Initial Plan: Subscription Email

Student	David Lowe
Module	CM0343
Credits	40
Year	2012/2013
Project supervisor	Dr. Frank Langbein
Moderator	Prof. Ralph Martin

Project Description

SMTP[1]-based email is the most popular and widely used method of receiving and delivering messages on computer networks such as the Internet. Its popularity has led it to become entrenched. Over the years, its drawbacks have become increasingly clear, the two most obvious being its inherit insecurity and its susceptibility to huge volumes of spam.

In the original design, electronic messages are transmitted in plain-text, from the user's client, through SMTP relay servers, to the recipient's storage server, finally arriving at the recipient's email client. This leaves the user's email vulnerable to snooping and man-in-the-middle attacks at any one of these intermediary points, or at the wire. These vulnerabilities are what is meant by the insecurity of email.

SSL[2] only partly solves this problem, as it must be applied at every link in the chain, and the user must still trust the relay servers. PGP[3] and other end-to-end encryption technologies are the only way to guarantee privacy and authenticity, but it has proven too cumbersome and confusing for rapid adoption by end-users. Its critical flaw is that both parties must agree to use PGP before the technology can be enabled.

Similarly, existing solutions to combating spam and fraudulent emails are lacking. The two most effective and commonly used techniques are blacklists and content-based filtering. Blacklists of known spamming servers are costly to generate and maintain, and they greatly increase the difficulty of creating new legitimate email servers. As for content-based filters, their success rate is still too low to be considered a perfect solution.

I hold that these problems are inseparable from the design of SMTP. As a result, the only way to eliminate insecurity and spam is to design a completely new protocol for receiving and delivering electronic messages. My project to fulfil the requirements of module CM0343 will be to design such a new protocol, and to complete an implementation of it.

Inspired by proof-of-concept systems like Internet Mail 2000, the protocol's defining characteristic will be that messages are stored on the sender's server, rather than the recipient's server. The recipient’s client only fetches her messages from a list of pre-selected servers. Because a user would never voluntarily approve a spammer's address, she will never receive any spam, solving one of the problems with email. However, as a consequence of this design, it will be impossible to send messages to users who have not already approved you, unlike email.

To protect the confidentiality and the authenticity of the user's conversations, all electronic messages will be encrypted using PGP keys. This will be done automatically and invisibly to the users of the system, to avoid the usability problems of handling PGP keys manually. Users will be able to vouch for other users' authenticity, implementing a simple web of trust. This requirement has

1 Simple Message Transfer Protocol
2 Secure Sockets Layer
3 Pretty Good Privacy

drawbacks: for example, if a user loses her PGP key, she will be unable to decrypt old messages, and she will have to rebuild the reputation of her online identity.

In summary, the new protocol should eliminate two of email's problems: spam and insecurity. However, its design will introduce two new problems: a higher barrier to initial contact, and a higher risk of losing archival and identity data. My task is to evaluate this trade-off, to complete a proof-of-concept implementation of the protocol, and to test and evaluate it.

Project Aims and Objectives

- To write an interim report by the 14th of December 2012, which would contain:
  - An exploration of the limitations of SMTP-based email, particularly spam and insecurity
  - A background study on other alternatives to email, especially ones with a subscription and a decentralised model, such as Internet Mail 2000
  - The specification of the first version of the protocol
  - The rationale for the design choices made
  - Examples and walk-throughs of specific parts of the protocol
  - The first prototype of the server and client
- To write a final report by the 5th of May 2013, which would contain:
  - A description of the results of the project, especially the implementation of the server and of the client
  - A discussion of the drawbacks of the design of the protocol
  - An elaboration on how these problems were approached
  - An evaluation of the design and of the implementation
  - The source code of the proof-of-concept server-side and client-side code, which would demonstrate the decentralised nature of the protocol and its viability

Work plan

Term	Week	Starting date	Assignment
Autumn	Week 1	1 October 2012	Prepare the initial report
	Week 2	8 October 2012	Prepare the initial report
	Week 3	15 October 2012	Review the initial report with supervisor and initial background research
	Initial report deadline on the 19th of October		
	Week 4	22 October 2012	Complete background research on the current alternatives to email, especially subscription-based protocols such as Internet Mail 2000
	Week 5	29 October 2012	Design the first iteration of the protocol
	Week 6	5 November 2012	Implement the first iteration of the server-side code with a simple client
	Week 7	12 November 2012	Refine the protocol, and test and improve the implementation
	Week 8	19 November 2012	Refine the protocol, and test and improve the implementation
	Week 9	26 November 2012	Focus on resolving the first-contact problem in the protocol and the implementation
	Week 10	3 December 2012	Write interim report
	Week 11	10 December 2012	Write interim report
	Interim report deadline on the 14th of December		
	Christmas recess (15 December 2012 to 6 January 2013)		
	Revision and examination period (7 January 2013 to Friday 25 January 2013)		
Spring	Week 1	28 January 2013	Focus on resolving the first-contact problem in the implementation
	Week 2	4 February 2013	Examine the security of the protocol and the implementation
	Week 3	11 February 2013	Explore data loss problem
	Week 4	18 February 2013	Implement a possible data recovery option
	Week 5	25 February 2013	Focus on making the web of trust easily comprehensible and accessible by the users
	Week 6	4 March 2013	Refine the implementation
	Week 7	11 March 2013	Evaluate working system
	Week 8	18 March 2013	Evaluate working system
	Easter recess (23 March 2013 to 14 April 2013)		
	Week 9	15 April 2013	Write final report
	Week 10	22 April 2013	Write final report
	Week 11	29 April 2013	Write final report
	Final report deadline on the 5th of May		