Global Information Assurance Certification Paper

Achieving E-mail security with PKI

GSEC Practical Assignment Version: 1.4b Option 1

Dany Rahal

7th of March 2003

Abstract

The goal of this paper is to discuss Public key Infrastructures (PKI) and basic cryptography concepts then to explain how these elements can intervene in securing one of the most common means of communication these days: E-mail.

We will also briefly discuss the S/MIME protocol that enables e-mail clients to support cryptographic security services.

Introduction

" Security is a chain; it's only as strong as the weakest link"[1].
If we analyze this saying, we realize how difficult it is to have a secure system: different components ranging from software to hardware, processes, procedures and even the human factor act together to compose this chain. The concepts of Defense in Depth, layering and best-of-breed products, when applied carefully, may be very useful in achieving a secure system. Defense in depth is “the concept of protecting a computer network with a series of defensive mechanisms such that if one mechanism fails, another will already be in place to thwart an attack” [2].
Also, choosing different products from different security vendors can help achieve a good level of security.

Humans also constitute an important link in this chain. Some statistics say that 90% of security threats come from inside the company. Social engineering is an important factor that we have to pay attention to.

With all of this mentioned, does this mean that protection may sometimes become futile?

If we really need to be 100% secure, one can buy a desktop, strip it off from its floppy-drive, disconnect it from the network and work safely behind his screen.

But is this what we want to achieve? The answer is NO!

We want to be part of this global network called the Internet and interact with the 400 million people using it each day...However, the question remains the same: can we identify these people securely enough?

Since e-mail is one of the major means of communication these days that is very subject to compromise, let’s look at the risks and the ways to mitigate these risks by the use of PKI.
However, it is important to mention that this paper does not attempt to provide a comprehensive discussion of PKI with all its elements. A work of that sort is obviously far beyond the scope and intention of this essay. This paper instead discusses some PKI and cryptography basic concepts and how these can be applied to secure e-mail communications.

Understanding e-mail risks

It is clear that e-mail is a very important tool for business (and personal) communications these days but we must understand that email is also a function that is very vulnerable to compromise.
When it comes to e-mails, at least three major security risks exist: snooping, forgery and tampering.

- **Snooping:** Imagine yourself while on vacation sending a postcard to friends or relatives. Anybody who has contact with that postcard can read the message since there is nothing covering it. With emails, the same thing happens so snooping is merely the unauthorized reading of email when sending it unsecured (in clear text).
- **Forgery:** Forged email means that the e-mail received appears that it came from a certain email address (i.e., individual or company) different from the real sender of that e-mail. This is extremely easy to realize in practice.
- **Tampering:** Unlike forgery, the tampered email will appear as though it came from a known source; however, the content within the actual email is different.

If we want to qualify these risks in a more technical manner, we can speak about confidentiality, authentication and integrity.
Now that we know that risks do exist when we communicate through emails, let's discuss how PKI can mitigate these risks.
Since PKI is based on cryptographic algorithms, let us first discuss these cryptography concepts.

Cryptography basics

Cryptography is the science of using mathematics to hide the meaning of messages and other data.
A cryptographic algorithm is a mathematical function used for encryption and decryption. All major cryptographic algorithms use Keys. Basically, a key is a variable or a large prime number used to encrypt and decrypt.

Cryptographic algorithms are mainly classified under two families:

Symmetric key cryptography

In a symmetric key system (also called private key system), encryption and decryption are performed using the same key called symmetric or secret key.

Pros:

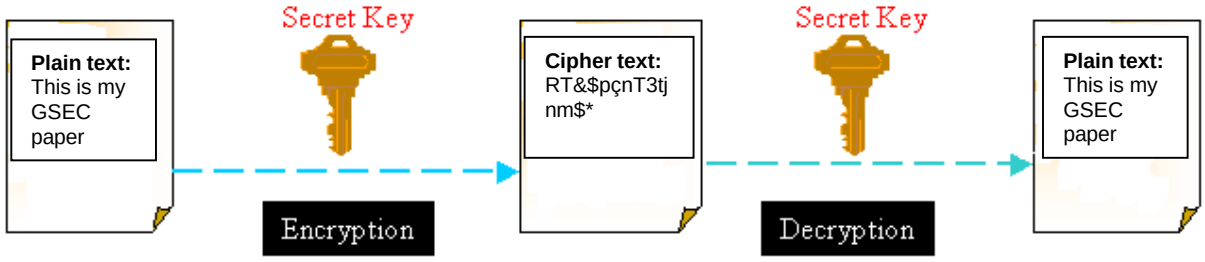
- This method is really fast since the keys used are short (128 bits for example). This would result in good computational efficiency (Example: for DES, the computational speed is around 1-100 million bits/sec)

Cons:

- Easier to crack
- Communicating parties must exchange their keys first, risking exposure
- You have to have a key with each party involved in the communication so you will have to manage a large number of keys in case you communicate with different parties.

DES, triple-DES, IDEA, Blowfish, AES, RC4 and RC5 are examples of symmetric key algorithms.

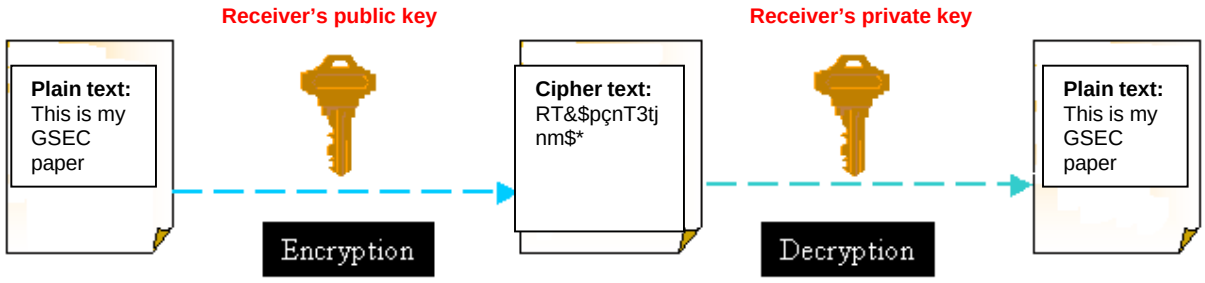
The figure below shows a basic example of the system: the sender will encrypt a message to the receiver who will decrypt it using the same secret key.



Asymmetric key cryptography

In an asymmetric key system (also called public key system), encryption and decryption keys are different (asymmetric key pairs). The encryption key turns plaintext into cipher text. The corresponding decryption key reverses the transformation. Encryption keys are made public (this could be done through publishing these public keys in an LDAP-compliant directory for example). Decryption keys are held securely by the owner (might be stored on the hard disk, on a token or a smart card for more security).

RSA, DSA, Diffie-Hellman and elliptic curve are examples of public key systems.



For example, if someone wants to encrypt a message, he would take the receiver's public key, encrypt the message and send it. Only the receiver can decrypt this message because he's the only one holding his private key securely.

Pros:

- Better key management
- Increased security and convenience
- Provides digital signature that cannot be repudiated

Cons:

- These algorithms are slow since they use long keys (1024 bits for example). This would result in a bad computational efficiency (Example: due to mathematical problems, early versions of the RSA algorithm have a computational speed of about a few thousand bits/sec).

A third cryptography method is worth mentioning here which is the "hybrid" cryptography method. It combines both symmetric (better computational efficiency and speed) and asymmetric methods (better key management and security).

This would make us conclude that "Public-key cryptography is not meant to replace secret-key cryptography, but rather to supplement it, to make it more secure." [3]

In this method (used in SSL for example), we use the destination's public key to encrypt a secret key that is generated locally. Once sent, the involved party at the destination's side will decrypt the secret key by using his private key, which makes this the best way to exchange secret keys. After that, all messages exchanged between the two parties will be encrypted by using the secret key.

PKI uses asymmetric key cryptography.

PKI

What is a PKI

Public-Key Infrastructure (PKI) is the combination of software, hardware, encryption technologies, policies and services that will enable users to secure their communications and transactions over different networks.

It is a key component to the overall email security as it is the solution recommended so that users can:

- Send and receive encrypted messages
- Send and receive digitally signed messages
- Check the authenticity of a sender
- Check the authenticity of a message

- Non repudiation

The S/MIME protocol is usually used by E-mail clients to support these functionalities. We will explain what S/MIME is later in this document.

Role of a Public Key Infrastructure

In order to ensure secure communications and transactions over unsecured networks, the use of PKI is recommended. This will provide protection of the information assets in several essential ways, which is very important for communications:

- **Authenticate** all the parties involved in communications or transactions meaning that every individual concerned must prove he is the one he claims to be.
  In order to do this, digital certificates issued by the Certification Authority are used to allow different entities holding these certificates (users, organizations, servers…) to confidently validate the identity of each party in the transaction through the use of digital signatures.
  Later in this document, we will explain what a digital certificate and a digital signature are and how we ensure authentication by using these two elements.

- **Verify Integrity**: This means that all the data transmitted must remain unchanged during the whole transaction. Here also digital certificates ensure that the message or document the certificate "signs" has not been changed or corrupted in transit online through the use of hashing techniques.

- **Ensure Privacy**: The data transmitted can be hidden from everyone that is not concerned. This can be done through encryption. No E-mail messages will be sent in clear.

- **Authorize access**: PKI digital certificates replace easily guessed and frequently lost user IDs and passwords to streamline Intranet login security. The certificate can be stored on a smart card for example and the user can access the authorized physical or logical resources.

- **Authorize transactions**: With PKI solutions, access privileges to every online resource can be controlled.

- **Support for Non Repudiation**: A given party can never deny any data he has sent. Digital certificates validate their users' identities, making it nearly impossible to later repudiate a digitally "signed" transaction, mail or document. We won't discuss in this essay the legal impact of non repudiation.

The Public Key Infrastructure assumes the use of public key cryptography, which is based on the use of two keys: a public one (published in a Directory; LDAP compliant preferably) and a private one (only known by its holder). We will go through some cryptography basics used for email encryption and signature later in this essay.
As for the roles of a PKI, they are the following:

- Management of public keys on a large scale
- Registration of users(…and other entities) and issuance of certificates
- Storage and distribution of keys
- Revocation and status verification of certificates
- Backup and recovery

Components of a PKI

Several components act together to form a PKI.

- Entities
  - End Users (Key Holders)
  - Registration Authority (RA) and Local Registration Authorities (LRAs) if we are working on a large scale PKI
  - Certificate Authority (CA): Certificate issuer and CRL (certificate revocation list) issuer
  - Directory (Usually LDAP compliant)
  - Validation service

- Hierarchy

  Several levels of certification (depending on a common root CA with possible cross certifications).

- Tools
  - Management and communication protocols CMP, PKCS, CRMF, OCSP, SCVP, and LDAP...
  - Many "standards", some unfinished, unstable or not implemented yet
  - Cryptographic APIs

- Rules
  - Certificate Policies (CPs): Policy under which the certificate was issued and purposes for which the certificate may be used by end-users (applications).
  - Contracts and other legal documents (warranties and limitations on liability).
  - Certification Practice Statement (CPS): statement of the practices, which a certification authority employs in issuing certificates now generally extended to the whole infrastructure.

Defining the CA practice statements, policies and procedures is outside the scope of this document and it is important to mention that defining this is one of the most complicated components of a PKI.

Definition of a certificate

A certificate is a data structure, which associates a name to a public key by means of a signature (i.e. it carries its own authentication). A certificate is tagged as suitable for one or more uses (encryption, signature, key exchange, and certificate generation) and contains only one key, which can be RSA, DES…
The current standard for certificates is X.509 v3.
A certificate usually contains the holder's information (his distinguished name), a serial number, a validity period, the name of the revocation list, the holder's public key and information about the CA.

Uses of a PKI

We already mentioned the utility of a PKI; practically the most common applications are the following:

- Secure e-mail with S/MIME
- Secure Web-based applications with HTTPS (Specially with E-commerce applications)
- IPsec VPNs
- Global authentication and signature mean
- Desktop and file server encryption

In the next section, we will focus on how can we use PKI and S/MIME to secure e-mails.

How do we achieve email security with PKI?

Two important key words that we must stop at when discussing email security and PKI: digital ID and digital signature. Although both are necessary in sending secured email, they perform different functions.
A digital ID is actually a pair of keys that is used to sign/verify, encrypt and decrypt messages. One of these keys is available to everyone and is typically known as a public key. The other key is called a private key, since it should not be made available to outside sources. In order to use these two keys and have secured email, there must be a Certification Authority (CA) in place to verify the digital ID.
Usually and for most of the PKI products, each user has 2 pairs of keys:

- 1 pair for signing/verification. The verification key is made public.
- 1 pair for encryption/decryption. The encryption key is made public.

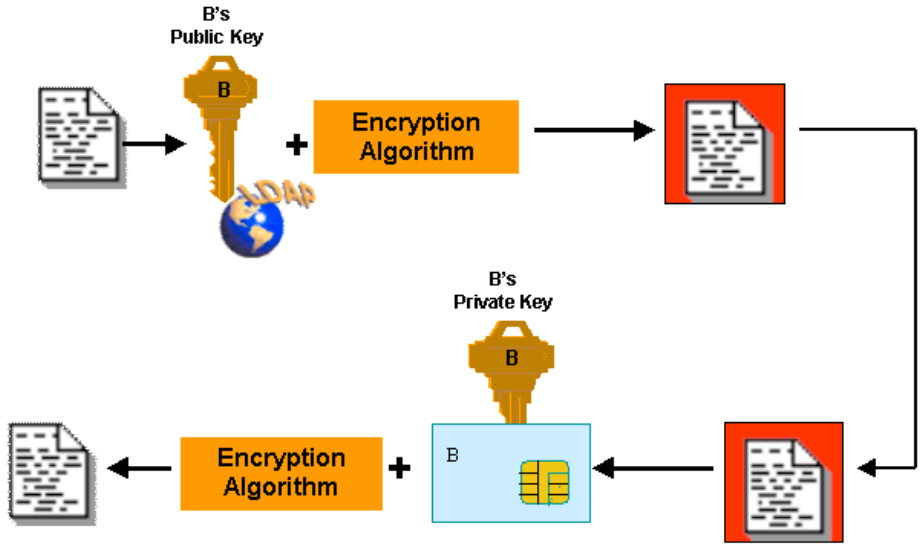
A digital signature is actually the end result of an electronic signing process. With the right software, individuals can place digital signatures on documents and files. A digital signature is similar to a stamp that is unique to each private key holder. A digital signature is the result of computing a message digest (also known as a hash function) via special algorithms and the private key. The result of this process is a digital signature (also known as a digital fingerprint).
Digital IDs are typically stored on computer discs (within your browser and email software) or smart cards.

To achieve email security we must then encrypt and sign mails:

- **Mail encryption (to ensure confidentiality and access control)**

Let's say A wants to encrypt a mail and send it to B. A would write down his mail and click on the 'encrypt' icon in the mail client. Every other operation is transparent to the user.
Practically, what is happening is that the mail client would search for B's public encryption key (in the directory for example or in A's address book); so A will use B's public encryption key to encrypt the mail.
Only B can decrypt this mail because he only holds his private decryption key (on his hard disk or smart card).

Most of e-mail clients have plugins enabling them to access LDAP and fetch the destination's public key.



In case B isn't PKI deployed (i.e. doesn't have a key pair), encrypting emails is impossible. A warning message will be displayed and the mail has to be sent in clear.

- **Mail signing (to ensure authentication, integrity and non repudiation)**

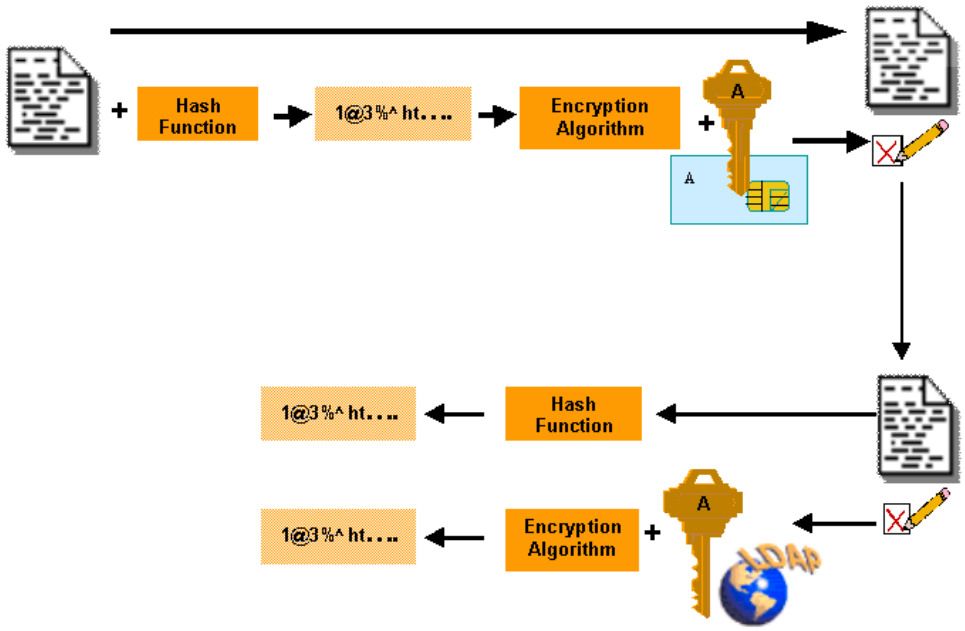
If A wants to sign a mail for B, he will have to hash the mail then encrypt the hash with his private signing key to obtain the digital signature (once again, all these operations are transparent to the user, all he has to do is to click on the 'sign' icon).
A hash algorithm reduces variable-length input to fixed-length (128 or 160bits, also called a message digest or fingerprint) output and it has the following characteristics:

- Can't deduce input from output
- Can't generate a given output
- Can't find two inputs which produce the same output

Hash algorithms are used to:

- Produce fixed-length fingerprint of arbitrary-length data
- Produce data checksums to enable detection of modifications
- Distill passwords down to fixed-length encryption keys

The most common hashing algorithms are SHA-1, MD4 and MD5 (MD4 is weak).
Once the digital signature is obtained, it will be attached to the message and sent in clear or encrypted with the destination's public encryption key along with the message.
B will use A's public verification key (published in LDAP or attached to the message itself in a certificate) to decrypt the signature and to get the hash.
He would hash the message again (if it is encrypted, he will decrypt it first with his private decryption key), compare the two hashes and if they are equal, this means that A has been authenticated and that no one has modified the e-mail message.



For B (the recipient), the signature is a guarantee that:

- There is no spoofing: **authenticity** of the sender.
- There is no tampering: **Integrity** of the message data.
- A can't deny sending the message: **non-repudiation**.

S/MIME

What is S/MIME

S/MIME is an extension of the MIME (Multi-Purpose Internet Mail Extensions) protocol. MIME itself is an extension of the original Internet e-mail protocol (SMTP) that lets people exchange different kinds of data files on the Internet: audio, video, images, application programs, and other kinds, as well as the ASCII originally handled in SMTP.

Following a proposition from Nathan Borenstein of Bellcore in 1991, the IETF accepted to extend SMTP so that Internet (but mainly Web) client and server could recognize and handle other kinds of data than ASCII text. As a result, new file types were added to "mail" as a supported Internet Protocol file type.
What happens in practice is that a MIME header is inserted by servers at the beginning of any Web transmission. This header is used by clients to select an appropriate "player" application for the type of data the header indicates. Some of these players are built into the Web client or browser (for example, all commercial browsers come with GIF and JPEG image players as well as the ability to handle HTML files); other players may need to be downloaded.

Now that we know what MIME is, let's discuss S/MIME.

S/MIME stands for Secure/Multipurpose Internet Mail Extensions and is a secure method of sending e-mail that uses the RSA encryption system provides: it is a consistent way and a data encapsulation format to send and receive secure MIME data. S/MIME provides the following cryptographic security services for electronic distributed messaging applications:

- Authentication
- Message integrity
- Non-repudiation of origin (using digital signatures)
- Privacy and data security (using encryption).

The main advantages of S/MIME are the following:

- Can be used by traditional e-mail clients to add/interpret cryptographic security services to mail that is sent/received.
- S/MIME is not restricted to mail; it can be used with any transport mechanism that transports MIME data, such as HTTP.
- Further, S/MIME can be used in automated message transfer agents that use cryptographic security services that do not require any human intervention, such as the signing of software-generated documents and the encryption of FAX messages sent over the Internet.
- S/MIME is included in the latest versions of the Web browsers from Microsoft and Netscape and has also been endorsed by other vendors that make messaging products (see examples in the next paragraph).

- S/MIME follows the syntax provided in the Public-Key Cryptography Standard format #7 (PKCS #7) entitled Cryptographic Message Syntax Standard, which is a standard that describes general syntax for data that may have cryptography applied to it, such as digital signatures and digital envelopes. This will enable high interoperability.

To conclude, MIME itself, described in the IETF standard RFC 1521, spells out how an electronic message will be organized. S/MIME describes how encryption information and a digital certificate can be included as part of the message body.

Examples of S/MIME compliant email software

The following e-mail clients support S/MIME:

- Microsoft Outlook Express (Windows OS version only)
- Microsoft Outlook 98
- Netscape Messenger version 4.x and 7.x on all operating systems
- Mozilla versions after 0.9.7
- Eudora
- Deming
- Frontier
- Premail
- Opensoft
- Connectsoft
- Lotus Notes

Conclusion

Securing sensitive communications via e-mail is a critical issue for many organizations today.
More and more, emails are becoming a critical means of communication that is very used but at the same time very subject to compromise.

This paper focused on the uses of PKI to ensure secure e-mail exchanges. Basic cryptography concepts and the S/MIME protocol were also presented briefly.
PKI remains a very complex and costly solution and many aspects should be taken into account (deployment, management, trust, cost...) before choosing such solutions.

Glossary

AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certification Authority
CMP	Certificate Management Protocol
CP	Certificate Policies
CPS	Certification Practice Statement
CRL	Certificate Revocation List
CRMF	Certificate Request Message Format
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
GIF	Graphics Interchange Format
HTML	HyperText Mark-up Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IDEA	International Data Encryption Algorithm
IETF	Internet Engineering Task Force
IPsec	Internet Protocol Security
JPEG	Joint Photographic Expert Group
LDAP	Lightweight Directory Access Protocol
LRA	Local Registration Authority
MD4 (5)	Message Digest
MIME	Multipurpose Internet Mail Extensions
OCSP	Online Certificate Status Protocol
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
RA	Registration Authority

RC4 (5)	Rivest Code
RFC	Request For Comment
RSA	Rivest-Shamir-Adleman
S/MIME	Secure/Multipurpose Internet Mail Extensions
SCVP	Simple Certificate Validation Protocol
SHA	Secure Hash Algorithm
SMTP	Simple Mail Transport Protocol
SSL	Secure Sockets Layer
VPN	Virtual Private Network

Bibliography

1. Ellison, C. and Schneier, B. " Ten Risks of PKI: What You're Not Being Told About Public Key Infrastructure ". Computer Security Journal, v 16, n 1, 2000, pp. 1-7.
URL: http://www.counterpane.com/pki-risks.html (Jan 26, 2003)

2. McGuiness, Todd. "Defense In Depth". November 11, 2001.
URL: http://www.sans.org/rr/securitybasics/defense.php (Feb 2, 2003)

3. “Domain 4 Cryptography NtWaK0 CISSP Study”. URL:http://www.safehack.com/Textware/cissp/Crypto/Domain4_Crypto-197.htm (Feb 25, 2003)

4. Schneier, Bruce. Applied Cryptography Second Edition: protocols, algorithms, and source code in C. John Wiley & Sons, Inc. 1996.

5. Kiran, Shashi; Lareau, Patricia; Lloyd, Steve.“PKI Basics - A Technical Perspective”. November 2002. URL:http://www.pkiforum.org/pdfs/PKI_Basics-A_technical_perspective.pdf (Feb 2, 2003)

6. "S/MIME Version 2 Message Specification". IETF homepage. March 1998. URL:http://www.ietf.org/rfc/rfc2311.txt (Feb 24, 2003)

7. “S/MIME Frequently Asked Questions”. RSA Security homepage. URL:http://www.rsasecurity.com/standards/smime/faq.html (Feb 24, 2003)

8. Kelm, Stefan. “The PKI page”. February 11th, 2003. URL:http://www.pki-page.org/ (Feb 25, 2003)

9. Dartmouth College PKI Lab. “Using S/MIME e-mail”. 2 Feb 2003. URL:http://www.dartmouth.edu/~pkilab/pages/Using_SMIME_e-mail.html (Mar 3, 2003)