Stopping Outgoing Spam

Joshua Goodman [*]
Microsoft Research
One Microsoft Way
Redmond, WA 98052
joshuago@microsoft.com

Robert Rounthwaite
Microsoft Anti-Spam Technology and Strategy
Group
One Microsoft Way
Redmond, WA 98052
robertro@microsoft.com

ABSTRACT

We analyze the problem of preventing outgoing spam. We show that some conventional techniques for limiting outgoing spam are likely to be ineffective. We show that while imposing per message costs would work, less annoying techniques also work. In particular, it is only necessary that the average cost to the spammer over the lifetime of an account exceed his profits, meaning that not every message need be challenged. We develop three techniques, one based on additional HIP challenges, one based on computational challenges, and one based on paid subscriptions. Each system is designed to impose minimal costs on legitimate users, while being too costly for spammers. We also show that maximizing complaint rates is a key factor, and suggest new standards to encourage high complaint rates.

Categories and Subject Descriptors

H.4.3 [**Information Systems Applications**]: Communications—*Electronic Mail*

General Terms

Security

Keywords

Junk email, spam

1. INTRODUCTION

1.1 The Outbound Spam Problem

Spam has become an increasingly large problem. For instance, in a recent Infoworld [8] article, over 40% of respondents listed spam as the worst IT problem of the past

---

[*]This document represents the personal views of the authors, which are not necessarily the same as the views of our employer, Microsoft Corporation.

year. The vast majority of research on spam has focused on helping the recipients of spam avoid receiving it [6, 13, 16] (to cite a few). We instead focus on helping Email Service Providers (ESPs) prevent their users from sending spam. ESPs include most consumer ISPs (e.g. AOL and MSN), free email systems (e.g. Hotmail and Yahoo), universities, etc. Stopping outbound spam at ESPs will not prevent all outgoing spam, as many spammers own direct internet connectivity and cannot be forced to adopt these techniques, but it will stop a substantial source of spam. ESPs want to stop outgoing spam to lessen the load on their own servers, to prevent their systems from being blocked from sending mail, and to avoid bad publicity.

One now well known technique to help stop outgoing spam, first used at Yahoo and now used at Hotmail, is to require users to solve a simple test (a so-called reverse Turing test) to prove that they are human. A spammer can amortize this account creation cost over perhaps 1000 messages: this technique imposes only a cost of about .002 cents per message on spammers, while spammers often earn or charge .01 cents per message or more. One approach to counter this is to use low daily volume limits on outbound mail per account. Unfortunately, we will show that this often has almost no effect at all on the cost to spammers. An alternative technique is to impose some cost for every message sent, i.e. require senders to prove they are human for every message, or every 10'th or 100'th message. This would certainly work, but would be very annoying.

We will show that only moderately intrusive techniques can work as well as requiring a cost for every message forever. In particular, users are required to pay some cost, such as proving they are human, or having their computers solve a difficult computation, for, say, every 100 messages, but only for the first 1000 messages they send. After they have performed the test or solved the computation 10 times, they need never solve it again. We will show that surprisingly, this works about as well as imposing the cost forever. For legitimate users who use the system for an extended period, this cost can be amortized over many thousands of messages, making the cost per message to legitimate users very small. Moreover, we describe a system in which users who wish to exceed their daily limit can solve additional tests, and have their limit raised on that day. After solving a certain number of puzzles, their limit is permanently raised.

1.2 Overview

In the next section, we will explain why spammers like to abuse ESPs, especially free email services, and then we will

explain in more detail why ESPs have incentives to prevent their abuse – even though stopping outbound spam does not help the ESP receive less spam, it results in other important savings, and is critical for allowing mail from the ESP to be trusted.

Next, we we will briefly discuss previous work on stopping outbound spam, of which there has been only a little. Unfortunately, we are not aware of any previous work being published in almost any sense; it has simply been implemented in shipping products. Also, to thwart spammers from understanding and working around the systems, some of the systems have been intentionally kept secret.

We then discuss the various techniques for stopping outbound spam in more detail, including the various ways costs can be imposed (reverse Turing tests, computation, etc.) as well as the different ways of imposing these costs (account signup costs, per message costs, etc.) We introduce our novel technique with a limited number of initial costs and show that it is a sufficient disincentive.

Finally, we discuss complaint mechanisms, and advocate a standardized complaint mechanism that helps ESPs quickly learn about accounts being abused by spammers, and helps verify that the complaints are legitimate. We also discuss the problem of list cleaning, in which spammers record the complaints about their account, and remove those recipients from their lists, and a new technique for avoiding this.

We conclude that our system of imposing initial costs can be an effective deterrent to spammers, and only a bit annoying to users, and that by combining it with standardized complaint reporting techniques, the initial costs can be lowered even further, making them quite acceptable.

2. ESPS AND SPAMMERS: A PARASITIC RELATIONSHIP

We begin by describing the relationship between spammers and ESPs, first explaining the advantages to a spammer of using an ESP to send mail, rather than sending the mail directly. Then, we explain why ESPs care about stopping outbound spam, even when none of this spam is received by their own customers.

2.1 Why Spammers Use ESPs

There are many good reasons that spammers use ESPs, rather than sending spam directly. Doing so lets the spammer avoid blackhole lists, take advantage of any safelists the ESP is on, avoid his own ISP's limits on mail volume, make the spammer more difficult to trace, and multiply his bandwidth.

Many email systems today deploy blackhole lists – lists of IP addresses from which they do not accept mail. These include known sources of spam, open proxies, and sometimes dialup or DSL lines. A spammer may find himself on such a list, but still be able to send mail indirectly through an ESP such as Hotmail or Yahoo. ESPs could reject connections from some of these, but certainly not from, say, dialup and DSL lines.

In addition, there are now several proposals and companies organized around creating lists of known good senders. Efforts such as IronPort's BondedSender program and ePrivacy Group's Trusted Sender program, as well as a recent announcement from a group of large ISPs [10] are all moving in this direction. ESPs would very much like to be on such

lists, ensuring that their users' mail is delivered. We envision a time when such lists are widely used, and not being on such a list means a presumption that you are a spammer. In such a world, spammers will have a huge incentive to send mail from ESPs on the lists. In other words, for these lists to be truly successful, and for the ESPs to satisfy their customers, outbound spam must be stopped.

Some consumer ISPs route any data on port 25 (the SMTP email port) to special relay servers, where it can be examined, rate limited, etc. This blocks spammers from using these ISPs to send spam. However, users of these ISPs can still connect to the internet, and thus connect to ESPs like Hotmail, from which they can spam.

In addition, sending mail through an ESP may make it more difficult to trace the spammer, since it adds one more level of indirection. This means it may take longer to terminate the spammer's account at his ISP.

Spammers also may get bandwidth advantages through abusing ESPs. In some cases, they can send one message to the ESP, with many recipients listed; the ESP will then forward it to the many different receiving domains. In effect, the ESP multiplies the spammer's bandwidth.

Given all of these advantages to abusing ESPs, minor impediments will not be enough to stop spammers from using them. In the long run, it is not enough to make using ESPs inconvenient: they must be made unprofitable.

2.2 Why ESPs Want to Stop Outgoing Spam

It is not surprising that most previous research on spam has looked at how email receivers can avoid receiving spam: they are the ones who pay the cost of getting spam, while senders pay almost no cost at all. Other than altruism, it may not be obvious why an ESP would want to stop outgoing spam. In fact, however, there are several very good reasons.

1) Spammers use the servers at the ESP. While each individual message costs a small fraction of a cent to send, the millions or billions of messages that spammers send add up quickly.

2) A larger factor is probably the cost of responding to complaints, and terminating accounts. In particular, many ESPs (Hotmail, Yahoo, etc.) host advertising supported free email systems. Spammers who use automated means to create accounts do not read advertising, so the revenue is zero, and the small cost of responding to a complaint and terminating the account generates a loss. If thousands or millions of such accounts need to be terminated, the costs become large.

3) The ESP risks being put on a blackhole list through systems such as the MAPS RBL [11]. Blackhole lists are lists of IP addresses that many mail systems do not accept mail from. To give one of the most extreme examples, some administrators actually block all mail from China and Korea [14]: if entire countries are blocked, then even large ISPs are at risk of being blocked if they do not control their outgoing spam. Similarly, programs to create lists of known good senders are gaining momentum. ESPs can only be included in such lists if they can effectively prevent outbound spam.

4) Assuming the spammer cannot or does not use a falsified email address, recipients will receive large amounts of spam obviously from the ESP, causing substantial damage to the ESP's reputation.

In short, while most ESPs have more incentive to stop

inbound spam than outbound, stopping outbound spam is still an important goal.

3. PREVIOUS WORK

There is little documented previous work on stopping Outbound spam. Most previous work on spam has focused on stopping inbound spam. Several ideas from stopping inbound spam can be adapted to stopping outbound spam, so we describe them in more detail here.

One of the earliest attempts at stopping inbound spam was to impose a computational cost on spammers [6, 9]. In the computational approach, spammers are in some way required to prove that they have performed a computation. For instance, they could be sent a challenge if their mail does not include computation, or all mail without attached computation could be rejected. The computation is chosen to be easy to verify, and time consuming to compute. Typically, for stopping inbound spam, the computation required is a function of various header fields, such as From, To, Subject and Date. One possible computation is to combine together these fields, and then require the sender to find a number, which, when prepended to this combined string has a hash whose first k bits are 0. (This is roughly how the Camram [9] system works for inbound spam, as inspired by Hashcash [3].) k can be chosen so that a certain amount of time is required, on average, to find such a hash value. If, for instance, we choose k such that the time required on average is 1 minute, and assuming it costs roughly $1000 to purchase a computer, maintain it, power it, etc. for one year, then the cost of solving such a puzzle is approximately $100000/(365 \times 24 \times 60) = 0.2$ cents. Another option is to use memory bound puzzles [1, 5], which are more robust to variations in CPU speed. For both CPU and memory-bound puzzles, most legitimate users have many unused cycles on their computers, and there is effectively zero incremental cost to performing such a computation (which can be done in the background at low priority, where it will not be noticed), while a spammer trying to send millions of messages must actually purchase the computers (or otherwise acquire the cycles, such as by stealing them – a problem we will discuss later.) Later we will describe how this approach can be leveraged for stopping outbound spam as well.

Another very interesting technique for stopping inbound spam, and one that has already been partially used for stopping outbound spam, is Human Interactive Proofs (HIPs) [13] (also known as a Reverse Turing Test, or as a CAPTCHA [4]). In this technique, mail from a previously unknown sender is challenged. The sender is required to solve a puzzle that would be difficult or impossible for a computer, but not too hard for a human. The typical puzzle is text that has been obscured in some way, so that it is too difficult for most existing OCR software. An alternative puzzle is a spoken list of letters with added noise, reverberation, etc: this can be used by the visually impaired.

There have been very few previous attempts to stop outbound spam. The best known is an adaptation of the HIP idea. On many free email systems, a HIP must be solved to create an account. This effectively imposes a cost to create the account (a spammer must use his own valuable time, or pay for someone else's time, or provide valuable services in exchange for solving the HIP.) Later we will analyze this technique and show that it is an insufficient disincentive for stopping spam.

Other than small account creation costs, the only other technique that appears to have been used is to terminate accounts as complaints come in, and perhaps to track down the spammer for legal prosecution. An article in the Wall Street Journal [2] describes one example of this, Earthlink's attempts to track down and stop a known spammer. Earthlink users are allowed to send as many messages as they want per day, until account termination. The Earthlink spammer used new stolen identities regularly, a total of 343, and sent almost a billion spams, before being caught. The main techniques Earthlink used to stop him were to terminate his account in response to complaints – something they could not do quickly enough to be an effective deterrent; and to try to manually determine a common pattern in his accounts (the accounts used similar passwords, and, for a while, used the same phone number.) Besides being ineffective, the techniques were also extremely expensive for Earthlink, using 20-30 hours per week of their employees' time for a single spammer.

4. SOLUTIONS TO OUTBOUND SPAM

In this section, we discuss possible solutions to outbound spam, focusing first on solutions for free ESPs such as Hotmail and Yahoo, and later on solutions for paid ESPs. We will use an economic framework for our analysis. Our goal is to determine, for a specific system, what the cost per message will be for the spammer. We will show two somewhat surprising results. First, the number of messages per day, often considered a critical factor in stopping outbound spam, typically plays a minor role in the cost. Second, the number of messages until a complaint is filed, a number that receives little attention, plays a critical role. We will also analyze existing solutions (a one time account creation cost of some sort) and show that current creation costs are too low. An alternative would be to require senders to pay a cost for every message they send, which would work, but would be unnecessarily burdensome. We will show that a system that periodically imposes a cost initially, but then allows users to send for free, can work just as well as a system which imposes costs forever. Finally, we will show that such a system can also be used to allow users to increase their daily limits, while still remaining a sufficient disincentive.

4.1 Account Creation Costs

There are several different ways to "pay" for an email account. In the case of free accounts, there is often no cost at all. More recently, many free ESPs have started requiring users to solve a Human Interactive Proof (HIP) [13, 4] in order to receive an account. While there is no monetary cost to users, spammers must spend their own time, presumably worth something, or pay or induce others to spend time solving these puzzles, also presumably at some cost. In some cases users pay actual money, perhaps $20 per year, such as for Hotmail premium accounts. The most expensive common account type is an email account included with an ISP, such as MSN or AOL, in which there is a fixed monthly charge, perhaps $20 per month. For now, we simplify these three scenarios by simply assuming that there is some cost to create an account. Later, we will consider more complex scenarios in which the recurring fee is modeled.

Consider the following scenario: a user creates an account on an ESP by paying some fixed cost, C. He can then send some number of messages per day, D. (We always measure

messages by number of recipients; e.g. a message sent to 3 recipients counts as 3 messages.) We assume that recipients complain about any spam messages with probability p. The account is terminated when some number of complaints is reached. For simplicity, we assume that one complaint is sufficient to terminate an account. Initially, let us assume that users complain the moment that a spam message is received, and the account is immediately terminated. Later, we will consider a slightly more complex model with a lag of L days between when spam is sent, and when the account is terminated (comprising the time for the recipient to read and respond to the mail, and the time for the complaint and termination process.)

Given this framework, what will the cost to a spammer be to send a single message? Initially, let us consider the simplified case of instant account termination, in which an account is terminated as soon as a complaint is filed. After sending each message, the account will be terminated with probability p, so on average, a spammer can send $1/p$ messages before account termination. The cost to create the account was C. The cost per message is thus Cp. Notice that in this simple scenario, the number of messages per day does not even enter into the equation: lowering this number has no effect. On the other hand, the probability of a user complaint is a critical factor: making it easier for users to complain is one of the two most important factors in stopping outbound spam. Later, in Section 5, we will discuss how industry can substantially increase this probability.

We can also now see that some initial attempts to stop outbound spam were unlikely to succeed. The earliest attempts to stop outbound spam took systems with essentially zero signup cost $C = 0$ and added a HIP test, raising the cost C. How much was the cost raised? In the next section, we will discuss likely actual values for C, p, D, and L. For now, however, we note that $C = 2$ cents is an upper bound on the cost of solving a signup HIP; $p = 1/1000$ is a good approximation to the probability of a complaint, leading to $Cp = 0.002$. However, the lowest price spammers charge that we have seen is .0025 cents per message, so even the cheapest spammer might be able to make a profit on such systems, and prices of 0.01 cents or more are common. Given this analysis, it is unsurprising that signup HIPs alone have failed to stop spammers from using free email systems for outbound spam. (In Section 4.2 we will describe where these numbers come from.)

Now, let us consider the more complex model in which accounts are not terminated instantly, but rather only after users complain, and complaints are responded to, which combined takes time L. Let us assume that the spammer each day sends D messages all at once, i.e. there is no chance of the spammer sending $< D$ messages, and a fast complaint causing account termination. This is the optimal strategy for a spammer. Given these assumptions, a spammer can always send minimally $D \times L$ messages before his account is terminated (we're assuming a deterministic time L rather than a distribution over complaint/termination times.) Each day after the $L'th$ day, his account is terminated if any of the messages resulted in a complaint. The chance of a complaint on a given message is p, so the chance of a complaint on a given day is $1 - (1-p)^D$. We will call this quantity q. Notice that when D is small (compared to the number of messages a spammer can send on average before a complaint, which is $1/p$) $q \approx pD$. Thus, the expected number of messages sent before account termination, for small D is

$$LD + D/q \approx LD + D/(pD) = LD + 1/p$$

If L and D are both small compared to $1/p$, then

$$LD + 1/p \approx 1/p$$

In other words, for small D (compared to $1/p$), the number of spams that can be sent does not approach 0 – it approaches $1/p$, and the overall cost per message does not depend on the number of messages per day!

To some people, this is a somewhat surprising result. After all, one might expect that by lowering the number of messages that an account can send per day, one might have an impact on the cost to spammers – and yet there is often no such impact. Why is this? Because the cost to a spammer is primarily based on how many messages he can send before the account is terminated, and the number of messages per day often has only a small impact on this number. As D is reduced, a spammer must create more accounts to send the same amount of spam, but each account lasts longer.

Another way to look at this is that when a spammer receives an account, they have not received a license to send an infinite amount of mail; instead, they can send mail until they are caught, which will on average be $1/p$ messages. We need simply ensure that the average cost to the spammer makes this unprofitable.

In actual practice, there may be some effect from lowering D. Spammers with limited resources may not be able to afford to pay people to create the larger number of accounts, so there may be an effect. Also, spammers may be unwilling to risk spending more resources to create more accounts (since there is always a chance a commonality between the accounts will be discovered, and they will all be terminated, or that the ESP will find other ways to block outbound spam.)

On the other hand, there is a cost to an ESP of maintaining each of these accounts. (They may, for instance, receive spam, and disk space needs to be allocated for any mail they receive.) Encouraging spammers to create more accounts, each of which lasts longer until termination, costs an ESP money too. So, the primary effect on an ESP of reducing D much below $1/p$ may be to simply raise their own costs, with only a small effect on the amount of outbound spam.

This analysis is for small values of D, e.g. 100, typical of a free ESP like Hotmail. For large values of D, e.g. thousands, typical of a paid service, D does indeed play a large role.

4.2 Estimates of Actual Values

It is very difficult to get data on actual values for any of these parameters, but it is important to at least have rough estimates to understand what techniques are likely to work in practice, and which are likely to be impractical. Here are our best estimates for each parameter.

C (cost to create a new account.) With no signup costs, this will be near 0. With the requirement of a signup HIP, the cost goes up. I personally can solve 8 HIPs per minute, extrapolated to 480 HIPs per hour. (We assume that the other parts of account creation are completely automated so that only the HIP portion of signup takes time.) Assuming very rough US labor rates of \$10/hour, we get 1000 cents / 480 HIPs = 2 cents per HIP. If HIP solving can be moved

to countries with lower labor costs, the cost may be reduced substantially. (In very low cost countries, the predominant cost may become the computer and internet access, rather than labor.)[1]

p (probability that an individual message generates a complaint) This is an especially difficult figure to find. The best estimate we have is from MSN TV, who very kindly gave us estimates of their complaint rates (for inbound spam) and amount of inbound spam. (Note that MSN TV does not generate outbound spam – because of their TV-based architecture, it is very difficult to do so.) Based on their data, we estimate complaint probabilities of 1/800 to 1/900. Clearly this is only a rough approximation. In addition, note that on some systems, spammers can alter the From information to be another domain, making it less likely that the true ESP will receive a complaint, while on others, this is not possible. Note also that we have generally assumed accounts are terminated after the first complaint. If, say, three complaints were required for account termination, this would be roughly equivalent to a p value 1/3 as large. We use 1/1000 as our complaint rate estimate throughout the text.

L (the lag time between sending mail, and a complaint being registered.) Unfortunately, we do not have complaint timing data. However, we have excellent data on a reasonably close proxy. We have been surveying Hotmail users to classify mail as spam or good, using a random sample. As soon as the mail comes in, for a certain percentage, we send them a request to classify the mail. We know exactly the time from when the mail was received to the time at which users classify this mail. The mean time of classification is 2 days 9 hours; the median is only 1 day 5 hours. This indicates the amount of time it takes users to read and respond to their mail. Based on these numbers, we will use a value of 2 days between complaints in our examples.

D (the number of messages that can be sent per day) Hotmail currently has a 100 message per day limit [15]. The Yahoo maximum is not clear – Yahoo has an algorithm that is not public that occasionally requests additional HIPs to be solved and has additional limits. In a quick test, a newly created Yahoo account could send 150 messages immediately. Then an additional HIP needed to be solved. After 220 messages were sent, a hard hourly limit was reached, with no more mail allowed, although apparently this limit increases over time. It does not appear that Yahoo is using the algorithm we describe here, although we are not sure what their algorithm is. After a break, another 250 messages could be sent without additional HIPs. We will use 100 as our example for this value.

Finally, the most important question is how high must the cost be, in order to be a sufficient disincentive to spammers? If we can raise the cost of spamming through legitimate ESPs above the cost of spamming in other ways (purchasing a domain, setting up mail servers, etc.) then spammers will not typically spam through ESPs. If we can reach a higher bar, and raise the cost of spamming through legiti-

mate ESPs above the profit that spammers expect to make, per message, then we can go a step forward: ESPs with good outbound spam stopping measures can put each other on safelists, and allow all mail from such ESPs to reach their users, without the danger of a spam filter triggering. This eliminates the so-called false positive problem, one of the largest problems with most spam filtering technologies. In Appendix A we analyze various media reports of per message costs and profits. We find that the costs and profits range widely, from about .0025 cents to about .05 cents.

4.3 Initial Challenging

From this discussion, it should be clear that a single HIP leads to too small of a cost. A natural inclination would be to try to impose a cost on every message. This approach has been suggested for incoming spam [6, 13, 9]. We will first give a trivial analysis showing that this approach works fine for outbound spam, but we will then show a much less burdensome approach that will also work. In particular, in our less burdensome approach, we initially require payment (such as money, computation, or solving a HIP) for every n messages, but eventually stop charging for any additional messages, until a complaint is received. We show that this can be a sufficient deterrent to spam. We also describe a system in which users can pay to increase their daily limit.

First, imagine a system in which users pay a cost C for every n messages. (The cost must be paid before the first message in the batch of n is sent.) It should be clear that such a system imposes a minimal cost of C/n per message, and as long as this cost is more than the profit the spammer can make, will be prohibitive.

This cost can be charged in various ways. For instance, it could be paid with actual money, perhaps using a micropayment system. Alternatively, users could be asked to solve a HIP (as suggested for incoming spam by Naor [13]) . Spammers wishing to send millions of messages will need to pay someone to solve the HIP, costing them actual money. Alternatively, the cost could be paid with computation (as suggested for incoming spam [6, 9] and discussed in Section 3). As we showed in Section 3, a one minute puzzle costs a spammer about .2 cents worth of computer time, and a 1 hour puzzle costs him about 11 cents. For computational costs, most legitimate users have many unused cycles on their computers, and there is effectively zero incremental cost to performing such a computation (which can be done in the background at low priority), while a spammer trying to send millions of messages must actually purchase the computers (or otherwise acquire the cycles.) In this paper, we'll mostly be agnostic to the actual form of the cost – money, computation, or HIP, and simply describe all of these in terms of their equivalent monetary cost.

Now, consider an alternative system for stopping outbound spam. In this system, users are charged a cost of C for every n messages, but they are charged a maximum of k times; after that all messages are free, although still subject to a daily limit of D messages per day, and, the account is suspended if a complaint is received. One might think that such a system is an invitation to disaster: spammers will send nk good messages, and then start spamming, or spammers will spam from the beginning, and a few will get lucky (no complaints) and be able to keep spamming at no cost.

Let us start by determining the spammers optimal strat-

---

[1]Note that it has been widely rumored that some spammers provide free porn in exchange for solving HIPs. We have been unable to find any examples of this attack in the wild. Even if the attack does exist, given the availability of free porn, such spammers must spend money to attract people to their site, and to provide them with content that is not available elsewhere for free. It is hard to estimate what the cost to the spammer per HIP would be, but it would be at least a fraction of a cent.

egy in this system. Is there some way he can intersperse good messages and spam to do better than simply sending as much spam as he can as soon as he gets his account? No: we can show that his optimal strategy is to send as many spams as possible as quickly as possible. A sketch of the proof is as follows: consider a spammer with some schedule X of sending good mail and spam, where X is not simply sending as much spam as possible as quickly as possible. Consider schedule X' which is identical to X except at the first time at which the spammer sent good mail, or simply passed up an opportunity to send a message, the spammer instead sends spam. Consider the time t when the spammer has sent m spams with schedule X and the time t' when the spammer has sent the same number of spams m with schedule X'. Notice that t' is at least as early as t. The cost that the spammer incurred at time t' with schedule X' is no larger than the cost incurred at time t with schedule X (because the spammer has sent no more messages at time t' than at time t.) The probability of account termination by that time is also no higher, because the spammer has sent as many spams in less time. Therefore, the spammer is at least as well off with schedule X'. By an inductive proof on the time of the first good message or missed opportunity, we see that the strategy of always sending as much spam as possible as quickly as possible is at least tied for best.

Now, we will analyze the cost to a spammer of using this strategy. The analysis is much simpler if we assume that there is a fractional payment scheme, i.e. you pay per message. The spammer's cost is slightly lower if he pays C/n per message for the first kn messages than if he pays C for each set of n messages for the first k sets. Thus, this analysis lower bounds the spammer's cost.

Given this per message cost, the daily cost is DC/n. The daily number of messages is D. Assume that L (the number of days it takes to complain) is less than nk/D. (nk/D is the number of days it takes to pay for all messages.) Assume for simplicity that nk is a multiple of D and that $L \leq \frac{nk}{D}$.

Day 1: no probability of termination. Cost is DC/n.
Day 2: no probability of termination. Cost is DC/n.
...
Day $L + 1$: termination prob today = q. Cost is DC/n.
Day $L + 2$: termination prob today = q. Cost is DC/n.
...
Day nk/D: termination prob today = q. Cost is DC/n.
Day $nk/D + 1$: termination prob today = q. Cost is 0.
Day $nk/D + 2$: termination prob today = q. Cost is 0.
Day $nk/D + 3$: termination prob today = q. Cost is 0.
...
So, the expected cost is

$$\frac{LDC}{n} + \frac{(1-q)DC}{n} + \frac{(1-q)^2 DC}{n} + ... + \frac{(1-q)^{nk/D-L}DC}{n} =$$
$$\frac{\left(L + \frac{(1-q)-(1-q)^{1+nk/D-L}}{q}\right)DC}{n}$$

The expected number of messages sent is

$$LD + (1-q)D + (1-q)^2 D + (1-q)^3 D + ... =$$
$$(L + (1-q)/q)D$$

So, the expected cost per message is

$$\frac{\left(L + \frac{(1-q)-(1-q)^{1+nk/D-L}}{q}\right)DC/n}{(L + (1-q)/q)D}$$
$$= \frac{\left(L + \frac{(1-q)-(1-q)^{1+nk/D-L}}{q}\right)C/n}{L + (1-q)/q}$$
$$= \frac{(L + \frac{1-q}{q})C/n}{L + (1-q)/q} - \frac{\frac{(1-q)^{1+nk/D-L}}{q}C/n}{L + (1-q)/q}$$
$$= C/n - \frac{\frac{((1-q)^{1+nk/D-L}}{q}C/n}{L + (1-q)/q}$$

Notice that if $1 + nk/D - L$ is reasonably large, the second term approaches 0. Among other things, as the number of messages per day approaches 0, this approaches 0. Once again, lowering the messages per day has essentially no effect if the number is already reasonably small. In particular, for many reasonable values of $1 + nk/D - L$, the cost per message to a system with initial payments only is almost the same as the cost per message with payments for all messages. Consider a reasonable example: $n = 100$, $k = 10$, $D = 100$, $L = 2$, $p = 1/1000$, and $C = 2$. That is, users must solve a HIP for each 100 messages they send, up to a maximum of 10 times. They may send up to 100 messages per day, and we assume a 2 day lag between when spam is sent, and when complaints cause account termination. Plugging these variables in, we get a cost of .012 cents per message. This is only a bit below what the cost would be if we required solving a HIP for every set of 100 messages, instead of just initially. In that case, the cost would be .02 cents per message, and this system, which challenges a maximum of 10 times, is far less annoying to users. A system challenging 20 times yields a cost of .017 cents per message. One that challenges 30 times yields a cost of .019 cents. In short, systems based on challenging initially only can yield costs almost as high as those that require solving HIPs for every n messages. Note also that we are well into the range where lowering the daily maximum has almost no effect (assuming we still require a HIP for every 100 messages initially.) Changing D from 100 to 300 changes the cost per message from .0126 to .0121. In other words, with this system, we can also raise the daily volume limit.

Note that an alternative to this system would be to simply require a large initial cost from users, such as solving 10 HIPs in a row. We think the system described here, where users only occassionally solve HIPs, is likely to be far less annoying. At the very least, legitimate users are more likely to give up when confronted by a single large annoying task and simply find a different email system to use. Many legitimate users will not even send 1000 messages over the lifetime of their account.

For legitimate users who send a lot of mail, a system with initial challenging is far cheaper. In our 10 challenge example, imagine a user who sends 10,000 messages over the lifetime of the account: he pays only 20 cents/10000 = .002 cents per message, while the spammer pays .012 cents – six times more. Initial challenging keeps the cost of whatever sort low for legitimate users, and relatively high for spammers.

This system becomes far more compelling if the cost paid is a computational one. Requiring a large initial compu-

tational cost, say 2 hours (about 22 cents) before the user could send mail for the first time, would be pretty annoying. Instead, consider a system with initial challenging with 30 seconds per message (about .1 cents.) In this case, consider $D = 300$, $n = 1$, $k = 1000$ – the user must solve a 30 second challenge per message for each of the first 1000 messages. This raises the cost per message to a spammer to .06 cents per message, beyond the point of profitability.

With computational costs, users could, of course, prepay. When a user signs up for a new account, computational challenges begin running in the background on their system. The challenges can all be sent to the user at account creation time – the user need not even be online during this process, as long as it can resynchronize its results before the user sends another message. For the first 500 minutes (about 8 hours) after signup, this process runs in the background at low priority. During that 8 hour period, the user can send up to the lesser of 100 messages and however much computation he has finished by that time. In general, the user will not even realize this computation is going on, and once it is finished, he can switch machines without having to reinstall the computation-solving software (presumably a plug-in of some sort.) This is a compelling solution, leading to high costs to spammers, and low costs to users.

One problem with computation is that it might be possible to steal cycles, using, e.g. zombie machines, or even to buy cycles. How much does a zombie cost? How much of a disincentive is the risk of prosecution? It is hard to estimate the size of this problem. One partial solution is to drastically increase the amount of computation required, perhaps to 5 minutes per message, or even 30 minutes per message. This will not inconvenience most legitimate users, and may not be a profitable use of zombie machines.

Consider an alternative system using money: there is a one time charge of \$1. In this case, we can allow an even larger number of messages per day; let's say $D = 400$, and as always assume $L = 2$, $p = 1/1000$. Then we get a cost of .05 cents per message, at the cutoff of profitability. Any user who can afford a dollar can use this sytem, though presumably most would prefer a computation based system.

4.4 Increased Limits and Banking Credits

Consider a user who has paid his cost k times. Now a complaint comes in. His account is terminated. Notice that there is not much disadvantage to letting him get his account back, assuming that he simply starts over. A spammer could always create a brand new account, and is in no better position by being allowed to keep the old one, for the same price as a new one. Thus account termination from complaints should be only temporary: users simply start over on the payment schedule.

Now, consider a user who has reached his daily account limit and wishes to send more messages. He could create a new account, inconveniencing himself. We might as well simply allow him to send more messages, assuming he meets the same requirements as someone who creates a new account, namely solving additional HIPs, performing more computation, or depositing more money. When he has made his k payments, we can raise his limit permanently.

If a user with a raised quota receives a complaint, we can reduce his quota, as if he had had one account terminated. That is, we can think of the user as having multiple virtual accounts, and we treat each virtual account as if it were a single real account, in terms of quota, etc. We will call these virtual accounts "streams."

In the case of our initial payment scheme, we can use the following technique. Associated with each user's account is a set of unused "tokens" (e.g. solved HIPs, solved computation, paid money, etc.) Also associated with each user's account is a set of streams – the right to send D messages per day. When a user tries to send more messages than he currently has the right to send, either because he has hit the daily limit on all of his streams, or because all of his streams require an additional payment in our initial payment system, a token is used up. The token is used either to pay for the next payment in the initial payment system, or, if he has used up the daily limit on all streams, it is used to create a new stream. For each stream, we keep track of how many payments in the initial payment system have been made; eventually, streams pass their initial payment period and can be used without further payments, up to their daily limits. When a complaint comes in, we terminate the stream that was used to send the message in question. Clearly, the system with multiple streams is no cheaper for spammers than the simpler one in which each account has a daily limit that cannot be raised. However, users will find it more convenient, since all of their mail comes to one place, and they need not manually determine if they have an account which has not hit its daily limit, etc. Also, since this does not increase the mailbox quota for the user, it is cheaper for the ESP.

We do not expect users to understand the complex accounting used with streams. A user would see only that he occassionally receives new challenges, but would see that they come only occassionally, or when he tries to send large volumes of mail. He can also be told that these challenges will only come for a limited period of time.

4.5 Paid Accounts

Hotmail and other ESPs currently charge approximately \$20 per year for premium email services. Combining our analysis using \$1 per account (in Section 4.3) with the idea of allowing multiple streams, we could for example give such users 20 streams. Initially, they could send 400 * 20 = 8000 messages per day. Each time they receive a complaint, one stream would be deleted, and their daily limit appropriately reduced. This seems like a very reasonable system.

There are many different parameters one could use, such as $D = 800$, and \$2 per stream would also work – the key in choosing parameters is to remember that even legitimate user sometimes make mistakes, and there is a non-trivial error rate on complaints. We would not want to use a single stream with a very large D that could result in the user's entire account being suspended.

For ISP services that are also ESPs, such as AOL and MSN, assuming a charge of \$20/month, they can initially award 20 streams (of 400 messages each) and award an additional 20 streams per month. This means initially a user could send 8000 messages per day, and the amount would increase monthly.

A new small business that immediately needed a larger quota could prepay their ISP. For instance, they could pay in advance for one year, and immediately receive 240 streams, and the right to send almost 100,000 messages per day. If they spam with this right, their prepaid account may be terminated very quickly, in as little as a day.

The number of streams should probably be capped. For instance, there have been many reports of viruses and trojans infecting legitimate user's machines to send email. Alternatively, a user who has not previously sent much mail but who has had an account for a long time may decide to make money by spamming, given the low incremental cost. Capping the number of streams per account limits the damage from these problems.

5. A STANDARD FOR COMPLAINTS

While we have suggested some very viable techniques for stopping outbound spam, there are a number of reasons we want even better techniques. First, the most onerous HIP-based system we think a user would accept, 10 challenges on the first 1000 messages, still only raises the cost to .012 cents. Second, our computational approach has a cost of about .06 cents per message, which is excellent, but users of older slower machines may find it too inconvenient. Finally, we are afraid that spammers may find ways at reduced cost to get users to solve HIPs (e.g. by providing porn or letting them play games) or may be able to purchase or steal unused computation at reduced rates. Raising the cost of spamming further helps protect against all of these problems.

We have shown that the probability of a complaint is the most important factor in keeping the cost of outbound spam high. How can we increase this probability as much as possible? We suggest a standardized mechanism for complaints that makes it easy to complain, and easy to handle complaints.

Currently, complaining is a semi-tedious process for both the complainer and the sending ISP. Some users forward mail, or write a hand-crafted message. If the mail is not forwarded as an attachment, relevant headers are lost and it may be difficult or impossible for the sending ISP to know if the recipient even received the alleged message, and whether the alleged message actually originated in their system, or was forged as coming from the alleged sender. Finally, currently, almost all complaints are made by hand, by the complainant, and processed manually by the sender ESP.

Another problem is list cleansing: some spammers already attempt to determine the list of users likely to complain. They then remove these users from their lists. This can drastically lower the probability of complaints. On the other hand, we need a way to prevent recipients from receiving further mail from the sender. (In some cases, the sender's account is terminated, but in the case of senders who have paid large fees, their account is debited some amount, and they are allowed to continue sending mail.) Thus, we suggest that rather than notifying the sender that the recipient has complained, the recipient ESP notifies the spammer's domain, and adds the particular sender-receiver relationship to a block list; all mail from that sender to that receiver is put in a junk folder, or deleted. Only the sender's domain, not the individual sender, finds out about the complaint, and the recipient does not receive more mail.

Mail clients should contain a "Report Spam" or similar button or menu option. Pressing this button should cause the message in question to be sent as a MIME attachment, complete with headers, to `abuse@example.com` (where `example.com` is the sender's domain, possibly found through reverse DNS lookups, when available.) The subject line of such messages should be standardized, e.g. "Automated abuse report compliant with RFC xxx." The body of the

message should contain a human readable message (in case the particular postmaster does not support automated complaint handling), e.g. "The attached message contains unsolicited mail to the following recipient: recipient@test.com" If the complainant system supports automated blocking, and wishes to avoid list cleaning (which all systems should), the message body also contains

```
AUTOMATED-SENDER-RECIPIENT-BLOCKING: TRUE
```

In this case, the sender's ESP must not notify the sender of the complainant's identity.

Notice, that, unfortunately, our solution to list cleaning does not always work: if a spammer can acquire a domain, or inside information at an ESP, he can clean his lists. He may be able to perform list cleaning on a system he controls, and then send mail using a victim ESP, who receives few if any complaints. There does not appear to be a good way around this. We considered various information hiding strategies, but cannot get around the fact that the complaint must contain the sender's email address. Spammers who create one sending email address for each recipient can always clean their lists.[2]

We also suggest that recipient systems also create so-called "honeypot" or "trap" accounts. These are accounts that should never receive legitimate mail. They can be exposed to spammers through dictionary attacks, or by posting the addresses on the web in a place unlikely to be seen by humans (e.g. white on white, unused web pages, etc.) Sometimes "retired" accounts are used: old accounts that have been left unused for a while. (This last type occassionally receives legitimate mail.) Any mail sent to these accounts should automatically generate a complaint. (But, since it is important to prevent spammers from learning the identities of the honeypot accounts, perhaps these complaints should only be generated for mail from trustworthy ESPs, or following the idea of footnote 2.)

Users sometimes make mistakes, forgetting that they have signed up for a list, or misunderstanding the definition of spam (e.g. a joke they don't think is funny.) However, mail to honeypots is extremely indicative of a spammer, and sending ESPs may wish to treat such complaints especially harshly. However, not all honeypots are created equal. Mail to a retired account might be from an old friend who does not know the recipient moved. Thus, we suggest the following additional fields in the message body of complaints generated from honeypots: `HONEYPOT-RECIPIENT: TRUE`; also `HONEYPOT-UNUSED: TRUE` if the account was not previously used; and `HONEYPOT-RETIRED:` n where n is the number of days since the account was retired. (The longer an account is retired, the worse it is to send to it.)

6. DISCUSSION AND CONCLUSION

We have performed an economic analysis of outbound spam, and shown that current common techniques are unlikely to be succesful. In particular, current techniques such

---

[2]Perhaps a somewhat complex system could work, in which, for half of the recipients, all complaints are sent, while for the other half of recipients, complaints are only sent to known trustworthy ESPs; the rest of the complaints are used simply for blocking. A list cleaner might be able to cleanse his list of half the complainers, but when he moved to an oft-abused (but hopefully trustworthy ESP), his account would be terminated.

as requiring a signup HIP only impose a maximal cost of about .002 cents per message, while our minimum estimate for the costs/potential revenue from sending spam are around .0025 cents, with many spammers charging or earning 5 to 10 times that.

We have shown that we need not charge per message to deter spammers. As long as the average cost for an account versus the expected number of messages a spammer can send before a complaint is kept high, we can make it unprofitable for spammers to abuse ESPs. We have used this observation to develop three systems that can be as effective as a per message charge, but much less bothersome to users. In one, users are given signup HIPs not just once, but perhaps 10 times, once every 100 messages for the first 1000 messages. In the short term, this may or may not be a sufficient disincentive to spamming. This can raise the cost to .012 cents, at the edge of profitability. This would certainly be a much larger disincentive than the single HIP approach, which seems doomed. We could require 5 times as many HIPs to raise the cost to a true disincentive, but we think requiring one HIP for every 20 messages for the first 1000 messages would be overly burdensome.

We prefer techniques based on computational puzzles. In this system, computational puzzles would be solved on the sender's computer for approximately 8 hours (at low priority in the background), raising the cost to spam to about .06 cents per message. This is above current costs/revenue, and would truly stop spammers. After the initial 8 hour computation, no further computation would be needed, and parts of the computation could be delayed, e.g. performed only when the user is actively using the system. Other than being asked to install a piece of software, users would not notice. Users could start sending mail immediately, as long as they did not send to more than one person every 30 seconds.

In situations where users can pay money, even small amounts, there are also good disincentives. Even a $1 cost to create an account is a reasonable disincentive to spamming, raising the cost per spam to about .05 cents, although for every complaint, the user must pay another $1. However, in systems where the user is already paying a moderate amount of money, e.g. to an ISP or for a premium account, large volumes of mail can be allowed while keeping it unprofitable to spam.

We have identified complaint rates as a critical factor in the cost to spammers. By increasing the complaint rate, we can reduce the cost to legitimate users of any of these systems, or further increase the cost to spammers. We suggest a system in which complaints are automated and standardized to include all needed information.

We have introduced several novel ideas to the problem of stopping outbound spam. First, and perhaps most important, for the first time we analyze solutions to outbound spam from an economic perspective. Second, we have shown that existing techniques are unlikely to work. Third, we have described systems using initial challenging that can be as effective as systems that challenge every message, but in the long run, are far less annoying and costly to legitimate users. Fourth, we have described computational costs as an option for stopping outbound spam: for users with reasonably fast machines and extra cycles, this may be the least burdensome technique. Fifth, we have shown that complaint rates are a critical component in cost analysis, and described possible new standards that would help raise complaint rates. We believe that some combination of these ideas will be successful in stopping outbound spam.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] M. Abadi, M. Burrows, M. Manasse, and T. Wobber. Moderately hard, memory-bound functions. In *Proceedings of the 10th Annual Network and Distributed System Security Symposium*, 2003 February.

[2] J. Angwin. Hunting 'buffalo': Elusive spammer sends web service on a long chase. In *The Wall Street Journal*, May 7 2003.

[3] A. Back. Hashcash, May 1997. `http://www.cypherspace.org/hashcash/`.

[4] M. Blum, L. A. von Ahn, J. Langford, and N. Hopper. The CAPTCHA project: Completely automatic public turing test to tell computers and humans apart, November 2000. `http://www.captcha.net`.

[5] C. Dwork, A. Goldberg, and M. Naor. On memory-bound functions for fighting spam, August 2003.

[6] C. Dwork and M. Naor. Pricing via processing or combatting junk mail. In *Lecture Notes in Computer Science 740 (Proceedings of CRYPTO'92)*, pages 137–147, 1993.

[7] P. Griffin. Spammers remain unrepentant as they make money. In *The New Zealand Herald*, March 21 2003.

[8] Infoworld. What is the worst IT disaster of the last year, July 2003.

[9] E. S. Johansson. Camram, 2002. `Available at http://www.camram.org`.

[10] J. Krim. E-mail providers devising ways to stop spam. In *The Washington Post*, October 30 2003. `http://www.washingtonpost.com/wp-dyn/articles/A38051-2003Oct29.html`.

[11] MAPS. Mail abuse prevention system realtime blackhole list, 2003. `http://mail-abuse.org/rbl/`.

[12] J. M. Moran. Spam king living high in the bayou. In *The Hartford Courant*, June 30 2002. `http://www.ctnow.com/technology/hc-sp1scelsonjun30.story`.

[13] M. Naor. Verification of a human in the loop or identification via the turing test, 1996. Available from `http://www.wisdom.weizmann.ac.il/~naor/PAPERS/`.

[14] OKEAN. Chinese and korean net blocks, 2003. See `http://www.okean.com/asianspamblocks.html`.

[15] S. Olsen. Hotmail restricts outgoing messages. In *CNET News.com*, March 24 2003. `http://news.com.com/2100-1025-993774.html`.

[16] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A bayesian approach to filtering junk e-mail. In *AAAI'98 Workshop on Learning for Text Categorization*, July 1998.

[17] M. Wendland. Spam king lives large off others' e-mail troubles. In *The Detroit Free Press*, November 22 2002. `http://www.freep.com/money/tech/mwend22%5F20021122.htm`.

APPENDIX

A. SPAMMER PER MESSAGE PRICES AND COSTS

How much must outbound spam cost per message, in order to be a sufficient deterrent to spammers? There are two possible answers to this question: more than the cost of sending spam through other means, e.g. by creating their own domains, buying their own mail servers, etc. The other possible answer is more than they can earn by sending the spam. Unfortunately, it is very hard to get good estimates of either value. We give here a sampling of our (widely varying) estimates of these numbers, derived by analyzing published interviews with spammers (who are not known for their honesty.)

There are two ways to analyze spammer costs. The best is to get reports of their actual costs, but this can be difficult. A different technique is to look at what they charge for mailings, which presumably upper bounds their costs, assuming they make a profit (which they apparently do.)

The Detroit Free Press reports [17] that Alan Ralsky, a notorious spammer, charges $22,000 to send to his entire database of 250 million addresses, or just about .01 cents per message.

In a New York Times interview, (now reformed) spammer Richard Colbert says that he used to charge $900 for one million spams: about .1 cents per message. However, Colbert reports that prices have dropped precipitously recently, as low as $25 per million: .0025 cents per message.

An article in the Hartford Courant [12] says that Ronnie Scelson has revenue of $30,000 to $40,000 per 80 million messages, for revenue of as much as .05 cents per message, but is willing to spam for products that bring as little as $1000 per mailing (presumably also to 80 million people), or as little as .00125 cents.)

A Wall Street Journal Article says that Howard Carmack earned $360 for sending 10 million messages, or .0036 cents per message.

The New Zealand Herald [7] reports that one spammer is paid US$300 per million messages (.03 cents).

Based on these figures, there does not appear to be a clear or constant going rate for spam. Indeed, we expect the numbers to change over time. We hope that spam filtering efforts such as our own will be successful. We also hope that well distributed safelists of good senders will become widely used, and will include large ESPs. If very little spam gets through, the spam that does get through will be worth more. And if large ESPs are on safelists, then their spam will get through, making it especially valuable. Based on this analysis, and the numbers here, we need a long term plan that minimally aims to raise the cost of spamming from ESPs to .01 cents, and ideally .05 cents or more. Current account signup costs, which cost at most 2 cents, are far too low to achieve this, assuming complaint rates of 1/200 or less.

B. SAFE MESSAGES

Some messages can be detected as "safe." For instance, a piece of plain text email – no image, no HTML – with no links, no phone numbers, and not detected as containing suspect words or odd obfuscations (e.g. misspellings or unknown words) by a spam filter – is almost certainly not spam. If it is spam, the techniques used are so convoluted that they likely get a much lower response rate than traditional spam. If the best response rate such a convoluted message can get is 1/10 the rate for traditional spam, then we can allow 10 times as many such messages to be sent, while keeping it uneconomical for spammers. We thus suggest assigning different costs per message: low costs for plain text mail that spam filters do not find suspicous, medium costs for difficult to process mail; high costs for mail that appears to be spam. The filter might be a probabilistic, machine learning filter [16]. These costs are counted against the number of messages per day and against the number of messages that need to be sent before the next payment is due. The details are left to implementors, and depend on the filtering technology (e.g. probabilistic or not, how easily defeated) they have available.