather than the underlying concepts, since user interfaces generally present the terminology without explanation.

[7] Specifically, 35% of all respondents thought that personal email sent or received at work did not require any protection, although 10% agreed with the statement that personal email "should never be sent or received at work." At home, 51% thought that personal email did not need any cryptographic protection.

E-commerce related email:

Bank or credit-card statements	65%
Receipts from online merchants	59%
Questions to online merchants	33%
*Savvy**	26%
*Green**	36%
Advertisements	17%

General Email:

Tax returns or complaints to regulators	74%
Personal mail sent or received at work	40%
Personal mail sent or received at home	40%
Mail to political leaders voicing opinion	38%
Newsletters from politicians	22%

*$p < .05$

Figure 5-8: Percentage of respondents in the August 2004 Mail Security Survey who thought a particular kind of email required the use of digital signatures, by mail type. Most respondents thought that digital signatures should be used for financial statements, receipts from online merchants, and official correspondence to government agencies sent through email. No statistically significant differences were seen between the Europe and US groups, or between the Savvy and Green groups, except where noted.

Surprisingly, when summary statistics alone were considered, no statistically significant difference was seen in the answers of those in the *Europe* and *US* groups with respect to the appropriateness of digitally signing email. Only statistically significant difference was seen between the *Savvy* and the *Green* groups: roughly 40% more *Green* people thought that questions to online merchants should be digitally signed than *Savvy* people. Apparently, familiarity with the technology made these respondents think that the technology was less important to use in this application.

Summary results of all email appropriateness questions are shown in Figure 5-8.

5.2.7 Why don't people use email security?

Despite the fact that the majority of respondents thought that security should be used, it appears that very few of them actually use the technology. The evidence for this claim is drawn from the first page of the survey, in which we asked our users whether or not they send email that is digitally signed or sealed with encryption. These results are presented in Tables 5.6 and 5.7, respectively. It turns out that very few (33 out of 470) of our respondents indicated that they digitally signed or sealed their mail "sometimes" or "always."

Although roughly half of our respondents indicated that they didn't use cryptography because they didn't know how, the free-response answers from the more knowledgeable respondents indicated that they either didn't think that encryption was necessary or else that the effort, if made, would be wasted.

> *I don't because I don't care. (30154, US)*

Survey Response (multiple selections allowed)	
I always send email that is sealed for the recipient.	0.9%
I sometimes send email that is sealed.	3.5%
I rarely send email that is sealed because it is not necessary for the kind of mail that I send.	16.7%
I rarely send email that is sealed because I just don't care.	7.9%
I don't send email that is sealed because it is too hard to do.	5.7%
I don't send email that is sealed because I don't know how.	41.0%
I don't send email that is sealed because I am worried that the recipient won't be able to read it.	14.3%
I'm sorry, but I don't understand what you mean by "sealed" or "encrypted".	22.0%
Other	3.3%
Total Respondents	454
No Response	(16)

Table 5.6: "Do you send email that is sealed with encryption so that it can only be read by the recipient? Please check all that apply."

Survey Response (choose one)	
I always send my email digitally signed.	2.2%
I sometimes send email that is digitally signed.	4.2%
I rarely send email that is signed because it is not necessary for the kind of mail that I send.	19.2%
I usually don't because I don't care enough to sign my email.	9.9%
I don't ever send email that is digitally signed because I don't know how.	44.8%
I'm sorry, but I don't understand what you mean by "digitally signed."	24.1%
Other	3.8%
Total Respondents	453
No Response	(17)

Table 5.7: "Do you send digitally signed mail? Please check all that apply."

> *I doubt any of my usual recipients would understand the significance of the signature. (30468, Europe)*

> *Never had the need to send these kinds of emails. (30391, US)*

> *I don't think it's necessary to encrypt my email & frankly it's just another step & something else I don't have time for! (30220, US)*

These statistics and free-form comments are particularly significant in light of the fact that 25.2% of our respondents thought that receipts sent by online merchants should be digitally signed, while 33.6% thought that they should both be signed and sealed![GSN$^+$05] Remember, all respondents are themselves Amazon.com online merchants!

5.2.8 Signature interfaces and metaphors

As the S/MIME RFCs are silent as to how the presence of a valid digital signature should be displayed, different programs employ visible indications, as shown in Figures 5-10 and 5-13.

We asked our respondents how they would like their email programs to indicate that a message has a valid digital signature. Roughly equal numbers (44% vs. 41%) said that they would like the one-line of text added to the header interface (as shown in Figure 5-13) as a ribbon or certificate that is shown when the message is displayed in a list (as shown in Figure 5-10). Roughly a quarter (24%) agreed with the statement that they "would like to see a signature at the bottom of the message, as if it was signed in ink." Users of encryption favored the ink metaphor to non-users, 31% to 22%, a statistically significant difference ($p < .05$).

We also asked what respondents thought a "good description" of a digitally signed message would be. Respondents could chose one of five choices or provide their own answer; a plurality of respondents (37.3%) agreed that a digital signature is "like signing your name at the bottom of a message." Next were the 30.7% who believed that a signature is "like putting your fingerprint at the bottom of a message," followed by the 27.5% who agreed that a signature was "like having the message notarized," No statistically significant differences were seen between users and non-users, although we did see statistically significant differences the *Europe* and *US* samples, with more Europeans (43% vs. 28%) preferring the fingerprint metaphor, and more Americans (30% vs. 15%) prefering the notarized metaphor.

Our analysis of the metaphor question indicates that users don't have strong metaphors or analogies for what it means to digitally sign mail. This may be a reflection of the fact that the technology itself is somewhat ambiguous, providing both *integrity protection* and *sender identification*. What is frequently left unresolved, in both user interfaces and documentation, is whether or not sending digitally signed mail is meant to convey some form of intentionality as well. This confusion is mirrored in the offline world. For example, to have a document notarized in the United States merely means that the signature on the document was witnessed by a commissioned officer of the state; it is no guarantee of the veracity of the document's contents. Nevertheless, the idea that notarized documents are somehow more trustworthy is a misconception that is commonly presented in American media. In fact, notarized documents are not more likely to be truthful—and neither are messages that are digitally signed.

5.2.9 Free-format responses

Our survey contained many places where respondents could give free-format responses. Many wrote that they wished they knew more about email security. For example:

> *I wish I knew more about digitally signed and sealed encrypted e-mail, and I wish information were more generally available and presented in a manner that is clear to those who aren't computer scientists or engineers. (30346, US)*

> *This is an interesting topic... I had not thought about the need to send/receive signed or sealed e-mail for other than tax info. (30391, US)*

Others do not understand cryptography and do not want to learn:

> *Most sellers do not care about digital signatures when selling on on-line marketplaces unless they are dealing in big sums of money in the transaction, even then I still do not care. (30014, US)*
>
> *I think it's a good idea, but I'm lazy and it's too much trouble to bother with. (30154, US)*
>
> *It still seems too complicated for ordinary home-based computer users. More and more encryption and other safeguards seem increasingly necessary. However, the technology still has some wrinkles to iron out in making it more user-friendly. (30076, US)*
>
> *I would be somewhat scared to use encryption as I often forget passcodes now and would most likely lose the "key" (30222, US)*

These comments, and many others, reinforce our belief that the usability standards for a successfully deployed email security system must be extraordinarily high. It is not enough for systems to be easily learned or used, as Whitten argues. [Whi04a] Security information should be conveyed passively, providing more detailed information on demand, but should not otherwise impact on standard operations.

Spam, viruses and phishing

Many respondents used the free-format response sections to complain about spam, viruses, and phishing—sometimes to the point of chastising us for not working on these problems:

> *I hope this [survey] will help to stop the viruses, spam, spyware and hijackers all too prevalent on the web. (30029, US)*
>
> *[I] feel the topic is somehow "phony" because of the way viruses are transmitted by email. I'm more concerned with attacks by future NIMDAs*[8] *than I am with sending or receiving signed email. (30281, US)*
>
> *Digital signatures would cut down on SPAM and the Nigerian scams. Moreover, encryption would protect receipts, credit card card and billing statements, as well as those from banks. (30082, US)*
>
> *I have received many "phishing" e-mails through the years. Although I always forward them to the appropriate authorities, I worry about others who may fall prey to them. I think digital signing would be a way to help the problem, but I don't think it would end the problem. There are still far too many people who will willingly give their banking information to "Nigerian Officials" or other scammers. (30265, US)*

Several respondents noted that there is little need to send sealed email, since such messages can be sent securely using feedback forms on SSL-encrypted web sites.

[8] W32/Nimda was an email worm that was released in September 2001 and affected large parts of the Internet.[CER01]

5.2.10 Survey conclusions

We surveyed hundreds of people actively involved in the business of e-commerce as to their views on and experience with digitally signed email. Although they had not received prior notification of the fact, some of these individuals had been receiving digitally signed email for more than a year. To the best of our knowledge this is the first survey of its kind.

It is widely believed that people will not use cryptographic techniques to protect email unless it is extraordinarily easy to use. We showed that even relatively unsophisticated computer users who do not send digitally signed mail nevertheless believe that it should be used to protect the email that they themselves are sending (and to a lesser extent, receiving as well).

We found that the majority (58.5%) of respondents did not know whether or not the program that they used to read their mail handled encryption, even though the vast majority (81.1%) use such mail clients. Given this case, companies that survey their customers as to whether or not the customers have encryption-capable mail readers are likely to yield erroneous results.

We learned that digitally signed mail tends to increase the recipient's trust in the email infrastructure. We learned that despite more than a decade of confusion over multiple standards for secure email, there are now few if any usability barriers to receiving mail that's digitally signed with S/MIME signatures using established CAs.

Finally, we found that people with no obvious interest in selling or otherwise promoting cryptographic technology believe that many email messages sent today without protection should be either digitally signed, sealed with encryption, or both.

5.2.11 Future work

Comments from merchants make it clear that there are many opportunities for future survey work to document needs and current business practices:

> *The concepts of digital signing & encryption for email new to me. Glad your working to give the bad guys a harder time. Shame we need it. Would it stop spam? Need simple info & guidelines for learners like me. I get confused by computer jargon, glad this survey did not use it. (31085, Europe)*

> *I receive digitally signed email only from a couple people, and it's mostly annoying and time-wasting, and I'm not sure those aren't using it because they don't know how to turn it off. I'm sure these applications are useful to particular businesses, but I'm not aware that they affect most computer users at all. (30642, US)*

Although the Pew Internet Life Project[PEW05] has done numerous surveys on Internet use and opinions, the project has not addressed specifics of security technology to the extent that we have. A follow-up survey that looks specifically at the need, use and acceptance of security technology would be helpful. Such work could be done with Pew, as the organization has significant research and methodological tools that are unavailable to individual researchers.

5.3 Signatures Without Sealing

Given the acknowledged difficulties that have been encountered in trying to deploy secure mail that provides both signing and sealing for every message, it seems reasonable to instead shoot for an attainable intermediate goal. Once such goal would be for organizations sending large quantities of automated or do-not-reply email to simply commit that this mail be sent with S/MIME signatures.

"Automated email" is a large category of electronic messages that are automatically generated, usually in the course of an e-commerce transaction, but which are intended to be read by an individual. Do-not-reply mail is mail that is sent out by a sender with an explicit note telling recipients something to the effect of "do not reply do this message." Examples of such messages includes auction bid confirmations, messages from payment providers, routine messages from credit-card companies and advertisements.

Although digital signatures do not protect the contents of an email message from being intercepted while that message is *enroute*, there are nevertheless many benefits that can be had from signing alone:

- A digital signature on an advertisement allows the recipient to verify the sender of the message and to know that the advertisement's prices in the advertisement have not been inadvertently altered.
- A digital signature would allow the recipient to readily distinguish between a message that was actually sent from the machine of the sender and one in which the sender's `From:` address was forged by a third-party. Many worms in the Klez family use this technique to make it difficult to locate machines that they have infected. Although digital signatures do not prevent an infected machine from sending out messages that are signed with a private key that resides on the machine itself, such messages will point directly back to the infected machine and make it easier to eradicate the infections.[Sym04]
- Digital signatures would complicate phishing attacks. Currently those engaged in phishing can send out official-looking messages that claim to have a return address of something like `support@paypal.com`. Although attackers could send out messages that are signed from such a domain, they could not send out messages signed with the same key as official messages. Client-side software could distinguish messages signed with one key from messages signed with another.
- By sending a message that is digitally signed, the sender would be giving the recipient the option of responding to the message with a message that is digitally sealed by distributing the sender's Digital ID.
- A majority of the merchants who responded in our survey believe that it is appropriate for invoices, bills, statements, and other kinds of financial e-mail to be signed.
- Sending out signed messages may convey the impression that the sending organization is concerned about security issues and is employing technologically advanced measures to help combat spam and phishing attacks.

If there are so many advantages to sending out email that is digitally signed, why aren't organizations doing so? Three factors may be at work:

1. Institutional inertia.
2. A fear that the S/MIME signature may cause usability problems for some of the recipients.
3. A fear that the organization may be held to a higher legal standard for the content of signed email than the content of email that is not signed. Such a belief may be bolstered by the digital signature laws that were passed in the late 1990s.

The remainder of this section will examine the second and third points. The hope is that by responding to these criticisms, organizational inertia may be overcome in light of the advantages offered by signed email.

5.3.1 Choosing a signature standard

Signed mail is something that cannot be sent in the abstract: email messages must be signed using a specific signature standard with a specific private key. The corresponding public key can be not certified at all, it can be self-certified, or it can be certified by a third-party. Any concrete proposal for sending signed mail needs to clearly specify these parameters before it can be seriously considered.

Complicating the decision of which signature standard to use is the fact that there are three different signing standards currently in use:

1. PGP clear-signed signatures, in which the signature is placed in a text block at the bottom of an ASCII text message (Figure 5-9).[Zim95] PGP's clear signed signatures were adopted early on by CERT for signing the organization's bulletins. Although CERT has now largely stopped the practice of sending out signed ASCII text messages, other organizations such as the FreeBSD foundation continue to do so.
2. OpenPGP MIME, in which the message and the signature are sent as two separate MIME parts in a single message.[ASZ96, Elk96] This message format is supported natively by the Evolution mailers and by the PGP plug-ins.
3. S/MIME signed messages, in which the message and the signature are sent as two separate MIME parts in a single message. This format is supported natively by all S/MIME-enabled mailers.

In addition to these standard, PGP supports two other signature types: the PGP signed message format, in which the signature and the signed message are bundled together in a single binary archive; and the PGP *detached signatures*, in which the file being signed is left unmodified and the signature is placed in a separate file. Although these PGP formats are widely used on the Internet today for signing software distributions, they are not generally used for signing email messages. Other message formats for signing messages includes PEM's provisions for signed messages and the failed S-HTTP standard[Sho95] for signing web pages. Lotus Notes has its own standard for digitally signed messages, but these messages are converted to S/MIME when they are sent over SMTP.

Unfortunately, the design of the OpenPGP and S/MIME formats appears to preclude signing a single message with both signatures. This didn't have to be the case—the designers of the OpenPGP format could have made their implementation orthogonal to the message protection features in

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

This is a message that will be signed
with PGP. It is a very simple message.

-----BEGIN PGP SIGNATURE-----
Version: PGP 6.5.8

iQA/AwUBQjOIL/KaG0LR8e7UEQIosACgus8rixeaxmaF/4dRSeiRlwBCc1YAoMbB
Ot+iT3LqmdZjLz2lVnNdKnLN
=27D3
-----END PGP SIGNATURE-----
```

Figure 5-9: A PGP clear-signed signature

S/MIME, as will be discussed below—but it is a decision that was made. As a result, an organization sending signed messages must choose whether to send each message signed with S/MIME or signed with OpenPGP. (It is possible to use S/MIME to sign a PGP clear-signed message, but this mode of operation has not been widely observed.)

Deciding which signature standard to choose is simplified somewhat by the fact that support for S/MIME is widespread while support for OpenPGP is not. Many programs, including Microsoft Outlook, Outlook Express, Communicator, Thunderbird, and Apple Mail, have both support for S/MIME and are furthermore distributed with CA keys for major CAs such as Thawte and VeriSign that make available Digital IDs to interested parties. Thus, it would seem that messages signed with S/MIME signatures have the highest possibility of being successfully decoded by the recipient.

On the other hand, there is no support for S/MIME in any readily available webmail system with the exception of Microsoft's Outlook Web Access. Likewise, AOL does not support signed messages. In choosing which digital signature standard to use, one must consider the impact of signed mail on these webmail systems as well as on mail clients that do not support the standard in question.

As we shall see in the following sections, it turns out that S/MIME is in fact an excellent choice for a signature standard—not because of any inherent brilliance in the format, but because support for S/MIME is widespread and because S/MIME signatures seem to have minimal usability impact when they are viewed in mail systems that do not have S/MIME support.

5.3.2 Evaluating the usability impact of S/MIME-signed messages

Once a decision is made to send messages with the S/MIME signature standard, a number of questions need to be answered:

1. How do properly signed S/MIME messages appear in S/MIME-enabled readers?
2. How do properly signed S/MIME messages appear in e-mail systems that have no support for

S/MIME?

3. How do S/MIME enabled readers handle messages that are signed with the S/MIME standard, but which cannot be verified for some reason or other?
4. What are the opportunities for an S/MIME-signed message to be damaged while it is *en route*, and how would damage affect signatures?

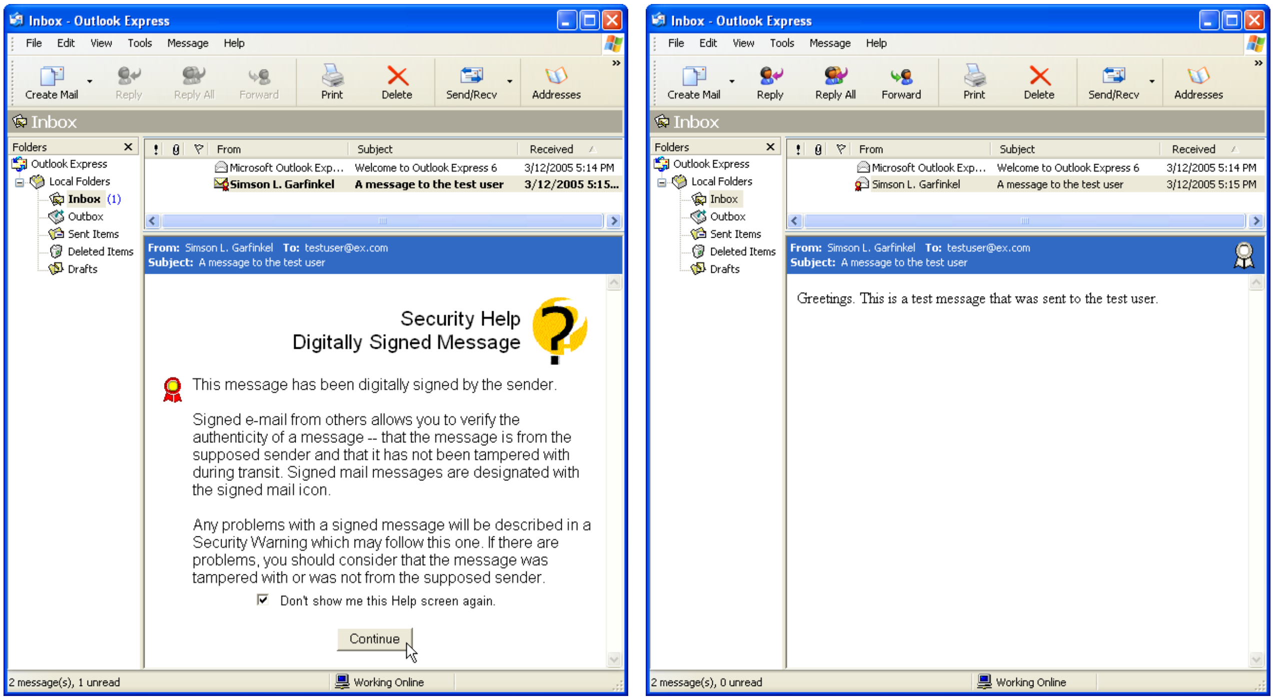
To answer these questions, Thawte FreeMail certificate `0x0d04d8` (#853208) was obtained September 10, 2004, and used to send 6,226 signed S/MIME messages to hundreds of distinct email addresses during the following nine months. Messages were sent using Microsoft Outlook Express, Microsoft Outlook, and Apple Mail to both individuals and mailing lists. Complaints by correspondents were noted. Many test messages were further sent between the mail clients—sometimes with messages passing through mailing lists. Finally, a series of informal interviews were conducted with other users who had similarly tried sending mail that was digitally signed. The results are presented in the remainder of this section.

S/MIME reader, S/MIME-signed message
Today's S/MIME-enabled mail readers differ in the way that they display signed S/MIME messages. The first time that Outlook or Outlook Express receive a signed message, these programs display an informative message to the user that gives a brief explanation about digital signatures, as shown in Figure 5-10 (left). This screen can be thought of as a primitive example of Whitten's "Safe Staging" technique. Outlook Express also annotates a signed message with a small red icon that resembles a second-place ribbon awarded in a dog show. This icon is displayed in the message summary area and in the message preview area.

Clicking on the dog ribbon displays a panel titled "View Certificates" that allows the user to view the Sender's certificate, as shown in Figure 5-11. Confusingly, this panel includes two buttons that perform the same function of viewing the sender's certificate. Pressing either of these buttons causes the Microsoft standard dialogue for viewing certificates to be displayed (Figure 5-12). The panel also includes a button for adding the sender's certificate to the user's address book, which is odd considered that S/MIME certificates are automatically added to the address book when they are received.

At first blush, the "General" certificate properties tab looks more or less reasonable but the "Details," "Certification Path," and "Trust" tabs seem to offer information in a manner that is too detailed for most users to understand. The use of X.509 abbreviations "CN," "O" and "C" (which stand for Common Name, Organization and Country) in "Issuer" line of the "Details" tag are particularly troubling; how is a user supposed to know what this means and what they should do with the information? Indeed, one of the secondary findings of the *Johnny 2* user test described in Chapter 7 is that naïve users who clicked on this dialogue had no idea what to make of any of the information that it presented. Simply seeing lots of numbers, letters and words convinced many of the users that the certificates must be legitimate.

Apple's Mail application displays signed messages with a subtle line saying "Signed:" that is added to the mail header when the message is displayed (Figure 5-13). It is not possible using Mail 10.3 to display the certificate that was used to sign the message. However, receiving a signed message causes the certificate to be added to the user's keychain, where it can be viewed with the



From	Subject	Received
Microsoft Outlook Exp...	Welcome to Outlook Express 6	3/12/2005 5:14 PM
Simson L. Garfinkel	**A message to the test user**	**3/12/2005 5:15...**

From: Simson L. Garfinkel **To:** testuser@ex.com
Subject: A message to the test user

Security Help
Digitally Signed Message

This message has been digitally signed by the sender.

Signed e-mail from others allows you to verify the authenticity of a message -- that the message is from the supposed sender and that it has not been tampered with during transit. Signed mail messages are designated with the signed mail icon.

Any problems with a signed message will be described in a Security Warning which may follow this one. If there are problems, you should consider that the message was tampered with or was not from the supposed sender.

☑ Don't show me this Help screen again.

Continue

2 message(s), 1 unread

From	Subject	Received
Microsoft Outlook Exp...	Welcome to Outlook Express 6	3/12/2005 5:14 PM
Simson L. Garfinkel	A message to the test user	3/12/2005 5:15 PM

From: Simson L. Garfinkel **To:** testuser@ex.com
Subject: A message to the test user

Greetings. This is a test message that was sent to the test user.

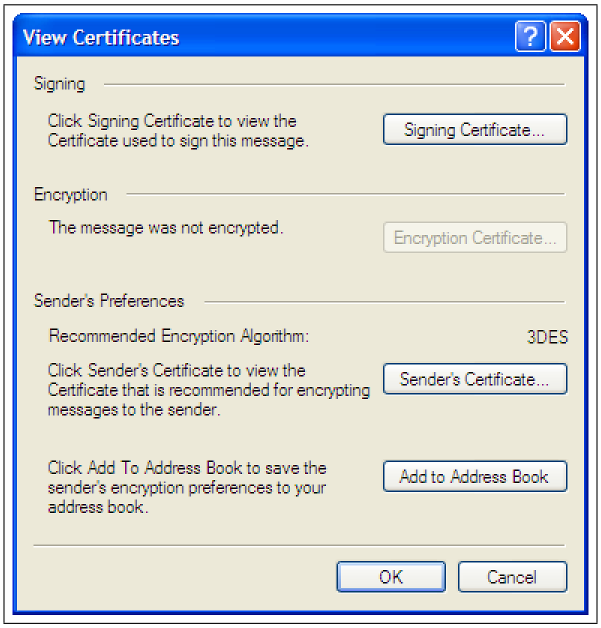
2 message(s), 0 unread

Figure 5-10: The first time the an Outlook Express user receives a digitally signed message, Outlook Express displays this informational message. To prevent the screen from displaying again, the user must click the check-box labeled "Don't show me this Help screen again."

MacOS Keychain application (Figure 5-14). This user interface has many of the same problems as Microsoft's interface: information is not presented in a manner that makes sense to a person who is not a security professional.

The Mozilla tool for viewing certificates is shown in Figure 5-15. An advantage over the Microsoft panel is that the X.509 abbreviations are spelled out in the General tab (although they are still not spelled out in the Details panel). Disadvantages are the fact that the panel displays black text on a dark gray background, that the information presented in the "Details" tab is shown in a tree control which uses a lot of space but doesn't present much information, and once again the fact that the information is not presented in any understandable context.

It is likely that considerable progress could be made in developing a user interface for displaying certificates. For example, the hash visualization techniques discussed Section 2.4.6 on page 62 could be used to augment the display of the certificate fingerprints. (Visualization algorithms would need to be standardized so that a fingerprint displayed in different browsers displayed with the same visualization.) Instead of displaying information like certificate serial numbers in hexadecimal, they could be displayed in decimal notation. Instead of displaying dates using a form that can be misinterpreted (is 9/10/2004 September 10th or October 9th?), the could be displayed in an unambiguous notation (e.g. 2004-SEP-10). The Safari and Mozilla certificate displays could clearly indicate if the date is valid or not, the way the Windows display does. The interfaces could display more information about certificates directly in the interface, rather than hiding it underneath a "help" button.



Click Signing Certificate to view the Certificate used to sign this message.

The message was not encrypted.

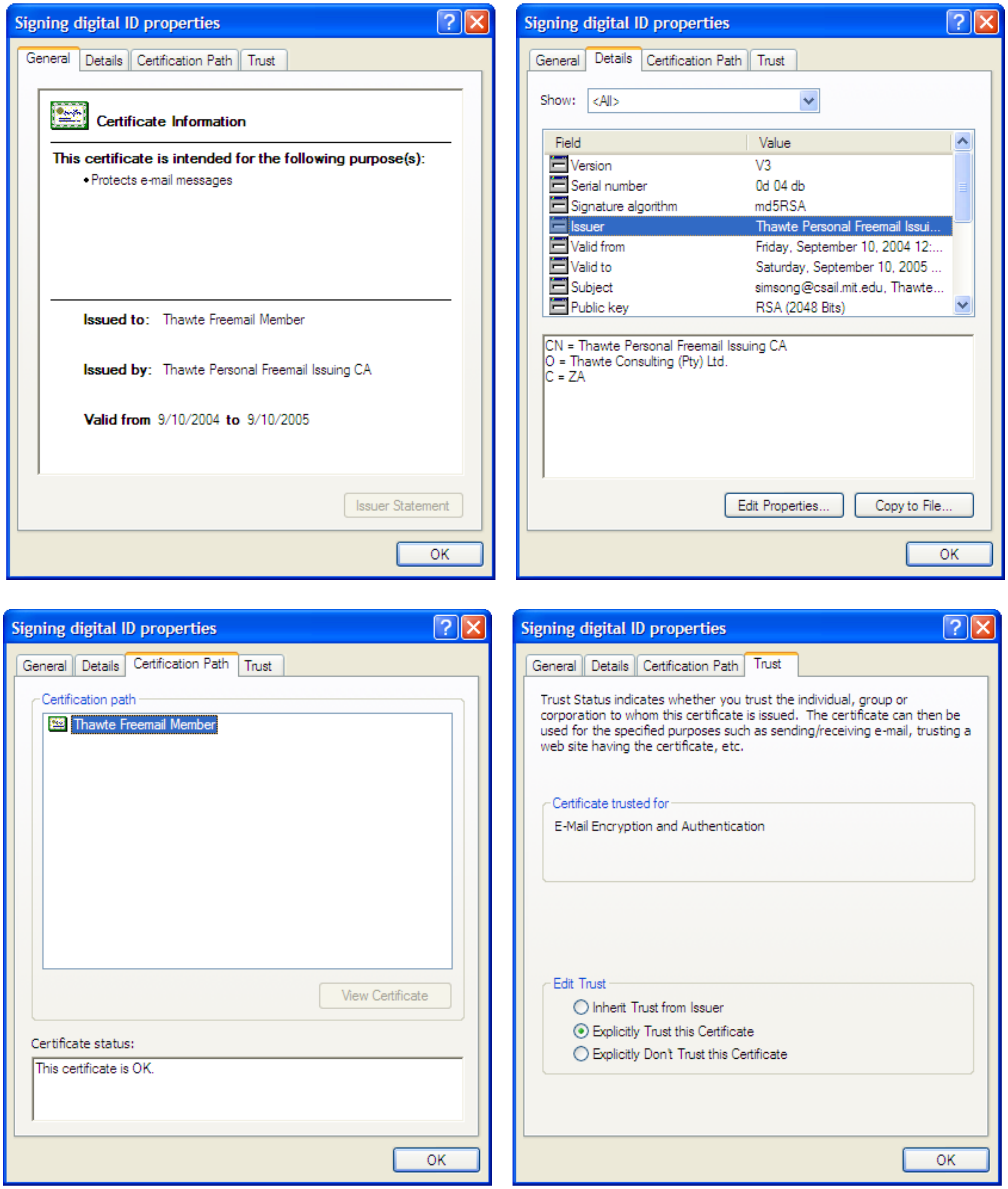
Recommended Encryption Algorithm: 3DES

Click Sender's Certificate to view the Certificate that is recommended for encrypting messages to the sender.

Click Add To Address Book to save the sender's encryption preferences to your address book.

Figure 5-11: Pressing the certificate icon causes Outlook Express to display this dialogue for viewing certificates. Pressing the “Signing Certificate...” button or the “Sender’s Certificate...” button causes the certificate to be viewed using the dialogue panel shown in Figure 5-12.

Thus, while S/MIME-enabled mail readers such as Microsoft Outlook, Apple Mail, and Mozilla Thunderbird pose minimal burden on users upon receiving digitally signed mail, the programs do not do a good job showing people the contents of the digital certificates used to sign those messages.



Certificate Information

This certificate is intended for the following purpose(s):

- Protects e-mail messages

Issued to: Thawte Freemail Member

Issued by: Thawte Personal Freemail Issuing CA

Valid from 9/10/2004 to 9/10/2005

Show: <All>

Field	Value
Version	V3
Serial number	0d 04 db
Signature algorithm	md5RSA
Issuer	Thawte Personal Freemail Issui...
Valid from	Friday, September 10, 2004 12:...
Valid to	Saturday, September 10, 2005 ...
Subject	simsong@csail.mit.edu, Thawte...
Public key	RSA (2048 Bits)

CN = Thawte Personal Freemail Issuing CA
O = Thawte Consulting (Pty) Ltd.
C = ZA

Certification path

Thawte Freemail Member

Certificate status:

This certificate is OK.

Trust Status indicates whether you trust the individual, group or corporation to whom this certificate is issued. The certificate can then be used for the specified purposes such as sending/receiving e-mail, trusting a web site having the certificate, etc.

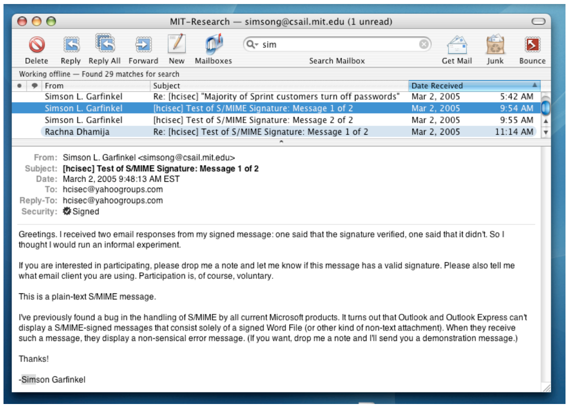
Certificate trusted for

E-Mail Encryption and Authentication

Edit Trust

○ Inherit Trust from Issuer
● Explicitly Trust this Certificate
○ Explicitly Don't Trust this Certificate

Figure 5-12: The Microsoft Windows standard dialogue for viewing certificates has four tabbed sub-panels. Certificates can be used even if they are not signed by a valid CA, but each certificate needs to be "explicitly trusted" using the dialogue on the Trust tab (lower right).



Working offline — Found 29 matches for search

From	Subject	Date Received	
Simson L. Garfinkel	Re: [hcisec] "Majority of Sprint customers turn off passwords"	Mar 2, 2005	5:42 AM
Simson L. Garfinkel	[hcisec] Test of S/MIME Signature: Message 1 of 2	Mar 2, 2005	9:54 AM
Simson L. Garfinkel	[hcisec] Test of S/MIME Signature: Message 2 of 2	Mar 2, 2005	9:55 AM
Rachna Dhamija	Re: [hcisec] Test of S/MIME Signature: Message 1 of 2	Mar 2, 2005	11:14 AM

From: Simson L. Garfinkel <simsong@csail.mit.edu>
Subject: **[hcisec] Test of S/MIME Signature: Message 1 of 2**
Date: March 2, 2005 9:48:13 AM EST
To: hcisec@yahoogroups.com
Reply-To: hcisec@yahoogroups.com
Security: Signed

Greetings. I received two email responses from my signed message: one said that the signature verified, one said that it didn't. So I thought I would run an informal experiment.

If you are interested in participating, please drop me a note and let me know if this message has a valid signature. Please also tell me what email client you are using. Participation is, of course, voluntary.

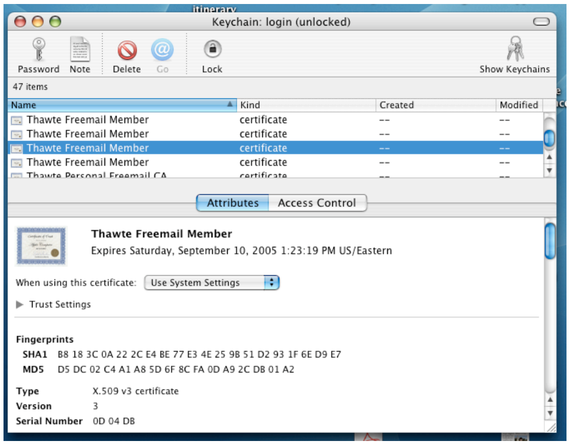
This is a plain-text S/MIME message.

I've previously found a bug in the handling of S/MIME by all current Microsoft products. It turns out that Outlook and Outlook Express can't display a S/MIME-signed messages that consist solely of a signed Word File (or other kind of non-text attachment). When they receive such a message, they display a non-sensical error message. (If you want, drop me a note and I'll send you a demonstration message.)

Thanks!

-Simson Garfinkel

Figure 5-13: Apple's OS X Mail application displays a special "Security:" header to indicate if messages are digitally signed. Unfortunately, there is no way to view the certificate that was used to sign the message.



47 items

Name	Kind	Created	Modified
Thawte Freemail Member	certificate	--	--
Thawte Freemail Member	certificate	--	--
Thawte Freemail Member	certificate	--	--
Thawte Freemail Member	certificate	--	--
Thawte Personal Freemail CA	certificate	--	--

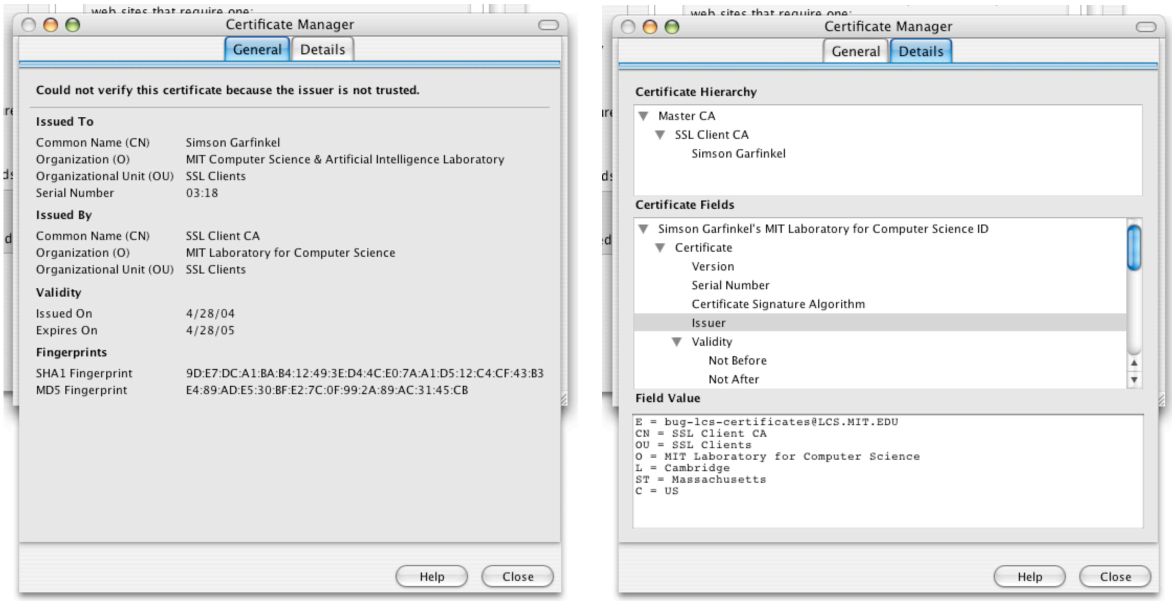
Thawte Freemail Member
Expires Saturday, September 10, 2005 1:23:19 PM US/Eastern

When using this certificate: Use System Settings

Fingerprints
SHA1 B8 18 3C 0A 22 2C E4 BE 77 E3 4E 25 9B 51 D2 93 1F 6E D9 E7
MD5 D5 DC 02 C4 A1 A8 5D 6F 8C FA 0D A9 2C DB 01 A2

Type X.509 v3 certificate
Version 3
Serial Number 0D 04 DB

Figure 5-14: Apple's Certificate Viewer is bundled into the MacOS 10.3 "Keychain" application. The program is surprisingly difficult to use—for example, view containing the certificate list and the Attributes/Access control are not embedded into an NSSplitView, which would allow the relative space devoted to each section to be adjusted. (The message list and the message preview area in the OS X 10.3 Mail application *are* embedded in an NSSplitView, as evidenced by the dimple in Figure 5-13.)



Could not verify this certificate because the issuer is not trusted.

Issued To
Common Name (CN) Simson Garfinkel
Organization (O) MIT Computer Science & Artificial Intelligence Laboratory
Organizational Unit (OU) SSL Clients
Serial Number 03:18

Issued By
Common Name (CN) SSL Client CA
Organization (O) MIT Laboratory for Computer Science
Organizational Unit (OU) SSL Clients

Validity
Issued On 4/28/04
Expires On 4/28/05

Fingerprints
SHA1 Fingerprint 9D:E7:DC:A1:BA:B4:12:49:3E:D4:4C:E0:7A:A1:D5:12:C4:CF:43:B3
MD5 Fingerprint E4:89:AD:E5:30:BF:E2:7C:0F:99:2A:89:AC:31:45:CB

Certificate Hierarchy

- Master CA
  - SSL Client CA
    - Simson Garfinkel

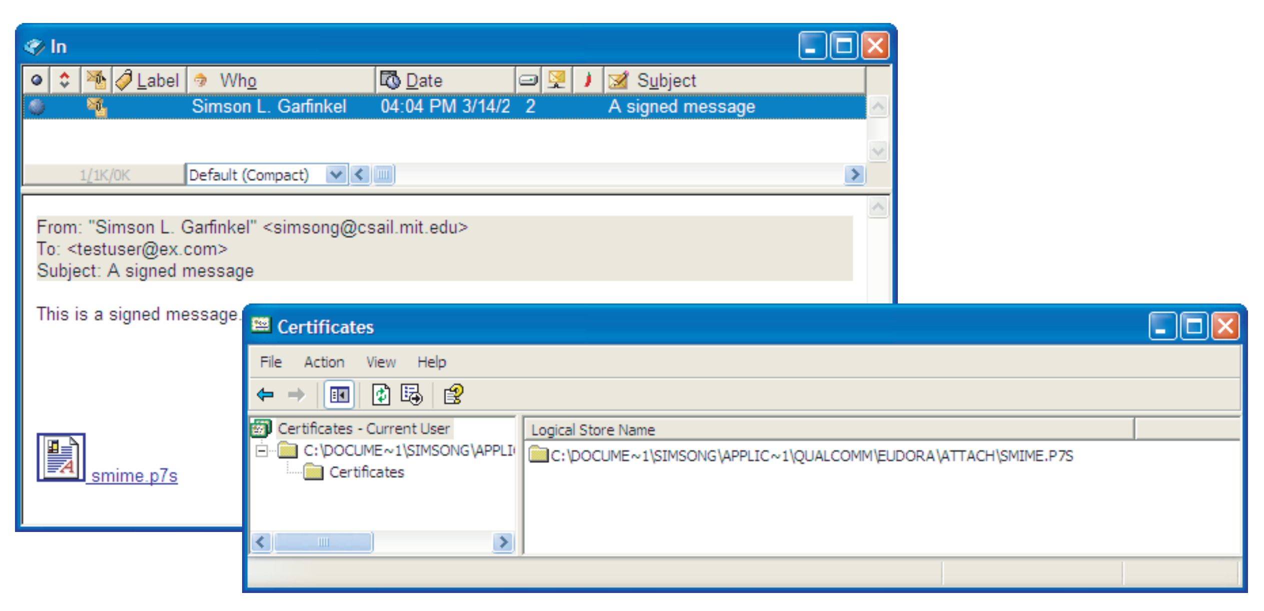
Certificate Fields

- Simson Garfinkel's MIT Laboratory for Computer Science ID
  - Certificate
    - Version
    - Serial Number
    - Certificate Signature Algorithm
    - Issuer
    - Validity
      - Not Before
      - Not After

Field Value

```
E = bug-lcs-certificates@LCS.MIT.EDU
CN = SSL Client CA
OU = SSL Clients
O = MIT Laboratory for Computer Science
L = Cambridge
ST = Massachusetts
C = US
```

Figure 5-15: The Mozilla certificate display dialogue, used in Mozilla Firefox and Thunderbird, makes it very difficult for the user to both see and understand the relevant information on a certificate. These problems are similar to the usability problems found on the Apple and Microsoft certificate viewers.



From: "Simson L. Garfinkel" <simsong@csail.mit.edu>
To: <testuser@ex.com>
Subject: A signed message

This is a signed message.

Figure 5-16: Eudora version 6 for Windows treats S/MIME signatures as attachments. Clicking on the attachment displays the Windows certificate viewer, but does not actually verify the certificate!

Non-S/MIME reader, S/MIME-signed message

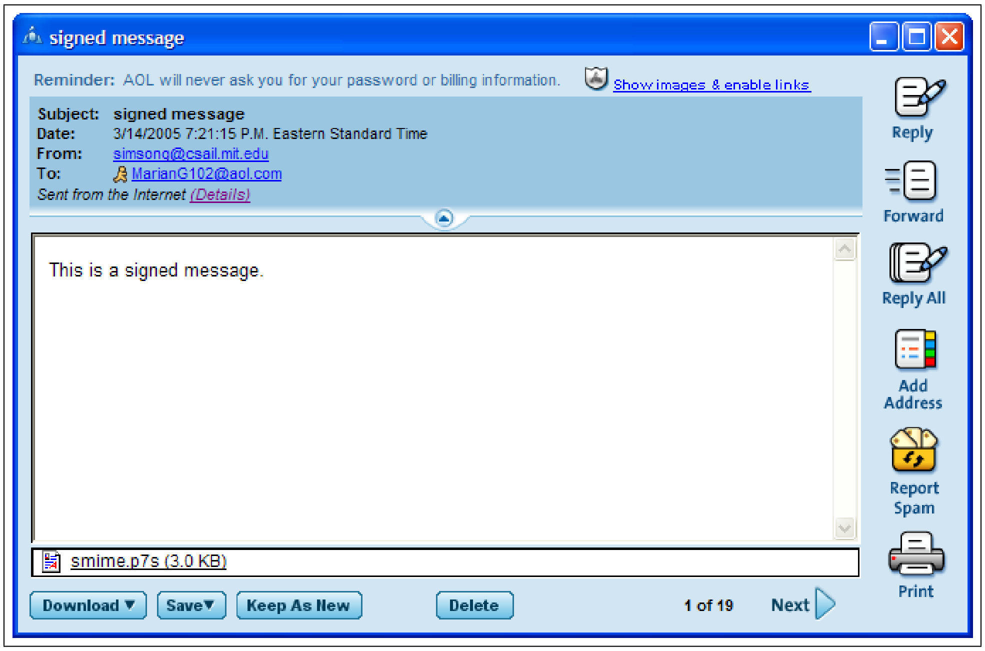
Most mail systems that do not directly support S/MIME display signatures as an attachment. In theory this allows an S/MIME signature to be saved into a file and verified independently of the mail reader. In practice nobody does this, and the S/MIME attachments frequently appear to be a source of confusion. An unfortunate aspect of this confusion is that many of the popular email systems that cater to the very individuals who are not sophisticated computer users—systems such as AOL and HotMail—are the same systems that do not have S/MIME support.

For example, when Eudora Version 6 for Windows receives an S/MIME signed message, the Eudora strips the signature attachment and places the file in its "Attachments" directory. Clicking on the icon causes the Windows certificate viewer to open, as shown in Figure 5-16. This may give the impression that the signature is valid, even though the signature is never actually checked!

Similar behavior is seen in both AOL version 9 (Figure 5-17), which the company heavily promotes as its "Security Edition," and in Microsoft's Hotmail (Figure 5-18). Microsoft's lack of support for S/MIME signatures is particularly disappointing, given that Microsoft does support the display of signed messages in the company's Outlook Web Access module.

S/MIME readers, non-verifying S/MIME message

One of the questions that the PEM committee couldn't answer back in the 1980s was what to do when a signed message didn't verify. Today's developers have solved this problem: messages are passed to the user with a warning. A related but different question is what to do when the message verifies but the key that was used to sign the message is not trustworthy, either because the key's certificate was signed by an untrusted CA, or because the certificate has expired or been revoked.



signed message

Reminder: AOL will never ask you for your password or billing information.

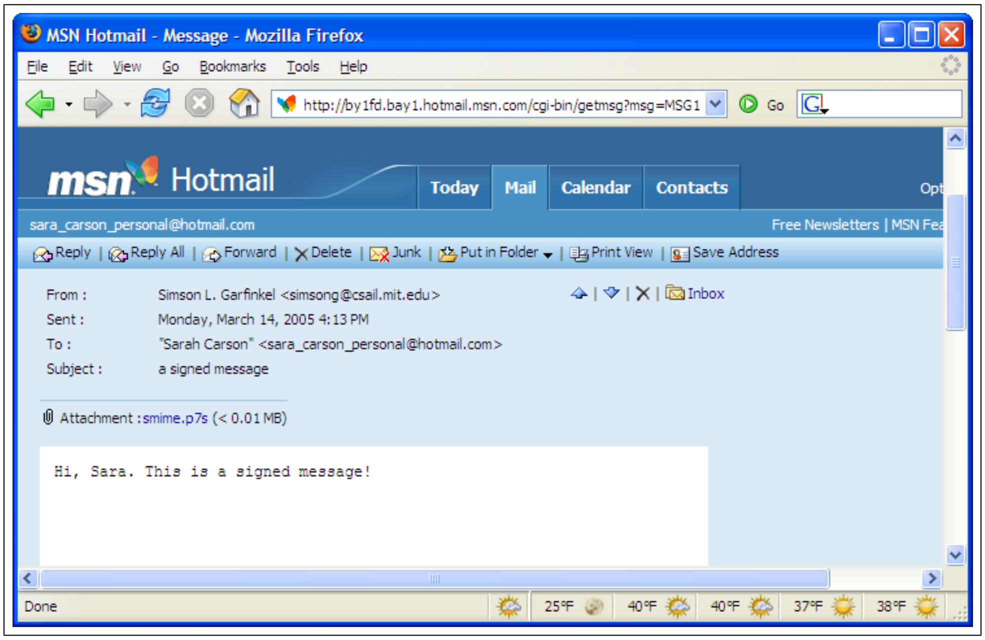
Subject: signed message
Date: 3/14/2005 7:21:15 P.M. Eastern Standard Time
From: simsong@csail.mit.edu
To: MarianG102@aol.com
Sent from the Internet (Details)

This is a signed message.

smime.p7s (3.0 KB)

1 of 19

Figure 5-17: AOL Version 9, the company's "Security Edition," displays S/MIME signatures as attachments. Although the AOL software will scan the S/MIME signature for viruses and spyware, it will unfortunately not verify the message to which it is attached.



MSN Hotmail - Message - Mozilla Firefox

http://by1fd.bay1.hotmail.msn.com/cgi-bin/getmsg?msg=MSG1

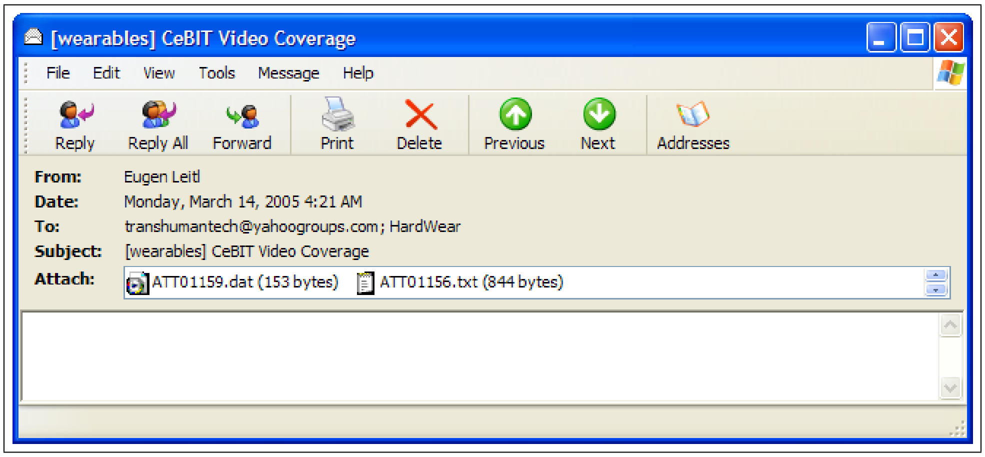
sara_carson_personal@hotmail.com

From : Simson L. Garfinkel <simsong@csail.mit.edu>
Sent : Monday, March 14, 2005 4:13 PM
To : "Sarah Carson" <sara_carson_personal@hotmail.com>
Subject : a signed message

Attachment : smime.p7s (< 0.01 MB)

```
Hi, Sara. This is a signed message!
```

Figure 5-18: In March 2005, Microsoft's Hotmail also displayed signed messages as simply having an attachment. In contrast, S/MIME signatures are properly decoded and displayed by Microsoft's Outlook Web Access, the company's webmail server for Microsoft Exchange.



[wearables] CeBIT Video Coverage

From: Eugen Leitl
Date: Monday, March 14, 2005 4:21 AM
To: transhumantech@yahoogroups.com; HardWear
Subject: [wearables] CeBIT Video Coverage
Attach: ATT01159.dat (153 bytes) ATT01156.txt (844 bytes)

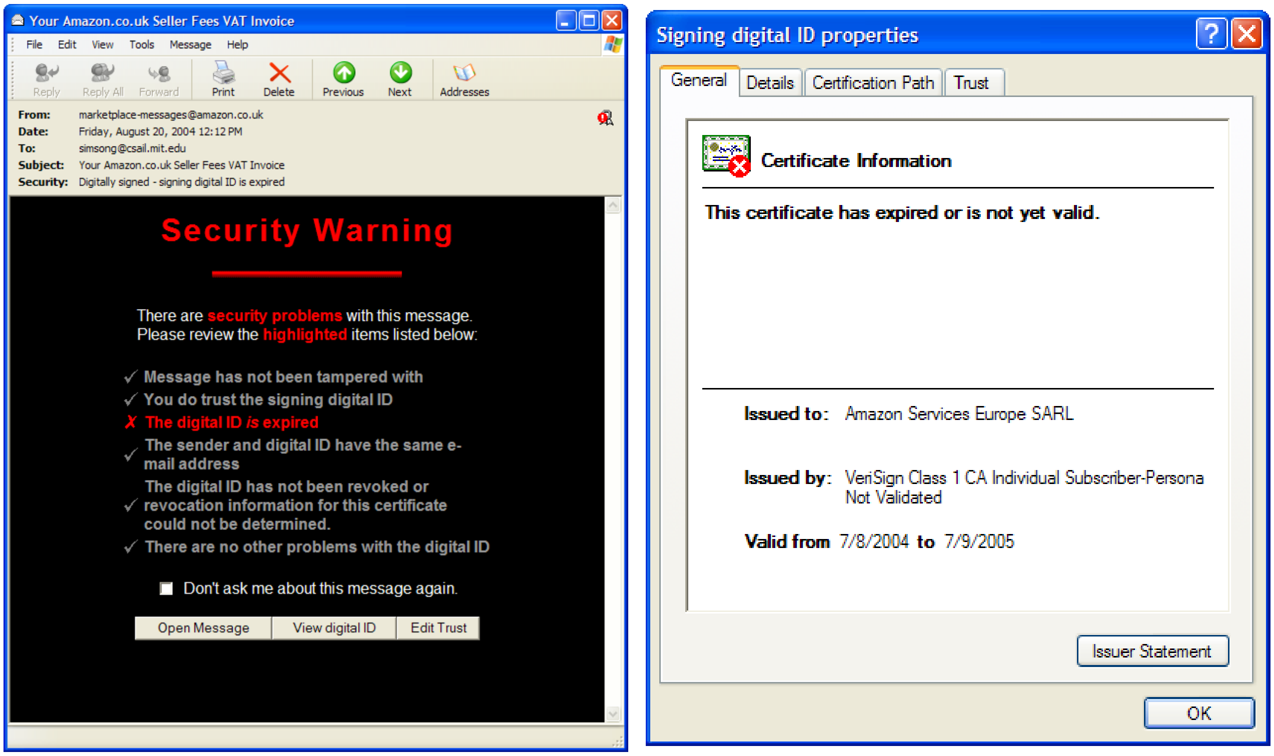
Figure 5-19: When Outlook Express 6 receives a message that is signed with the OpenPGP format, the program displays the message as two attachments.

Assuming that the S/MIME message was properly signed, the only reason that a message would not verify would be if the message was somehow modified in transit. Although signatures were created to protect against malicious modification, we have has never experienced such a modification. On the other hand, we have had many messages modified by mailing list systems. Such modifications have been very difficult to characterize and appear dependent on the message contents and the mailing list service. For example, some kinds of S/MIME-signed messages that were sent through some versions of the Mailman mailing list management system were modified, but other messages sent through the same Mailman system were not. Signed mail text messages sent through Yahoo Groups in March 2005 were passed without modification, but signed HTML messages sent through on the same day were modified by the inclusion of a small advertisement. (Yahoo could make such modifications without damaging signatures by adding the advertisement as an unsigned MIME attachment, but that might break other mail systems.)

One should also note that modifications that are not intended as malicious can still have significant results, and an advantage of using signed mail is that such modifications are easier to detect. For example, in 2002 it was observed that Yahoo's email service was silently changing the word "eval" to "review" in HTML messages. Other substitutions discovered were the words "mocha" being turned into "espresso" and "expression" being changed to "statement." These changes were apparently to defeat JavaScript attacks; one of the results of this typographical slight of hand was the coining of a new word, "medireview," as a synonym for medieval studies. [NTK02a] In some cases these automatic changes appeared in magazine articles, as the text of those articles had been sent from writers to editors through Yahoo and then not adequately checked. A complete list of the words can be found at [NTK02b].

Another reason that a message might not verify is that the certificate has expired. There are in fact two different permutations of an expired certificate:

- The certificate could have expired before the message was signed.
- The certificate could have been valid when the message was signed, but has since expired.



From: marketplace-messages@amazon.co.uk
Date: Friday, August 20, 2004 12:12 PM
To: simsong@csail.mit.edu
Subject: Your Amazon.co.uk Seller Fees VAT Invoice
Security: Digitally signed - signing digital ID is expired

Security Warning

There are security problems with this message.
Please review the highlighted items listed below:

- ✓ Message has not been tampered with
- ✓ You do trust the signing digital ID
- ✗ The digital ID is expired
- ✓ The sender and digital ID have the same e-mail address
- ✓ The digital ID has not been revoked or revocation information for this certificate could not be determined.
- ✓ There are no other problems with the digital ID

□ Don't ask me about this message again.

Certificate Information

This certificate has expired or is not yet valid.

Issued to: Amazon Services Europe SARL

Issued by: VeriSign Class 1 CA Individual Subscriber-Persona Not Validated

Valid from 7/8/2004 **to** 7/9/2005

Figure 5-20: Outlook Express 6 checks whether or not a Digital ID has expired based on when the message is displayed, rather than when it was signed (left). When the dog-ribbon with the exclamation mark is pressed, the certificate dialogue (right) displays the confusing message that the certificate "has expired or is not yet valid."—Doesn't the program know?

In tests, it was determined that neither Outlook Express nor Apple Mail handled certificate expiration in a sensible manner.

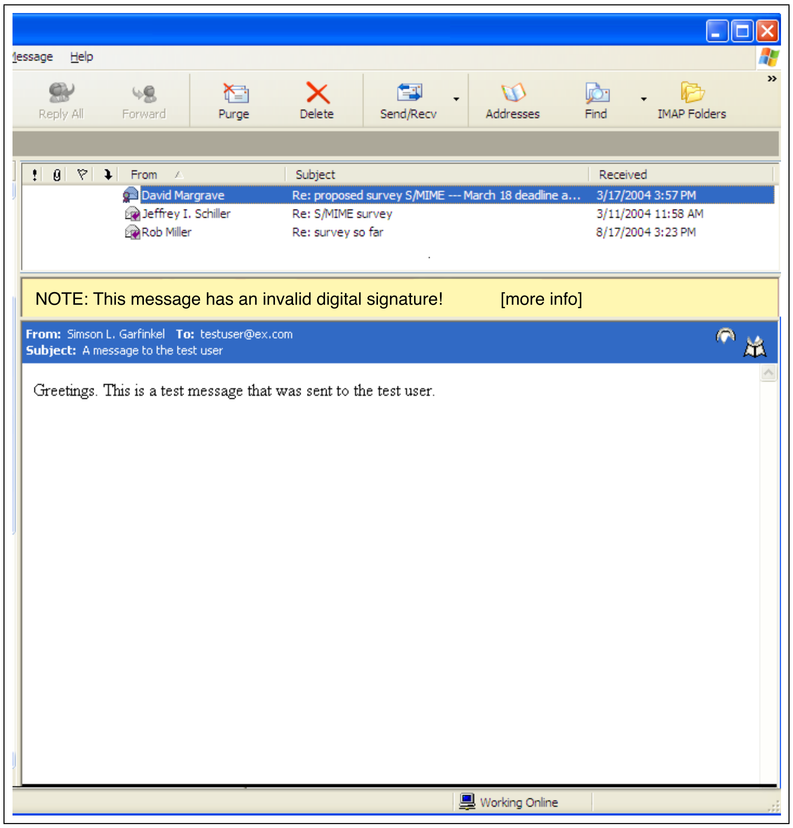
Microsoft Outlook Express declared that mail with a valid signature was no longer validly signed after the signing certificate expired, even if the signing certificate was valid when the signature itself was written. This happened even if OE had previously processed the mail and found it to be valid! Thus, a person who has valid S/MIME signed messages in an Outlook Express mailbox will find that these messages will become invalid over the course of time (Figure 5-20).

Apple's Mail takes a different approach and doesn't appear to check certificate validity at all on received messages. When sending messages, it was found that Apple Mail simply does not allow the sender to sign with a certificate that has expired.

Messages that do not verify because the Digital ID was signed by an untrusted CA are discussed in Chapter 6.

5.3.3 Problems from the field

In the course of researching S/MIME for three years and using S/MIME signatures on a daily basis for nearly nine months, many bugs were discovered in commercial S/MIME implementations. Some of the more interesting bugs are presented below:



From	Subject	Received
David Margrave	Re: proposed survey S/MIME --- March 18 deadline a...	3/17/2004 3:57 PM
Jeffrey I. Schiller	Re: S/MIME survey	3/11/2004 11:58 AM
Rob Miller	Re: survey so far	8/17/2004 3:23 PM

NOTE: This message has an invalid digital signature! [more info]

From: Simson L. Garfinkel **To:** testuser@ex.com
Subject: A message to the test user

Greetings. This is a test message that was sent to the test user.

Figure 5-21: A proposed interface for Outlook Express that would display security information using the same sort of informational bar that has been adopted for Internet Explorer and Mozilla Firefox. *Simulated screen shot.*

- S/MIME users in the US military have been frustrated by the fact that message decrypting keys are only present on their multifunction cards, that the cards are replaced every time they receive a new assignment, and the fact that S/MIME clients leave encrypted messages in the mail store. As a result of these decisions, access to old messages is lost unless the private keys are exported from the multifunction cards and transferred to new cards. As a result, technology to export unexportable keys had to be developed.[Hal03]

- A bug was discovered in the Microsoft S/MIME decoder (used in both Outlook and Outlook Express) used by the current and all previous versions of the two programs. When a signed multipart message is received that has only a single part (as is the case when a signed attachment is sent without a message body), a bug causes the Microsoft programs to refuse to display the message, even though the message is not encrypted.[Tre04] Microsoft never discovered this bug in its testing because Outlook and Outlook Express never send this kind of message, but Apple's Mail client does.

- Several users who had email systems that did not implement S/MIME were confused by the S/MIME signature attachment. Typical response was:

  > *"There is a strange attached file to your mail: smime.p7s... What's that?"*
  >
  > *"I couldn't open the attachment that you sent me."*

- A Canadian government agency configured its firewall to pass attachments named “smime.p7m” of mime type Application/X-PKCS7-MIME but to strip attachments named “smime.p7s” of mime type Application/X-PKCS7-SIGNATURE. It appears that the firewall had been configured to strip all attachments of types that had not been specifically registered; the firewall’s administrators knew of one S/MIME type but not the other.

- When the mutt mail reader on Linux received a message with a corrupted signature, it displayed the following information:

  ```
  [-- OpenSSL output follows (current time: Wed Mar  2 09:38:33 2005) --]
  Verification failure
  8135:error:21071065:PKCS7 routines:PKCS7_signatureVerify:digest
  +failure:pk7_doit.c:808:
  8135:error:21075069:PKCS7 routines:PKCS7_verify:signature
  +failure:pk7_smime.c:265:
  [-- End of OpenSSL output --]
  ```

  Following this display of OpenSSL output, mutt displayed the message “The following data is signed” and proceeded to display the message with the corrupted signature. Technically the message was correct, because the message was signed, although the signature did not verify.[Sam05]

- Some virus-scanning mail gateways append a tag line in mail messages to indicate that the message has been scanned for viruses. These tag lines break S/MIME signatures.[Mar05b]

- When users of some versions of Outlook attempt to reply to a message that is digitally signed, Outlook defaults to signing the outgoing message *even if the user does not have a Digital ID!* When the user hits the “Send” button, they then receive a message warning that they do not have a Digital ID and they are invited to press a button that says “Get a Digital ID” which, in turn, takes them to a web page that lists commercial Digital ID vendors.[Mar05b] (This is why we only recommend sending signed S/MIME messages for do-not-reply email at this time.)

- Many users were confused that today’s S/MIME implementations do not certify the Subject:, Date:, To: or From: lines of email messages. (Likewise, they do not encrypt the Subject: line of sealed S/MIME messages.) Although the S/MIME RFCs do provide for encapsulating these lines within a MIME object, none of the S/MIME clients tested for this dissertation implemented that functionality.

These errors all seem to indicate that the S/MIME standard has received relatively little use in the nine years that the software has been made widely available to businesses and consumers. After all, if the technology was being widely used, these bugs would have been found and eradicated.

5.4 Hidden Signatures

One of the fundamental problems with both S/MIME and the OpenPGP standards when used to sign messages is that these standards use MIME multipart attachments to convey metainformation about the messages themselves.

Although using the MIME standard was technically elegant and allowed the MIME standards and implementations to be re-used for security purposes, doing so created significant usability hurdles for individuals who had mail systems that understood MIME attachments but did not implement S/MIME. These users do not download the S/MIME attachments and independently verify them with helper applications: they are merely confused by the S/MIME attachments.

Another approach would have been to use specially crafted *hidden signatures* that are visible to the proper software but otherwise invisible. One technique for doing this is to hide the signature inside specially crafted header lines, as shown in Figure 5-9. This approach can also be used for distributing keys. This technique was developed for the Stream encryption proxy discussed in Appendix D on page 413. Two places where the headers can be placed are in the message header and in the headers of MIME body parts. Of these two approaches, hiding information in the MIME body part header was found to work better. This is because some programs (such as Eudora and RMAIL) display mail headers that they do not recognize. On the other hand, no program that was tested displays unrecognized MIME body part headers.

While the hidden signature approach has the advantage that it poses no usability burden on users who do not have the necessary decoding software, it has the disadvantage that nobody on the planet is currently running the necessary decoding software. Hidden signatures may be useful in putting forth new signature schemes, such as the separable identity-based ring signature system proposed by Adida, Hohenberger and Rivest. [AHR05a, AHR05b]

Given that S/MIME is widely deployed, it is almost certainly an easier task to get the few remaining hold-outs to adopt the S/MIME standard, rather than to try to put forth yet another secure email standard.

5.5 Conclusions and Recommendations

After nearly three decades of work on the secure messaging problem, the vast majority of email sent over today's electronic networks is without cryptographic protection. Nevertheless, great progress has been made. As the research presented in this chapter demonstrates, a significant fraction of the Internet's users have the ability to receive and transparently decode mail that is digitally signed with the S/MIME standard. It is within the capability of businesses to start sending S/MIME-signed messages today. Such practices are almost certain to do more good than harm.

What's more, the survey data presented in this chapter shows that a significant fraction of Amazon.com's merchants believe that financially related email should be signed (and sealed) as a matter of good business practices. Mail encryption is not possible using S/MIME technology unless the recipient obtains a Digital ID and somehow gets that ID to the sender. On the other hand, if organizations like eBay and Amazon started sending out signed mail today, their recipients could respond with email that was encrypted (but not signed) for the sending organizations.

5.5.1 Promote incremental deployment

Deploying email encryption systems is frequently seen as a chicken-and-egg problem. Senders can't encrypt messages for a recipient unless the recipient first creates a public/private keypair and

```
Mime-Version: 1.0 (Stream Encoded)
To: simsong@acm.org
Message-Id: <732b4c35ffa86d4f76b7e4967d599dd2@csail.mit.edu>
Content-Type: multipart/alternative; boundary=Apple-Mail-2--871523547
From: "Simson L. Garfinkel" <simsong@csail.mit.edu>
Subject: test message
Date: Mon, 14 Feb 2005 20:49:38 -0500


--Apple-Mail-2--871523547
PGP-sig01: Version: PGP 6.5.8
PGP-sig02:
PGP-sig03: iQA/AwUBQldqhBkGokKY4xwsEQKXRwCg5KCLs58HPFgPTWn6MC2F0udCMT8An3Pb
PGP-sig04: qSFf6Jy1wNyxTlNc9boojKhT
PGP-sig05: =hHEw
Content-Transfer-Encoding: 7bit
Content-Type: text/plain;
charset=US-ASCII;
format=flowed

This is a message that is signed
with PGP. It is a very simple message.

--Apple-Mail-2--871523547--
```

Figure 5-22: A digital signature hidden inside an S/MIME header. The signature, which covers the To:, From:, Subject: and Date: headers as well as the message content, is hidden from any MIME-enabled mail reader that does not know how to process the PGP-sig headers.

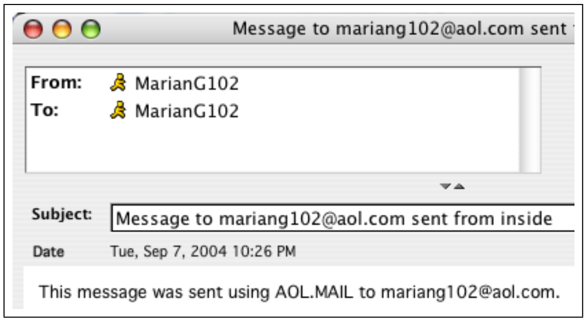
obtains the necessary certificate. But there is no incentive for a recipient to make this effort unless there is first a sender who wants to send encrypted mail.

No such chicken-and-egg problem exists for senders who wish to sign outgoing mail. Our survey shows that most Internet users have software that will automatically verify S/MIME signatures in a manner that is exactly analogous to accepting a CA-issued certificate during the SSL handshake. Companies sending email can begin adopting S/MIME now and incrementally deploy it.

Although in the 1990's digital signatures might have been seen as extravagant or expensive technology that required special-purpose cryptographic accelerators to implement on a large scale, those days have long passed. A 2GHz Pentium-based desktop computer can create an more than 700 S/MIME signatures every minute using the freely available OpenSSL package. S/MIME certificates are also cheap: a single VeriSign Digital ID purchased for $19.95 per year can be used to sign literally billions of outgoing messages, since VeriSign and other CAs charge for certificates by the year, not by the message.

5.5.2 Extending security from the walled garden

End-to-end encryption on the Internet was developed because the Internet computers and their links were not a secure infrastructure operated by a single management team. But many of encryption's benefits—identification of sender, integrity of messages, and privacy of message contents—can be accomplished for email sent within closed systems such as AOL and Hotmail. These so-called



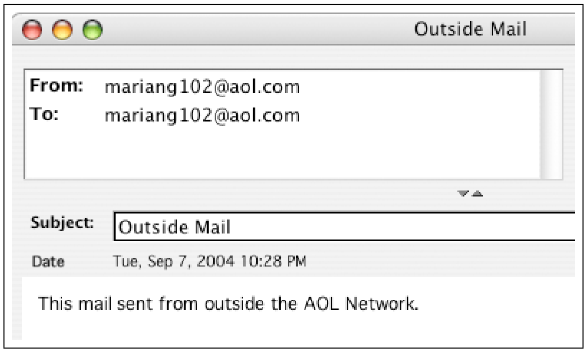
Message to mariang102@aol.com sent

From: MarianG102
To: MarianG102

Subject: Message to mariang102@aol.com sent from inside
Date Tue, Sep 7, 2004 10:26 PM

This message was sent using AOL.MAIL to mariang102@aol.com.

Figure 5-23: Addresses on messages that originate from within the AOL network, when viewed using AOL's webmail interface.



Outside Mail

From: mariang102@aol.com
To: mariang102@aol.com

Subject: Outside Mail
Date Tue, Sep 7, 2004 10:28 PM

This mail sent from outside the AOL Network.

Figure 5-24: Addresses on messages received from outside the AOL network appear differently than messages originating from inside.

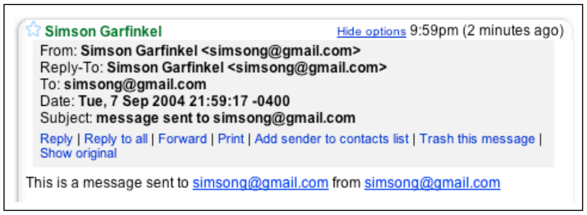
walled gardens can provide security assurances for their content because they use passwords to authenticate message senders and provide reasonable security for message contents.

Several online services are now providing some form of sender authentication, in that they are showing the recipients of some messages that the messages originating from within their services (their "walled gardens") were sent with properly authenticated senders. The services do this by distinguishing between email sent from within the service and email sent from outside—even when the mail sent from outside the service is sent with a `From:` address of an inside sider.

For example, both AOL's webmail and client interfaces identify email that originated within AOL with a little icon of a human being in the `From:` field, as shown in Figure 5-23. Mail that comes from the Internet is displayed with a complete Internet email address, as shown in Figure 5-24, and with the notation "Sent from the Internet" (not shown). This is true even when the email that arrives from the Internet has an `@aol.com` in *From:* field. The AOL network also has the ability to carry "Official AOL Mail," indicated by a blue envelope icon in the user's mailbox, an "Official AOL Mail" seal on the email message, and a dark blue frame around the message, as shown in Figure 5-27. All of these visual indications provide the user with cues that mail sent from within AOL is somehow different—and presumably more trustworthy—than mail from outside of AOL.

Other webmail providers do not follow AOL's practice. For example, Google's "GMail" service displays messages with `@gmail.com` addresses that originated *outside* GMail in exactly the same manner as messages that originated from *within* GMail, as shown in Figures 5-25 and 5-26. These two cases should be distinguished: mail originating within GMail was sent by a sender who provided a valid username and password, while no such verification was performed for the sender of mail sent from outside GMail. Inside mail is more trustworthy and should be distinguished from outside mail.

Users would benefit from having those systems make explicit guarantees about message integrity, authorship and privacy. An easy way to start is for walled gardens to distinguish between email originating within their walls and email originating from the outside, as AOL does. This recommendaiton is presented in Chapter 10.

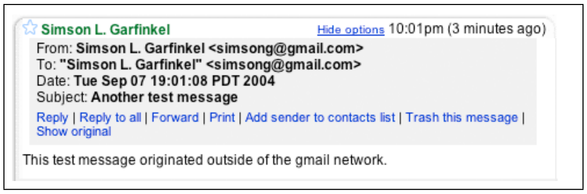


Simson Garfinkel 9:59pm (2 minutes ago)
From: Simson Garfinkel <simsong@gmail.com>
Reply-To: Simson Garfinkel <simsong@gmail.com>
To: simsong@gmail.com
Date: Tue, 7 Sep 2004 21:59:17 -0400
Subject: message sent to simsong@gmail.com

This is a message sent to simsong@gmail.com from simsong@gmail.com

Hide options
Reply | Reply to all | Forward | Print | Add sender to contacts list | Trash this message | Show original

Figure 5-25: Addresses on messages that originate from within the GMail network, when viewed using GMail's webmail interface.

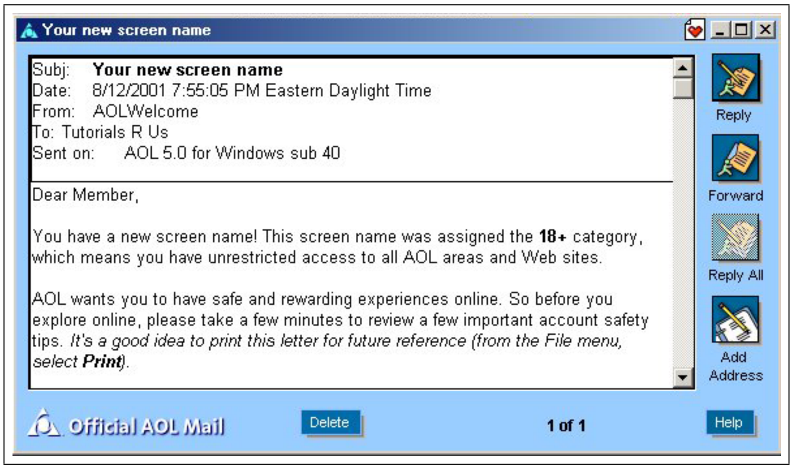


Simson L. Garfinkel 10:01pm (3 minutes ago)
From: Simson L. Garfinkel <simsong@gmail.com>
To: "Simson L. Garfinkel" <simsong@gmail.com>
Date: Tue Sep 07 19:01:08 PDT 2004
Subject: Another test message

This test message originated outside of the gmail network.

Hide options
Reply | Reply to all | Forward | Print | Add sender to contacts list | Trash this message | Show original

Figure 5-26: Addresses on messages received from outside the GMAIL network appear the same as messages that originate inside.



Your new screen name

Subj: Your new screen name
Date: 8/12/2001 7:55:05 PM Eastern Daylight Time
From: AOLWelcome
To: Tutorials R Us
Sent on: AOL 5.0 for Windows sub 40

Dear Member,

You have a new screen name! This screen name was assigned the **18+** category, which means you have unrestricted access to all AOL areas and Web sites.

AOL wants you to have safe and rewarding experiences online. So before you explore online, please take a few minutes to review a few important account safety tips. *It's a good idea to print this letter for future reference (from the File menu, select **Print**).*

Figure 5-27: The AOL network has the ability to transport "Official AOL Mail." Such messages cannot be spoofed by outsiders or other AOL members.

5.5.3 S/MIME for Webmail

The security of the Official AOL Mail system depends upon the security of the AOL network and the AOL client software. Although the implementation might use S/MIME or a similar digital signature system, it could be implemented with a variety of simpler means as well. Although proponents of cryptography might be tempted to argue that the S/MIME-based system would be more secure, such a system would still rely on the AOL client software to verify the S/MIME signatures.

Moving forwards, we believe that webmail providers such as Hotmail and AOL should work to support S/MIME directly in their systems. Today these services display S/MIME signatures as a small attachment that cannot be easily decoded and understood. Instead, we believe that they should validate the S/MIME signatures and display an icon indicating a signed message has a valid signature.

Once S/MIME messages are properly validated, we believe that the next step is for webmail providers to obtain S/MIME certificates on behalf of their customers and use those certificates to automatically sign all outgoing mail. This is ethically permissible because the webmail provider has verified the identity of the sender, at least to the point of knowing that the sender can receive email at the given email address. Major webmail providers could do this by establishing themselves as CAs and having Microsoft distribute their CA keys through the Windows Update mechanism; smaller webmail providers could work deals with existing CAs to obtain certificates that allow extension of the certification chain. This proposal is somewhat similar to Yahoo!'s DomainKey proposal, [Del04a] except that the signatures would be created with S/MIME and could be verified with software that is already deployed to hundreds of millions of desktops.

5.5.4 Improving the S/MIME client

Given that support for S/MIME signatures is now widely deployed, existing mail clients and webmail systems that do not recognize S/MIME-signed mail should be modified to do so. Existing systems should be more lenient with mail that is digitally signed but which fails some sort of security check. For example, Microsoft Outlook and Outlook Express give a warning if a message is signed with a certificate that has expired, or if a certificate is signed by a CA that is not trusted. Such warnings appear to both confuse and annoy most users; more useful would be a warning that indicates when there is a change in the distinguished name of a correspondent—or even when the sender's signing key changes—indicating a possible phishing attack. We shall return to this topic in Chapter 7.

This research presented in this chapter shows that there is significant value for users in being able to verify signatures on signed email, even without the ability to respond to these messages with mail that is signed or sealed. The technology has been deployed. It's time for us to start using it.