J. Zhou, W-Y. Chin, R. Roman, and J. Lopez, "An Effective Multi-layered Defense Framework Against Spam", *Information Security Technical Report*, vol. 12, pp. 179-185, 2007.
http://doi.org/10.1016/j.istr.2007.05.007
NICS Lab. Publications: https://www.nics.uma.es/publications

An Effective Multi-Layered Defense Framework against Spam

Jianying Zhou and Wee-Yung Chin
Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
{jyzhou, wychin}@i2r.a-star.edu.sg

Rodrigo Roman and Javier Lopez
E.T.S. Ingenieria Informatica, University of Malaga
29071, Malaga, Spain
{roman, jlm}@lcc.uma.es

***Abstract* — Spam is a big problem for email users, and more serious for wireless mobile users with limited bandwidth. The battle between spamming and anti-spamming technologies has been going on for many years. Though many advanced anti-spamming technologies are progressing significantly, spam is still able to bombard many email users. The problem worsens when some anti-spamming methods unintentionally filtered legitimate emails instead! In this paper, we first review existing anti-spam technologies, then propose a layered defense framework using a combination of anti-spamming methods. Under this framework, the server-level defense is targeted for common spam while the client-level defense further filters specific spam for individual users. This layered structure improves on filtering accuracy and yet reduces the number of false positives.**

Keywords – Spam, Network Security,Security Service

I. INTRODUCTION

With the widespread usage of the Internet, email has become a popular and useful way of conveying information across the world. Email has the ability to attach digital information such as files and images with little cost. The convenience of interchanging information almost instantly allows email to gain significant market share over postal mail.

Motivation of Spammers. Because bulks of email messages can be sent at little cost, this lucrative business has caught the attention of many businessmen. In contrast with the costly advertisements in newspapers, magazines or televisions, the email system provides an economical channel for strategic marketing of their products. With millions of email users currently available, even if the response rate is as low as less than 1%, this money-spinning business is difficult for many businessmen to ignore. In fact, there are approximately 8% of US email users who actually buy products from spam in 2002!

Spam is not restricted to only email advertisements. In fact, spam also consists of malwares such as virus, worms, spywares and Trojan horses. Since email is a common communication tool for many Internet users, it provides an excellent channel to spread malicious virus and worms such as the ILOVEYOU virus. With anti-virus software always a step behind the virus, irreversible severe damages have been done before their outbreaks can be controlled. In a more personal perspective, as Internet services such as Internet banking are gaining popularity, email spywares are also engaged in getting confidential information from unsuspecting victims through phishing. Some hackers even go as far as extorting money from large companies by threatening to make *Distributed Denial of Service* (DDoS) attacks using machine zombies taken over through email Trojan horses.

As a result, spam has grown significantly from an occasional nuisance email to an explicit daily menace over the years. There are even spamming companies that are specially set up to engage in the spamming activities. Most of them are operating in countries where there are no laws and controls over spamming. It is estimated that about 40% of all emails are spam, which cost the world around US$50 billion every year!

Co-evolution of Spam and Anti-Spam. For the past two decades, both camps of spamming and anti-spamming technologies have been progressing considerably. Although, a number of anti-spamming technologies have been proposed and deployed in email systems, there is no ideal solution yet. In the first generation of the spam reign, anti-spamming technologies start off with simple methods such as white list, black list and keyword matching to filter spam. These methods are effective at the beginning, but soon, spammers find their ways to escape detections. In respond to the keyword searching, spammers try to bypass the filter by altering the spelling of keywords or adding symbols between letters (for example, sex can be spelled as s-e-x). As for the white list and black list, spammers use spoofed legitimate addresses or new email addresses to send out the spam. Recently, spammers even use virus such as SoBig.F to control zombie machines of innocent victims to distribute spam. Since their email addresses are unlikely to be found in the black list, many email systems actually accept those spam.

Anti-spamming technologies soon move on to more statistical approach based on sentence structures and word frequency like the heuristics and Bayesian filters. However, spammers circumvent those defenses by means of using shorter sentences and synonyms, reducing their effectiveness. Other spamming tricks are inserting trusting good words or URL of non-spam sites in the spam messages, thus, making spam undetectable. The most worrying problem is when spammers place their spam messages into images, which is almost impossible for any anti-spamming method to detect.

The latest anti-spamming technologies are mostly based on artificial intelligence algorithms to differentiate between spam and genuine emails. Nevertheless, no matter how advanced the algorithms are, they are unable to filter all the spam. Because most of the time, the filtering accuracy is related to the false positive rate, some genuine emails are accidentally classified as spam instead! An almost ideal solution is the challenge-response method, which is a common authentication method. In this method, the sender will be issued a challenge from each receiver whenever he sends out an email, and need to provide the corresponding solution. Though this method is effective in stopping spam, it introduces new problems to the users such as email delay (especially in the case of email multicasting) and denial of service (when the spammer uses the victim's email address as the sender's address). Does an ideal anti-spam solution really exist?

Our Contributions. In this paper, we analyze on the advantages and disadvantages of existing anti-spamming methods. Based on their pros and cons, we have derived a multi-layered defense framework against spam. When a combination of anti-spamming methods is used jointly in a layered structure, we can improve on the efficacy of spam filtering while reducing the number of false positives. In one of our sub-system, we use a pre-challenge method. A prototype of this method is built and implemented as an add-on in Microsoft Outlook 2002. The performance of the sub-system is being tested and analyzed. In our experiment, the sub-system is able to attain a remarkable 100% filtering accuracy of the spam. We believe that our layered defense framework is promising in eliminating spam thoroughly.

II. Analysis of Existing Anti-Spamming Methods

Simple Mail Transfer Protocol (SMTP) has been the fundamental email architecture since 1982 [1]. As the protocol is standardized and widely used, it is difficult to migrate to a new protocol. For the past years, most of the anti-spamming researches have been focused on application-level solutions. We now analyze the popular anti-spamming methods. Two indices need to be considered when evaluating the effectiveness of a method.

- *False Positive Rate* – the percentage of legitimate email misidentified as spam.
- *False Negative Rate* – the percentage of spam not detected.

Black List. *Black list* is one of the first generation anti-spamming methods. A list of recognized spamming email addresses and domain names is kept in the *Mail Transfer Agent* (MTA) or the email client system. Emails originated from these email addresses or domain names are discarded automatically. The method has the advantage of offering almost no false positive since every discarded spam that is detected is well-known to be a spam. However, this method is unable to attain a high filtering accuracy because spammers can either use new or spoofed email addresses to spam. Another problem with black list is that this service can be brought down easily when the MTA suffered from *Denial of Service* (DoS) attack [2]. Moreover, if a domain such as hotmail is blocked, the user might have unintentionally blocked 90% of the wanted email from that domain! As a final note, black list is too inflexible to be used alone.

White List. *White list* works similarly like black list, except that the list contains permissible email addresses or domain names that are known to the user instead. Most of the time, these email addresses are only either from the address book or previously sent to the mailbox, so the filtering capability of this method is fairly limited. As emails from fresh unfamiliar email addresses will be instinctively denied, this method introduces extremely high false positive rate. Besides, if spammers are able to access to the list, they can readily bypass this filter with spoofed addresses in the list. A common spoofed email address can be a well-known mailing list address that is white-listed by many users. Hence, this method also has moderately filtering rate. As this method requires lots of constant manual maintenance to work efficiently, it is usually used together with other anti-spamming techniques.

Keyword Searching. *Keyword searching* [3] is one of the most widely used methods to combat spam. It has the advantage of accomplishing high filtering accuracy. Through identifying keywords found in common spam messages, a large fraction of common spam can be eliminated. However, this method is ineffective in detecting word variations or context. Thus, there may be many false positives spam at the same time. For example, a legitimate email which contains the word "breast" can be mistakenly classified as spam. Besides, spammers can simply overcome this static filter by deliberately misspelling the words or using synonyms. In addition, this method will not be able to detect spam messages in images. (In fact, most anti-spamming methods will not be able to detect spam message containing in images.)

Reputation Services. *Reputation service* is an anti-spamming method used at the MTA level. A traffic monitor system will take note of the volume of email traffics of various email addresses or domain names. The reputation of the email addresses or domain names will increase or decrease according to any unusual change of volume, which may be an indication of spam. One of the most successful email traffic monitoring networks is SenderBase [4], which tracks about 25% of the world's email traffic. This service can identify and block 75% of incoming spam with about one false positive in a million emails. Nevertheless, one disadvantage of this method is that by the time the spamming email addresses or domain names are known to have bad reputation, they have already send out millions of spam. Another disadvantage is that innocent email addresses or domain names may be spoofed by spammers and their reputation tarnished.

Challenge-Response. For *challenge-response* method [5], after sending an email to the receiver, a sender receives a challenge through a reply email. The challenge can range from a simple question to a CAPTCHA [9]. The sender is obliged to provide the correct solution to the challenge in his/her reply email. While this method is effective in catching spam from automatic systems, it is unable to identify automatic legitimate response systems' replies after an online purchase or a mailing list registration. Besides, this method introduces an email delay in the handshaking process which is undesirable. Further-

more, the email address can be easy target of DoS attack when spammers spoofed the target email address as the source address. The attractiveness of this method is that it can easily achieve a 100% rejection rate of all spam from bot, regardless of what spamming tricks are added into the spam messages. Nevertheless, the disadvantage is the inconvenience caused to the senders since they have to send an email twice to every new recipient. Another disadvantage is lack of a feasible solution to handle emails from automatic mailing lists given that they are not able to answer all their members' challenges.

Micro-Payment. *Micro-payment* method [10-13] deals with the root of spam. The user or the client MTA is required to perform a resource-consuming process or pay a small sum of money in order to gain access to the server MTA. Because it will be time-consuming or non-profitable to send out each mail, spammers will be refrained from distributing bulk of emails. Such an approach may create a problem for those client devices with very weak computing capability such PDA and mobile phones. In addition, Internet Service Provider that has implemented the method may lose out customers to its competitors, which are still giving free email services. There is also some argument on who is going to build the micro-payment infrastructure since there is a possibility that the cost of each transaction may be much more than the amount received from a single payment. What about emails from mailing lists and internal email systems? If we were to exclude them from the micro-payment, spammers will still be able to spoof these addresses to send out spam.

Hash/Signature Filter. In this method, the hashes of previously identified spam messages are kept in a database at the MTA level. All incoming emails will be checked against these hashes to distinguish spam apart from normal emails. This method is effective in filtering a fraction of spam. Nonetheless, it is one step behind newly generated spam, they will still be able to get past this filter. Moreover, spammers have already found a workaround to bypass this scheme by introducing a random string into the spam messages to generate different hashes. Another more alarming problem is the swelling of the database over time since there will be thousands of newly generated spam everyday. The checking process time will increase significantly over the years.

Header Analysis. Every email has a header attached to it which contains its routing information. Spammers may insert invalid routing information to protect their identities from being tracked. Therefore, the header of an email can be analyzed to determine if it has a wrong format to find out if it is a spam. Although this method can indicate spam, it can also indicate a wrongly configured mail server. On the other hand, a well-formed header does not signify that it is not from a spammer. In addition, spammers can always take control over machine zombies to send out undetectable spam for them. As a result, this method has a rather poor filtering accuracy but low false positive. It must be used with other anti-spamming techniques to be effective.

Heuristics. In this approach [14], a combination of previously mentioned anti-spamming techniques such as header analysis and signature filter can be used to determine if an email is a spam. Based on a threshold level set by the user, enough evidence will be needed to suggest whether an email is a spam. Since the filtering accuracy is proportional to the false positive rate, this method requires complex fine-tuning to prevent the occurrence of unwanted false positive. Besides, this method is yet to be foolproof and can be bypassed by new evasion techniques employed by spammers such as text hiding, alternate character encoding and messages in images. Nevertheless, this method has a better performance than most of the traditional anti-spamming filters and laid the foundation for future anti-spamming techniques.

Artificial Intelligence. In recent years, there are multiple works dealing with email content analysis based on *Artificial Intelligence* (AI) [15], machine learning and statistical techniques. The main advantage of these methods is the ability for the system to retrain itself while it is put in use. Thus it lessens the intervention of any manual work while keeping a superior filtering accuracy. Although such techniques are more effective than most of the other methods mentioned earlier, new problems are introduced. Firstly, this method requires complicated fine-tuning and testing before they were put in use. Secondly, there is a need for complex analysis on the receiving end, making the process of receiving email laborious and time-consuming. Thirdly, even with the best AI algorithms, perfect spam detection is hardly possible. Lastly, this method may lead to high false positives, which is unquestionably not desirable.

Obfuscation. Spammers usually harvest email addresses from the Internet. Similar to micro-payment, obfuscation method tries to work on the root of spam. It prevents Internet harvesting by displaying the email addresses in an altered but obvious form (e.g., *alan@hotmail.com* can be displayed as *alan* at *hotmail* dot *com*). This method is easy to apply since no changes are required for the email system. However, as there are limited combinations, it allows AI-based harvest programs to retrieve real addresses effortlessly. Moreover, given that spammers may obtain email addresses from other sources, the method is practically ineffective. Furthermore, once the spammer obtains the email address, the scheme does not offer any spam protection. In conclusion, this method can be used to reduce email harvesting but it cannot be used as the main protection against spam.

Methods	False Negative	False Positive
Black List	High	Low
White List	Medium	High
Keyword Searching	High	High
Reputation Service	Medium	Medium
Challenge-Response	Medium	Low
Micro-Payment	-	-
Hash/Signature Filter	High	Medium
Header Analysis	High	Low
Heuristics	Medium	Low
Artificial Intelligence	Medium	High
Obfuscation	-	-

Figure 1: *Summary of Anti-Spam Methods' False Rates*

In Figure 1, we summarize the false negative and false positive rates of the anti-spam methods reviewed in this section. An ideal method is one with zero false negative and zero false positive. Obviously, no single method is able to achieve this. To counter spam effectively, a combination of the appropriate methods in a layered structure is necessary.

There are a number of commercialized anti-spamming products using a combination of the above anti-spamming methods, like Symantec Brightmail [16], Yahoo SpamGuard [17], Apache SpamAssassin [18], CipherTrust IronMail [19]. Some of the products have been deployed in popular email systems such as hotmail, yahoo mail and gmail. They can filter about 95% spam and the false positive rate is not negligible. So spam is still able to bombard many email users, and they are also concerned that some important emails might be filtered by the email firewall.

III. Layered Defense Framework

An ideal anti-spamming solution will be one that can eliminate all spam without causing legitimate emails from being falsely classified (false positive). In fact, falsely filtered legitimate email is more undesired than spam in the mailbox. Here, we propose a defense framework using some of the existing anti-spamming methods at both server and client levels in a layered structure. At the server-level, we try to maintain a low false positive rate while removing the spam. At the client-level, we will further reduce the remaining spam to improve our performance. In other words, the server-level is to filter the common spam while the client-level is to filter the specific spam to each user.

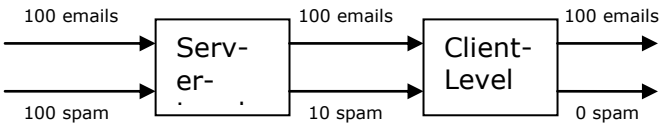

Figure 2: *Ideal Scenario of an Anti-Spam System*

4.1 Defense at Server-Level

We wish to filter the spam while ensuring the false positive rate as low as possible, if not zero, at the server-level defense. Based our analysis in Section 2, the following combined anti-spamming methods could be used here.

Black List. Since black list can filter quite a number of spam from notorious email addresses or domain names, we use this method at the forefront of our layered defense structure. We can incorporate some of the readily available Realtime Black Lists and Open Relay Lists such as Spamhaus Block List and Open Relay Database into our server filtering system. In addition, the corporate administer can include his/her organization-level black list based on past spam history of his/her organization. At this filtering layer legitimate email will not be falsely classified as spam.

Reputation Service. Some newly generated spam may be able to get past the black list. It is annoying, if not tedious, to add new records into the black list manually all the time. With a reputation service, the system will compute a reputation score to every incoming email address and domain name based on user complaints. When the reputation score reaches a certain threshold, emails from the address or domain name will be blocked indisputably. However, individual email address is preferred in this case so as to avoid an occasion when 90% of the valid emails are blocked from a black-listed domain name.

Heuristic. The core defense at the server will be the heuristic method. By using various properties of an email, evidence is collected to determine if the email is a spam. The results of header analysis, reverse DNS lookup and many other filtering rules are collected to make a judgement. This method is able to eliminate most of the spam at the server-level. Nevertheless, it requires tedious fine-tuning with large test data to reach its finest performance. By this layer, we expect our system to filter at least 90% of the spam.

4.2 Defense at Client-Level

At the client-level, we want to further reduce the number of spam that has past through our server-level defense.

White List. White list is our first layer of defense at the client-level. Each user only maintains a concise white list which contains known harmless email addresses from address book or past accepted senders on individual basis. All other email whose address is not in the white list will be passed to the next layer of defense.

Reply List. When a user sends out an email, he/she will expect replies from the recipients. The reply list will contain all the recipients' addresses not found in the white list. Emails from these addresses will be accepted bypassing the rest of the filters.

Pre-Challenge Method. This is our main protection against spam at the client-level [20]. In this method, each email user will define a challenge exclusively for his/her email address. A suitable challenge will be one which is easy for human to answer but yet impossible, or at least difficult, for AI to solve. For the initial contact, an email sender will obtain the receiver's email address together with its associated challenge. The sender will be required to provide the corresponding solution for the receiver's challenge for the very first time of sending an email to the receiver. Only email with the correct solution will be accepted by the receiver's email system. The email address passed the test can be added into the receiver's white list.

With the pre-challenge method, the receiver's challenge is readily available in advance, so the sender can directly solve the challenge and send the email to the receiver. Unlike the challenge-response method, there is no delay even for receiving emails from unknown senders. This also avoids DoS attack in the challenge-response method when spammers forge a sender's address in their mails thus directing all the challenges to the victim's address.

Another benefit of the pre-challenge method is the continuous protection against email harvesting. When the email address is obtained by a spammer, it is useless without getting the solution of the pre-challenge. Even if the spammer acci-

dentally gets the solution, the user can always change his/her challenge any time to invalidate the old solution. The goal of this method is to check whether there is really a human sending the email. Thus it ensures that emails from bots, which do not contain the solution, are undeniably discarded.

Mailing-List Solution List. We assume that the owner of a mailing list will define a challenge to be associated with the mailing list address. All members will obtain the challenge of the mailing list when they sign up to join the list. (They will also receive the new challenge from the owner if it is updated.) The solution will be kept in this mailing-list solution list. Any member who wishes to send to the mailing list will be required to include the solution in the subject field of the email. Therefore, spammers who obtain the mailing list address without the solution are unable to spam the members in the list. Thus, this layer ensures the mailing list is safe from spam.

Warning List. The purpose of the warning list is to prevent DoS attack. When a sender provides an old solution in the email, the updated challenge will be sent to him/her. This warning list ensures that this process will only be done once. The sender's address will be added to the warning list for the first time when the updated challenge is sent. Subsequently, if the sender's address is found in the warning list, the system will no longer send the updated challenge. So the receiver's address will not be subjected to DoS attack if the spammer spoofed the source address as the receiver's address as seen in the challenge-response method.

IV. Performance of Our Defence at Client-Level

We had built a prototype of our client sub-system as an add-on in Microsoft Outlook 2002. Figure 3 shows the sequences of filtering events when our system receives an email.

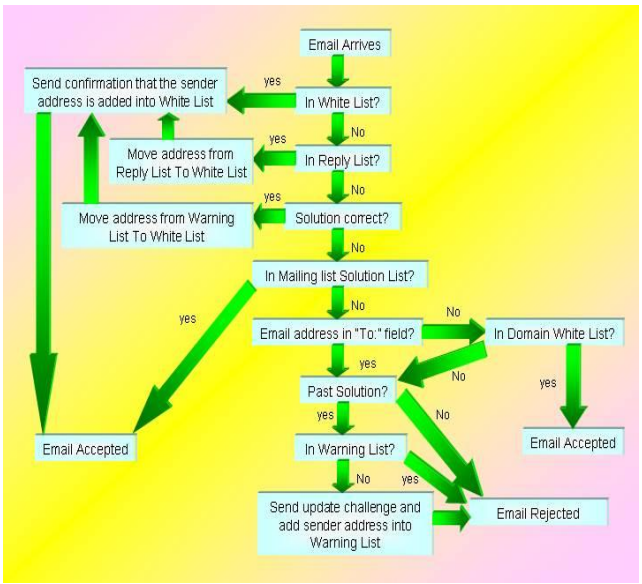

Figure 3: *Flow Chart of Email Filtering at Client-Level*

The system was tested with an email account from Institute for Infocomm Research. Sample of emails were collected for analysis for a period of 3 weeks. In this period, 232 emails were received by the email account, of which 181 of them were spam. Our system is able to filter 100% of the spam. In addition, there is no occurrence of falsely filtered email. For the same period, emails received by a hotmail account were observed. There were 120 emails altogether. Out of the 120 emails, 6 of them were spam. However, hotmail is only able to filter 4 out of the spam. Moreover, 2 legitimate emails are actually mistakenly classified as spam! The result is shown in Figure 4.

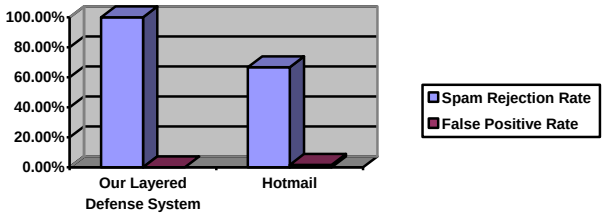

Figure 4: *Comparison of Spam Filtering Performance*

The remarkable performance of our system has been expected. Since our system utilised the pre-challenge method, spam, which were unable to provide the solution, would be all filtered. On the other hand, as long as the senders were aware of our scheme and able to provide the correct solution, their emails would definitely be accepted.

In contrast, hotmail, which uses a combination of Brightmail and IronPort [21], performs poorly in this experiment. One possible reason might be because some of the spam had already been filtered at the mail servers before they actually reached the user's mailbox. Nevertheless, it could not be deniable that a fraction of spam was still able to bypass its anti-spam systems. In addition, a small percentage of legitimate emails had been accidentally classified as spam.

The only setback for our system is the little hassle for the senders to include the solutions in their outgoing emails and for the receivers to define their own challenge for their email addresses. Nevertheless, we wish to emphasize that the sender will only be required to insert the solution for the very first time of sending an email to a receiver. Subsequently, their emails will be just like known senders in the white list or reply list, bypassing the check. Moreover, our system has a function that allows the users to import their address books into their white lists. Therefore, the inconvenience is not as severe as it has been anticipated.

V. Conclusion

We have analyzed the advantages and disadvantages of many anti-spamming methods, and proposed an effective layered email defense framework. At the server-level, spam is restricted by black list, reputation service, and heuristic methods. At the client-level, we build white list, reply list, and warning list filters around a pre-challenge method to counter spam. By using mailing-list solution list and the pre-challenge method, our system has a feasible solution for mailing lists, which are prospective targets of spammers. Our layered protection can be easily integrated into existing email architec-

ture. A prototype of our client sub-system is developed as an add-on to Microsoft Outlook 2002. In our experiment, we are able to filter 100% of the spam without having a single false positive. This result can be achieved at the expense of some trivial inconvenience for the email users. We believe that the tradeoff is justified and, maybe one day, spam will be completely wiped out from where they have started.

REFERENCES

[1] J. Postel. *Simple Mail Transfer Protocol*. IETF RFC 821, 1982.

[2] P. Gray. ZDNet Australia. http://news.zdnet.co.uk/internet/security/0,39020375,39115930,00.htm.

[3] http://www.theworldjournal.com/special/nettech/news/fightspam.htm.

[4] SenderBase. http://www.senderbase.org/.

[5] M. Jakobsson, J. Linn, and J. Algesheimer. *How to Protect against a Militant Spammer*. Cryptology ePrint archive, Report 2003/071, 2003.

[6] M. Iwanaga, T. Tabata, and K. Sakurai. *Evaluation of Anti-Spam Method Combining Bayesian Filtering and Strong Challenge and Response*. In Proceedings of CNIS 2003

[7] SpamArrest. http://spamarrest.com/faq/.

[8] SpamCap. http://www.toyz.org/cgi-bin/wiki.cgi?SpamCap.

[9] Telling Computers and Humans Apart. http://www.captcha.net/.

[10] C. Dwork and M. Naor. *Pricing via Processing or Combatting Junk Mail*. In Proceedings of Crypto 1992, pages 139-147.

[11] C. Dwork, A. Goldberg, and M. Naor. *On Memory-Bound Functions for Fighting Spam*. In Proceedings of Crypto 2003, pages 426-444.

[12] M. Abadi, A. Birrell, M. Burrows, F. Dabek, and T. Wobber. *Bankable Postage for Network Services*. In Proceedings of ACSC 2003.

[13] Penny Black Project, Microsoft Research. http://research.microsoft.com/research/sv/PennyBlack/.

[14] D. Calloway. *Guide to Fighting Spam. The Latest Tools*. http://www.theworldjournal.com/special/nettech/news/fightspam.htm.

[15] D. Strickler. Network World. http://www.networkworld.com/news/tech/2003/0414techupdate.html.

[16] Symantec Brightmail. http://enterprisesecurity.symantec.com/products/products.cfm?ProductID=642%20.

[17] Yahoo SpamGuard. http://antispam.yahoo.com/tools.

[18] Apache SpamAssassin. http://spamassassin.apache.org/index.html

[19] CipherTrust IronMail. http://www.ciphertrust.com/products/spam_and_fraud_protection/.

[20] R. Roman, J. Zhou, and J. Lopez. *Protection against Spam Using Pre-Challenges*. In Proceedings of IFIP SEC 2005, pages 281-293.

[21] Todd R. Weiss. IronPort, Microsoft team on anti-spam effort for Hotmail. http://www.ironport.com/company/pp_computerworld_05-05-2004.html.