CMSGu2012-06

National Computer Board

Mauritian Computer Emergency Response Team

Enhancing Cyber Security in Mauritius

E-mail Best Practices

CERT-MU

National Computer Board
Mauritius

Version 1.0

August 2012

Issue No. 6

Table of Contents

DISCLAIMER: *This guideline is provided "as is" for informational purposes only. Information in this guideline, including references, is subject to change without notice. The products mentioned herein are the trademarks of their respective owners.*

1.0 Introduction

1.1 Purpose and Scope

The aim of this guideline is to provide users with a secure online experience when dealing with e-mails.

1.2 Audience

The target audience include everyone who uses e-mail at work, at home, at university or at school.

1.3 Document Structure

This document is organised into the following sections:

Section 1 contains the document's content, the targeted audience and the document's structure.

Section 2 gives a background on e-mail and e-mail clients available.

Section 3 talks about e-mail security concerns.

Section 4 explains e-mail signature and encryption.

Section 5 provides a few e-mail security tips.

Sect*ion 6* concludes the document.

Section 7 contains a list of references that have been used in this document.

2.0 Background

E-mail is easy, fast, cheap and practically universal. This is what really makes it a vital tool in daily operations. It also requires a little education and awareness and can be a safe means of communication if used properly. Users should learn what to look for and how to avoid becoming victims and take all necessary precautions to protect themselves.

There are many e-mail clients available for use. It is important to understand features of e-mail clients when making a choice. Clients need to be securely configured and kept current with the latest patches. Below is a list of commonly used clients:

- Microsoft Outlook
- Apple Macintosh Mail
- Eudora
- Microsoft Outlook Express
- Thunderbird/Mozilla/Netscape
- Lotus Notes
- UA Webmail

3.0 E-mail Security Concerns

3.1 Scams and Spam

An increasingly common type of scam using spam is phishing. Phishing attacks use “spoofed” e-mails and fraudulent websites designed to fool recipients into divulging personal financial data such as credit card numbers, account usernames and passwords, social security numbers, etc. E-mails can be forged, including the send and return address, the e-mail body, and other e-mail information.

3.2 E-mail spoofing

E-mail spoofing may occur in different forms, but all have a similar result: a user receives e-mail that appears to have originated from one source when it actually was sent from another. E-mail spoofing is often an attempt to trick the user into releasing sensitive information (such as passwords), or making a damaging statement. Spoofed e-mail can range from harmless pranks to social engineering ploys.

Examples of spoofed e-mail that could affect security include:

- E-mail that appears to come from a known source but contains a virus or other malicious code or instructions.
- E-mail claiming to be from a system administrator requesting users to change their password to a supplied word or phrase and/or threatening to suspend their account unless they comply.
- E-mail claiming to be from a person in authority requesting users to send them a password or other sensitive information.

If someone can obtain the username and password used to access an e-mail account, they can read and send e-mail messages impersonating the user of that account.

It is very easy to construct messages that appear to be from someone other than who they are actually from. Many viruses use this method to propagate themselves. In general, there is no simple way to be sure that the apparent sender of a message actually sent it.

Note that while service providers may occasionally request that users change their password, they usually will not specify what a password should be changed to. Legitimate internet service providers will never ask users to send any personal information via e-mail. If it is suspected that an e-mail is spoofed by someone with malicious intent, contact the internet service provider's support personnel immediately.

3.3 E-mail-borne viruses

Viruses and other types of malicious code are often spread as attachments to e-mail messages. Before opening any attachments, verify the source of the attachment. It is not enough that the mail originated from a recognized address. For example the Melissa virus spread because it originated from a familiar address. Also, malicious code might be distributed in amusing or enticing programs.

Only run programs that are created by a trusted person or company. Forwarding programs of unknown origin to your friends or co-workers simply because they are amusing -- could spread a worm or Trojan horse.

3.4 Hidden file extensions

Windows operating systems contain an option to “Hide file extensions for known file types”. The option is enabled by default; it is recommended users disable this option in order to have file extensions displayed by Windows. Multiple e-mail-borne viruses are known to exploit hidden file extensions. Examples include:

- Downloader (MySis.avi.exe or QuickFlick.mpg.exe)
- VBS/OnTheFly (AnnaKournikova.jpg.vbs)

The files attached to the e-mail messages sent by these viruses may appear to be harmless text (.txt), MPEG (.mpg), AVI (.avi) or other file types when in fact the file is a malicious script or executable (.vbs or .exe, for example).

4.0 Signing and Encrypting E-mail

Organisations often want to protect the confidentiality and integrity of some of their e-mail messages, such as preventing the exposure of personally identifiable information in an e-mail attachment. E-mail messages can be protected by using cryptography in various ways, such as the following:

- Sign an e-mail message to ensure its integrity and confirm the identity of its sender
- Encrypt the body of an e-mail message to ensure its confidentiality
- Encrypt the communications between mail servers to protect the confidentiality of both the message body and message header

The first two methods, **message signing** and **message body encryption**, are often used together. For example, if a message needs to be encrypted to protect its confidentiality, it is usually digitally signed as well, so that the recipient can ensure the integrity of the message and verify the identity of the signer. Messages that are digitally signed are usually not encrypted if the confidentiality of the contents does not need to be protected.

The third cryptography method listed above, **encrypting the transmissions between mail servers**, is typically applicable only when two organizations want to protect e-mails regularly sent between them. For example, the organizations could establish a virtual private network (VPN) to encrypt the communications between their mail servers over the Internet. Unlike methods that can only encrypt a message body, a VPN can encrypt entire messages, including e-mail header information such as senders, recipients, and subjects. In some cases, organizations may need to protect header information. However, a VPN solution alone cannot provide a message signing mechanism, nor can it provide protection for e-mail messages along the entire route from sender to recipient.

4.1 Public Key Cryptography

Most e-mail messages are protected individually by digitally signing and optionally encrypting them. The most widely used standards for signing messages and encrypting message bodies are Open Pretty Good Privacy (OpenPGP) and Secure/Multipurpose Internet Mail Extensions (S/MIME). Both are partially based 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. As public key cryptography is computationally complex, it is used in moderation in e-mail security; symmetric key cryptography, which is much more efficient, is much more heavily used.

4.2 Symmetric Key Cryptography

Symmetric key cryptography requires a single key to be shared between communicating parties, the sender and recipient of an e-mail message. The process involves the sender generating a random key and encrypting the message with it using a symmetric key encryption algorithm. The sender then encrypts the symmetric key with a corresponding public key encryption algorithm using the recipient’s public key, and sends both the encrypted message and encrypted symmetric key together to the recipient. This hybrid process uses public key encryption only to encrypt the symmetric key. Because only the intended message recipient holds the private key that is needed to recover the symmetric key, no other party can decrypt the message and read it. Digital signature techniques rely on the creation of a digest or fingerprint of the information (i.e., the message being sent) using a cryptographic hash, which can be signed more efficiently than the entire message.

4.3 Open PGP

OpenPGP is a protocol for encrypting and signing messages and for creating certificates using public key cryptography. It is based on an earlier protocol, PGP, which was created by Phil Zimmerman and implemented as a product first released in June 1991. The initial PGP protocol was proprietary and used some encryption algorithms with intellectual property restrictions. In 1996, version 5.x of PGP was defined in IETF RFC 1991, PGP Message Exchange Formats. Subsequently, OpenPGP was developed as a new standard protocol based on PGP version 5.x. OpenPGP is defined in RFC 2440, OpenPGP Message Format and RFC 3156, MIME Security with OpenPGP.

Although certain aspects of OpenPGP 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 OpenPGP (some steps may occur in a different order):

- OpenPGP compresses the plaintext, which reduces transmission time and strengthens cryptographic security by obfuscating plaintext patterns commonly searched for during cryptanalysis.

- OpenPGP creates a random session key (in some implementations of OpenPGP, users are required to move their mouse at will within a window to generate random data).
- A digital signature is generated for the message using the sender’s private key, and then added to the message.
- The message and signature are encrypted using the session key and a symmetric algorithm (e.g., 3DES, AES).
- The session key is encrypted using the recipient’s public key and added to the beginning of the encrypted message.
- The encrypted message is sent to the recipient.

The recipient reverses the steps to recover the session key, decrypt the message, and verify the signature. Popular mail clients such as Mozilla Thunderbird, Apple Mail, Eudora, and Microsoft Outlook require the installation of plug-ins to enable the user to send and receive OpenPGP-encrypted messages.

There are also security gateway servers available that can use OpenPGP to encrypt, decrypt, sign, and verify signatures on e-mail messages on behalf of users. If two organizations exchanging e-mails both use compatible security gateway servers, then the use of OpenPGP is essentially transparent to users. If only one organization has such a gateway, it can still be used to protect messages, but it is not a transparent process at all to users at other organizations. If a gateway user sends an e-mail to a recipient at another organization, that recipient will actually receive a notification e-mail from the gateway that explains how to retrieve the protected e-mail, typically through an SSL-encrypted HTTP session. Some gateways can also perform these functions for e-mails sent to lists of users. For example, a single user could send an encrypted and signed e-mail to a mailing list address. The gateway would decrypt the e-mail and re-encrypt it for all the individual recipients of the mailing list. Each recipient can then decrypt the e-mail and verify the original signature.

4.4 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, and the X.509 version 3 standard for digital certificates.18 S/MIME version 2 achieved wide adoption throughout the Internet mail industry. Although it is not a recognized IETF standard, it is specified by informational RFCs 2311, 2312, 2313, 2314, 2315, and 2268.

S/MIME version 3 was developed by the IETF S/MIME Working Group, which now coordinates all development of the S/MIME standard, and adopted as an IETF standard in July 1999. S/MIME version 3 is specified by the following RFCs:

- Cryptographic Message Syntax (RFC 3852)
- S/MIME Version 3 Message Specification (RFC 3851)
- S/MIME Version 3 Certificate Handling (RFC 3850)
- Diffie-Hellman Key Agreement Method (RFC 2631)
- Enhanced Security Services for S/MIME (RFC 2634).

The most significant feature of S/MIME is its built-in and nearly “automatic” nature. Because of heavy industry involvement from manufacturers, S/MIME functionality exists with default installations of common mail clients such as Mozilla and Outlook Express. The actual process by which S/MIME-enabled mail clients send messages is similar to that of OpenPGP. S/MIME version 3.1 supports two symmetric key encryption algorithms recommended by FIPS PUB 140-2: AES, which is recommended but optional for compliant implementations to support, and 3DES, which is mandatory for implementations to support. Organizations using S/MIME to protect e-mails should use AES or 3DES (preferably AES, which is considered a stronger algorithm than 3DES).

As with OpenPGP, there are security gateway servers available that can use S/MIME to encrypt, decrypt, sign, and verify signatures on e-mail messages on behalf of users.

4.5 Key Management

Both OpenPGP and S/MIME use digital certificates to manage keys. A digital certificate identifies the entity (e.g., a user) that was issued the certificate, the public key of the entity’s public key pair, and other information, such as the date of expiration, signed by some trusted party. However, differences exist in the key management models used by OpenPGP and S/MIME to establish trust using digital certificates. The default and traditional model that OpenPGP uses for key management, is referred to as the “web of trust,” which has no central key issuing or approving authority. The web of trust relies on the personal decisions of users for management and control. For example, if Alice trusts Bob and Carol trusts Alice, then

Carol should trust Bob’s e-mails. While this is suitable for individual users and very small organizations, the overhead of such a system is unworkable in most medium to large organizations. Some organizations deploy key servers that users can access to get others’ keys and store their own keys. Although this does promote scalability, the process is typically controlled mainly by individual users, and organizations are often not comfortable trusting key servers to provide sufficient assurance of user identity.

Conversely, S/MIME works on a classical, more hierarchical arrangement of authorities that the organization chooses to trust. Typically, there is a master registration and approving authority, referred to as a root Certificate Authority (CA) that issues a public key certificate for itself and any subordinate CAs it sanctions. Subordinate CAs normally issue certificates to users and also to any other subordinate CAs that they in turn sanction, forming a hierarchy. Such a public key infrastructure can be used to establish a chain of trust between any two users holding valid certificates issued under it. By default, S/MIME-enabled mail clients depend on the trust of their immediate master CA when processing S/MIME transactions. This authority can be either a third-party CA or a CA that is controlled by the organization issuing the certificates.

Having an organization exchange OpenPGP or S/MIME-protected e-mails with other organizations is usually extremely complicated, especially when attempting to maintain transparency for the users. The biggest challenges are key exchange and establishing trust relationships between the organizations. Organizations can connect their PKIs or use a mutually trusted third-party PKI, but in either case there are often technical and legal or regulatory challenges. Also, support for OpenPGP and S/MIME varies considerably depending on the mail client in use.

Third-party services are available that allow organizations to exchange encrypted e-mail without having to establish trust relationships or worry about mail application compatibility. However, the use of such services necessitates placing sensitive messages on third-party servers, which itself can be a security concern. The use of mail encryption gateways between two organizations typically has lesser key management concerns because the keys are maintained on the gateways and a trust relationship already exists between the gateways. Work is currently underway on a possible method of reducing key management concerns for e-mail signing and encryption. Identity-based encryption (IBE) is a form of public key encryption that allows any string to be used as a public key. By using e-mail addresses as public keys, IBE could simplify key management, making it much easier for senders to protect the e-mails that they send. However, there are serious barriers to adoption of IBE, including no open standards for IBE and no FIPS-approved IBE products. Informational Internet-Drafts have been started that propose how IBE could be performed using S/MIME.

4.6 Issues with E-mail Encryption

Although encrypting e-mail provides additional security, it does involve a cost, so organizations should carefully consider the issues associated with encrypting e-mail messages:

- Scanning for viruses and other malware and filtering e-mail content at the firewall and mail server is made significantly more complicated by encryption. If the firewall or mail server does not have a method for decrypting the e-mail, it cannot read and act upon the contents. Some malware scanners can decrypt e-mails if the scanner is a recipient of the e-mails or if the sender specifically encrypts the e-mails for the scanner, but such solutions are technically complex and often hard to enforce. Also, giving the malware scanner the ability to decrypt many or all e-mails could have serious consequences if the malware scanner host is itself infected or otherwise compromised. If having the malware scanner decrypt e-mails is not feasible, scanning might have to be performed on the hosts of the mail clients that perform decryption.

- Encryption and decryption require processor time. Organizations might need to upgrade or replace equipment that is not capable of supporting the load of encryption and decryption.

- Organization-wide use of encryption can require significant ongoing administrative overhead. Examples of this include key distribution, key recovery, and revocation of encryption keys.

- E-mail encryption can complicate the review of e-mail messages by law enforcement and other investigative parties.

- Encrypted e-mails sent to or received from other organizations may be insufficiently protected if those organizations do not support the use of strong encryption algorithms and key sizes. Organizations should ensure that their users' mail applications notify them when they receive a weakly encrypted message or when they are attempting to send an encrypted message to a recipient that only supports weak encryption methods. Users can then contact the relevant party to notify them of the problem and request that they either use a stronger encryption algorithm or transfer the information that needs protected through a mechanism other than e-mail.

5.0 E-mail Security Tips

- Treat all e-mails and attachments with suspicion.
- Be cautious when using a link in an e-mail to get to a web page. If you must visit the website, type the URL directly into your browser's address bar.
- Never send personal or financial information to anyone via e-mail.
- Regularly scrutinize bank, credit and debit card statements to ensure that all transactions are legitimate. If anything looks suspicious, contact financial institutions or card issuers.
- Make sure that software (antivirus, antispyware and firewall) have all security updates installed.
- Ensure that you have adequate safeguards against fraud before accessing financial information online – Contact your financial institution for information.
- Beware of all attachments. Only open those that you are expecting from a trusted source or have confirmed with the sender.
    - **The Know test:** Is the e-mail from someone that you know?
    - **The Received test:** Have you received e-mail from this sender before?
    - **The Expect test**: Were you expecting e-mail with an attachment from this sender?
    - **The Sense test:** Does e-mail from the sender with the contents as described in the Subject line and the name of the attachment(s) make sense?
    - **The Virus test:** Does this e-mail contain a virus?
- Beware of software updates sent by e-mail. Corporations and financial institutions will never send any security patches, updates, distributions or other executable files by e-mail – they will send notifications instead.
- Beware of URL's within an e-mail message. Best practice is to open the browser and type in the URL rather than click it from within the e-mail message.
- Use e-mail filters to move identified suspected spam mail to a spam folder for later verification.
- Keep Operating System and application software updated with latest security updates/patches in the computer system used for e-mail to prevent the exploitation of the weakness in the system.
- Scan an e-mail attachment before opening/downloading to minimize the risk of downloading malware (e.g., virus).

- Use encryption for sending and receiving confidential e-mail to ensure that message can only be read by the intended recipients.
- Keep strong password with minimum of eight characters, comprising a combination of alphabets (both upper and lowercase), numbers and special characters.
- Do not keep your computer unattended to avoid misuse

6.0 Conclusion

It is no doubt that e-mail provides many benefits such as it is easy, cheap and fast. People make use of e-mail for various purposes, for example to send personal messages, documents related to work or even images. However, if not used cautiously, users can be exposed to many threats. That is why it is imperative to educate users about the potential dangers of using e-mail in the improper way and how they can overcome these.

7.0 References

- Safety Tips for using E-mail, CERT-In
- E-Mail Client and Usage, The University of Arizona
- Paul McFedries' Web Home: **http://www.mcfedries.com**
- Guidelines on Electronic Mail Security, NIST