IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 12, December 2015

Email Privacy with Encryption Standards

M. Kundalakesi[1], V. Magesh Babu[2], R. Naveen Chakarvarthy[3]

Assistant Professor, Bachelor of Computer Application, Department of Computer Application &

Master of Software Systems, Sri Krishna Arts and Science College, Coimbatore, Tamil Nadu, India[1,2,3]

Abstract: Email privacy is the broad topic deals with problem of unauthorized access and inspection faced in electronic mail. This unauthorized access may occur while an email is in transit, as well as when it is stored on email servers or on a user computer. Mail servers are often the most targeted and attacked hosts on an organization's network, second only to Web servers. The problem is that e-mail has to go through numerous untrusted computers before reaching its destination, and there is no way to tell that it would be accessed by an authorized entity. There are certain technological workarounds that make unauthorized access to email hard, if not impossible. However, since email messages frequently cross nation boundaries, and different countries have different rules and regulations' governing who can access an email, email privacy is a complicated issue. This paper may be used in enhancing security on existing and future email systems, in an effort to reduce the number and frequency of email related security incidents.

Keywords: Email, Encryption, Public Key, Pretty Good Privacy (PGP), S/MIME, Security.

I. INTRODUCTION

Electronic mail (email) is perhaps the most popularly used system for exchanging information over the Internet (or any other computer network). At the most basic level, the email process can be divided into two principal components:

(1) Mail servers, which are applications that deliver, forward, and store mail,
(2) Clients which interface with users and allow users to read, compose, send, and store email messages.

After Web servers, mail servers are often the most targeted and attacked hosts on an organization's network. This is because the computing and networking technology that underpins email is ubiquitous and allows attackers to exploit such systems to a somewhat greater degree. These circumstances result in the need to secure mail servers and mail clients and the network infrastructure that supports them.

II. BACKGROUND

Before one can understand the concepts of email security, it is necessary to fully understand how email messages are composed, delivered, and stored. For most email users, once a message is composed and sent, it leaves their system and magically appears in the intended recipient's inbox. This may seem to be the case, however, the handling and delivery of an email message can be as complex as that involving physical mail – with processing and sorting occurring at several intermediary locations before arriving at the final destination.

Once the mail server is processing the message, several events occur: recipient server identification, connection establishment, and message transmission. Using Domain Name Services (DNS), the sender's mail server determines the mail server(s) for the recipient(s). Then, the server opens up connections to recipient mail server(s) and sends the message using the same process that the client used to supply the message initially.

At this point, one of two events could occur.

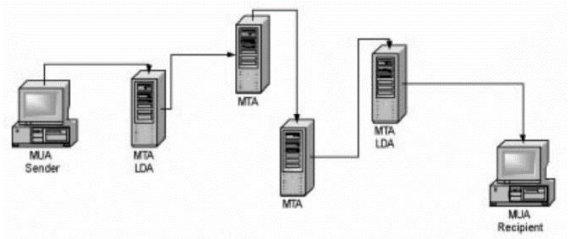

Fig 1: Example of Message Flow

If the sender's mail server is the same location as the recipient's mailbox, the message is delivered using a local delivery agent (LDA). If the sender's mail server is not at the location of the recipient's mailbox, the send process is repeated from one MTA to another until the message reaches the recipient's mailbox.

III. EMAIL-RELATED ENCRYPTION ALGORITHMS

The two primary mechanisms for securing email content end-to-end are Pretty Good Privacy (PGP) and Secure Multipurpose Internet Mail Extensions (S/MIME). Both are based, in part, on the concept of public key cryptography, which involves a user having a pair of related keys: a public key that anyone can hold, and a private key that is held exclusively by its owner. The recipient's public key is used for sending him encrypted information that can be decrypted only with the private key. The sender's private key is used for sending digitally signed information whose authenticity can be verified by anyone holding the corresponding public key. Digital signature techniques rely on the creation of a digest or fingerprint of the information (i.e., the message being sent) using a cryptographic one-way hash, which for efficiency is signed instead of the entire message.

Because of the computationally intensive operations involved in public key cryptography, more efficient, symmetric key cryptography is also used in securing email. Symmetric key cryptography requires a single key to be shared between communicating parties – in the case

of email, the sender and recipient of an email message. The process typically requires the sender to generate a random key and encrypt the message with it using a symmetric key encryption algorithm. Then, the sender encrypts the symmetric key, using the recipient's public key with a corresponding public key encryption algorithm, and sends both the encrypted message and encrypted symmetric key to the recipient. Because only the intended message recipient holds the corresponding private key that is needed to recover the symmetric key, no other party can decrypt the message and read it.

Two of the most prevalent email encryption mechanisms used today, though many mechanisms have been proposed since the invention of email is PGP and S/MIME.

3.1 CHOOSING AN APPROPRIATE ENCRYPTION ALGORITHM

Choosing an appropriate encryption algorithm depends on several factors that will vary with each organization. Although at first glance it might appear that the strongest encryption available should always be used, that is not always true. The higher the level of the encryption the greater impact it will on the mail client resources and communications speed (encryption can increase the size of an email considerably).

In addition, a number of countries still maintain restrictions on the export, import, and/or use of encryption. In addition, patents and licensing issues may impact which encryption schemes can be used in a particular country. Finally, the choice of email encryption standard (PGP, S/MIME, etc.) may limit the choice of encryption algorithms. Fortunately, for federal organizations, the choice is simple and clear – choose either 3DES or AES.
Overall, common factors that can influence the choice of an encryption algorithm include the following items:

A. Required security

- Value of the data (to either the organization and/or other entities – the more valuable the data, the stronger the required encryption).
- Time value of data (if data are valuable but for only a short time period (e.g., days as opposed to years), then a weaker encryption algorithm can be used – an example would be passwords that are changed daily basis because the encryption needs to protect the password for only a 24-hour period).
- Threat to data (the higher the threat level, the stronger the required encryption).
- Other protective measures (if other protective measures are in place they may reduce the need for stronger encryption – an example would be using protected methods of communications such as dedicated circuits instead of the public Internet)

B. Required performance (higher performance requirements may necessitate weaker encryption, but normally a consideration with email)

C. System resources (less resources such as processor speed and memory size may necessitate weaker encryption, but are not typically a factor in email)

D. Import, export, or usage restrictions

E. Encryption schemes supported by mail client applications and operating systems.

3.2 PRETTY GOOD PRIVACY (PGP)

PGP was first released in June 1991. Originally freeware, both free and commercial versions of PGP have become available. The current commercial version of PGP supports a number of cryptographic algorithms recommended by NIST and the Federal Government, including the following:

1. Data Encryption Standard (DES) in triple DES mode (3DES)for data encryption
2. Advanced Encryption Standard (AES) for data encryption
3. Digital Signature Algorithm (DSA) for digital signatures
4. RSA for digital signatures
5. Secure Hash Algorithm (SHA-1) for hashing.

If an organization chooses PGP, they should apply the following guidelines:

For encryption:

i. 3DES for maximum compatibility
ii. AES with a key size of 128 bits or higher, for maximum security and future compatibility

For digital signatures:

i. DSA, with a minimal key size of 1024 bits or higher
ii. RSA, with a minimal key size of 1024 bits or higher, and for compatibility with users having legacy RSA keys

For hashing:

i. SHA-1, which is the most secure of the available alternatives

Although certain aspects of PGP do use public key cryptography, such as digitally signed message digests, the actual encryption of the message body is performed with a symmetric key algorithm as outlined earlier. The following is a brief description of signing and encrypting a message with PGP (some steps may occur in a different order):

1. PGP creates a random session key
2. Message is encrypted using session key, and a symmetric algorithm (e.g., 3DES, AES)
3. Session key is encrypted using recipient's public key
4. SHA algorithm generates a message digest (hash); and this hash is "signed" with the sender's private key
5. Encrypted session key is attached to message
6. Message is sent to the recipient.

3.3 SECURE MULTIPURPOSE INTERNET MAIL EXTENSIONS (S/MIME)

S/MIME, which was originally proposed in 1995 by RSA Data Security, Inc., is based on their proprietary (although widely supported) Public Key Cryptography Standard (PKCS) #7 for data format of encrypted messages. Because S/MIME was originally developed in 1995, the S/MIME standard had to conform to the existing U.S. export controls for cryptography code. This meant that S/MIME implementations were forced to support the insecure 40-bit RC2 algorithm. These controls have since been relaxed. However, because of the standing requirement to support 40-bit RC2, S/MIME is often

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 12, December 2015

criticized as being "cryptographically weak." This is only accurate if a weak algorithm is chosen. S/MIME is compatible with a number of encryption algorithms that allow it to support secure encryption. The actual process by which S/MIME-enabled mail clients send messages is similar to that of PGP.

The most significant feature of S/MIME is its built-in and nearly "automatic" nature. Similar to PGP, no flaws have been discovered in the actual. The current version of S/MIME supports two cryptographic algorithms are DES and 3DES. Algorithm performance is rarely an issue with S/MIME, since encryption and decryption usually takes place on a desktop. When security is paramount, 3DES is the strongest algorithm currently supported by S/MIME, though AES will soon be incorporated.

IV. PRETTY GOOD PRIVACY (PGP)

PGP is a remarkable phenomenon. Largely the effort of a single person, Phil Zimmermann, PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications. From its beginnings about 15 years ago, PGP has grown explosively and is now very widely used. A number of reasons are sited for such growth:.

It is available free worldwide in versions that run on many different platforms, Windows, UNIX, Mac etc. In addition the commercial version satisfies those who want vendor support.

1. It is based on algorithms that have survived extensive public review and are considered secure. Specifically, the package includes RSA, DSS and Diffie Hellman for public-key encryption; CAST-128, IDEA, and 3DES for symmetric encryption; and SHA-1 for hash coding.
2. It has a wide range of applicability, from corporations that wish to select and enforce a standardized scheme for encrypting files and messages to individuals who wish to communicate securely with others worldwide over the Internet.
3. It was not developed by, nor is it controlled by, any government or standards organization. For those with an instinctive distrust of "the establishment", this makes PGP attractive. In the last few years commercial versions have become available.
4. PGP is now on an Internet standards track (RFC 3156). Nevertheless, PGP still has an aura of an anti-establishment endeavor.

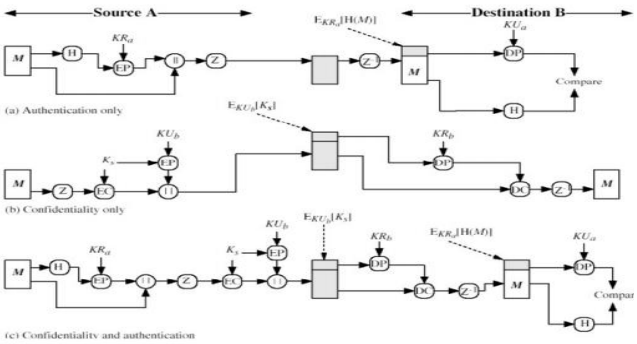

Fig 2: PGP cryptographic functions with confidentiality and authentication.

4.1 CRYPTOGRAPHIC KEYS AND KEY RINGS

PGP makes use of four types of keys:

1. One-time session symmetric keys
2. Public keys
3. Private keys
4. Passphrase based symmetric keys

Three separate requirements can be identified with respect to these keys:

i. A means of generating unpredictable session keys is needed
ii. We would like to allow a user to have multiple public-key/private-key pairs. As a result there is not a one-to-one correspondence between users and their public keys. Thus, some means is needed for identifying particular keys.
iii. Each PGP entity must maintain a file of its own public/private key pairs as well as a file of public keys of correspondents.

4.1.1 Session key generation

Each session key is associated with a single message and is used only for the purpose of encryption and decrypting that message. Recall that message encryption/decryption is done with a symmetric encryption algorithm. Assuming it is a 128 bit key that is required, the random 128 bit numbers are generated using CAST-128. The input to the random number generator consists of as 128-bit key (this is a random number using the keystroke input from the user) and two 64-bit blocks that are treated as plaintext to be encrypted. Using CFB mode two 64-bit cipher text blocks are produced and concatenated to form the 128 bit session key. The algorithm that is used is based on the one specified in ANSI X12.17.

4.1.2 Key Identifiers

As mentioned it is possible to have more than one public/private key pair per user. Each one therefore needs an ID of some kind. The key ID associated with each public key consists of its least significant 64 bits. That is, the key ID of public keyKUa is ($KUa mod\ 2^{64}$).

This is a sufficient length that the probability of duplicate key IDs is very small. A key ID is also used for the PGP digital signature as the sender may use one of a number of private keys to encrypt the message digest and the recipient must know which one was used.

Private Key Ring

Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
T_i	KU_i mod 2^{64}	KU_i	$E_{H(P_i)}[KR_i]$	User i
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

Public Key Ring

Timestamp	Key ID*	Public Key	Owner Trust	User ID*	Key Legitimacy	Signature(s)	Signature Trust(s)
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
T_i	KU_i mod 2^{64}	Ku_i	$trust\_flag_i$	User i	$trust\_flag_i$		
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.

Fig 3: General structure of private and public-key rings.

DOI 10.17148/IARJSET.2015.21230

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 2, Issue 12, December 2015

4.1.3 Key Rings

Key IDs are critical to the operation of PGP. It can be seen that two key IDs are included in any PGP message that provides both confidentiality and authentication. These keys need to be stored and organized in a systematic way for efficient and effective use by all parties. The scheme used in PGP is to provide a pair of data structures at each node, one to store the public/private key pairs owned by that node and one to store the public keys of other users known at this node. These data structures are referred to, respectively as the private-key ring and the public key ring.

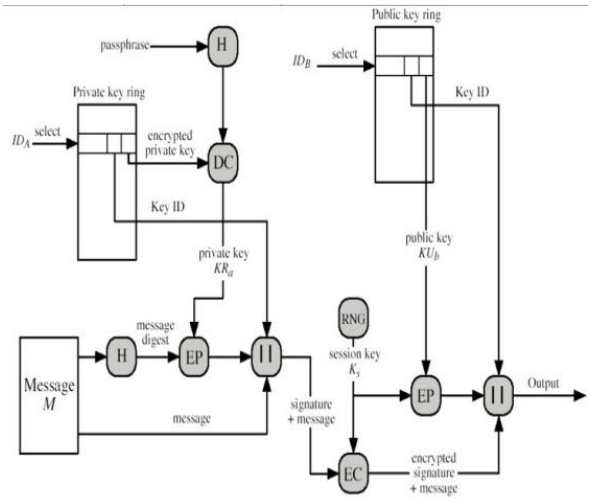

Fig 4: PGP Message generation

V.SECURE MULTIPURPOSE INTERNET MAIL EXTENSIONS (S/MIME)

S/MIME (Secure Multipurpose Internet Mail Extensions) was originally proposed by RSA Data Security, Inc. in 1995, which then led an industry consortium including most of the major email software and Internet browser vendors, such as Microsoft, Netscape and Lotus. Development work is now being coordinated by the IET F S/MIME Working Group.By contrast with PGP's "web" model, with inter locking trust relationships which can be assigned a "weight" or value by the user, S/MIME was designed from the outset as a purely hierarchical model. Keys or certificates are trusted based on the "trust worthiness" of the issuer, which is assumed to be of a higher value than that of the user.

RFC 822 defines a format for text messages that are sent using electronic mail. SMTP/RFC822 scheme limitations:

1. SMTP cannot transmit executable files or other binary files.
2. SMTP cannot transmit text data that includes national language characters because these are represented by 8-bit codes with values of 128 decimal or higher, and SMTP is limited to 7-bit ASCII.
3. SMTP servers may reject mail message over a certain size.
4. SMTP gateways that translate between ASCII to EBCDIC suffer translation problems.
5. Some SMTP implementations do not adhere completely to the SMTP standard defined in RFC 822.

S/MIME FUNCTIONS

S/MIME is based on the Cryptographic Message Syntax(CMS) specified in RFC 2630.

Enveloped data:

This consists of encrypted content of any type and encrypted content encryption keys for one or more users. This function provides privacy and data security.

Signed data:

A digital signature is formed by signing the message digest and then encrypting that with the signer private key. The content and the signature are then encoded using base 64 encoding.

This function provides authenticity, message integrity and non-repudiation of origin.

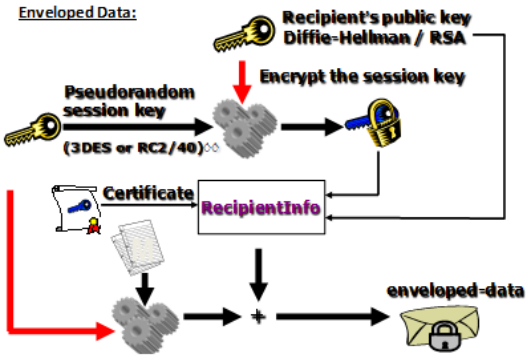

Fig 5: Enveloped Data

VI. CONCLUSION

As the Internet Mail Consortium, which has been involved in the standards process since 1997, says: having two protocols that do the same thing is much worse than having one. It is unclear which standard will prevail –the "Internet community" is fond of PGP because of its age and its impeccable (i.e. non-commercial) pedigree, but the weight of the market is pushing toward S/MIME.As we have seen in a wide variety of markets, the "best" standard is often not the prevailing standard. Only one fact seems clear: the vast majority of Internet e-mail is not routinely encrypted. This seems unlikely to change in the near future.

REFERENCES

1. https://en.wikipedia.org/wiki/Email_privacy
2. https://books.google.co.in/books?id=xCDZAgAAQBAJ&pg=PA304&lpg=PA304&dq=pretty+good+privacy+tables&source=bl&ots=ziM_q4x8bn&sig=ZMXxvuapf6sM_eZPkn4GYKXsXvY&hl=en&sa=X&ved=0ahUKEwjZmt7L7-HJAhVlIKYKHXYCAjIQ6AEILTAE#v=onepage&q=pretty%20good%20privacy%20tables&f=false
3. http://www.pcworld.com/article/254338/how_to_encrypt_your_email.html
4. http://www.math.utah.edu/~beebe/PGP-notes.html
5. http://www.google.co.in/search?ie=UTF-8&oe=UTF-8&sourceid=navclient&gfns=1&q=Email+security&gws_rd=cr&ei=TB1yVvnkKqG8mgX1pKzwDw
6. http://www.google.co.in/search?ie=UTF-8&oe=UTF-8&sourceid=navclient&gfns=1&q=pretty+good+privarcy+tables&gws_rd=cr&ei=CCFyVpv7D4TemAX6p63oAQ