Taughannock Networks

An Overview of E-Postage

June 2003 (Updated February 2004)

Executive Summary

E-postage, a cash payment for each e-mail message sent, has often been proposed as a solution to the junk e-mail problem. Can e-postage work? We think not, for both fundamental technical and social reasons. The technical barriers to a working e-postage system appear insurmountable, and even if a system could be constructed, the unforeseen side-effects to the mail system and the Net at large could be a cure worse than the disease.

Why e-postage?

Unlike postal mail, Internet e-mail has always been a "recipient pays" system. That is, the cost of each message is borne almost entirely by the message's recipient (or the recipient's network provider), rather than by the sender. This historical quirk has had considerable benefits, making it possible to send both individual one-to-one messages and multi-recipient mail cheaply and easily to willing recipients.

Of course, it has also had the unfortunate effect of making it equally cheap and easy to send vast amounts of unsolicited mail to increasingly overwhelmed recipients. Unsolicited bulk e-mail (UBE), informally known as *spam*, has become the scourge of the Internet.

Many observers have noted that the amount of postal mail we receive is throttled by the cost to the sender of each piece. Bulk postal mail costs about 50 cents per piece, counting printing and mailing as well as postage, giving mailers an incentive to send mail only to recipients likely to be interested. If senders had to pay a similar amount for each piece of e-mail, wouldn't it solve the spam problem? Unfortunately, we don't think e-postage can work.

E-postage scenarios

Various e-postage proposals have different details, but most of them follow the same general plan. Each sender buys a supply of stamps, coded tokens created by a bank, that can be sent along in the headers of a message. The recipient mail system checks for the presence of a stamp, verifies that it's real (not a simple task, as we shall see), and accepts mail with valid stamps. The recipient or the recipient's mail system collects the value of the stamps on incoming mail.

Taughannock Networks provides strategic consulting, design, and prototype implementation on e-mail and Internet infrastructure. +1 646 570 1224 or info@taugh.com

Most proposals have some provision for waiving or refunding postage on some mail, for mail from regular correspondents, non-commercial mailing lists, and other sources of mail that the recipient welcomes. In some models, a recipient can choose to accept mail without e-postage from senders on a *white list* of known correspondents. In others, all mail must bear postage, but the recipient can refund the postage to the sender.

In effect, e-postage acts as a sort of "reputation system", in which a sender who is unknown to the recipient offers a stamp as an indication of good faith. There are a lot of other reputation systems ranging from DNS blacklists such as the MAPS RBL to the web-of-trust signatures in Pretty Good Privacy. Any e-postage system that holds itself out as a reputation system needs to show that it's a better alternative than other reputation systems.

Creating electronic stamps

All schemes (except for hashcash, discussed later) require a *micropayment* system to handle the exchange of value for e-postage stamps. It's instructive to look at paper postage stamps and see how they do and do not serve as a model for e-postage. For paper mail, there is a monopoly post office in each country from which all mailers buy stamps. The post office inspects each letter at the time it is mailed to ensure that it bears adequate postage, cancels the stamp, and sends it along through the mail system. All mail within the system is presumed to have adequate postage. International mail has to be transferred from one post office to another, with the hand-offs negotiated either bilaterally or through the Universal Postal Union in Berne, but mail handed off is again presumed already to have adequate postage. The 200 or so national post offices pay each other monthly settlements based on the relative volumes of mail in each direction. Until 2000, there was one central clearinghouse for all post offices but now they're moving to a more complex system due to the failure of some post offices to pay what they owe.

How analogous is this to e-mail? Other than the problem of deadbeats, not very. Fifteen years ago, closed e-mail services such as MCI Mail and Compuserve worked on the postal model, charging a fee for every message introduced. This model collapsed when the people building the Internet built an e-mail system where mail messages were too cheap to meter. As the closed systems connected to the Net, per message charges disappeared. On today's Internet, nearly every network runs its own e-mail post office, from the largest (AOL, Hotmail/MSN, and Yahoo) down to tiny businesses and individuals' systems with only a handful of users. It is a triumph of the Internet's design that these hundreds of thousands of separate mail systems all inter-operate. But most mail systems are strangers to each other, and any particular pair

of servers will rarely exchange more than a trickle of mail. This means that setting up direct agreements between each sending and recipient mail systems is impractical, so they need a mutually trusted intermediary, that is, a bank.

It's important to realize that unlike paper mail, the postage is paid to the ultimate recipient or the recipient's ISP, not to a third-party post office, perhaps with a cut taken by the bank. This has the benefit of potentially reimbursing the recipient for handling the mail (remember that the recipient bears the bulk of the cost), but the disadvantage that it gives recipients an incentive to maximize the amount of incoming postage they collect, possibly by unscrupulous means.

Banks and micropayments

E-postage is an example of micropayments, which we loosely define as payments individually too small for conventional payment systems like credit cards or bank transfers. There are two general approaches to any kind of payments: *book entry* payments in which every user has an account with the bank, and payments are made by the payor telling the bank to move money to the recipient's account, like depositing a check, or *bearer* payments where the payor directly gives the payee a token which the payee can later redeem at the bank.

Bearer payments are much faster than book entry, since the bank doesn't have to be involved in every transaction, but present greater problems of fraud. It's not hard for a bank to create electronic stamps and sign them with a digital signature that any user can check against the bank's published key. (In the literature, these are usually called coins, but the application here is postage, so we'll refer to them as stamps.) Since a sender can send the same valid stamp to many recipients, a recipient who gets a stamp from an unknown sender needs to check to see if it has already been used, by asking the issuing bank. Since so much incoming mail is spam (more than half now), and assuming that most spam will have forged postage, in practice recipients will have to check with the issuing bank for all incoming stamps.

Since the Internet goes all over the world, we can expect stamps to be issued a large number of banks located all over the world, with mail often arriving from a sender unknown to the recipient, bearing stamps issued by a bank that the recipient doesn't know either. Most likely the majority of banks will be competently run, but some won't, deliberately or inadvertently issuing stamps that they can't later cash. When a customer presents a check on an unknown foreign bank to a U.S. bank, the usual procedure is to send the item for collection, wait a month to find out whether the check was good, and charge a $20 fee for the extra handling. Usable international e-postage will need a system that lets recipients rapidly decide whether they're willing to accept stamps from unknown far-away banks. We can imagine some possibilities,

such as various forms of deposit insurance, organizations that will vouch for the validity of banks' stamps and cash them if the banks can't, but this adds yet another layer of complexity and cost to any micropayment system.

The micropayment infrastructure

Knowing that each mail delivery will need a to validate the stamp with a bank, we can estimate of the size of the transaction system needed. The largest mail systems, AOL and Hotmail, each report dealing with upwards of a billion messages a day. We use 100 billion messages a day as a conservative estimate of the number of daily deliveries in the U.S. By comparison, there are about 100 million credit card transactions a day.

This means that widely deployed e-postage will involve a thousand times as many transactions as the entire credit card system. Even assuming that the transactions are a lot simpler than a credit card transaction, say 1/10 as hard, e-postage would still need a system 100 times the size of the credit card system. The credit card system took many billion dollars of investment and four decades to build. No micropayment system that's even large enough to serve as a prototype has yet been built. One of the largest deployed micropayment systems, e-gold, recently reported only 50,000 transactions a day.

Even though the number of transactions would be enormous, the total amount of money involved would not be. If a stamp costs a penny, which is on the high side of proposed prices, and 10% of the total stamps presented are real (the other 90% being spam), and that the bank's cut on each stamp is 10%, the bank has only 1/10 cent to spend to cancel each real stamp and 1/100 cent to reject a fake stamp. Even granting that computers are cheap and these are very simple transactions, this still strikes us as an unrealistically low budget for a transaction system where each error will cost someone real money.

As a result, we conclude that: **Creating a transaction system large enough for e-postage would be prohibitively expensive.**

Hashcash

Hashcash is an alternative form of e-postage that uses computer CPU time rather than real money. When a mail system receives a message from an unknown sender, it poses a complex computational problem to the sender's computer, and won't deliver the message until the sender's computer provides the answer. The idea is to pose problems that take a few seconds to solve, so they wouldn't delay mail from individual senders much, but would be impossibly slow for mail sent in bulk.

Hashcash was an elegant idea when it was first proposed by Dwork and Naor at IBM in 1992 and it's still elegant now. Unfortunately, it also suffers from technical and social problems. The technical problems are that some computers are a lot faster than others, and that on the current Internet, spammers have vast numbers of other people's computers at their disposal. High volume senders use high-end servers with multiple Xeon CPUs that run at 3000 MHz, while the archetypical grandmother exchanging mail with friends and relatives can get by perfectly well with an old PC with a 100 MHz 486. A problem that takes a second on the Xeons would take several minutes on the 486, whereas one that takes a second on the 486 would take 10 milliseconds on the Xeons. There's no way to size a problem to be of appropriate difficulty for the wide range of computers that people use for mail. (Some schemes attempt to use memory bandwidth rather than CPU time, but memory speeds vary almost as much as CPU speeds.) Furthermore, using viruses and worms, spammers have vast arrays of hijacked "zombie" computers at their disposal. Blacklist maintainers report adding 10,000 newly hijacked computers per day to their blacklists. Spammers are already using zombies to send the majority of spam, and they could easily program the zombies to solve hashcash problems. Even zombies that can't send mail due to blacklisting or ISP firewalls can be used as compute servers to solve hashcash problems for spam sent from elsewhere. No legitimate mailer has anything like 10,000 computers dedicated to sending mail, much less 10,000 added computers a day.

The social problem, shared with monetary e-postage systems, is that hashcash is impossibly intrusive unless users use a whitelist to skip the hashcash for all of their regular correspondents, but hostile senders can forge the address of someone on the whitelist. We address this issue in more detail in the next section.

Postage and identity games

Spammers have consistently manipulated and gamed the e-mail system for their own ends, forging origin information to evade responsibility, and appropriating innocent parties' equipment via open relays, proxies, and deliberately compromised hosts (known as Trojan horses), both to hide the origin of their spam and to pump it out faster. Many spammers also engage in plain old financial fraud with stolen credit card numbers and the like. What would they do with e-postage?

We don't claim to have any great insight into the criminal mind, but we can immediately see several varieties of e-postage scam. One class of fraud lets spammers send mail without paying the postage, using missing or fake stamps, or by charging the postage to someone else. If successful, this would make e-postage ineffective. Since e-postage is collected by the recipient, thereby making mail valuable to the recipi-

ent, a second class of fraud would collect postage from unwilling senders, either by tricking people into sending mail to strangers under false pretenses, or by impersonating someone to whom they do want to send mail. If successful, these frauds would make e-postage actively dangerous.

- To send mail without paying postage, one might send spam with fake stamps hoping recipients won't check them, send mail with forged return addresses that are on recipients' whitelists, send a little innocent mail to get recipients to whitelist them followed by a lot of spam, set up a fake bank that deliberately issues uncashable stamps, or trick a legitimate bank into issuing postage without paying for it.
- To charge e-postage to third parties. one might sneak spam into other people's mailing lists, or use a virus or Trojan horse software that sends spam from the third party's computer.
- To collect postage from unwitting senders, one might seed chain letters of the "Bill Gates will pay you $20 if you send mail to this address" variety, or use the proven "click here to get free porn" web sites.

All of these tricks lead to administrative and legal problems. If someone plants a virus on your machine that sends out spam, who pays the postage? If the answer isn't "you do", who decides whether to waive the postage, and how do they tell a genuine virus victim from a spammer who planted a virus on his own system? If, on the other hand, users do have to pay for any mail that viruses send, how many users would be willing to accept the unknown extra costs of having an e-mail account?

Address forgery is rampant in spam now both to defeat whitelists and to hide the spam's origin. Although cryptographic schemes such as PGP and S/MIME that authenticate senders have existed for many years, almost nobody uses them. A few current signature proposals seem likely to be deployed, but they are in effect shared whitelists for high-volume senders, and no more easily extended to individual users than PGP and S/MIME are, and even a signature that identifies an individual user is easily stolen if the user's computer is hijacked.

No doubt it would be possible to come up with a set of laws and procedures and tribunals to deal with all the scams, and rating or discount services to keep track of all the issuing banks of varying reliability, but there is no reason to assume that the resulting situation would be any less expensive or more satisfactory than what we have now. Hence we conclude: **The true financial, administrative, and social costs of e-postage are completely unknown.**

Users hate micropayments

Finally, users of all kinds of communication systems have shown over and over again that they prefer flat rate to metered service, even if the flat rate service costs more. Andrew Odlyzko has published a seminal series of papers looking at the history of pricing of mail, telephone, and other communication media including the Internet, and has found that they all consistently move from per-message or per-minute pricing to flat rate. In his recent paper *The Case Against Micropayments* he argues that for this and other reasons, micropayments are unlikely to succeed except in small niches where they can piggy-back on a payment scheme that already exists, such as mass-transit smart cards.

One of the reasons that e-mail has been so popular is that it's unmetered, and you don't have to literally or figuratively hunt for a stamp each time you send a message. It's hard to see users voluntarily moving backward to the metered systems they abandoned a decade ago, so we conclude: **Users will strenuously resist using e-postage.**

Conclusion: is there any hope for e-postage?

For all the reasons described above, we see no likelihood of e-postage deployed broadly across the Internet for general e-mail.

We do see some niche applications. For mail and mail-like services that are expensive to provide, such as e-mail to fax or paper mail gateways, or mail to satellite phone terminals, users set up accounts and pay per-message now. That's not likely to change, but it's not a big growth market.

We also see Reputation Purchase Systems (RPS) for bulk mail, in which mailers pay ISPs for guaranteed or preferred delivery of mail, which the ISPs might pass along as reduced prices to its users. We do see this as a growth market, initially with direct bilateral agreements between advertisers and ISPs. There may well also be room for intermediaries, negotiating blanket agreements with ISPs, and handling complaints about the mail. Ironport's Bonded Sender program is an early example of such a RPS intermediary. It's hard to see RPS as more than a niche, though. After a while, a mailer either will have earned a good reputation, in which case it will be able to get whitelisted without paying, or it'll have a bad reputation in which case ISPs will reject its mail regardless of offers to pay, since no plausible per-message payment could come close to the cost of handling a customer spam complaint.

Further Reading

Cynthia Dwork and Moni Naor, *Pricing via Processing or Combatting Junk Mail*, 1999. The original hashcash proposal.
http://research.microsoft.com/research/sv/PennyBlack/junk1.pdf

Andrew Odlyzko, *The Case Against Micropayments*
http://www.dtc.umn.edu/~odlyzko/doc/case.against.micropayments.pdf

Andrew Odlyzko, *The history of communications and its implications for the Internet*
http://www.dtc.umn.edu/~odlyzko/doc/history.communications0.pdf

Revision date: 2020/04/25 21:40:43