A Multifaceted Approach to Spam Reduction

Barry Leiba, IBM Thomas J Watson Research Center, Hawthorne, NY

Nathaniel Borenstein, IBM Lotus Division, Office of the CTO, Cambridge, MA

{barryleiba, nborenst}@us.ibm.com

1 Introduction

As we think about the history of spam reduction, we can see a gradual change in the approach over time, as the spam problem has changed. Many of us may think of spam as a new problem, but in fact, it goes back at least to 1975, as noted by the late Jon Postel.[1] At the start users were mostly "techies", and spam mostly referred to Usenet newsgroup posts that got out of hand, wherein someone would post a message to dozens or hundreds of newsgroups – a message that was unrelated to most or all of the newsgroups to which it was posted. Then, social and administrative action was sufficient: the perpetrator was castigated, perhaps privately, perhaps publicly; repeat offenders would quickly be added to "kill lists". And so, early spam filtering simply identified "bad senders".

The World-Wide Web opened the Internet to a great many people, to a great many non-techies throughout the world, by enabling access to information and services in a way that had never before been possible. Within two or three years, we saw an enormous expansion of Internet usage, of the number of users of the Internet, and, consequently, of "marketing" opportunities thereon. The spam problem has grown astronomically since, and the earlier techniques for keeping it under control no longer work.

Further, spam goes beyond commercial solicitations (unsolicited commercial e-mail, or UCE). The term, and the perception of the recipients, includes such items as chain letters, urban legends, and the many ubiquitous jokes that we've all seen, as well as e-mail-borne viruses and worms, and even those mailing-list subscriptions we no longer want (or neglected to opt out of). And beyond e-mail, instant-messaging spam and SMS spam on mobile phones have also become problems. Some of the mechanisms we discuss here are useful in these areas as well.

2 The Layers of Spam Fighting

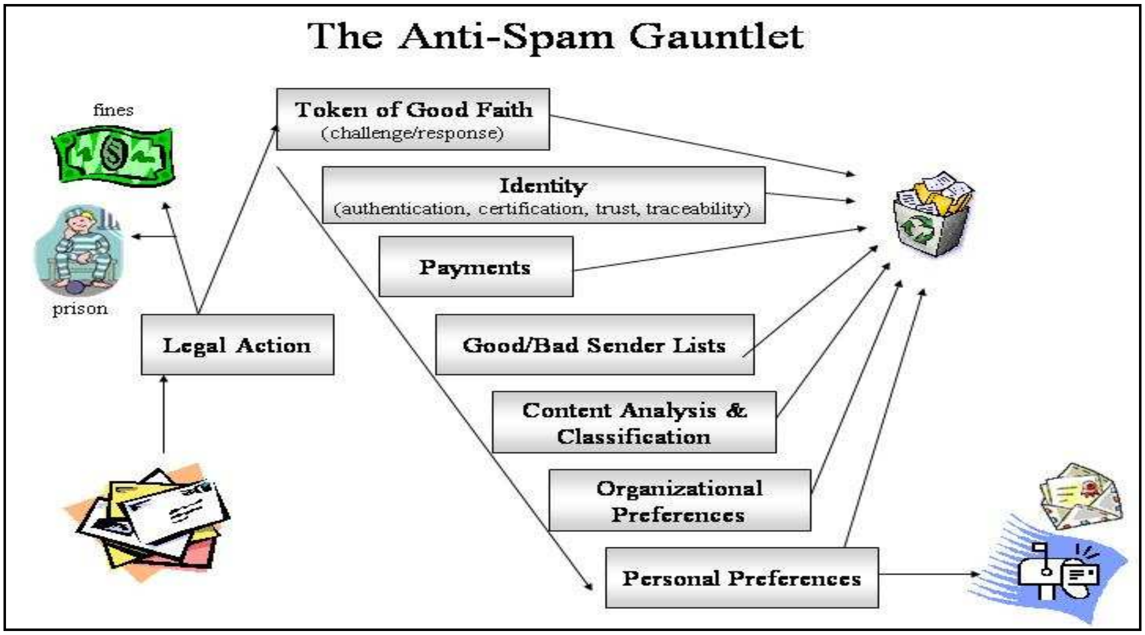
As we develop new techniques, we quickly find not only that no one technique solves the problem fully, but that different techniques excel in different ways. We are in a constant battle between those who want to send spam and those who want to stop it; to be most effective we must fight it in many ways, winning now here, now there, so that as spammers succeed against one shield, another will still stop them. In thinking about how to keep many shields up, we have conceived it as a set of layers, each strengthening the armor, or providing customization and flexibility.

Note that we talk about "spam **reduction**"; we believe that it is impossible to **eliminate** spam. This is partly because spammers, as they aggressively pursue their goals, always remain ahead of us in some areas. Still, with good techniques and customization we could come close to elimination. What is more to the point here is that the **definition** of spam – what is, and what isn't – is not something on which we completely agree. What is spam to one recipient is legitimate mail to another. One organization may ban an entire category of mail that another might welcome. Is **all** unwanted mail spam, or is it possible to have some mail that we don't want, and that we discard, but that isn't really "spam"? Do we consider it spam if it complies with certain rules we set out (for example, it admits where it comes from, and has a working "unsubscribe" function)? We don't, as a community, agree on the answers to these questions, so we can't possibly agree when to declare that spam has been "eliminated".

The next sections discuss some layers, as we consider them (see Figure 1). As we describe the layers, we stress a few key points:

- There is no "right answer" or "best way". The best way to reduce spam is to use as **many** ways as possible in a coordinated and cooperative manner.
- The system as a whole works best when the "good guys" cooperate to reduce spam for all.
- Open standards are key to implementing anti-spam mechanisms. Proprietary algorithms and products are fine, and have advantages in certain contexts, but they must plug together and interoperate in standard ways. The world must be able to implement each layer – and more layers that we haven't considered yet.

Figure 1



It is also important to note that, while we discuss the layers in a particular sequence – those closest to the user first, moving up toward the domain boundary, providing progressively coarser granularity – our intent is that the layers feed information back through each other, so that while layers far from the end user may (should) remove spam without passing it through to the next layer, their actions are expected to take instructions, directly or indirectly, from the subsequent layers. Similarly, the layers closest to the user will have available to them the accumulation of information available to and developed by the prior layers. Legal action, for example, does not happen on its own, but is driven by the preference of a user (or organization) to file a lawsuit.

3 Personal and Organizational Preferences

As we said earlier, not everyone agrees about what is and isn't spam. Individual users, as well as organizations, also differ in their tolerances for variations in spam-reduction results. While one user may accept a one-in-a-hundred false-positive rate in order to block essentially all of her spam before she ever sees it, another might insist on a one-in-ten-thousand false-positive rate and accept the resultant bits of spam. A third might demand **no** false positives, requiring mail to be tagged or segregated, rather than deleted. These desires also have variations for the same user – tolerance for false positives aimed at eliminating porn, but more acceptance of messages about low mortgage rates.

Moreover, similar considerations impose varying spam-control preferences within organizations. A CEO may prefer strong protection from junk mail even at the cost of more false positives in the spam filter, but she would probably prefer the opposite for her company's customer service representatives. Since users are not uniform in their desires or business needs, a good spam-reduction system should take the different users' needs and desires into account, allowing personal or organizational preferences to influence the decisions and the behavior of the system.

A user might be presented with an interface for configuring the system to her specifications explicitly. In one filter prototyped by IBM Research, mail is categorized, and a user may decide what categories to reject or to subject to increased scrutiny by other layers of the anti-spam system. Some categories deal with the broad subject matter: this message appears to refer to pornography, that one to mortgage offers. Other categories address spamming techniques – messages may be categorized as appearing to be trying to hide the true message from spam filters, or as using numeric IP addresses in URLs. One user may decide to throw away suspected porn, file mail with numeric URLs in a "junk" folder, and allow mortgage offers into the inbox – another user could have different preferences.

In general, we believe that it is of primary importance to allow users to specify how they want their spam to be handled, because it is this point that builds trust in the system. Of course, a system that makes a great deal of mistakes is not a trustworthy one in the first place, but if we assume that most systems today will do a reasonable job, then allowing users to choose between, say, deletion or filing to a "junk" folder gives them a way to assure themselves that they are not losing important mail to false positives. Some users will demand that, while others will accept the loss of a small amount of real mail in order not to be bothered by solicitations and offensive subjects.

At the next level of importance is allowing users to customize individually the criteria and thresholds used to filter mail. If users can configure "spamminess" thresholds, or choose specific criteria that are or are not important to them (as in the example above, selecting more aggressive filtering of pornography and less aggressive filtering of mortgage offers), it allows them better to manage their tolerances for false positives and for spam that gets through. It might fairly be argued that such configurability might quickly exceed the interest or capacity of the average email user. However, the major target of such tools is the mail system administrator, who will be able to make use of such flexible configuration to implement filtering policies appropriate to business needs (consider, for example, the needs of spam filters for a company that sells mortgages or medications and discusses them often by email).

We can also infer implicit preferences from other settings and behavior. Access to the user's address book provides a good set of white-listed senders that varies from user to user. Alternatively, the outbound gateways can keep track of the recipients of mail users send, and use it as a customized white list. Message-id headers from outbound mail can be collected, allowing properly tagged replies to get through.

An aspect of user behavior can be considered to provide both explicit and implicit preferences: the anti-spam system can let the user provide feedback on that user's perception of the accuracy of the decisions. We call this "voting", wherein the user tells the system that a piece of mail that was classified as spam is not so in his opinion, or that a piece of mail that got through is, in fact, spam to that user. A well-tuned voting system can learn individual preferences from these votes, and can aggregate the votes of many users to improve accuracy for the full population. In addition to explicit voting, we can use certain other user behavior to infer implicit votes. If a user actively files a message, we can take that as a "good" vote, for instance. Perhaps if a user does not delete a message, but re-opens it a number of times, that also constitutes a "good" vote. Some study is needed here, and, of course, this requires instrumentation of the client application, which may be difficult in some environments, but it could be a useful source of feedback for the system.

One aspect of aggregated voting, whether implicit or explicit, is that it exposes the system to errors caused by incorrect voting, and even to attacks perpetrated by malicious voting. Our experience with deployment of a voting system indicates that the effects of errors are insignificant – that few enough users will vote the same set of messages incorrectly, so that, while a user's own votes over time will affect future classifications for that user, the aggregation algorithms do not find a "trend" here. The case of malicious voting by an organized group, however, is more worrying. Once users are satisfied with how the system works, and they are confident in a low false-positive rate, they do not tend explicitly to vote mail as “good”. A significant group of users who then decided to subscribe, say, to a mailing list run by their political opponents, and then vote every message as spam, could possibly cause a fast-enough effect in the aggregation to eliminate that mailing list’s messages as spam for everyone. This problem can be mitigated in a few ways, and we are investigating the efficacy of these:

- Obtain “good” votes implicitly, as described above.
- Detect and act on anomalies in a user’s voting patterns, or in the voting patterns on a class of mail. If a class of messages has been accepted for some time with few spam votes, and is now getting many spam votes, something may be wrong. This is similar in some ways to implicit voting.
- Limit the rate at which voting is aggregated. With this, for instance, users who select a “maybe spam” category will start seeing these messages appearing there, and can vote them “good” or report the situation. This method has the disadvantage of limiting the system’s ability to respond quickly to a new class of spam.

Since we have not seen such attacks (yet?) in our deployment, we will be simulating them in our experiments.

4 Classifying and Filtering

The heart of any anti-spam system is the part that classifies messages, and filters them (based on the user and organizational preferences we discuss above). There are many classification mechanisms, and we will not discuss details of specific ones here, but the goal of all classifiers is to give messages a “score” and/or “category” (or set of categories) that can be used as input to filters. Classifiers may analyze the message’s headers, the body of the message, its structure, or a combination of all. They may consider only this one message, or may compare it to other messages. They may give a numeric score that indicates how likely they determine this message to be spam, or they may attach a set of keywords or categories, highlighting aspects of the message that might be considered by filters.

While a single classification mechanism can provide a reasonable barrier to spam, any one algorithm is more easily defeated. Instead, an amalgamation of techniques may be used, giving multiple levels of classification and providing a better, more attack-proof shield to spam.

It is the filters that determine which mail should “get through” and which should be considered spam, and a good set of filters provides a great deal of flexibility and customization. Using all the information available, input from all layers of the anti-spam system, provides more options here and, so, the best balance between aggressive spam reduction and a low false-positive rate. Here we decide that for this user, a message with a spam score of 65% that has been categorized as “porn” should be deleted, a message with a category of “contest” needs a score of 80% to be deleted, and a message with a category of “mortgage” and a score of 80% should go to the user’s “junk” folder.

We find multiple levels to be useful and efficient, as a message proceeds down the chain of anti-spam methods. One classifier may determine that a message is very likely to be good, and result in skipping subsequent checking that might delay the message or burden the sender. Or it might see a message as suspicious, and subject it to extra scrutiny at the lower levels. Such layers are particularly important with challenge/response systems and payment systems, reserving the challenge or the payment only for those messages not adequately classified in other ways.

5 The Good, the Bad, and the Honey

At the next layer closer to the Internet, we come to spam collectors and sender lists. Spam collectors, often called “honeypots”, are e-mail addresses that have been created with no legitimate use, and so all mail sent to them is considered spam by definition. The addresses are spread around the Internet in the hope that spammers will harvest them and send spam to them, and that the spam can then be analyzed and that analysis used to improve classification algorithms. Bayesian classifiers make significant use of spam collectors, comparing each message with known spam to determine "similarity" – and spammers have quickly adapted to this and are using various random-text techniques to try to fool the Bayesian classifiers. Still, a well-tuned Bayesian classifier can provide important input to the filters when blended with other techniques. Other classifiers, too, can benefit from spam collectors.

Related to spam collectors are lists of "good" and "bad" senders, sometimes called "white lists" and "black lists". We find that "bad" lists can do more harm than good if not used with great caution. In the absence of stronger authentication (discussed below), the purported sender of e-mail and its domain of origin are often not related to the actual sender. Thus, a black list may be ineffective, and often the use of these lists causes innocent senders, and sometimes entire domains, to be blocked inappropriately. There are a number of such shared lists on the Internet, some listing "senders of spam", some listing "open relays", some listing ISPs that are known to be "spammer-friendly". All suffer from problems related to list maintenance – ensuring that an innocent sender or domain that falls into the list can be quickly removed. The best of these lists can still be used as some part of the input to the classification and filtering, but one must be careful.

Rather better are "good" lists, especially when these lists are customized for the individual user. The problem of spoofing is still there, as is the issue of a "good" sender whose machine has been infected by a virus or worm, but good-lists are more likely to be successfully used by filters than are bad-lists.

Overall the preferred balance between "good" and "bad" lists is dependent on the needs and preferences of the recipient, much like the tradeoff between false positives and false negatives in filtering messages. The tradeoff is much less clear for third-party list services. Because the consequences of poor administration are much worse when the result is false positives than false negatives, we recommend extreme caution in any use of blacklist services.

6 Payments

The financial truth that enables spam in the first place is that sending spam costs so little that even a negligible response rate makes it worth the expense of sending. With paper mail, there is a significant cost to printing the mailing, to handling it, and to paying for the postage. The cost of sending one million pieces of junk mail is such that a reasonable response rate is needed before the mailing pays for itself, so bulk mailers usually target their mailings, for a higher probability of interest, response, and business. With junk e-mail this is not the case, and the cost model is such that just a few responses in a million make it work economically. In fact, the cost of collecting data and maintaining databases make targeting of spam **more** expensive than simply blasting it out to the world.

It is a big social change to make, but a system of collecting payment for sending e-mail would turn the financial model around, and would make an enormous difference in the amount of spam sent, and in the content of it. Even a charging a small fraction of what is charged for paper mail would make a large difference in how spam is paid for. Much of the debate over postage schemes centers on who collects the postage. One scheme -- proposed by IBM Research and called *charity seals* – would let each user direct postage to charities of the user's choosing. Another , known as *attention bonds*, lets each recipient set a price for which an unknown sender can post a bond to get email through; for wanted email, the bond is returned, but for spam it is kept by the recipient as payment for time wasted.

The politics of payments, though, is problematic. Some legitimate email senders, from list administrators to airline frequent flyer programs, are likely to strongly oppose the imposition of monetary charges for sending email. On the other hand, many institutions have expressed a community-spirited willingness to take charge of collecting e-postage fees. The political problems of email payment probably dwarf the technical ones, though the latter certainly remain, and we consider it unrealistic to expect payment systems to be part of the solution in the short term.

7 Identity – Authentication, Certification, and Traceability

One of the most difficult problems in dealing with spam is the inability to determine accurately, and with any degree of trust, who sent the message and how it was routed through the Internet. This problem, as with the problem of spam itself, goes back to the very beginning of the use of electronic mail.[2] The ability to trace e-mail reliably to a specific sender, a specific sending domain, and/or a specific route through the SMTP infrastructure would increase the effectiveness of many classification mechanisms, including the obvious good-lists and bad-lists, and would also increase the feasibility and effectiveness of legal remedies.

Domain verification schemes, of which there are several proposals currently outstanding, including SPF,[3] Caller ID,[4] and DomainKeys,[5] all attempt the same basic task: choose an indication in the message of who the "sender" is (which choice varies among the schemes), get information directly from that sender's domain that helps verify that this message really is coming from that domain, and use that as input to the decision about how to handle this message. In theory, this means that a sender outside of IBM, say, could not send mail that appeared to come from ibm.com, without having that mail flagged as "suspicious". The recipient's filtering system can then use the results of this verification ("the domain is verified", "the verification failed", or "the sending domain does not provide data"), along with some "reputation" of the sending domain (if verified), as a factor in its filtering decision.

All of the proposed domain verification systems have weaknesses, some of which relate to the ways in which e-mail is transmitted today and some of which can easily be sorted out as the mechanisms are refined. Despite any weaknesses, an Internet standard domain verification system is a good tool in the anti-spam toolbox, and we support – and are participating in – the definition of a scheme that puts the best features together and minimizes the weak points. We stress here that the resulting domain verification system *must be an Internet standard* – and we further note that all three schemes we refer to above are have prepared Internet drafts through the IETF, to be considered by the new MARID Working Group.[6] Overall, we think that the differences between these proposals are minor in comparison with the benefits of convergence, and we urge everyone to work together and accept compromises. This does seem to be happening through MARID, as shown by the merged SPF/Caller-ID spec, recently released.[7]

Another issue that arises in any attempt to more broadly deploy an Internet authentication structure is the question of certification authorities. Most schemes, S/MIME included, allow users or organizations to choose one or more certification authorities to vouch for the association between a cryptographic token and some name or identity that has a real-world meaning and (presumably) verification. However, such authorities need to be scrutinized closely. In the best case, such certification authorities are open and competitive, but they can also become monopolistic and exclusionary. (In the extreme, one can imagine a few certification authorities effectively blackballing "undesirables" of their own definition from any Internet access.) An alternative approach, the "web of trust" pioneered by PGP, distributes the responsibility for certification across a dynamic network of "friends of friends." Such decentralization is inherently less open to abuse, but it can be challenging for large organizations to work with.

In addition to validating the sending domain and possibly the sender, it may be useful to validate the path that the message took through the Internet. That tracing is provided today with the RFC 2822 [8] "Received" lines, but there are limitations with that:

- The specification makes most of the information optional, thus they can not be relied upon to provide necessary information. Moreover the syntax is complex, with much variance among implementations.
- They can be inserted, removed, or altered at any stage of relay through the Internet, thus they can not be trusted. They also tend not to interact well with mechanisms that sign the message headers.

Even once the mail enters a trusted environment, and we believe the entries from that point, they are intended for human use, not designed to be parsed by machine. Thus the variations in text layout and content makes the process error-prone and difficult to maintain. There are proposals in progress for reliable, trustable route tracing; we support those efforts and believe a trusted route trace to be a useful feature.

8 Is It Human, or Machine?

Challenge/response systems have become popular of late. The basic idea is that if a particular sender is not on the recipient's good list, a challenge is sent to the sender and the mail is only delivered if the sender successfully meets the challenge. The challenge is designed to be easy for a human to satisfy, but difficult for a computer to do automatically. An example is asking the sender to analyze an image and answer a question about the content: What is the word printed on the barn door? How many birds are perched on the tree branch? The use of audio files can yield similar functionality for the visually-impaired, and should be considered in implementing such systems. Another kind of challenge/response system is a computational challenge, in which the receiving mail server increases the effective cost of delivery by posing a computational challenge to the sender.[9] In the absence of adequate authentication mechanisms, this is a way to ensure that this message is not part of a mass-mailing.

To be effective, a challenge/response system must have a sufficient variety of challenges (and responses) to provide a strong defense against trial-and-error attacks. On the other side, such systems may cause problems in certain contexts. Recipients of business messages, particularly those who will often hear from new or potential customers, may find that the challenges annoy correspondents and result in lost business. Someone doing one a favor might not respond to a challenge. And there **are** reasons for an automaton to send mail – not all of it is spam.

These effects can be mitigated by – and here is our refrain, again – combining this technique with others. A challenge might only be sent once a message is determined, by other means, likely to be spam. If the classification is uncertain, or if the recipient's preferences specify it, a challenge may then be sent. The problem of having challenges sent to unwitting third parties, whose e-mail addresses have been "spoofed" in spam, may at least partially be resolved with a domain verification mechanism: the challenge is not sent if the domain fails to verify.

9 Legal Action

The controversy over the appropriateness and usefulness of the CAN-SPAM Act of 2003 [10] notwithstanding, one final weapon in our anti-spam arsenal is the legal system, and the law is beginning to be used in fighting not the spam, but the spammers. In recent cases in New York [11] and Florida,[12] spammers were sued under older fraud statutes, while in March, a consortium of ISPs filed six CAN-SPAM lawsuits in federal courts in four states against a number of spammers. [13] Some of the techniques already described, particularly those that validate the sender and the message routing, can help support legal action, providing evidence that may lead to convictions.

Clearly, though, the biggest problem with legal remedies is the multiplicity of jurisdictions involved, again highlighting the need for cooperation in the anti-spam effort, here not by technologists, but by legislators. International laws restricting spamming will be more effective than will the laws of any individual country. For these reasons, many in the technical community have expressed strong skepticism about the value of legal measures to counter spam, predicting instead that anti-spam laws will only increase the regulatory and compliance burden on legitimate email senders. However, laws against spam must be understood not just as efforts to prevent spam from happening, but also as tools to punish spammers after they are identified. Convicting spammers of crimes such as fraud is difficult, since it requires a burden of proof that much spam simply doesn't meet. We believe the criminalization of spam, even via flawed measures such as CAN-SPAM, is a first step towards chasing spammers underground and out of countries with such laws. While this, and even the arrest of some prominent spammers, will not stop spam, it could provide enough of a deterrent to substantially reduce the quantity of spam in the future.

10 Conclusions

Spam is a large and complex problem today, and there is a significant financial incentive for spammers to learn to defeat any spam-reduction techniques we develop. Because of that, any robust, long-term anti-spam solution must use multiple techniques in several layers, must incorporate social and legal aspects as well as technical ones, must involve cooperation among all parties interested in finding solutions, and must be rooted in open Internet standards. We believe that, though they be controversial today, payment systems very likely have a place in the long-term solution. Laws specifically aimed at spam will take time to develop properly and internationally, and they, too, will have a firm place in the anti-spam arsenal. We must pursue all these aggressively, because as computer technology advances, spammers are able to mount ever-stronger attacks. We need a solid, ever-stronger defense.

Acknowledgments

The authors want to thank the many in IBM who have participated in discussions and technical work that have contributed to the ideas herein, and to the software used to experiment with and prototype some of them. Those involved include Steven Adler, Ed Bradford, Jason Crawford, Robert Filepp, Mike Gagnon, Mike Halliday, Tien Huynh, Anne Jackson, Jeff Kephart, Mathew Nelson, Michael R. Nelson, Calvin Powers, V. T. Rajan, Isidore Rigoutsos, Richard Segal, Mark Wegman, and Mary Ellen Zurko.

References

1. Postel, J. “On the Junk Mail Problem”, RFC 706, Internet Engineering Task Force, November, 1975
2. Thomas, B. “On the Problem of Signature Authentication for Network Mail”, RFC644, Internet Engineering Task Force, July, 1974
3. Lentczner, M., Wong, M. “Sender Policy Framework (SPF)”, Internet Draft, http://www.ietf.org/internet-drafts/draft-mengwong-spf-01.txt, May, 2004
4. Atkinson, B. “Caller ID for E-Mail” , Internet Draft, http://www.ietf.org/internet-drafts/draft-atkinson-callerid-00.txt, May, 2004
5. Delany, M. “Domain-based Email Authentication Using Public-Keys Advertised in the DNS (DomainKeys)” , Internet Draft, http://www.ietf.org/internet-drafts/ draft-delany-domainkeys-base-00.txt, May, 2004
6. Rose, M., Newton, A, co-chairs “MTA Authorization Records in DNS (marid)”, Internet Engineering Task Force Working Group, http://www.ietf.org/html.charters/marid-charter.html, chartered April, 2004
7. Lyon, J., Wong, M. “MTA Authentication Records in DNS”, Internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-marid-core-01.txt, June, 2004
8. Resnick, P., editor “Internet Message Format”, RFC 2822, Internet Engineering Task Force, April 2001 (a revision of RFC 822, August, 1982)
9. C. Dwork and M. Naor, “Pricing via Processing or Combatting Junk Mail”, Lecture Notes in Computer Science 740 (CRYPTO'92), 1993, pp. 137-147.
10. 108th Congress “Controlling the Assault of Non-Solicited Pornography and Marketing Act of 2003”, United States of America, January, 2003
11. IDG News Service “Microsoft, NY AG team on lawsuit against spammer”, InfoWorld, http://www.infoworld.com/article/03/12/18/HNmsnyag_1.html, December, 2003
12. IDG News Service “AOL, Earthlink sue alleged spammers”, InfoWorld, http://www.infoworld.com/article/04/02/18/HNaolspam_1.html, February, 2004
13. IDG News Service “Major ISPs sue hundreds of spammers”, InfoWorld, http://www.infoworld.com/article/04/03/10/HNspamsue_1.html, March, 2004