Global Information Assurance Certification Paper

E-MAIL ENCRYPTION FOR SMALL OFFICE / HOME OFFICE USERS:

GIAC Security Essentials Certification (GSEC) Practical Assignment v1.4b

Robert Dooling
April 19, 2004

Table of Contents

INTRODUCTION

Need for Security in E-Mail

It is generally accepted that, aside from its people, a company's most valuable asset is its Intellectual Property [33], much of which is carried in e-mail messages across the open, shared, and inherently insecure Internet. Aside from at large enterprises and organizations, the majority of these e-mails are sent in plain text – ordinary, human-readable text – potentially viewable by anyone.

Insecure e-mail can cause a multitude of problems for senders, recipients, and entire organizations. Unprotected e-mails provide a juicy target for malicious parties to "intercept, modify, spoof and turn to almost any criminal purpose possible." [33]

This paper will identify and discuss the pressing need for security in certain types of e-mail, and the options available to provide such service. A case will be made for one of the available options as the best solution; this solution will then be examined in greater detail, and an example implementation will be provided to help guide a user through the process. Issues of special concern and importance in this program will be highlighted, along with a set of “best practice” tips to address them. Lastly, this paper will provided a brief overview of the challenges to greater implementation of e-mail security.

There are four main security features that are absent or lacking in regular e-mail:
(1) **C**onfidentiality -- The ability to keep anyone except your intended recipient(s) from reading what you send;
(2) **I**ntegrity -- The ability to prove that the message arrived intact, as the sender created it, without any tampering or modifications;
(3) **A**uthenticity -- The ability to verify through proper identification the sender of a message; and
(4) **N**on-**R**epudiation – The ability to prove who sent the message, preventing the originator from denying they sent it. From the root word repudiate: “To cast off; to disavow; to have nothing to do with; to renounce; to reject.” [1]

These security features may not seem necessary for typical, day-to-day e-mailing activity, but from a business perspective they can be crucial, especially if you “consider the fact that the probability that a message will be modified while in transit is about 50-50. A modification doesn't have to be malicious or intentional to damage message integrity. It can range from the relatively benign conversion of spaces to tabs or vice versa, to thorough and complete violations of Internet e-mail standard.” [14]

[1] http://dictionary.reference.com/search?q=repudiate

But do businesses really transmit critical communications and information via e-mail? "According to a recent study commissioned by Evergreen Assurance… nearly 90 percent of companies conduct business transactions via e-mail, and nearly 70 percent say e-mail is tied to their means of generating revenue." [4]

Take the recent example of a regional U.S. bank that sent, completely unprotected, the personal data on 40,000 customers to an outside contractor[2]. The information included bank account numbers, Social Security numbers, and home addresses of bank account holders. The bank now faces regulatory scrutiny of their privacy practices, negative publicity, and almost certain customer backlash. Their customers face the prospect of knowing that their private information, which they had entrusted to their bank, was made accessible to anyone with an Internet connection.

Small organizations especially seem to rely heavily on e-mail, in lieu of more tailored, elegant, and, most relevantly – expensive, business software solutions such as Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), E-Commerce, and Accounting applications.

A simple matrix can illustrate the risks that face an organization utilizing e-mail insecurely. The table below presents a sample of these risks in the standard format of:

Vulnerability x (multiply) Likelihood of occurrence x Impact of occurrence = Risk

[2] http://www.miami.com/mld/miamiherald/business/national/8024074.htm

	VULNERABILITY	**THREAT / LIKELIHOOD** (Low - Moderate - High)	**IMPACT** (Low - Moderate - High)	**RISK** (Low / Moderate / Significant / High)
Risk scenario #1	Plaintext e-mail messages are read in transit by an eavesdropping third-party.	Moderate: Eavesdropping on e-mail sessions is slightly more complicated than, say, unsecured web sessions, but nonetheless the capability exists for almost anyone on the Internet to do so.	Low – High: Depending entirely on the subject and contents of the e-mail messages read, the impact could be negligible (someone found out your lunch plans!), to devastating (a competitor discovered your plans for an upcoming product). Consider this sobering statistic: 95% of organizations that lose their corporate data fail within 18 months. [4]	**Significant**: Although the majority of an organization's e-mail may be relatively unimportant, all it takes is for one critical message to be viewed by an outsider. If an eavesdropper listens in on *all* of your e-mails, rather than just a random one here or there, they can set up automated searches for keywords (such as financials, marketing plan, or password) to find the juicy materials and ignore the rest.

Risk scenario #2	Plaintext e-mail is inadvertently modified en route to recipient (by incompatible e-mail programs).	Moderate: One of the problems is that while there are lots of e-mail user agents...the number of e-mail applications that are fully compliant with Internet e-mail standards is very small. Compounding the problem is the fact that there are no accreditation or certification programs that confirm compliance with Internet e-mail standards. As a result, interoperability remains problematic, and the likelihood that any received message is identical to the sent message is increasingly dubious. [14]	Low – Moderate: Most likely, the recipient will still be able to understand a slightly deformed message; at worst, they can request retransmission since the e-mail will show obvious signs of corruption.	**Low**: This is more of a nuisance than a serious security problem.
Risk scenario #3	Plaintext e-mail is intentionally modified and retransmitted by a malicious "man-in-the-middle" en route to recipient.	Low – Moderate: This is slightly more difficult to accomplish than simply eavesdropping on a transmission, but still well within the capabilities of many unscrupulous individuals.	Low – High: Depending on the intentions of the attacker and success in deception, this could be minimal (miscommunication of lunch plans) to devastating (a change to an important speech document).	**Significant – High:** The possibilities for damage here are limited only by the imagination.

Risk scenario #4	A malicious imitator spoofs an e-mail as being sent by a known party.	Moderate – High: It is remarkably easy to change some of the e-mail header fields, including "From:".	Low – High: Ranging from an embarrassing practical joke to serious misunderstanding ("but Mr. Gates e-mailed me and said to go ahead and release the source code!").	**Significant:** Although spoofed e-mails can generally be detected without great difficulty, unsuspecting users still provide an easy target.
Risk scenario #5	An e-mail sender denies that they sent a message that has been relied upon by the recipient.	Low - Moderate: Many users would not even think that they could deny responsibility for an e-mail once it has left their send box, but certainly many might try.	Low – High: Ranging from botched lunch plans ("I never said I was going to meet you at 12:30.") to catastrophic business actions ("I never ordered 20,000 parts from you.").	**Significant:** Although there are ways to prove that an e-mail came from an individual computer, a user could still deny authorship. There is little to no legal precedent for assigning legally binding status to an unsecured e-mail.

The risks of insecure e-mails obviously depend highly on the intentions and success of attackers in exploiting the opportunities. But it cannot be denied that enormous potential for damage does exist.
Each of the risks listed above can be addressed by one or more of the four security features discussed previously – try to determine which one(s).

Many people would still scoff at the idea of e-mail security, considering it a technique only for the technical geeks and paranoids.
"Perhaps you think your email is legitimate enough that encryption is unwarranted. If you really are a law-abiding citizen with nothing to hide, then why don't you always send your paper mail on postcards?" [34]

It is a disturbing thought to consider that these or a wide variety of other exposures may have been, or could be in the future, exploited within one's e-mail.
Article 12 of the Universal Declaration of Human Rights addresses this concern directly: "No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honour and reputation. Everyone has the right to the protection of the law against such interference or attacks." [3] Yet, as high an authority as the U.S. District Court in Pennsylvania has ruled that "...there is not an expectation of privacy in ... email". [4] Therefore, if an

[3] http://www.amnesty.org.uk/udhr/udhr.shtml
[4] http://axion.physics.ubc.ca/email-privacy.html

individual desires or requires privacy in their e-mail communications, technological solutions must be utilized.

Solution: Encryption

"With the speed of modern or even old processors, there's no reason that there should be any cleartext transmissions on the Internet at all." [11]

Cryptography is defined as the science of secret writing. [26]
Cryptographic technologies can address each of the risk areas identified above, by providing each of the four security features in the context of e-mails, all at minimal expense:
"...effective products may be obtained at the sort of price tag that puts cryptography well in the range of small to medium sized enterprises..." [33]

Solutions that provide these benefits are available to cover a wide range of differing technology infrastructures and needs. The solution chosen should provide in some shape or form each of the two essential pieces for secure e-mail: the encryption of the message text itself, and digital signatures.

Encryption is desirable from many perspectives for e-mail users across the spectrum: from multinational, Fortune 500 enterprises, to home users, to non-profit organizations and government entities.
This paper endeavors to make the case for e-mail encryption for a fairly specific subset of the e-mailing population: small office and home office (SOHO) users with remote (crossing the Internet) e-mail communication needs. This may include independent contractors, freelancers, telecommuters, relatively small organizations with multiple branches, or those who outsource their mail hosting function. [27]

Introduction to Encryption

Encryption of messages – "coding a message in such a way that its meaning is concealed" [26] -- ensures **C**onfidentiality of e-mail, preventing any unauthorized person from reading a message while in transit. This aspect is crucially important when the communication must be kept private for any reason – competitive interests, privacy of personal information (heard of Identity Theft?), legal obligations, protection of financial data, authentication information, etc.

Digital signatures, another structure of encryption, maintains **I**ntegrity -- assurance that data has not been altered from the original, provides **A**uthenticity -- verification of one's identity, and enforces **N**on-**R**epudiation – preventing the sender from denying their authorship. These capabilities are especially valuable when important, precise transactions – purchases, confirmations, contracts – are completed via e-mail.

To define a few terms that a user should be familiar with to understand and effectively use encryption technology:
"The process of disguising a [plaintext / cleartext] message in such a way as to hide its substance is encryption. An encrypted message is ciphertext. The process of turning ciphertext back into plaintext is decryption". [1] How exactly the message is transformed from plaintext to ciphertext and back again "depends on a mathematical formula called an encryption algorithm or a cipher." [26] A Digital Certificate is a "digitally signed message from the issuer (signer) to the verifier (user) associating a name with a public key." [25]
Public Key Cryptography relies on two mathematically related keys, known as a public key and a private key. The important thing about the public and private keys is that they are complementary. Specifically:
- Data encrypted with the public key can only be decrypted with the private key.
- Data encrypted with the private key can only be decrypted with the public key.
Further, it is computationally infeasible to deduce the private key from the public key. [22]

Information to Consider Encrypting

Effectively any information can be encrypted, although different file systems and file types may require different encryption.
SOHO e-mail users may want to encrypt the occasional message, the majority of their messages, or all of their e-mail correspondence. How much encryption is needed may depend on the nature of their work, and the privacy and security requirements of their locality, industry, or organization, and any business partners or 3rd parties with whom they must communicate. For example, an independent contractor may not be concerned with the security of their data themselves, but the other party to the contract may have strict requirements guiding the security of their contractors' work and communications. As a general rule, it is better to be safe than sorry when it comes to securing business communications. If sender and recipient have the capability to communicate securely, why not take advantage of it?

Specific examples of types of information that should generally be encrypted include:

- personally identifiable information (e.g., transmitting your application information to a Human Resources department may include Social Security Number, Birthdate, Criminal record, etc.);
- personal financial information (e.g., submitting an order to an e-commerce company via e-mail with either check (routing and bank account number) or credit card information);
- organizational financial information (e.g., communicating to your boss the sales figures for last month and next month's projection);
- strategic information (e.g., a business plan for a startup company);
- information required by law to be protected (e.g., health patient information); or

- any other type of information that, if viewed, intercepted, modified, spoofed, or denied by any other party, could have damaging effects.

(This discussion obviously ignores illegal or immoral uses for e-mail encryption, of which there are many: organized crime, extortion, fraud, etc. Encryption has also been used by dissidents in politically repressive environments.)

Options for encryption standards and programs

A multitude of encryption algorithms, methods, protocols and programs have been proposed and utilized at one point or another over the (relatively short) history of e-mail. However, none has been agreed upon as *the* uniformly-accepted standard of e-mail encryption – this lack of agreement has led to confusion, interoperability conflicts, and duplication of efforts, and has proven to be a considerable hindrance to the development of high-quality and easy-to-use encryption software, and therefore, the widespread acceptance and use of such technology.

Before describing e-mail encryption options, it may be helpful to understand why these options are needed in the first place – why can't e-mail be protected in the same manner that, for example, web transactions are?
"E-mail service is asynchronous; all the regular security protocols…are synchronous." [7] What this means is that while web transactions require the user and server to be in consistent, two-way communication, the two parties to an e-mail can interact at different times – an e-mail sent by A may not be downloaded and received by B until tomorrow. Because of this different structure of communication, traditional security protocols (e.g., SSL – the "lock" icon in a web browser) are not feasible solutions for protecting e-mail messages from one end to the other.

Briefly described below are four e-mail encryption options: one was proposed in the early days of e-mail, but failed to gain widespread acceptance; two currently vie for the status of 'most accepted', and the final standard has been proposed and developed very recently, but has not yet gained marketplace acceptance.[5]

PEM

PEM – Privacy Enhanced Mail, was first developed as a standard in 1985, by the Privacy and Security Research Group (PSRG). It is defined in RFCs 1421-1424.[6] PEM allows for both private- and public-key infrastructures, and has the ability to run on a wide range of platforms, programs, and interfaces. PEM Digital certificates are based on the X.509 standard.

RIPEM (Riordan's Internet Privacy Enhanced Mail) is one application of the PEM standard that provides each of the main features of secure e-mail: "disclosure protection [**C**onfidentiality] (optional), originator authenticity [**A**uthentication],

[5] Descriptions will not delve into the benefits and drawbacks of various encryption algorithms (e.g., RSA vs. DH) or frameworks (Public-Key vs. Private-Key); instead, references will be provided for additional reading and research into these topics.
[6] http://www.faqs.org/rfcs/rfc1421.html

message integrity measures [**I**ntegrity], and non-repudiation of origin [**N**on-**R**epudiation]."[7]

Although PEM evolved through several revisions, it never gained widespread acceptance from the Internet community, even though as many as three public domain implementations existed at one point in time. [14]
So why did PEM fail to become the first widely-accepted standard for e-mail security, and why is it almost non-existent in use today?
In 1996, an effort was undertaken by Network Associates Laboratories to address this issue: "...the Internet community endorsed a secure email technology standard called Privacy Enhanced Mail....However, the adoption of the PEM secure email technology...has been limited. Instead, the Internet community has endorsed and adopted an email technology that supports arbitrary contents, including images, voice, video, and structured combinations of contents: Multi-purpose Internet Mail Extensions (MIME). Unfortunately, MIME did not include any security services."[8]

PEM does not support security services to multimedia files, as laid out in the MIME standard, which provides richer content in e-mails than had previously been possible. Essentially, the promise of "rich" content in e-mail won out over the promise of secure e-mail in this case.
A list of "frequently noted" vulnerabilities in PEM can be found at:
http://www.ja.net/CERT/VanHeyningen/RIPEM_Vulnerabilities.html

PGP
PGP – "Pretty Good Privacy" – was introduced to encourage the widespread use of encryption in order to resist government efforts to restrict private use of such technology – as "encryption for the masses".
Phillip Zimmermann wrote and published PGP in 1991, in response to the possibility of government intervention into encryption production development.

PGP can utilize several algorithms, and is a "hybrid" solution, in that it utilizes both private- and public-key cryptography. Digital certificate verification uses a "web of trust" model, relying on all of its users to assure trust, rather than a centralized commercial entity.

PGP implementations are now available on Macintosh, Unix/Linux variations, several versions of Windows platforms[9], and command-line and graphical user interface (GUI) based programs. It can be integrated with e-mail systems such as Microsoft Outlook, Microsoft Outlook Express, Lotus Notes, Eudora; versions have been available to plug in to instant messaging programs such as ICQ as well. [16]
Versions available from PGP, Inc. include Corporate, Personal and Freeware versions. Freeware is available only for non-commercial uses. Federal export restrictions on PGP have been significantly lightened, meaning that it is available

[7] http://wombat.doc.ic.ac.uk/foldoc/foldoc.cgi?RIPEM
[8] http://www.networkassociates.com/us/nailabs/research_projects/network_security/infosec.asp
[9] http://www.pgpi.com

for use worldwide. Perhaps most importantly, in 2002, the source code [programming instructions] was once again made public, to allow for peer review, after a period of time when the code was kept secret under the control of Network Associates, Inc. This move has helped to restore the faith of the cryptography community to the program: [20] "... any cryptographer can tell you that a well-designed encryption algorithm does not have to be classified to remain secure. Only the keys should need protection." [3]

S/MIME

S/MIME – Secure Multi-Purpose Internet Mail Extensions – is a secure method of sending e-mail that incorporates encryption information and a digital certificate within the message body, in accordance with the MIME standard itself, as laid out in RFC 1521[10]-[11]. S/MIME was first proposed by RSA Data Security, Inc. in 1995. It uses the RSA encryption algorithm, based on Public Key Cryptography; Digital certificates are based on X.509. Verification follows a purely hierarchical model: this means that encryption key generation and registration must be done through commercial Certificate Authorities (CAs), such as Verisign. This model does not always lend itself to thoroughness, however: large CAs must process a high volume of these transactions at a very low premium, leaving limited resources for validity checking. [9]

S/MIME is not restricted to mail; it can be used with any transport mechanism that transports MIME data [30], such as Web transactions.
Built-in support for S/MIME has been programmed into recent versions of Netscape Communicator and Internet Explorer, the two most widely-used web browsers, and their associated mail programs – this has provided a major boost for S/MIME in the marketplace. However, it also has not become an agreed-upon standard for e-mail encryption, as some had expected it would. The current working version at the IETF (Internet Engineering Task Force) is S/MIME version 3, handled by the S/MIME Working Group[12].

IBE

Alternatives to certificate-based message security (such as PGP and S/MIME) have recently been developed, including Identity Based Encryption (IBE), which allows e-mail senders to use arbitrary strings (such as e-mail addresses) as public keys. For example: the recipient's e-mail address is hashed, which serves as the beginning of the public key generation process. Upon receipt, the recipient uses a private key generator (PKG) to construct a private key to decrypt the message.
In 2001, a team from Stanford University developed a proof-of-concept system based on Weil Pairings on Elliptic Curves. More information on the research and development of this system can be found at the project team's website: http://crypto.stanford.edu/ibe/

[10] http://www.faqs.org/rfcs/rfc1521.html
[11] http://whatis.techtarget.com/definition/0,,sid9_gci214187,00.html S/MIME def
[12] http://www.imc.org/ietf-smime/index.html

Under this system, the encryption components are truly "certificateless", but a specialized server must still be established to store the encryption parameters and act as a private key generator. Once a client in this system caches the public parameters of a recipient's "security district", they can send encrypted messages to any member of that district, even while offline. The major drawback of IBE is that it provides no **A**uthentication or message **I**ntegrity protections -- these features would have to be layered on top of an IBE solution to provide capabilities equivalent to PGP or S/MIME, adding to the complexity of the system.
The main draw of IBE systems is expected to be for applications where users must send encrypted mail to recipients who are not enrolled in an e-mail security system.
Voltage Security (http://www.identicrypt.com/) offers three products as part of their IBE solution: a key server, mail client, and policy server. These products currently run on several versions of Windows systems, and are expected to be offered for several Unix varieties soon. Voltage offers integration capabilities for Yahoo! and Hotmail accounts as well.

Choosing a solution

"The algorithm used must be a well-known, established, scrutinized, tested, accepted method of encryption." [26]

In addition to these four options, there are literally hundreds of lesser-known solutions available in the marketplace, most of them available either for free or at very low cost. So why limit one's choices to one of these? In general, lesser known standards and programs provided by third parties are not based on one of the two main current standards (PGP and S/MIME). They are probably not widely used for good reasons, and typically should not be relied upon, as Phil Zimmermann explains:

> "I learned how easy it is to fall into a false sense of security when devising an encryption algorithm. Most people don't realize how fiendishly difficult it is to devise an encryption algorithm that can withstand a prolonged and determined attack by a resourceful opponent. Many mainstream software engineers have developed ... naive encryption schemes, and some of them have been incorporated into commercial encryption software packages and sold for good money to thousands of unsuspecting users." [3]

This is not to say that all of the other programs available are no good; rather, it is to say that they have not been in existence long enough, tested thoroughly enough, and proven their mettle enough to be considered secure on the same level as the options presented here. Choosing one of these lesser-known solutions may suit your needs fine, and the solution may, in the long run, be considered equally secure, but the risks of relying on a potentially insecure program are enormous:

> "This is like selling automotive seat belts that look good and feel good, but snap open in the slowest crash test. Depending on them may be worse

than not wearing seat belts at all. No one suspects they are bad until a real crash. Depending on weak cryptographic software may cause you to unknowingly place sensitive information at risk when you might not otherwise have done so if you had no cryptographic software at all. Perhaps you may never even discover that your data has been compromised." [3]

PEM is no longer a relevant competitor in the field of e-mail security. IBE, on the other hand, is too relatively "young", and therefore has not gained enough of a following to be effective, nor has it been tested thoroughly enough.
PGP and S/MIME perform effectively the same functions; however, they do not interoperate. In fact, they are considered "fundamentally incompatible" from "technical, practical and policy perspectives" [14]. Users of PGP cannot communicate securely with S/MIME participants using their respective programs. Therefore, a choice has to be made between the two. PGP remains the most widely *used* secure e-mail protocol, while S/MIME is actually more widely *deployed* – built-in integration with the two most popular web browsers. [14]
"PGP vs. S/MIME, S/MIME vs. PGP. On the one hand, it really doesn't matter which of the two technologies you choose. From a user's perspective, both provide the same set of security services, and neither really has any significant advantage over the other." [14]
From a high-level perspective, this statement is accurate. However, when you are able to identify the preferences or needs of a relatively small, unique population segment, such as SOHO e-mail users, the small differences between the two become more important, and the choice becomes more clear.

- The fact that S/MIME is integrated with the two most popular web browsers, and yet still lags PGP in amount of use, speaks volumes about the lasting popularity and confidence in PGP, and the uncertainty and indifference towards S/MIME.
- PGP was originally designed for use by individuals, so its products include all the ancillary support necessary for an individual to get started – in this case, the ability to generate a public/private key pair and issue one's self a PGP certificate.
- The Public-Key infrastructure of PGP lends itself well to networks of more than a handful of users in a decentralized, distributed structure (no centralized key protection required), with dynamic hosts.
- The longevity and open source nature of PGP lends it credibility and confidence among serious cryptographers that simply does not exist when considering a proprietary, closed-source option such as S/MIME:
    - "Since we can't prove there are no weaknesses in either implementation, the probability of there being such weakness is a straightforward function of the expert man-hours spent searching for them. One doesn't have to assert there ARE such weaknesses to make this argument. Thus the risk in using PGP is less than the risk in using S/MIME implementations that are not available with source." [14]

- PGP also provides two additional features that S/MIME does not include: local file encryption and secure deletion.

A case study of a real-world decision process comparing and choosing between PGP and S/MIME can be identified with the Gas Industry Standards Board (GISB) decision to adopt PGP for its 165 corporate members. Although certainly not a small organization, the GISB chose PGP for similar reasons that a SOHO user might: flexibility, ability to work in a decentralized, widespread environment, confidence in the program and its algorithms, cost effectiveness, e-mail and local file-encryption, and, most importantly: "the group felt it was better suited for its requirements, which include data privacy [**C**onfidentiality], [**I**]ntegrity, [**A**]uthentication and [**N**]on-[**R**]epudiation."[13]
The remainder of this paper will concentrate on PGP as the choice of e-mail encryption for Small Office / Home office users.

[13] http://www.internetweek.com/case/study081699-1.htm

Pretty Good Privacy

"There's nothing wrong with asserting your privacy." [34]

History of PGP

Zimmermann wanted to create an encryption program "for the masses", as mentioned above:

"…no one draws suspicion by asserting their privacy with an envelope. There's safety in numbers. Analogously, it would be nice if everyone routinely used encryption for all their email, innocent or not, so that no one drew suspicion by asserting their email privacy with encryption. Think of it as a form of solidarity." [34]

He was under federal investigation for three years following the publication and worldwide distribution of PGP in response to the threat of tightened federal restrictions on cryptography development (cryptography was considered munitions, and therefore subject to tight controls, especially in terms of international distribution).

"Despite the lack of funding, the lack of any paid staff, the lack of a company to stand behind it, and despite government persecution, PGP nonetheless became the most widely used email encryption software in the world." [18]

PGP has transformed significantly since its early days, gone through several phases of ownership, become proprietary and then switched back to open source again – but its popularity has never significantly faded.

How it works

Message Encryption

PGP relies on the Public Key Cryptography system, utilizing a public key and a private key, as previously defined.

The process of encrypting messages for **C**onfidentiality purposes occurs as follows:

Essentially, the sender's software chooses a session-long secret key to be used; encrypts the message with this key, then encrypts this secret key with the recipient's public, shared key. The recipient decrypts the secret key with their private key (which is mathematically-related to their public key, but, of course, cannot be determined from knowledge of the public key), and decrypts the message contents with the newly decrypted secret key. Understanding the details of this complex arrangement is not essential, and, fortunately, PGP accomplishes all of this fairly transparently, from an end user's perspective.[14]

Digital Signatures

[14] Algorithms supported by PGP include:
- either SHA-1 or MD5 for message hashing;
- DES, CAST, Triple DES, or IDEA for encryption; and
- RSA or DSS/Diffie-Hellman for key exchange and digital signatures.

Digital signatures provide verification of the **I**ntegrity of the message, the **A**uthenticity of the sender, and **N**on-**R**epudiation of the message.
Digital signing is a two-stage process in PGP:

- first, the message is "hashed". Hashing is a one-way function that converts the original text into a 'message digest', from which retrieving the original text is not possible; identical text hashed with the same algorithm will always produce identical message digests, and (theoretically), no two different text inputs will ever result in the same message digest. These message digests are often referred to as "fingerprints" of the message.
- The resulting hash is encrypted using the sender's private key. The result is a digital signature (a.k.a. Message Authentication Code, MAC), which can be sent along with the original message, and used by the recipient to verify the message by decrypting the digital signature using the sender's public key, to get the hash value. A hash of the actual received message is then computed using the same algorithm, and compared to the decrypted hash; if they match, then the message is confirmed. [26] The hashing process is what actually confirms the **I**ntegrity of the message; the private key encryption provides the **A**uthenticity and **N**on-**R**epudiation aspects.

Other Capabilities
PGP also performs compression of e-mail messages, reducing file sizes, which speeds the transmission and reduces storage requirements. The PGPdisk Volume Security component offers the ability to encrypt local file system data and "Wipe" files or hard drive freespace – a method of repeatedly scrubbing data away to securely and permanently delete information.
PGP has also now been extended to encrypt Internet phone (Voice-over-IP -- VoIP) conversations, although that is beyond the scope of this paper.

Benefits for SOHO Users

Because PGP does not require the same platform, e-mail application, or any intervening infrastructure, it is considered very easy to deploy. The only requirement is that both users have PGP software installed, and know one another's public keys. This allows for users to be added to the PGP "network" very quickly and easily. The decentralized nature of certificate distribution ("Web of Trust", described below) provides great flexibility and control for individuals and small groups of users to arrange communications among themselves, without having to rely on unwieldy, costly bureaucratic structures.

Programs
PGP software for commercial use can be purchased directly from PGP, Inc.[15], where Subscription Licenses start at $125 per year for Corporate desktop versions[16], or from a number of third parties with integrated products.

[15] http://www.pgp.com/products/index.html
[16] https://store.pgp.com/default.php?cPath=65

"Companies seeking an inexpensive [$59] yet easy-to-use encryption solution to secure e-mail, instant messages or shared files will find AritcSoft's FileAssurity OpenPGP 2.0.2....well worth a look." [10]

Hushmail (http://www.hushmail.com/) offers web-based e-mail service using the OpenPGP encryption standards, starting at $29.99 per year. Plug-ins are available for several e-mail clients as well, for an additional cost.

IMPLEMENTATION

"PGP empowers people to take their privacy into their own hands." [34]

In order to demystify the process and demonstrate how simple installing and setting up PGP can actually be, the following steps will be illustrated:

1) Download and Install
2) Configuration Wizard / Key Generation
3) Send key to public servers
4) Sign and Import others' public keys

These four steps are enough to get PGP fully installed, configured and prepared to begin sending and receiving secure e-mails. Actual transfer of e-mail using PGP is actually quite easy and intuitive, and is best learned by using the program first-hand.

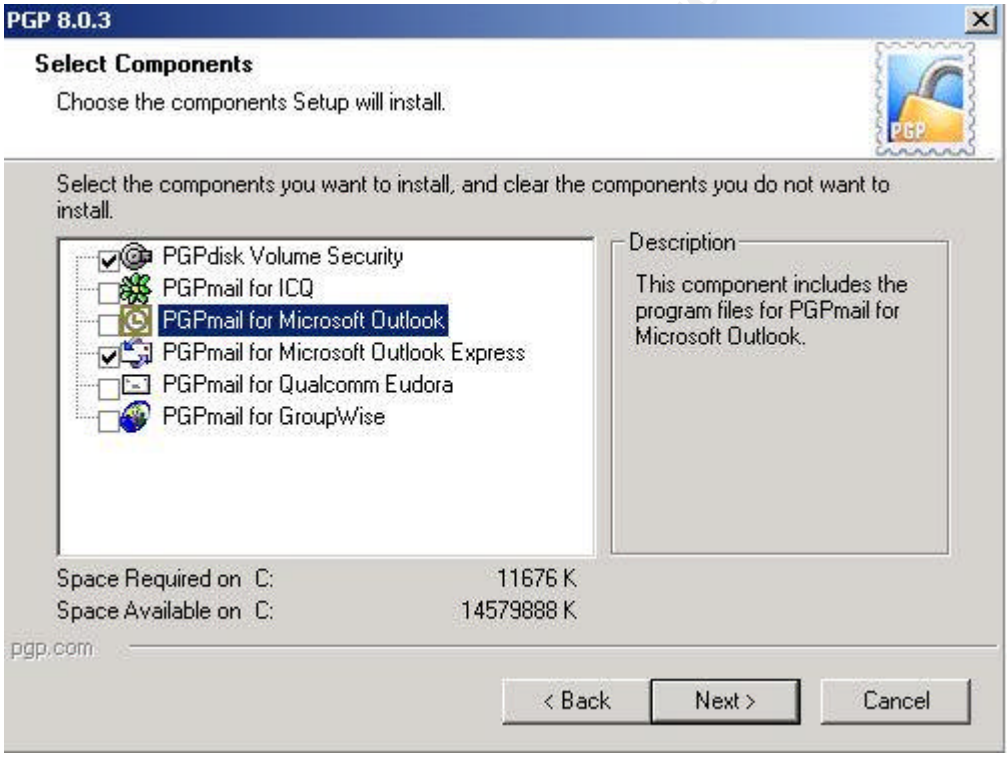
Installation and Configuration

1) Download the appropriate version of PGP by following the links from www.pgpi.com.
   (For this example, PGP 8.0.3 for Windows 2000)

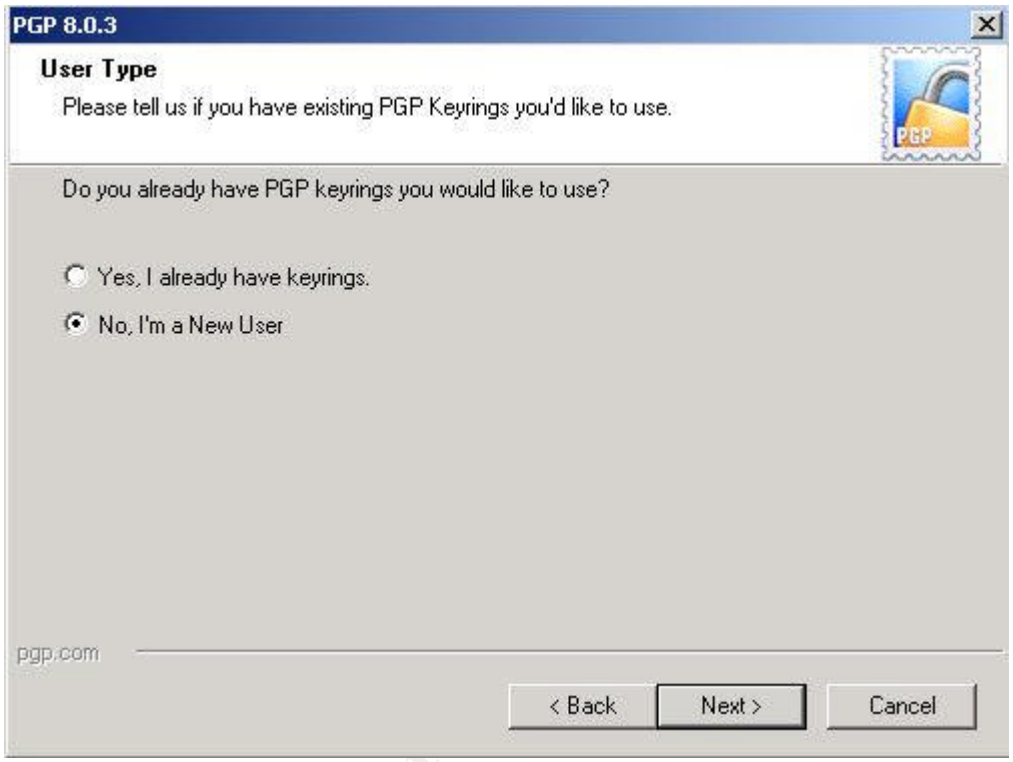
   - Unzip the setup files if necessary; Begin install by double-clicking the appropriate setup file (e.g., PGP8.exe);

   - Read and Accept the End User License Agreement (EULA);

   - Select the Components to install, based on your e-mail client and needs (Make sure to select 'PGPdisk Volume Security' if you would like to encrypt files on your local system with PGP):
     (In this example, PGPdisk Volume Security and PGPmail for Microsoft Outlook Express)



2) Select ‘No, I’m a New User’ when asked if you already have keyrings to use:

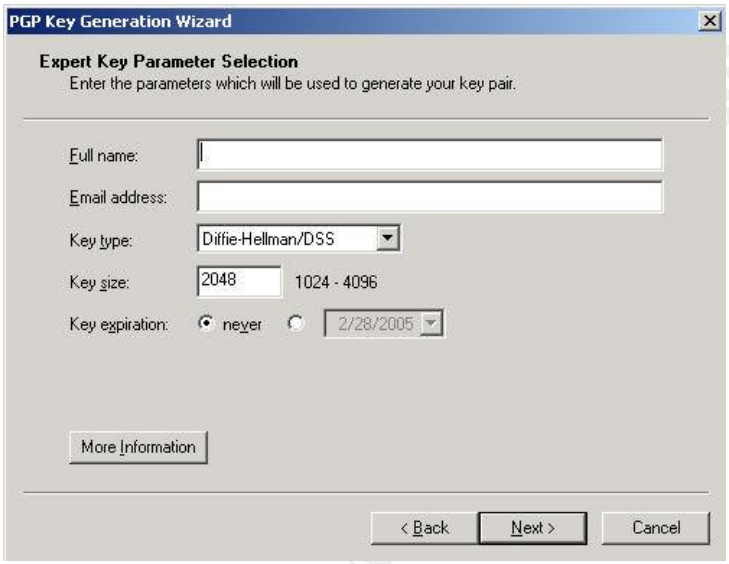


- Reboot the machine if prompted;

- Upon restart, either enter license information when prompted or choose “Later”, if using the freeware version;

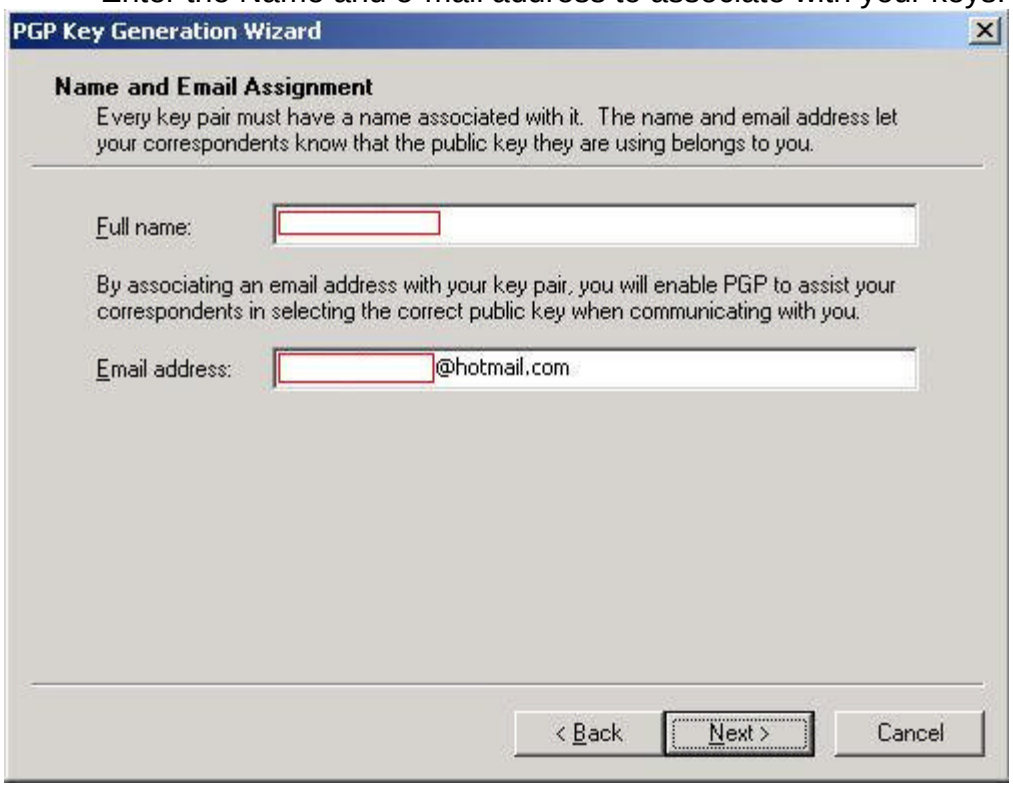
    - **Note:** Users preferring advanced control over their key generation process, rather accepting all default values, may choose the “Expert” option from the next screen, which presents a screen of selectable parameters such as algorithm, key size, and expiration.
      Here you’re given a choice of how strong you want your key to be. The bigger the number, the more secure it is, and the longer it will take when encrypting, decrypting, or signing messages. This may be set arbitrarily, depending on how long you may require your information remain secret. Experts currently tend to agree that 2048 bits provides a good balance between performance and security. [26]

You can also choose to have your key expire at a specific time in the future. This may be useful when creating a key for a temporary employee, contractor, or business partner where you want to have limit over how long it is valid. Expiration time can also be related to the sensitivity of the data it protects: less sensitive data can have a longer key lifetime; more sensitive data should have a shorter key lifetime.



- Begin Key Generation by clicking “Next” to start the “PGP New User Configuration Wizard”

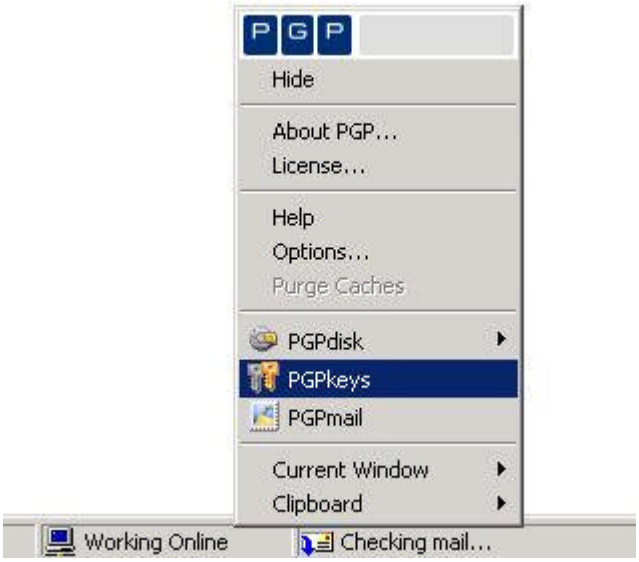
- Enter the Name and e-mail address to associate with your keys:



- Next comes the passphrase generation. The passphrase protects the local storage of your private key, which is encrypted to prevent anyone with access to the machine from using it. You must create a good, strong (read: complex) passphrase, but be sure you will not forget it: if a passphrase is forgotten, any data that has been encrypted with your private key will be rendered inaccessible. [26]
  The passphrase must be entered any time access to the private key is required for an operation in PGP.
  Note: The passphrase strength bar will increase as you make your passphrase longer and more complex, meaning more difficult to guess. This is a helpful guide to pay attention to.

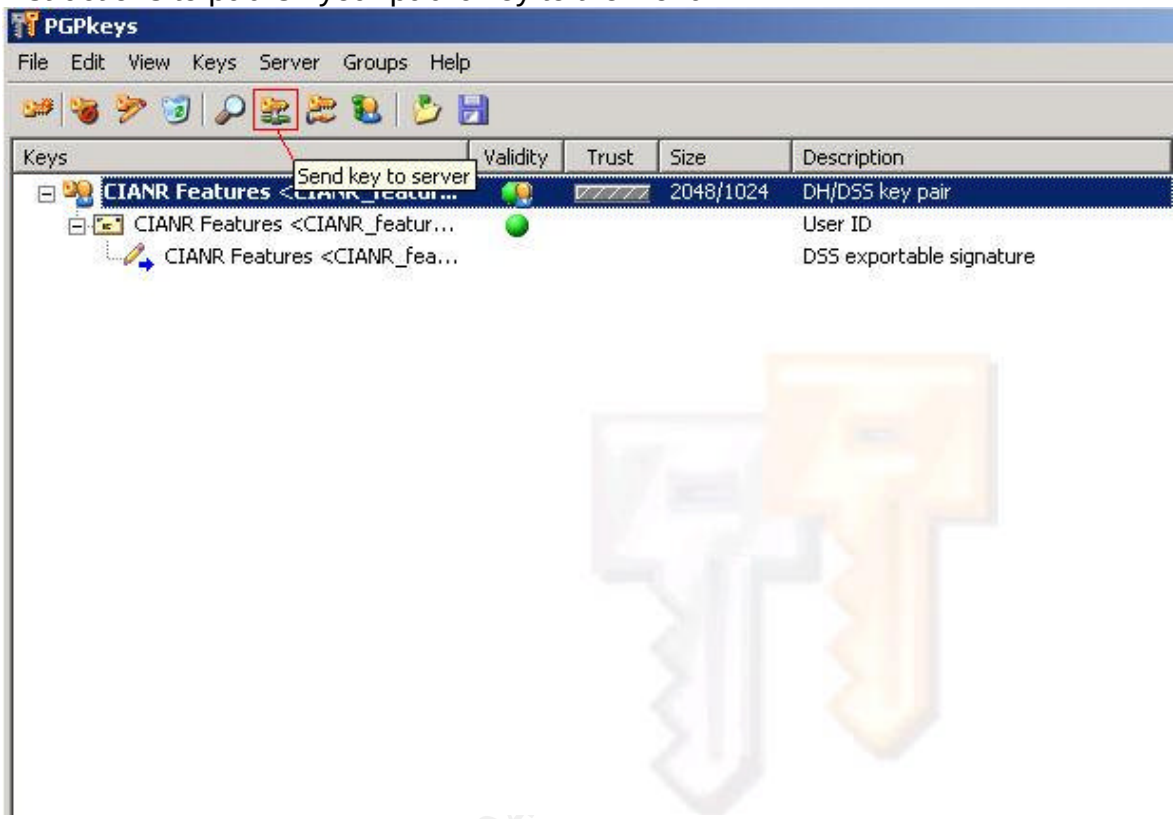
- Successful Key Generation message:

- You are now able to view your PGP keys setup from the toolbar menu in the lower-right of the screen:

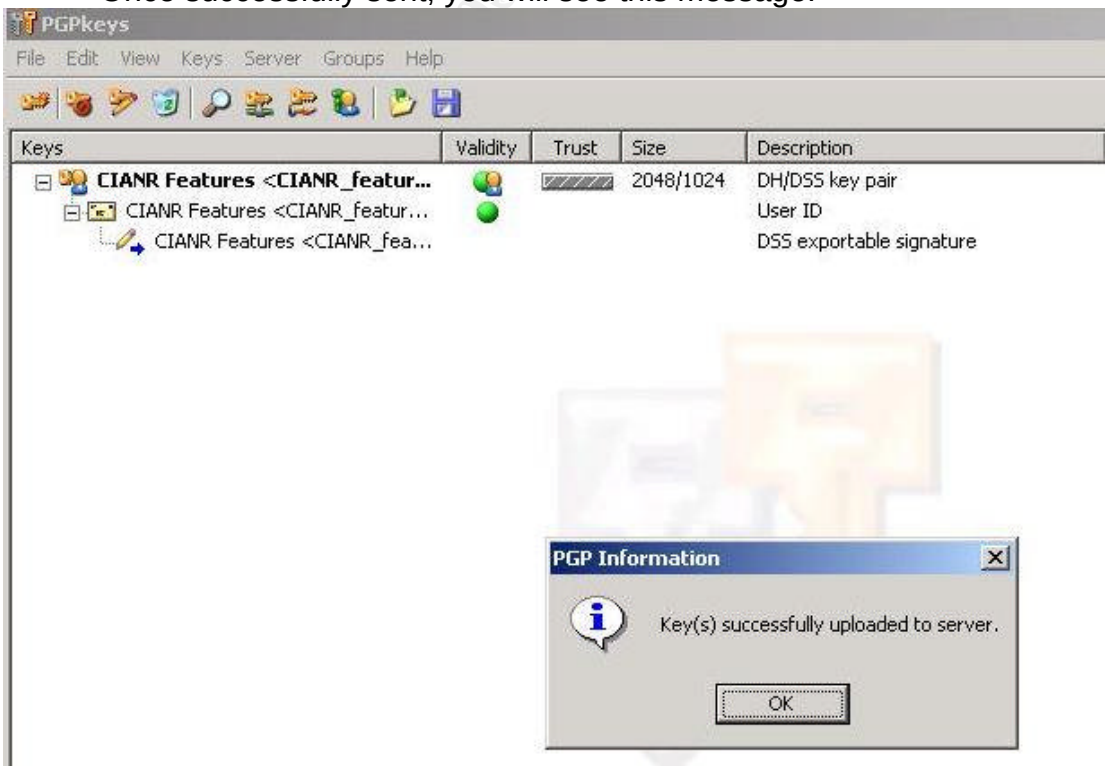


3) To send your public key to a server from within your e-mail client:

- click on the “PGP keys” icon in your client, or, alternately, the PGPkeys item from the toolbar menu shown above. Click on the icon highlighted below and follow the instructions to publish your public key to the world.



- Once successfully sent, you will see this message:

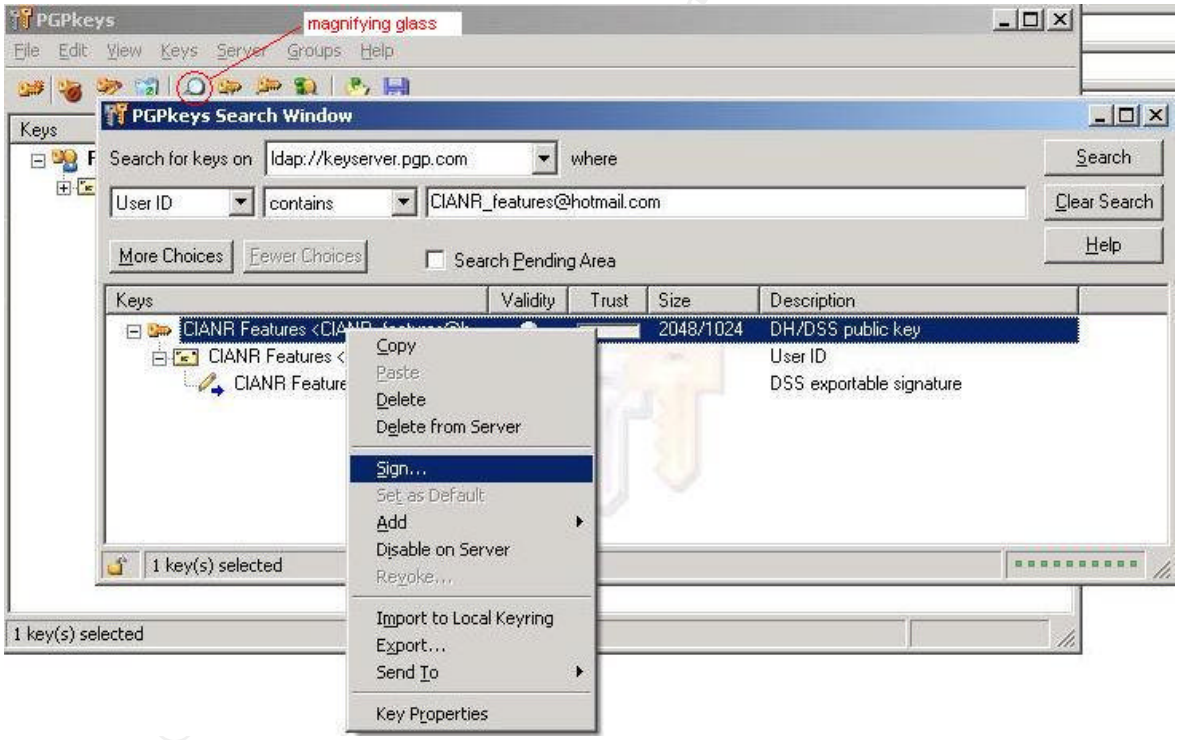


- In the future, when someone requests your key(s), one may either: directly send the key file with a signature inside, the key file with the accompanying signature file, or direct them to the appropriate key server and key ID (found in key properties).

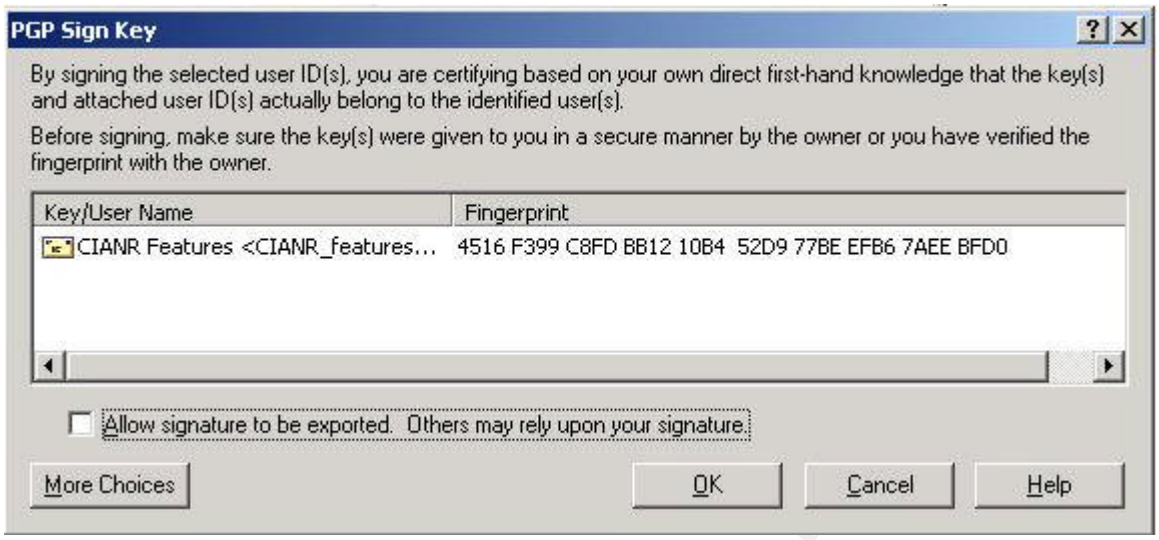
4) Finally, In order to begin sending encrypted e-mails, a user must also locate, (sign, optionally) and add to their key ring other users' public keys as illustrated in the following several images:

- Search for a recipient's public key on a public server by clicking on the "Open key search window" icon (magnifying glass). Searching is available by User ID, Key ID, Creation Date, etc.:

- Once the key has been located, right-click and select "Sign" from the context menu if you would like to add your signature to the key, asserting your knowledge that the key is, in fact, valid:

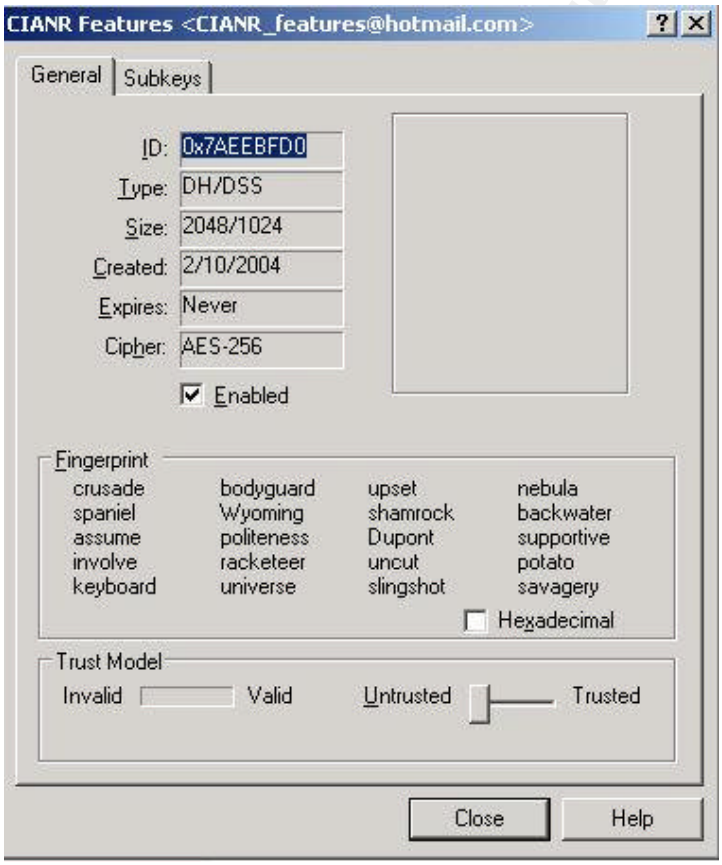


- Before signing the key, the following message will be displayed to remind you of what you are attesting to, and to view the certificate fingerprint:

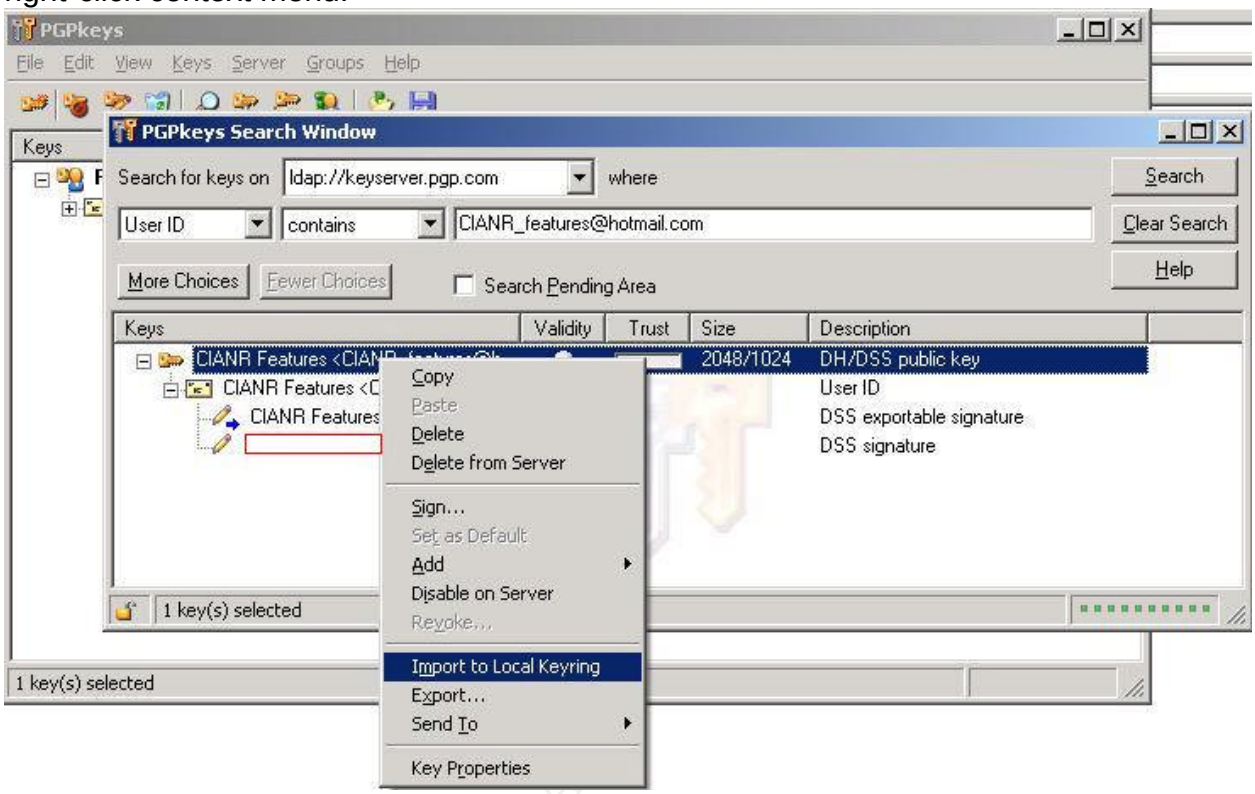


- **Note:** A PGP user can also sign their own key at any time, and resubmit it to public servers.

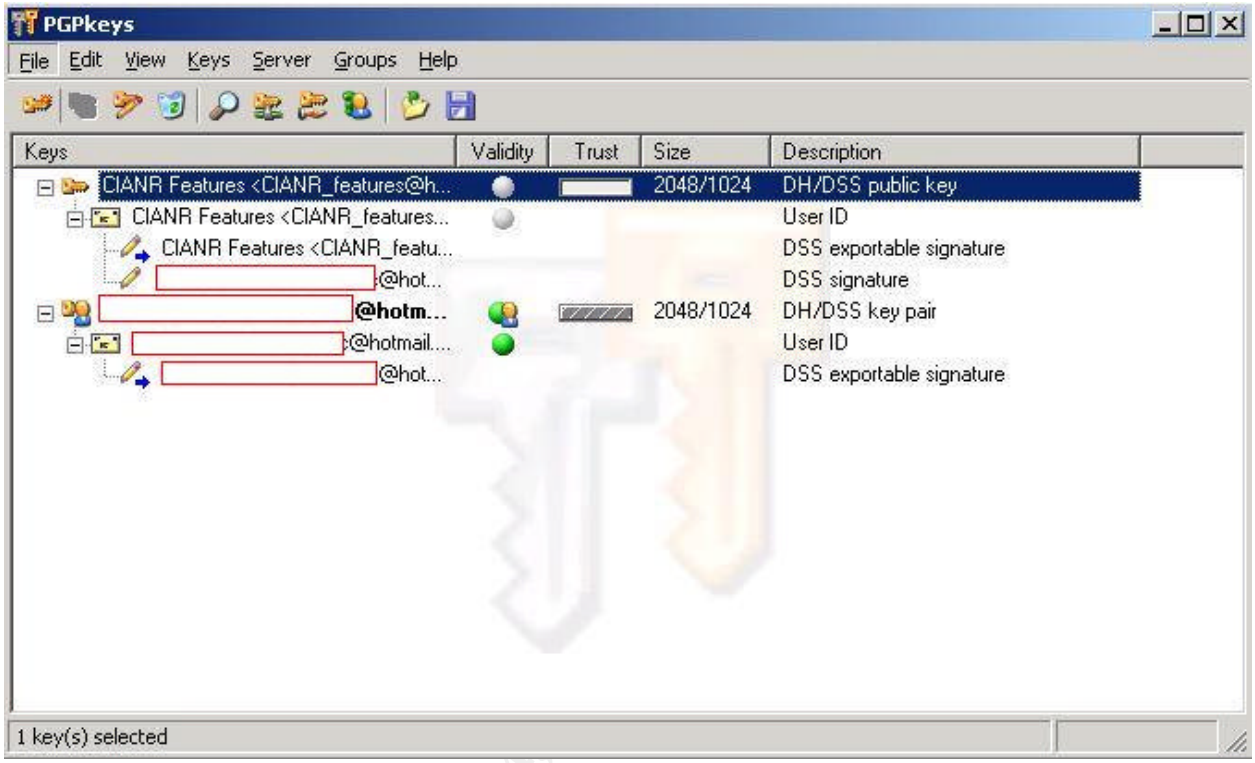
- Additional key properties can also be viewed from the search results window: right-click on the key of interest, and select “Properties” from the context menu:



- When you are ready to import the key, select “Import to Local Keyring” from the right-click context menu:



- After a successful import, your personal keyring will now have two entries – your own, and your newly added key:



- In the future, in addition to the method just illustrated, other users' keys may be found on personal websites, public websites (such as http://pgp.mit.edu), received directly from the user via e-mail, received from another, trusted user via e-mail, or received "out-of-band" somehow -- on a floppy disk, for example. (refer to [19] for instructions)

Key Management Issues

Web of Trust

The issue of certificates and trust is central to any public key cryptography system. If you do not have a high level of confidence that a public key is owned by the person who claims it, you cannot have confidence that your private e-mails are reaching your intended recipient.

PGP is based on the idea that trust is a social concept. [8] PGP places the burden on the end-user, who must maintain a list of trusted issuers. This “Web of Trust” means, essentially, that each user is able to decide what other users they will trust to introduce other parties, or keys, to them. Any user can "sign" the public key of any other user – by signing, the user asserts that they believe the key does belong to the person assigned to it, and that they know how to correctly use PGP. Users

can decide whether or not to trust, or accept, a particular public key based on the number of other users who have signed the key, and the "quality" of these users, a subjective measurement. There are four levels of trust available to sign to other users.

When a certificate is obtained, a cryptographic hash, or "fingerprint", should also be obtained from the user, ideally via a different channel (e.g., certificate via website, fingerprint via telephone).

PGP computes a trust level for each public key in your keyring – this is partially controlled by the user's input:
- Key legitimacy field: Measures the degree to which this PGP user trusts that the key is valid for its user. The higher the level of trust, the stronger is the binding of this user ID to this key.
- Signature trust field: Measures how far the PGP user trusts the signer to certify public keys. (The key legitimacy field for an entry derives from the signature trust fields.)
- Owner trust field: Indicates the degree to which this PGP user trusts the key's owner to sign other public-key certificates. User assigns this value.

Key Security
It is generally agreed that the security of an encryption deployment is based upon the security of the key(s) involved -- "not the secrecy of the algorithm, the inaccessibility of the ciphertext, or even the inaccessibility of the plaintext." [29] Users should always be aware of the physical location of their private key file and should not store the file on any machine that they do not have complete physical control over. It is strongly recommended that a user back-up their key pair immediately after creation (refer to [19] for instructions), and encrypt these with conventional encryption.
Additionally, users should not store their passphrase on the computer where their private or secret key file is located.

Key Revocation
When a user wishes to no longer trust a key, there are two options:
Signature revocation or Certificate revocation.
When an employee quits, revoking your signature of their certificate is a way to inform other users that your original opinion of the key is no longer valid – the ex-employee should no longer be using a key associated with an e-mail address at your company. This does not, however, actually invalidate the key. Others may still trust the key (possibly unaware of the occurrence that led you to invalidate), and it can continue to be used.
Revoking the certificate itself, on the other hand, invalidates it entirely. This operation can only be performed by the person who created the certificate or has been assigned by the owner as a designated revoker. Therefore, if you create certificates for employees, it is a wise idea to assign yourself as a designated revoker, should the need arise.

Certificate revocation may be necessary in the situation where a user learns or suspects that their passphrase or secret key file has been compromised. In this situation, all people that the user has exchanged PGP encrypted messages with should be contacted, warned of the compromise and instructed to stop using the user's public key. [2] Revoking the certificate will ensure that no one unwittingly communicates private information to the attacker. Another situation where revocation would be helpful, although not as urgent, would be when a user chooses to retire an e-mail address and the key(s) associated with it.

Sending / Receiving PGP Messages

After navigating the key generation, distribution, and storage process, one can finally get down to business: sending and receiving PGP-secured e-mail messages. This is relatively simple, and should be a straightforward process for a new user, with one important note:
Messages can be either signed (providing **I**, **A**, and **N**-**R**), encrypted (providing **C**), or both signed and encrypted (**C**, **I**, **A**, **N**-**R**). The order of operations when both signing and encrypting is exactly that – sign first, then encrypt.

Problems and Inhibitors to Implementation

"The availability and affordability of cryptography, added to the frequency of e-mail security breaches, makes it mystifying why so few companies currently take effective precautions. The answer to the low take-up so far of e-mail security lies in misconceptions and corporate complacency." [33]

While PGP does provide the best option for secure e-mail for SOHO users at this time, it is not without flaws.

Usability

PGP still has a reputation for being difficult for beginners -- especially non-technical users -- to comprehend, understand, and correctly implement and use consistently.

An study conducted by two researchers in 1999 [35] showed that a majority of users in their sample could not correctly use PGP 5.0, and in some cases actually compromised security by improper use or misunderstanding. Since that time, the usability of PGP has improved considerably, but the complexity of the scheme still intimidates many novice users.

Misconception of Conflicting Security Technologies

There is a widely held belief that encrypted e-mail messages negate other areas of security because they are able to by-pass anti-virus and content-checking software....The reality, however, is that anti-virus software and cryptography are fully compatible. Anti-virus software can be hampered by encryption when it is

installed on a network server. When the installation is made at the desktop level, the two sets of software are completely complementary. [33]

Costs of Implementation

Action Item	Cost in Time	Financial Cost
Research of various encryption options, programs, and licenses to determine which is most suited to needs.	5 hours	$0
Purchase of encryption technology license(s).	1 hour	~ $30 - $200 / user
Reading and Education	4 hours	$0
Installation and Configuration	2 hours	$0
Totals:	**~12 hours**	**~ $30 - $200 / user**

Considering the value of data being transported and potential impact of a security breach in e-mails, these costs seem like a small price to pay for the security offered.

Potential Attacks Against PGP

According to its creator, Phillip Zimmerman, PGP was developed with "the best algorithms from the published literature of civilian cryptologic academia. These algorithms have been individually subject to extensive peer review....But you don't have to trust my word on the cryptographic integrity of PGP, because source code is available to facilitate peer review." [3]

Several practical and theoretical attacks have been illustrated that may provide avenues to compromise the security provided by PGP. These are not based on any fundamental flaws within the PGP structure or the algorithms it utilizes; rather, they are typically aimed against user implementations on real-life systems. Such attacks include: obtaining a user's passphrase and/or private key file, subverting user's trust of public keys by tampering with personal public key rings, operating system-based attacks, trojaned software versions, brute force attacks against cryptographic elements, and low-tech attempts such as "shoulder surfing", breaching physical security, or social engineering. Countermeasures may be adopted to help protect against the possibility of these threats, where necessary. [2] More detailed, technical information is available for reading at:
http://www.schneier.com/paper-chotext.pdf
http://www.schneier.com/paper-pgp.pdf
http://www.sans.org/rr/papers/index.php?id=1092

Additionally, research should be done on the particular product and version chosen for implementation to learn about any known vulnerabilities and countermeasures to take. A keyword search for “PGP”, “GnuPG”, etc. at http://www.securityfocus.com/bid/keyword/ will list information from the BugTraq database on known security issues.

Complacency

The answer to the complacency issue can be found in the findings of the CBI's 2001 cyber crime survey. While 73% of companies acknowledged that cyber crime was rising, only 42% felt it would increase in their own business. In layman's terms, the take-up of e-mail security is being blighted by a widespread outbreak of "it won't happen to me" syndrome. [33] Anyone can understand the concept of clear-text messages and wide-open access to data on the Internet, but few people can comprehend why anyone would want to eavesdrop on *their* data, or believe that an attacker would be able to find *their* meaningful data from among the enormous amount of data flow in the Internet. Much in the same way that people have a difficult time imagining that *they* will ever be the victim of a flood or identity theft...

Simply implementing PGP or comparable cryptography is not a guarantee of security, however. An axiom of the information security community is that "Security is a process, not a product." This fully applies to encryption solutions – processes, policies, and practices must be implemented and followed on an ongoing basis to receive the continued benefit of e-mail security. Some of the biggest obstacles to overcome in a successful PGP implementation include key management, training and awareness, and consistency in use.

Future of cryptography for this market

Groups such as the IRTF (Internet Research Task Force) are working on countermeasures for such abuses, but the challenge is daunting. Even with the promise of new technical approaches, the implementation of new standards will take a lot of time and effort because of the global nature of e-mail. And because e-mail is already built into so many applications, implementing a new standard could "break" a lot of systems. [4] Small businesses and organizations cannot afford to wait for new standards to be proposed, discussed, agreed upon, developed, tested, and implemented to solve their privacy and security needs. ***Take existing e-mail protocols and bundle on top the best of existing privacy and security solutions -- PGP***

Some industry observers suggest that security, including that of e-mail, will begin to be taken more seriously by organizations as legal claims are won against companies failing to secure information, and insurance policies are created to protect against such claims.[17] Enhanced security could become a financial advantage, in the form of lower insurance premiums. [33] In addition, laws and regulations related to information security continue to be a hot topic of discussion in Washington, D.C., and throughout the I.T. industry.

Ultimately, the success of affordable, simple encryption will rely on the commercial success, based on Total Cost of Ownership (TCO) and Return on Investment (ROI) calculations in organizations, as much as the continued innovation of

[17] http://www.schneier.com/crypto-gram-0204.html#6

cryptographers, and continued relaxation of government restrictions. Strong and affordable solutions such as PGP offer an optimistic view that investments in cryptography pay off with security dividends that can easily justify and outweigh the costs involved.

PGP remains the encryption tool of choice for the majority of technical users and open-source advocates. On the other hand, the weight of such vendors as Microsoft and Netscape incorporating S/MIME into their dominant products provides a very strong endorsement of this standard. Emerging cryptosystems such as IBE provide exciting glimpses into the future of e-mail encryption technology.

Conclusion

"The only way to hold the line on privacy in the information age is strong cryptography." [34]

As technological innovation continues to advance the communication and collaboration capabilities of parties across the globe, electronic communications such as e-mail will only grow more important. Many organizations already rely on this medium to transmit sensitive information without any form of security, blissfully or deliberately ignorant to the potential dangers involved. As disrupting e-mail communications grows to become a more profitable endeavor for malicious users on the Internet, incidences of security breaches in e-mail will become more commonplace and costly. Tools are available and easily-implemented to prevent this kind of breach: encryption using strong, proven systems such as PGP, when properly implemented and maintained, provide a nearly impenetrable defense against e-mail security attacks.

The choice rests with each and every organization and individual: maintain the e-mail status quo, crossing fingers and hoping not to become a victim (although perhaps never knowing if you have), or take the initiative and empower yourself or your organization to utilize the excellent tools that have been provided by pioneers such as Phillip Zimmermann. Secure your communications from anyone, everyone – and use e-mail to help advance your communications and promote your organization's goals without worry.

Works Cited

[1] Schneier, Bruce. Applied Cryptography. 2nd ed. N.p.: John Wiley & Sons, 1995.

[2] Thomas, Ryan. Attacks on PGP: A User's Perspective. GSEC Practical Assignment, 08 Jan. 2004 <http://www.sans.org/rr/papers/index.php?id=1092>.

[3] Zimmermann, Phillip. Beware of Snake Oil. 1997. 19 Dec. 2003 <http://www.philzimmermann.com/essays-SnakeOil.shtml>.

[4] Metz, Cade. "Can E-Mail Survive?" eWeek 17 Feb. 2004. 17 Feb. 2004 <http://www.eweek.com/article2/0,4149,1483793,00.asp>.

[5] Katz, Jonathan. Schneier, Bruce. A Chosen Ciphertext Attack Against Several E-Mail Encryption Protocols. June 23, 2000. 09 Dec. 2003 <http://www.schneier.com/paper-chotext.pdf>.

[6] Court ruling in USA of ' Privacy ' of Email. 1996. University of British Columbia. 22 Nov. 2003 <http://axion.physics.ubc.ca/email-privacy.html>.

[7] Elias, Ilan. E-Mail Security. Student Lecture. 18 Jan. 2004 <http://www.cs.huji.ac.il/~sans/students_lectures/PEM.ppt>.

[8] Schneier, Bruce. E-Mail Security: How to Keep Your Electronic Messages Private. New York: John Wiley & Sons, Inc., 1995. 105-166.

[9] Kuzmowycz, George. E-Mail Security with S/MIME. GSEC Practical Assignment, 06 Oct. 2003 <http://www.sans.org/rr/papers/index.php?id=739>.

[10] Garcia, Andrew. "FileAssurity secures e-mail." eWeek 22 Dec. 2003: 49.

[11] Hatch, Brian. File and email encryption with GnuPG (PGP) parts one and two . 2003. 02 Jan. 2004 <http://www.hackinglinuxexposed.com/articles/20031204.html>, <http://www.hackinglinuxexposed.com/articles/20031231.html>.

[12] "Identity Based Encryption." Cryptonomicon.Net. 02 Feb. 2004 <http://www.cryptonomicon.net/modules.php?name=News&file=print&sid=631>.

[13] Jallad, Kahil. Katz, Jonathan. Schneier, Bruce. Implementation of Chosen-Ciphertext Attacks. 2000. 04 Dec. 2003 <http://www.schneier.com/paper-pgp.pdf>.

[14] Galvin, James M. "(IN)SECURITY FROM END TO END." Information Security Magazine Mar. 2000. 12 Jan. 2004 <http://infosecuritymag.techtarget.com/articles/march00/features2.shtml>.

[15] Intense School's CISSP Part 1 Boot Camp Workbook. Vol. 2. N.p.: IntensePrep, 2002. 489-546.

[16] Mail Client Capabilities: PGP-related. 19 Dec. 2003 <http://www.rjmarq.org/pgp/mail-clients-pgp.html>.

[17] Hamrick, Matthew S. "Matt's Quick Guide to Using PGP." Cryptonomicon.Net. 13 Dec. 2003 <http://www.cryptonomicon.net/howto/pgp.html>.

[18] Zimmermann, Phillip. Phil Zimmermann's Home Page. 03 Dec. 2003 <http://www.philzimmermann.com/index.shtml>.

[19] Hoel, Jeremy. PGP for Everyday Use. GSEC Practical Assignment, 12 Oct. 2003 <http://www.giac.org/GSEC.php>.

[20] PGP History. 2003. PGP Corporation. 18 Jan. 2004 <http://www.pgp.com/company/pgphistory.html>.

[21] Zimmermann, Phillip. PGP User's Guide Volume I: Essential Topics. 1994. 22 Nov. 2003 <ftp://ftp.pgpi.org/pub/pgp/2.x/doc/pgpdoc1.txt>.

[22] Edward, Skerke. A Practical Approach to Message Encryption. GSEC Practical Assignment, 02 Feb. 2004 <http://www.sans.org/rr/papers/index.php?id=583>.

[23] Garfinkel, Simson. Pretty Good Politics. 12 Dec. 2003 <http://hotwired.lycos.com/packet/garfinkel/97/18/index2a.html>.

[24] Avnet, Guy. Pretty Good Privacy: 'To PGP or not to PGP?'. Student Lecture. 18 Jan. 2004 <http://www.cs.huji.ac.il/~sans/students_lectures/PGP.ppt>.

[25] Ellison, C. Schneier, Bruce. Risks of PKI: Secure E-Mail. Jan. 2000. 08 Feb. 2004 <http://www.schneier.com/essay-insiderisks4.html>.

[26] Cole, Eric, et al. SANS Security Essentials with CISSP CBK. 2.1 ed. Vol. 2. U.S.A.: SANS Press, 2003. 882-1010.

[27] Fetissov, Nikolai N. Securing Electronic Mail in a Small Company. GSEC Practical Assignment, 09 Jan. 2004 <http://www.sans.org/rr/papers/index.php?id=1216>.

[28] Ploskina, Brian. Secure E-Mail Gaining. 29 Oct. 2001. EWeek. 28 Nov. 2003 <http://www.eweek.com/print_article/0,3048,a=17263,00.asp>.

[29] Setapa, Sharipah. Securing E-Mail. GSEC Practical Assignment, 18 Nov. 2003 <http://www.sans.org/rr/papers/index.php?id=581>.

[30] Keinan, Guy. S/MIME. Student Lecture. 18 Jan. 2004 <http://www.cs.huji.ac.il/~sans/students_lectures/S_MIME.ppt>.

[31] "S/MIME and OpenPGP." Online Posting. Internet Mail Consortium. 04 Feb. 2004 <http://www.imc.org/smime-pgpmime.html>.

[32] S/MIME Or OpenPGP? How Will You Secure Your E-mail? Worldtalk Corporation, 1998. 13 Jan. 2004 <http://www.worldtalk.com/Standards%20and%20Tech/PGP%20and%20SMIME.pdf>.

[33] Chandrasekaran, Vanessa. "Who's Reading Your Email?" SC Infosec 01 Mar. 2003. 28 Nov. 2003 <http://www.infosecnews.com/>.

[34] Zimmermann, Phillip. Why I Wrote PGP. 1999. 08 Dec. 2003 <http://www.philzimmermann.com/essays-WhyIWrotePGP.shtml>.

[35] Tygar, J.D. Whitten, Alma. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. Carnegie Mellon U, 04 Dec. 2003 <http://www.cs.cmu.edu/~alma/johnny.pdf>.

URL General Research Resources:

[36] URL: http://www.pgpi.com

[37] URL: http://cryptorights.org

[38] URL: http://www.openpgp.com