Master thesis for one student

Information flow control in world-wide, multi-party (store-and-forward) secure messaging systems

Contact: hauser@acm.org

Introduction

Standard eMail[1] is over 25 years old and is probably the number one application on the Internet. While the protocols used rely on the assumption of a benevolent environment, with the emergence of viruses and spam, the situation has changed dramatically. Moreover, users and institutions have become increasingly demanding in terms of their own requirements, such as confidentiality, integrity and non-repudiation. Standard eMail has been a champion in making information flow asynchronously through the internet in a point-to-point manner, but it miserably fails at any security goal except for “availability”. So while there is information flow, there is little control over it from a security point of view: no integrity, no confidentiality, no authenticity, and no accountability.

The secure eMail standards[2] specify content encryption approaches (hereafter called “SMIME”) to achieve these requirements. The original standard[3] also will celebrate its 10th birthday in June 2009 with surprisingly little acceptance, measured by the percentage of eMails using such protection. Difficult to solve legal issues regarding accountability for corporations[1] are one reason (e.g., the need to archive mail content) and the significant expenses entailed by running the required public key infrastructures (“PKI”) is another one. An alternative, which is less secure but still worthwhile, and also a lot cheaper to implement, is enhancing the standard eMail protocol[1] and mail client accesses with socket layer cryptography[4] (hereafter “STARTTLS”). While this cannot provide end-to-end guarantees, it can provide further, important security characteristics such as ‘relationship privacy’, i.e. an unauthorized observer cannot determine (at least without full-scale traffic analysis) which sender tries to reach which recipient and with what subject matter etc.

Being far from the predominant transport mode, this approach to at least secure links between mail servers (i.e. the sending part of it a.k.a. mail-transfer agents “MTA”) and/or also with mail programs[2] has experienced significant “opportunistic” adoption[3].

Since the early days, eMail has also been made more useful in business contexts with “signalling” information regarding the message status since the “fire-and-forget” approach of store-and-forward mail is often not meeting end-user expectations about their communication service. Therefore Delivery Status Notifications[5], vacation messages, bounce messages for non-existing users[4] are an increasing share of the eMail traffic. Modern MTAs [6] largely implement such features and also master STARTTLS at a high maturity level. As a conclusion, end-to-end eMail encryption so far has not lived up to its promise and there are little indications it will do so in a widespread manner anytime soon while link-layer encryption is implementable, but provides unsatisfactory overall protection results when used as the only measure.

Thus, for a long period of time, most institutions[5] practically only had the choice of either using eMail or complying with data protection laws but due to technological complexity not both. Such institutions recently

[1] Sarbanes Oxley Act in the U.S.

[2] = mail user agents "MUA". This means that also the POP or IMAP protocol are encrypted on the socket layer

[3] i.e. a roughly half of the professionally operated MTAs do offer STARTTLS-based security, albeit only very few making this a mandatory requirement

[4] produced from 5xx smtp return codes

[5] e.g. also government agencies themselves

started endeavours to both comply with the laws and “informationeller Selbstbestimmung” and still use modern internet technology thanks to having

a) strong crypto-libraries in mail clients such ThunderBird or Outlook
b) MTAs dealing reliably with STARTTLS and the underlying PKI prerequisites e.g. postfix
c) value added security service providers such as PrivaSphere[7]

Their end-users, however, cannot be expected to be significantly more sophisticated than average Internet users. Thus they continue to expect using all the features described above extensively (except maybe for SMIME).

While in this setup, it is achievable to protect original message delivery reliably, “signalling”-traffic in exception situations may jeopardize such achievements.

There are many ‘piecemeal’ possibilities to solve this problem, albeit often combined with a service level reduction. Therefore, the community is asking for an over-all plan[8] to address information flow control in secure messaging.

Project Objectives

Create an inventory of (secure) message flow scenarios based on current standards and infrastructure properties. Use this to develop an improved conceptual understanding of the problem and propose a solution (or set of solutions) to identified weaknesses.

Work Plan

To be refined with the student:

- Describe the achievable state-of-the-art information-flow protection levels (e.g. content confidentiality, relationship privacy, protection against heuristic analysis – e.g. by means of traffic analysis to determine message relations based on time or size patterns, etc.) and scenarios (recipient exists and reads, recipient exists and hasn’t read yet, recipient hasn’t read yet but informs about probable reading time by means of a vacation message, recipient exists but never reads, recipient doesn’t exist in recipient domain, recipient exists but reading domain isn’t reachable for a prolonged time, …) in secure messaging
- Describe the desirable message-flow protection levels and scenarios
- Identify gaps and weaknesses between the achievable and the desirable information flow protection levels and scenarios in secure messaging
- Devise solution approaches and evaluate them
- Prototype preferred solution approaches

Optional:

- Create a fully operational information-flow-controlling MTA
- Discuss them with the community to solidify the description of the previous work and get third party opinions on your solution hypotheses
- Draft an RFC

Prerequisites

- Interest in eMail security, threat analysis, end-user security processes, testing scenarios
- Strong conceptual and design skills for world-wide networking and messaging

Supervision: Prof. David Basin (basin@inf.ethz.ch), Dr. Ralf Hauser (hauser@acm.org)

References:

1. based on http://tools.ietf.org/html/rfc821
2. SMIME RFCs (3851 etc.)
3. Version 3 of SMIME (RFC 2633)
4. SSL/TLS as per RFC 2487
5. DSN RFC 3461
6. E.g. http://www.postfix.org
7. https://www.privasphere.com
8. http://marc.info/?l=postfix-users&m=123099471818688&w=2