Enterprise Email Security

by Paul Holman
Black Hat USA 2002

meta secura
pablos@metasecura.com

Introduction

Email has become the most fundamental communication mechanism for the modern enterprise. As such, email is the fulcrum against which all other security measures are leveraged. This presentation is based on our work architecting high-security email services for law-enforcement agencies, law-firms and human rights NGOs. These organizations have typically had difficulties adopting solutions, primarily due to poor usability. Recognizing this, our approach aims to keep email easy while making it vastly more secure for the whole enterprise.

Overview

This presentation is relevant for everyone interested in securing their organization's email. Many lessons have been learned in our experiences and we're eager to share them.

We'll classify the various approaches to securing email, discuss their comparative merits and consider their place in history and/or the future. The severe challenges of PKI trust models and key management will be made clear. Real-world solutions to countering spam and viruses are covered. We'll also predict how good email security could be in a perfect world versus our own.

The techniques we present can be applied regardless of the email infrastructure currently in use. Listeners will obtain a thorough understanding of practical measure's they can take to secure their organization's email infrastructure. We present a comprehensive architecture for email security that is in use by our clients, and discuss how other organizations can utilize this approach themselves.

Speaker

Paul Holman has been working to secure email since before most people heard of it. He is a member of The Shmoo Group of security, crypto & privacy professionals, supporting numerous open-source development projects. He started the Shmoo Seizable Mail Server project two years ago and now heads up Metasecura, a boutique security consulting firm that builds and manages these servers for enlightened customers.

Feel free to contact pablos@metasecura.com anytime.

Threats Made Real

Email has become the most fundamental communication mechanism for the modern enterprise. As such, email is the fulcrum against which all other security measures are leveraged. If email is not secured, it is unreasonable to expect most other aspects of an organization to to be secure.

Loss of privacy

The threat of email compromise runs deep in modern corporations. Consider the potential damage to your organization posed by routine unsecured email:

- Management and operational discussions
- Hiring/firing decisions
- Sales and other financial data
- Contract negotiations & proposals
- M&A discussions
- Proprietary innovations and trade secrets

According to the "Computer Crime and Security Survey," conducted by the Computer Security Institute and the FBI, 85% of companies and government agencies polled had experienced security breaches within the previous 12 months. Also, 64 percent acknowledged financial losses due to computer breaches, the most serious financial losses occurring through the theft of proprietary information.

Corporate Espionage

How much would you pay to read your competitor's email? Ethics aside, how much would your boss pay to read your competitor's email? This is probably about what they might be willing to pay to read yours.

Discovery Dilemma

Corporations are increasing exposed to litigation that hinges on email. A typical lawsuit will expose email archives, in discovery, which are scoured by attorneys. Often, this process reveals extensive data about a corporation that results in further litigation. This may be the greatest, measurable threat to large organizations posed by email.

American Home Products had to produce electronic archives that included 33 million emails. They paid $3.7 billion in legal damages as a result of information discovered.
http://cyber.law.harvard.edu/digitaldiscovery/library/mundy.html

When 3Com acquired U.S. Robotics, a suit was filed by shareholders claiming that top executives had dumped stocks and hidden large losses in the newly acquired subsidiary. Following a subpoena of its email archives, 3Com elected to pay $260 million to settle the case.
http://www.sfgate.com/cgi-bin/article.cgi?file=/chronicle/archive/2000/11/04/BU86960.DTL

History of Email Security

A number of efforts to secure email have been made over the history of the internet. Most of these initiatives have failed to make much progress.

Old Standards

Privacy Enhanced Mail (PEM) was the first real standard organized for internet email security. It has mostly failed to solidify in any way. PEM work eventually influenced PGP and S/MIME, but mostly lacks relevance today.

Services

Services such as Hushmail, Lokmail, MailVault and others attempt to improve email security by creating webmail interfaces with added security features. Most of them approximate Hotmail or Yahoo! Mail with far less advertising. They tend to take advantage of the SSL built into web browsers to secure a connection, and then use some server side encryption based on the login password to secure mail stored there. Most systems of this type began by creating their own encryption protocols, and a few have migrated to OpenPGP. Still, they tend to operate with no actual trust model for the keys and they have no real security value for email sent outside the system. Yahoo! merely uses SSL to secure the password, and then transmits all mail in the clear.

Proprietary Email Security Systems

Some companies are in the business of providing secure email with a software product or service. These all tend to work as “closed” systems, which provide no real trust model or security outside the system.

Zixit has a proprietary email system that attempts to secure mail between organizations who all buy into their system. They offer a patch to work with Microsoft Outlook or Lotus Notes, but no other mailers. Zixit has done a fantastic job of making email security seem easy to the end user, primarily by eliminating key and trust management tasks. Zixit operates a series of services, including timestamping and “certified” email. They claim to be steadily picking up new clients.

Lotus Notes is a very well entrenched corporate email platform that offers some kind of proprietary security

between email clients and the server. In addition, Notes now supports S/MIME.

Omniva (formerly Disappearing Inc.) pioneered work in creating email security systems oriented around helping corporations manage liability of their own email. Their product allows organizations to set per-message key expiration policies to provide forward secrecy. This can help combat the “discovery dilemma” that ensues during litigation. Omniva’s product integrates with Microsoft Outlook, but no other mailer.

The Mythical Encrypting Proxy

Every now-and-then somebody will dream up the idea of building a robotic encrypting proxy. A tool that you’d load on your mail server or mail client that will automatically look at all your mail and take care of key management and all the nasty innards of securing email. This is loosely approximated by some of the services mentioned, but mostly has yet to be built.

While some advantage can be be obtained here in providing compatibility with any mail client, the oversimplification of this approach will leave users subject to a wide range of attacks which existing standards were designed to protect against. Any work on an encrypting proxy should leverage one of the existing standards so as not to fragment the secure email userbase, and ensure a "scalable security" model is available when greater protection is necessary.

Current Standards

Today, two standards exist for end-to-end encryption and authentication between users of email.

S/MIME is a standard extending the MIME email attachment protocol which offers encryption and authentication using the X.509 model shared by SSL. It has been implemented primarily by Netscape, and Microsoft, though interoperability issues have plagued it from the start. S/MIME inherits the hierarchical trust model of X.509, and most of the time, this maps well to enterprise deployments. Sadly, the overhead of bootstrapping a usable and secure S/MIME deployment is far greater than most organizations can bear. It requires command of the full gamut of tools associated with PKI, including operation of a Certificate Authority, which is no small undertaking. As S/MIME implementations have generally been built by mail client developers, they've been tightly integrated and tend to operate in a more opportunistic manner, in some cases this makes them more usable than other systems

The OpenPGP standard has evolved out of Phil Zimmerman's original Pretty Good Privacy software. There are now multiple implementations which support the standard, available both commercially and under open source licenses. OpenPGP is by far the most widely used personal encryption system for email and probably the most relevant going forward. There are technical reasons why it is a more sound choice than S/MIME, but the main advantage is flexibility in the trust model. X.509 generally assumes a hierarchy, putting some trusted third party at the top of a pyramid which all trust calculations must resolve to. This can be seen in the basic key/certificate paradigm used by SSL or other X.509 derivatives. OpenPGP pioneered the web-of-trust approach to trust management. This allows everyone to create their own six-degrees-of-separation type trust calculations for keys. It is this grass-roots flexibility that has allowed OpenPGP to succeed thus far. Any two people can begin using OpenPGP on their own, and any three people can create a web-of-trust to establish the authenticity of keys.

Standards are Crucial

No email security system has succeeded in securing the bulk of the world's private email. None has come even close. Email security ultimately needs to be pervasive. Encryption should be a transparent part of the process of using email. Authentication should be made as unintrusive as possible for these exchanges. Invariably, as with every existing system, compromises need to be made between security and usability. The value of standards in this process cannot be understated. Many attempts to secure email do so at the drastic cost of real security. Usually the first thing to go is trust management for keys. While it is arguable that most trust management schemes are poorly implemented, the fact that they exist and can be taken advantage of when necessary is crucial. All attempts to improve email security should be based on existing standards, maintain interoperability, and respect the "scalable security" model inherent in the standard. Those who ignore this advice will not only fail to provide real security for their users, but will further faction and damage an already frustrated userbase of people who need their email secured.

Risk Management

Security for email, as with other services must be viewed from a risk management perspective. Most organizations have their most valuable data stored within their email infrastructure. Yet, email is often the least secure application in use. Mail is transferred over the internet in the clear, without encryption, using easily forged protocols. It is usually retrieved by sending passwords in the clear over the internet, and then stored on computers all over the place, with no real security measures protecting them, home computers, laptops, unattended machines at an office or internet café. The company mail server itself is almost always directly connected to the internet, and subject to a wide range of network attacks. Often managed by entry level system administrators with "it-ain't-broke-don't-fix-it" attitudes.

How much would you pay to read your competitor's email? Ethics aside, how much would your boss pay to read your competitor's email? This is probably about what they might be willing to pay to read yours.

To maintain any reasonable level of security, organizations need to assume they will be subject to attacks, determine what threats concern them most, and prepare for the worst. They need to take steps to secure themselves against attacks, and assume their security will be breached at some point. The best approach is to estimate the cost of damages from a typical attack, and use that as a guideline for budgeting security work.

Data Custodianship

Data custodianship refers to the enterprise's need to control email. Beyond the general need for preserving data with backups and ensuring high availability, an organization may wish to ensure that email access occurs only within a tightly controlled environment. Organizations may have a policy of destroying email after some limited time period. This can be a valuable approach to limit liability. Attackers cannot gain access to data that no longer exists.

The Key Management Problem

The best theoretical security for communication involves end-to-end encryption and authentication between users, who have established trust for the keys in use. Sadly, this implies a great deal of overhead. Processes must be established to ensure that the right key maps to the right human. All the products and services associated with PKI are aimed at managing this aspect of security. Despite our technical ability to create sound key management protocols, it is not possible to eliminate the liability of untrained users. PKI as a whole has no competition. To truly secure network communication of any kind, it will be necessary to get key management processes in place.

The Trusted Third Party meets The Man in the Middle

Many attempts have been made to simplify email communication. Usually the first thing that happens is trust management for the keys in use is abandon. By taking this single step, it becomes possible to make nearly transparent email encryption systems. Sadly, they just can't hold up to moderately sophisticated attacks.

Man in the middle attacks are often dismissed by security experts as impractical and unlikely to be performed in the real world. This is an unfortunate concession that has caused most systems on the internet to be left vulnerable to an entire class of attacks. Man in the middle attacks involve intercepting traffic between two parties. In the case of encrypted traffic, this gives the attacker a chance to replace keys being exchanged with forged keys of their own. As mail is passed back and forth between victims, the attacker decrypts and reencrypts messages in real time, keeping a plaintext copy unbeknownst to the users.

These attacks can be performed by anyone with access to any router or server involved in an exchange. They can be performed on SSH, SSL, and all our other secure protocols. The only defense we have is to pay close attention to the trust metrics built into these protocols. Email is particularly susceptible over other protocols due to the expected latency. Users send email with an expectation that it will take a while to get a response. This affords a greater amount of luxury to an attacker.

The "Trusted Third Party" provides assurance that a given key belongs to a given user. In the case of SSL, S/MIME, and other protocols that inherit X.509 conventions, this third party is usually a service provider or entity set up by the enterprise to act as a Certificate Authority. The CA issues certificates to establish that a given key should be trusted. Everyone who participates in a given microcosm trusts that CA to do it's job well. A

compromise of the CA's key would of course destroy trust for all keys in the system.

OpenPGP uses a "Web of Trust" which allows ad-hoc creation of these trusted third parties. A key can be easily certified by one or more parties, allowing a grass-roots trust model to develop. The enterprise can participate in this model by having it's own CA sign OpenPGP keys.

The Promise of Cryptographic Tokens

A big part of the problem with cryptosystems is finding ways for users to effectively keep track of their keys, and prevent the keys themselves from being compromised. Witness the rampant insecurity of credit cards for an example of keys poorly managed. The future promises to deliver a smart card or equivalent. A physical object with memory to store keys and a processor to perform cryptographic operations on board. These devices are meant to be non-forgable, and through the use of an associated password or pin, non-transferrable. While these may improve a user's ability to keep track of their own key, and reduce the possibility of theft from their workstation, they really do not offer significant usability improvements for the core problem of establishing a key's trust metric in the first place.

Our Approach

Operating network services such as email is always an exercise in making compromises between usability and security. Our goal is to secure email for an entire organization with little or no affect on usability. This section describes our choices, which we believe are the most practical, if not the most secure.

Philosophy

Networked Future
: Email remains the "killer app" of the internet, yet it's common use remains only incrementally improved over the last couple decades. Modern users are increasingly connected from their office, home, laptop, cellphone, and shared computers. Email resources become far more accessible and manageable if they are stored on a server and merely accessed by clients.

Impractical is Insecure
: Any email server can be made secure by unplugging the power supply and putting the hard drive through a wood chipper. This exteme measure represents the far right end of a spectrum that spans security and usability. While it sounds egregious, most people have opted for an equally ridiculous approach to email that represents the opposite extreme. Most email security systems have failed to gain popularity because of the overhead involved for users in managing them. Email security measures must be made practical and easy or they will fail.

Emphasis on Standards (Keeping Email Independent)
: Standards for email security must be embraced for many reasons. Primarily, factions created by competing schemes are the most damaging prospect for secure email. Developers should compete by building the best implementations, and strive to improve the standard wherever necessary. Innovations which break interoperability jeopardize the fragile possibility of making email security ubiquitous. Remember, email you send, can only be as secure as the recipient. In this case, it works in everyone's best interests to embrace the standards.

  Organizations that cannot decouple email from calendaring or other services cannot take security into their own hands. Unfortunately, vendors will always try to entrap customers with proprietary systems that ensure brand loyalty. This works directly against an organization's ability to prioritize security.

Architecture

In our experience, deploying end-to-end security with tools like PGP or S/MIME has always failed. Failed to gain a critical mass of acceptance within an organization. Users lose track of their keys, or stop using them. Usually, they lose track of other people's keys. While we hope to one day be able to operate in this mode, we can't wait for it to become practical. The measures we take to secure email do not work against end-to-end encryption systems. Users are welcome to layer them on top of our own measures.

Our approach revolves around creating a highly secured email server for an organization. This server (or group of servers) stores mail for users, allows only encrypted access, and provides security for mail against seizure or network break-ins by encrypting all mail on disk. Collectively, these measures create a comprehensive

security perimeter for the enterprise. All email is encrypted in storage, and in transport. Users merely access their mail the way they always have, but with added security features within the organization. Of course, once mail is sent off to Hotmail or other unsecured services no security is maintained. However, users sending email back and forth within the organization will always be fully secured.

Operating System Security

Mail servers by definition need to be accessible from the internet. This puts them on the front line for network attacks. Given the value of email in the modern enterprise, mail servers should be the most secure machines on the network. Unfortunately, they are often easy to set up and neglected until mail stops moving. Most of the attacks on servers these days take advantage of the fact that they run on well known, general purpose computers. Organizations should consider choosing an operating system that lends itself to being stripped down to only the essentials, has a strong security track record, and maintains stability through very frequent upgrading and patching. Administrators of highly secured machines should be in position to rebuild applications from source code if possible. This affords autonomous and rapid response to new vulnerabilities that cannot be achieved when relying on vendors or package managers. Many operating systems offer additional security features such as chroot which confine the applications within the machine, ensuring that a compromise delivers as little as possible to an attacker. We also highly recommend the use of host based intrusion detection tools such as Osiris or Tripwire to ensure that malicious software is never loaded or run.

Application Services

Over the last several years, nearly all mail software has begun to support SSL for existing email transport protocols. Requiring only minor configuration changes for clients and servers, this measure is the simplest and most effective means of improving email security. Our servers accept only encrypted connections from users, and use these to secure authentication credentials as well as mail in transit over the network. Users can continue to access their mail via POP, IMAP or even Webmail from anywhere on the internet in a secure fashion. Applying this approach to SMTP solves the problem of relaying mail for users as well. They can be authenticated before sending mail, instead of relying on their network address as is the most common practice. Requiring the use of encryption on these protocols is the first step in securing the email for an entire enterprise. By taking this step, the entire internet becomes a secured intranet for email.

Filesystem Encryption

Securing data while it is stored on a server protects against a massive array of passive attacks. Unfortunately, not a lot of work has been done on encrypted filesystems. The few that exist tend to offer very little flexiblity. Of the couple that do, immaturity renders them almost useless. The world needs a quality encrypted filesystem. In the mean time, we've developed a system to utilize OpenPGP for local mail storage. In our approach, all mail is stored on the server, encrypted on a per-user basis. Physically seizing one of these servers wouldn't be enough to get you the mail, hacking root on the box wouldn't help you read any of the mail either. On most mail servers, system administrators with root can read everyone's mail. In this model, it is encrypted, until a user logs in (via IMAP, POP, etc.) and then messages are decrypted in memory, and re-encrypted (with SSL) before being sent out over the network.

Responding to Viruses

A single company writes 99% of the code used by all viruses. This unfortunate fact makes it seem like the problem could be easily solved, but it hasn't been. Viruses that propogate through email always take advantage of an ability to trick the computer into executing some untrusted code. Using scripts on our mail servers, we're able to "inocculate" most email viruses by scanning their content for executable code. This causes the suffix to change on email attachments, and the Javascript or ActiveX controls within mail to be disabled. Users can manually assess the content of a message before deciding to execute any new code. Unfortunately, this requires some discretion that many users simply don't have. When choosing an OS or mailer that is immune to viruses is impractical, organizations should at least ensure their mail server cannot fall prey to viruses. Having an ability to scan outgoing email for viruses can be a great preventative measure to keep your organization from spreading the problem as well.

Handling Spam

Spam has become a plague of the internet. A universal nuisance that shouldn't be a problem for anyone these days. Every mail server should have a spam filter employed to automatically identify spam and handle it

accordingly. A common practice is to create a separate "Spam" folder for each user where these messages are preserved but wholly separated from the rest of a user's mail. Modern spam filters incorporate a variety of techniques and are very accurate. We recommend against the use of SMTP server blacklists, they tend to penalize legitimate users. Per-address and per-message blacklists such as Vipul's Razor have been fantastic. We use the free and extensible SpamAssassin package and have obtained fantastic results.

Compromises We've Made, How to Hack this System

We believe our design is the best that can be done currently without modifying user behavior. To get to that point, we've made security compromises along the way. In particular, we've reduced all encryption to the equivalent of symmetric encryption, relying on a password. So in effect, email is only as secure as a user's password. Beyond that, without the use of hardware security devices, we remain vulnerable to active attacks on the server. If an attacker can take over the mail server and continue running services, but replace them with modified versions, it will be possible for him to obtain the keys necessary to decrypt mail. This of course is far more work than would be necessary with an average mail server, but nonetheless, this possibility exists where it would not in an end-to-end encryption system, especially for a malicious system administrator running the server.