

A “Perhaps Naive” Suggestion for Dealing with Spams and Junk Emails

Min Chen

Department of Computer Science, University of Wales Swansea
Singleton Park, Swansea SA2 8PP, United Kingdom
m.chen@swansea.ac.uk

Created: Tuesday 2 March 2004

Updated: Friday 5 March 2004

Dear colleagues,

Like you, I am frustrated by spams and junk emails. I wonder if a two-stage email transfer protocol (see below) would work. I am an outsider to the research of emails, viruses and spams. I do not know if anyone has proposed such a solution before, and I am sorry that if it appears to have reinvented the wheel. I would be very interested to hear from experts if such a protocol could help combat spams and junk emails. If it is workable, I am quite happy for any company to make it into our daily life, as soon as possible.

A Two-stage Email Transfer Protocol

1. Basic Action.

A user **X** sends an email to user **Y** through an email user interface.

2. Sending.

X's email server will send the entire set of *envelope data* to **Y**'s email server, together with the first N bytes of the email body. N , which is a positive integer, is carefully chosen to eliminate the possibility of sending any meaningful or harmful attachment at this stage, and at the same time, it allows the transmission of simple messages such as “my URL is www.swansea.ac.uk”. This is so-called **Stage-One Transfer**.

It is the responsibility of **X**'s email server to deposit the entire email until it is retrieved by **Y**.

For emails with N or less than N bytes, the stage-one transfer is adequate, and there is no need for stage-two transfer and depositing.

3. Receiving.

Upon the receipt of the first stage email from **X**, **Y** may request the full body of the email, or simply ignore the email, according the envelope data and the first N bytes of data.

If **Y** chooses to receive the full body, an **ACCEPTENCE** control message will be sent to **X**'s server which will subsequently send the full body of that email, and free the storage space taken by the email. This is so-called **Stage-Two Transfer**. On average, this should take less time than downloading a web page.

If **Y** decides to junk **X**'s email without requesting the full body, **Y**'s email system may, or may not, sends a **NOT-INTERESTED** control message to **X**'s email server. Such a

message will help **X**'s server to remove unwanted emails. This **NOT-INTERESTED** control message can be sent automatically by setting the appropriate preference options.

X's sever may also automatically remove unwanted emails after a certain period based on appropriate preference options.

4. Trusted Correspondents.

Y's email system may automatically request stage-two-transfers for **trusted correspondents**. This automation may have the following five levels:

- Level (-1): Refuse all stage-two-transfers.
- Level (0): Automatic request all stage-two-transfers.
- Level (1): Automatic request stage-two-transfers for all stage-one emails from creditable sources identified by a list of email addresses and domains recorded in **Y**'s email systems. This can deal with junk emails with genuine source addresses, as well as most spams with random source addresses.
- Level (2): Automatic request for stage-two-transfers for all stage-one emails with creditable digital signatures. Only spams that may get pass digital signatures are those which have the means to obtain both encryption key and signature of a particular user.
- Level (3): Combining Level (1) and Level (2) mechanisms.

Levels (-1), 0, and (1) require little changes to the current email protocols, while Levels (2) and (3) would require the introduction of digital signatures to the envelope of the emails.

5. Digital Signature.

One suggestion is to have two new optional fields in each envelope, namely, signature string and decryption key. **X** may choose to send his/her email with:

- a) NO signature string, NO decryption key
- b) signature string only (perhaps the default)
- c) decryption key only
- d) BOTH signature string and decryption key

Y's email system should be equipped to record signature strings and decryption keys for the correspondents trusted by **Y**. A digital signature sent with (d) is not as trustworthy as that sent with (b) which is to be decrypted using a pre-recorded decryption key.

6. Comments.

Although this protocol may introduce some delays in transmitting emails, on the whole it reduces unnecessary email traffic considerably. It shifts some burden of handling junk emails from receivers to the senders. A commercial junk email service will need a lot storage space for unwanted emails, or risk being unreliable in providing stage-two transfers. By examining unwanted emails, an email server of a large organisation can monitor its problems of spams and worms more effectively. The introduction of digital signature and the concept of trusted correspondents can make spams much less effective.

Of course, this is just a rough sketch, and the protocol needs to be refined, for example, for handling emails with multiple destinations. It does not seem to be a difficult

issue. There are also user interface issues, especially in minimising users' effort in sending and receiving emails. It is believed that much of the additional effort can easily be passed onto the preference options, hence requiring little extra effort from users. However, there will be some necessary effort to educate ordinary users about the concepts of digital signature and how it works.

End Notes (5 March 2003)

On the 2nd and 3rd of March, I sent the first draft of this document to several colleagues in University of Wales Swansea, some companies that producing email software, server Internet colleagues who have made valuable contributions to the discussions of spams and junk emails on the Internet, and the experts in the IETF, including those organising *Enhancements to Internet email to Support Serve Environments WG*, and *S/MIME Mail Security WG*.

I have received some positive feedbacks. Special thanks to Ken Hollis (Gandalf The White) who gave the first encouragement message among several automatic messages from some companies, for instance:

“The feature you are inquiring about is not currently available. If you would like to see this feature in a future version of X please visit the following link to add your suggestion:

www.X.com/developers/feedback/

Thank you for choosing X.”