The Minstrel's Articles

Spam, a 21st Century Plague
Part B: Administrators

Version 1.2

26 April 2005

Table of Contents

1 Introduction

Spam is rapidly becoming an all-pervasive blight on our online lives, making e-mail less and less a useful communication mechanism. At the time of writing, it is estimated that around 75% of all e-mail clogging our bandwidth is spam, and this figure continues to rise. When will it stop? Can it be stopped? What can we do to protect ourselves?

In this two-part article, I discuss the problem in some depth – where does spam come from, who is sending it, why they are sending, and move on finally to what can be done, both by the community in general and by the individual Internet user.

Note to the pedantic: of course, I am using the word spam as a colloquialism for Unsolicited Bulk/Commercial E-mail (UBE/UCE), its use arising from the excellent Monty Python sketch. I am obviously not talking about SPAM™, the "delicious luncheon meat" from Hormel Foods Corporation, although they don't contest the use of the name to any great degree any more, having realised that their sales have soared since the term was first used to refer to UBE/UCE. Their product even enjoys something of a cult status now, and I'm sure their merchandising revenue rivals that of the product itself! You can see their statement on the subject at http://www.spam.com/ci/ci_in.htm, which is an interesting enough read in itself!

1.1 Sources and Accuracy

The structure of this article, along with some of the content, is loosely based on the NISCC 'Spam Mitigation' Technical Note (please see the References section at the end of the article), which was coincidentally issued just after I started writing notes on the subject. The remainder is based on personal and professional experience and some research.

It must be noted that I do not necessarily endorse any product or service mentioned, and that I only comment on those products and services of which I have personal experience. There are undoubtedly many others available, and I recommend you perform your own research before selecting an anti-spam product or service.

As with all my articles, I reserve the right to be wrong, but where you do spot an inaccuracy, I would very much appreciate hearing about it! Feel free also to disagree with any of my opinions – it should make for a lively discussion!

1.2 Audience

This series of articles is written primarily with the general Internet user in mind, and this one is no exception. This part of the article, imaginatively entitled 'Part B: Administrators', is aimed at anyone looking at implementing blocking, filtering or other defensive mechanisms at a corporate or community level. A detailed discussion on the sources of spam, the motives behind the spammers and the techniques they use to target us is presented in 'Part A: All Readers', and I would recommend starting with that before delving into the technical detail in this document. Part A also contains reference sources relevant to this article.

1.3 Contact Details

Please come and discuss this or any of my articles on the 'Front Line' discussion list:

http://lists.internetgremlin.com/mailman/listinfo/front-line

Or come and join the forums at Privacysense:

http://www.privacysense.com/

Alternatively, you can contact me privately at:

http://www.minstrel.org.uk/contact/

I look forward to hearing from you!

Finally on the subject of contact details, here's one for the address harvesters:

nullbox@internetgremlin.com

As the name suggests, anything sent to this address is automatically deleted, but only after being processed by our spam filter, which will learn from the experience!

2 Countering Spam

This section now presents the methods we (as a community, users, administrators and ISPs alike) can employ to combat it. It must be noted here, as it is at various points throughout this section, that the fight against spam is not something we can win in isolation – it affects everybody, and so everybody must be involved in countering it.

For details of action the average Internet user can take, please refer to Part A.

2.1 Concepts

To begin with, we will look at the legislative and technical measures we have at our disposal, as well as the problems with each approach. In reality, the most likely solution to the problem as a whole will be a combination of several, or even all, of the measures described herein.

There is, in essence, one fundamental aim in all of these countermeasures – if no spam message ever reached a valid recipient, or was always simply deleted without reading, the revenue stream to the spammer would eventually dry up, and the problem would naturally disappear. There have been attempts to trace spammers and take action against them, but this has been difficult for various legislative and technical reasons, and so 'mitigation' is likely to remain the primary focus.

2.1.1 Legislation

On 11 December 2003 the UK Government implemented EC Directive 2002/58/EC on Privacy and Electronic Communications, which aims to cover various different areas including:

- Protecting Privacy
- Security of networks and services
- Confidentiality of communications
- Spyware and cookies
- Traffic data
- Location data
- Public subscriber directories
- Unsolicited commercial communications
- Calling line identification
- Nuisance calls
- Emergency calls
- Automatic Call forwarding

More on many of these in later articles (I will be commenting at great length on this Directive!), but the consensus is that this particular piece of legislation is ineffective, for a number of reasons. Taking the 'opt in' approach, whereby consumers must already be customers of the organisation sending the message, or specifically request the information, it only addresses certain aspects of spam (i.e. only UCE), and even then only when targeted at the consumer. Businesses can still legally be spammed. Even if a message is found to be against the new rules, it is unclear exactly what could be done about it.

In the US, where the infamous CAN-SPAM laws have been in place for some time, the legislation has been generally interpreted as literally that – spammers **can** spam! It is incredibly difficult to legislate against UBE/UCE without inadvertently legislating against legitimate marketing communications.

In Italy, the Directive has been implemented in a more rigorous manner, in that anyone found to be spamming could be jailed and/or fined. Unfortunately, as we have already seen, tracking the culprit can be impossible, especially when it is estimated that at least half of all spam is sent through hijacked or subverted machines!

Another problem with the Directive is that its interpretation and implementation can vary enormously within the EU. As I mentioned earlier, spam is a global problem that must be fought on a global scale, and so the only useful legislation against it would have to be implemented *unilaterally*, an extremely difficult thing to achieve!

Until some means of developing an international standard and achieving global compliance to it can be found, there will always be a problem with boundaries. For example, consider a spammer based in Eastern Europe posting an advertisement for a US-produced medicine via an open mail relay in Australasia, targeting consumers in Western Africa. Where are the legislative boundaries, and who can take action against the individual concerned?

It is precisely this issue that is starting to be exploited by new 'spam-friendly' ISPs appearing in various countries around the world, where legislation is weak or non-existent.

Until a sensible legislative solution to spam prevention is found, about the only tool we have at our disposal is technical, and the two main areas we can focus on are discussed next.

2.1.2 System Security

If all computer systems in the world were as secure as they should be, spam would be very much reduced. Spammers would be left with few channels to exploit – essentially, they would either have to send the mail through a machine they are responsible for (and therefore would be accountable) or through a service offered by a 'spam-friendly' ISP, on whom pressure could be put, and I suspect they would not last long.

Unfortunately, the worldwide lack of security, and even security awareness, is not a problem we are going to solve overnight, and it may not be possible to solve it at all. These days, for a computer to be useful it has to be connected to at least one other, or to the Internet, immediately introducing security vulnerability.

In later articles, I will be presenting practical advice on security for the general user and the System Administrator, so hopefully we can gradually move towards a better situation. However, given that we cannot yet stop the source of spam, and cannot yet cut off the communication channels, with what are we left?

2.1.3 Filtering

Our second technical defence is to stop the spam arriving at its destination. In almost all cases, this will involve some form of filtering, and there are a number of techniques that can be used with varying degrees of efficiency.

Filtering of spam messages is becoming a complex field, increasingly using mathematical and statistical techniques to accurately identify unsolicited messages and (in most cases) delete them. There are several pieces of information in any particular e-mail message, some or all of which are used by the different filtering methods:

- Sender IP address (spoofed or hidden where possible)
- Sender e-mail address (usually spoofed)
- Recipient e-mail address (never spoofed, obviously!)
- Originating mail server (not usually possible to spoof)
- Chain of relaying mail servers (usually accurate, but may be spoofed)
- Content of e-mail – may be HTML, may contain hidden codes, will almost always contain hyperlinks to Web sites, and will have various lexical attributes

This information, in combination with public information (e.g. DNS records) are used in different ways by each filtering technique, and specific examples are given in each section.

Filters can be installed at your mail server, something that more and more ISPs seem to be doing by default as the plague escalates, on your local machine (there are many software products available to do the job), or even as a provided service (i.e. your e-mail is passed through a subscription service before you download it). In all cases, though, filtering is a balancing act – there is always a risk that legitimate mail is detected as spam and lost, or that spam is not detected as being so and still ends up in your mailbox.

I believe the most effective approach to filtering will combine several of the technologies discussed below. Gerylisting seems to be an interesting development, and combined with Sender Policy Framework (and perhaps controlled blacklisting) has the potential to eliminate spam. Again, though, any filtering will only be truly effective if applied unilaterally.

The main methods used to filter spam are discussed in the next few sections, and my thanks go to NISCC (see the References section) for prompting me to consider the more obscure ones. If you are not interested in these techniques, or have no means of implementing software to use them on your system, then please feel free to skip to the 'Actions' sections, in which practical advice is presented for the user and the System Administrator.

2.1.3.1 Blacklisting (Negative)

As the name implies, blacklists are lists of information that can be used to block any particular e-mail if some information within it matches an entry on the list. Although, in theory, there are several different types of blacklist, containing IP addresses, e-mail addresses, domain names or other information, the form most commonly used is the IP address-based blacklist.

Blacklists may be provided free or as subscription services, online or downloadable and updated at intervals or continually. Continually updated blacklists are commonly referred to as Real-time Blackhole Lists (RBLs), and it is these in particular that are in most common use by blacklisting software. Some reference information on RBLs is provided at the end of this article.

Whatever form of blacklist is used, the process followed by the software referencing the blacklist is the same:

- On receiving an e-mail for delivery, compare the relevant details with the blacklist
- If there is a match, take appropriate action, which may involve any of:
    - blocking the mail;
    - tagging it as spam so the user can filter if they want to;

  - quarantining it to be double-checked;
  - reporting it for further action (see the section on 'Taking Action').

The rules for matching against a blacklist have become increasingly complex over time. For example, at least one RBL lists **all** dynamically assigned IP addresses on the Internet (such as the one your ISP provides to you when you connect to the Internet). If such a list was used to find a simple IP address match, all of your e-mail could be blocked! Instead, if the *second* IP address in the mail relaying chain matches an entry on this RBL, then it should be handled as spam.

Blacklists are kept up-to-date in a variety of different ways, but are generally based on information from complainants – once a spam message is analysed, it is usually fairly clear where the problem lies (an open mail server or proxy, for example), and that IP address will be added to the RBL. Getting an address *removed* from an RBL can be more difficult (guilty until proven innocent!), which is one of the arguments against their use.

Use of a blacklist, if it is a well-managed one, can be very effective in blocking *potential* spam, providing you:

- Either double-check each message in case the blacklist is inaccurate, a resource-intensive task;
- Or accept that a certain level of legitimate mail being lost.

There are a number of other problems associated with blacklists, and a considerable amount of effort is being put into reducing or eliminating these so that they can become more effective:

- Spammers are usually one step ahead of the RBLs, since they make great use of dynamic addresses (dial-up, broadband, etc.).
- Over-enthusiastic addition of IP address ranges can cause large problems – an RBL administrator may think a particular ISP has a lot of spammers using it, for example, and add their whole address range to the RBL, meaning that legitimate users are also listed. Indeed, there have been several RBLs that have ended up listing almost the entire Internet!
- False reports can also cause problems – given the general difficulty in having an address removed from an RBL (there are exceptions), being falsely accused, either maliciously or otherwise, can cause a particular address to be blocked for some considerable time. One of my servers has been listed in the past through some misinterpretation of e-mail headers, but fortunately it was with a reactive RBL provider, so I could get it removed quite quickly.
- Blacklisting will not generally work against spammers that successfully spoof the information contained in an e-mail.
- RBL providers are increasingly becoming the target of Denial of Service attacks from the spammer community – if a mail service relies on the availability of a particular RBL, and that RBL is down for any time, e-mail communication can be severely disrupted.

2.1.3.2 Whitelisting (Positive)

Whitelisting, as I'm sure is obvious, is precisely the opposite of blacklisting. Instead of blocking any mail that has attributes matching an entry on a given list, it will *allow* mail that matches an entry on a list. Again, the match may be based on IP addresses, e-mail addresses, domains or other information, but this time the most common form of list is one containing sender e-mail addresses or domains.

There are several ways whitelists can be implemented, some automatic based on the e-mails sent out, and some requiring manual confirmation from a sender, but the main advantage is that the approach is generally fairly effective against spam. The disadvantages, however, are numerous, and include:

- The likelihood of legitimate e-mail being blocked, or at least delayed, is actually higher than with blacklisting, especially if you regularly receive messages from senders that have not contacted you before
- Whitelists require a great deal of management, since every blocked e-mail must be checked for legitimacy, and tend to be maintained by individuals or organisations, rather than by any central service or authority
- Sender-based whitelists have no defence against deliberately forged addresses, so targeted spam can still get through
- IP address-based whitelists are more difficult to bypass using forging techniques, but the management overhead is as high as with sender-based whitelists
- The technical resources required can be considerable – for an individual user, a whitelist may not be huge, but for an ISP handling whitelists for all their users, or an organisation whitelisting for their employees, each search through the list can be resource-intensive

2.1.3.3 Greylisting

The drawbacks associated with each of the previous techniques drove Evan Harris to develop a concept combining the two, in an attempt to reap the advantages and eliminate some of the disadvantages. Further information on his greylisting theory is referred to at the end of this article.

The concept is relatively simple:

- When an e-mail arrives at a mail server (greylisting cannot easily be used on a client machine), three attributes are examined (the triplet):
    - The IP address of the most recent server in the mail chain
    - The sender address
    - The recipient address
- If the triplet has not been seen by the server before (i.e. referenced against a list of 'known triplets'), or has been seen only a short time ago, then a temporary error is returned (i.e. 'Please Try Later', or 'TEMPFAIL')

The main aim of this approach is to target spamming software, whether malicious Trojans or software developed specifically for the purpose, which will generally not attempt to retry sending a message. Hence, no retry will come, and the message will not be delivered. Legitimate mail will continue to be delivered, with only a temporary delay on the first occurrence of a triplet. The processing overhead is also minimal – the e-mail content itself is not examined in the first exchange, and the only overhead is for the sending machine, which will have to queue the message for retrying later (the average retry interval is around an hour).

The testing performed by Evan and his colleagues in 2003 shows that the approach is very effective in blocking spam *and* viruses that have no built-in retry mechanism. Indeed, some statistics show that up to 95% of spam and viruses are delivered in this 'fire and forget' manner, in which case greylisting becomes an attractive solution to the problem.

There are, however, a few disadvantages to the technique:

- It is not possible to ensure rapid delivery of an e-mail that presents a new triplet without manual intervention (i.e. whitelisting)
- Spam sent through open mail relays will continue to be delivered, since they will obediently retry as requested – hence, blacklisting will continue to be a valuable measure
- There is risk of legitimate e-mail being rejected where an SMTP server is configured not to retry sending, or the sender address is always unique (discussion lists, for example) – whilst the 'TEMPFAIL' mechanism is defined in the relevant standards relating to SMTP, they are not strictly mandated, and so there is some administrative overhead in whitelisting these 'misbehaving' mail servers

Spammers would eventually adapt, as they always do, and begin bypassing greylisting through honouring the retry – it has been suggested that this would become a sufficient drain on their resources to reduce the problem in itself, but this is not easily measured. On the flip side, the retry interval does offer a window of opportunity in which the sender could be traced and then blocked, or further action taken if appropriate. Perhaps it would also be possible to increase the requirement to two or three retries before delivery, thus reducing the spammer's opportunity even further.

As mentioned earlier, I believe this approach, in combination with some level of other technologies (controlled blacklisting, some whitelisting and the SPF concept – see below) could be a viable solution for the Internet as a whole, but there is a considerable amount of work that will need to be done before such a global level of 'buy-in' can be achieved.

2.1.3.4 Bayesian Filtering

Paul Graham was also dissatisfied with simple blacklisting and whitelisting, but took a different approach to a new solution. Please refer to the Reference section for further information, but in essence, the Bayesian approach uses a complex statistical analysis of all e-mail passing through a server, and assesses a *probability* that it is spam. Based on this calculation, the server can then block, quarantine or pass each message to the final recipient.

The Bayesian analysis model proposed by Paul analyses not only the header information (i.e. IP address, sender, recipient, mail server chain, etc.) but also the e-mail content itself – using lexical analysis, the message is broken into its component parts, and each of those parts assessed for 'spamminess'. The message as a whole can then be assessed.

Bayesian analysis is designed to 'learn' for each recipient – the concept is that a mail user will tell the system when it receives a spam message, and the system will then increase the 'weighting' of that particular combination of message attributes. The user will also need to tell the server (or it can be assumed) that other messages are *not* spam (sometimes known as 'ham'), and the weighting of those attributes can be decreased.

This approach obviously has the advantage that a very refined filter can be created for each recipient, and the chance of a spammer being able to create a message that bypasses a significant number of these filters reduces over time. The overhead in tuning the filter, however, may be onerous for the user, particularly if they already receive a large amount of spam.

Other disadvantages of this approach include:

- It is not easily implemented in some environments, where users are not in a position to feed back to the server and tune their personal rules – applying decisions on each message to a server-wide filter will be time-consuming if done by an administrator, and unreliable if done by all users
- There is still a potential for false positives (i.e. ham being treated as spam), certainly in the early days as the filter is beginning its learning process
- The entire model is dependent on users' (or administrators') assessments of what is and isn't spam – this can sometimes be difficult (for those that haven't read this article, at least!)

Whilst the Bayesian approach has a great deal of potential, I feel it needs more work before it can be reliably used in a production environment, and in any case it should only be one of a series of measures.

2.1.3.5 Heuristic (Rule Based) Filtering

If you have ever set up filters in your e-mail client to delete spam based on subject lines or message content, then you have used Heuristic Filtering. As you can imagine, creating a rule set that will filter all spam would be a time-consuming task, and could block many legitimate messages.

Commercial and Open Source implementations of Heuristic Filtering (most notably SpamAssassin – see References section) modify this basic approach by using a weighting mechanism for each rule, resulting in a total score for each message.

For example, if a message contains the word 'discount', the rule might add 0.01 to the total score. If it came via an open mail relay, it might add 2.0. Use of plain text rather than HTML may reduce the score, and so forth. Once all the rules have been applied, the total score is totted up and then compared to a set of 'threshold' values set at a user or system level, to determine what to do with the message. The approach is usually along the lines of:

- For messages with a high score (e.g. 5.0 or more), delete it (or, preferably, quarantine, double-check and take further action – see later)
- For messages with a medium score (e.g. 3.0 to 5.0), mark it as potential spam, perhaps using mail headers or by modifying the subject line, but allow it to be delivered
- For all other messages, allow them to be delivered

Although this is a powerful approach to filtering, and easily implemented at the server level (as is demonstrated by SpamAssassin's colossal popularity in the industry), it does require constant maintenance in updating rules to account for new spamming techniques and attributes, adjusting the weightings to be more accurate, etc.

Other disadvantages include:

- The wide variety of different attributes both in spam and ham means that a certain level of error is inevitable – for example, many e-mail clients actually produce, by default, e-mail that is formatted in a very 'spammy' way. Similarly, some spam matches few heavily weighted rules, and will be passed – it is not uncommon for mail administrators managing heuristic filtering systems to gradually reduce their threshold value between rulebase releases.

- Since the rule base is static, and only the weighting for each rule or decision is easily adjusted, the spammer community can learn very quickly to bypass any particular set of rules. Some spammers are known to test their messages against SpamAssassin and other heuristic systems before sending them out, and there are even online services to allow them to do this!

Again, Heuristic Filtering is most effective when used in combination with other techniques – indeed, SpamAssassin also uses blacklists, whitelists and Bayesian technologies.

2.1.3.6 Sender Policy Framework (SPF) – Sender Authentication

Still in draft form (see the References section), this concept is an extension to the SMTP and DNS protocols, and is designed to target one particular aspect of spam messages – the spoofed 'From' address.

Once again, the theory is simple:

- The Domain Name System (DNS) is updated to contain details of legitimate mail servers for each domain
- When a message is received, servers check the sending machine against the registered 'approved' list of servers
- If there is a match, the e-mail is delivered, otherwise it is quarantined and examined, or tagged and then delivered

Unfortunately, there appear to be some fundamental flaws in the draft that would make the system easy to subvert or bypass, even if it were implemented on a global scale, which would be necessary for it to be useful. The implementation process itself could be extremely difficult (perhaps prohibitively so) for most ISPs.

As a first step in a chain of countermeasures, however, SPF could be a good approach, although I feel the other techniques above offer a more cost-effective balance between implementation, management and accuracy. Time will tell, though, and I will watch this draft with interest.

There are other emerging forms of Sender Authentication (see the References section), but they have a fair way to go before they can be considered a solution in themselves – as before, multiple techniques will need to be combined.

2.2 Actions For The System Administrator

If you administer a mail system, Web server or are responsible for users in any way, the following guidelines will help you protect them and your organisation from spam attack.

The first thing that can be done by any System Administrator is to raise awareness of the problem among the user base. This paper may be too detailed for your users, but a distillation of the relevant points, and in particular the 'Actions For The User' section, could be sent out to help users help themselves.

The next thing you need to do is define a realistic strategy to tackle the problem (if, in fact, it is even a problem for your users!). Installing software or subscribing to services without considering the cost/benefit ratio and potential impact on the system could be disastrous. What exactly you decide on can only be decided by you and any relevant management, and you will need to consider the type of systems you run, the size of your organisation or user base, the level of knowledge of those users, and a variety of other factors specific to your situation.

As a guide, though, the following may be of help in making your decision:

- At the most fundamental level, you should assess what functionality there is within your existing mail server software for preventing spam. Sendmail, for example, has a range of features that prevent unauthorised relay, perform reverse lookups on mail domains, etc. It would be worth considering enabling features you already have available, especially if they are low-maintenance options.
- Similarly, and more in an attempt to prevent your server actually being used for spamming, you should ensure that software is configured to enable *only* authorised users to use the service. This may involve validating sender and recipient addresses, but more reliably should be based on the IP addresses of your users. This recommendation applies to services of all kinds – mail relays, proxy servers, even Web-based feedback forms.
- If all of your users are relatively technical (for example, you run the mail server for a software house), they may be happy to manage their own filtering, either through functions in their mail clients, or by use of a server-level Bayesian filter.
- If your users are not so technical (for example, you run the mail services for a dial-up ISP or less technical organisation), it may be more beneficial to implement a form of filtering that requires no user intervention, such as black, white or grey lists. SpamAssassin is perfect in this scenario, as it still allows the individual more technical user to fine-tune their filtering.
- If your organisation is small, and the e-mail traffic levels are low, the administrative overhead (i.e. checking quarantined messages) of simple whitelisting should not be great. Since it is one of the most effective means of blocking spam, it is preferable in situations that can manage it. Of course, if your organisation is regularly contacted by new customers with unknown addresses, the delay in processing of new incoming messages may be unacceptable to the business.
- If your organisation has a much larger throughput of e-mail, or even if you are an ISP, a more manageable approach will need to be taken. As long as you have the resources to install, maintain and manage them, a combination of technologies will offer the best possible protection for you, and even allow some flexibility for the user. In my experience of running mail servers, the ideal combination has been:
    - Use of greylisting (potentially – I'm still experimenting with this) – in theory, most spam should be blocked before the server has to worry about the content
    - Use of a reputable and accurate blacklist – again, messages can be blocked before processing power is used analysing the content
    - Use of heuristic filtering (see earlier sections) to assess the likelihood a message is spam based on fixed rules
    - Use of Bayesian filtering (see earlier sections) to assess the likelihood a message is spam based on a 'learning' process per user

  Product-wise, the last three in this combination are well supported by SpamAssassin (which has the added value of using blacklist to 'weight', rather than to block outright), which is Open Source. Note, though: the fact that SpamAssassin is free is also its main drawback – it will require considerable expertise and management on your part to keep it running in the most effective manner.

- If your organisational resources are extensive, the simplest solution is likely to be to subscribe to a filtering service. In this scenario, the vendor you select will usually provide your Internet-facing mail servers, from which you download filtered mail to your own servers, or receive the messages relayed from the vendor after processing. MessageLabs is one such provider, and those customers of theirs I have encountered are generally very pleased, but you should always perform your own analysis of the marketplace to assess the most appropriate service for your organisation.
- If you manage a Web site for your organisation, or have any responsibility for distribution of organisational e-mail addresses (e.g. providing lists to other organisations), be as selective as possible about how and where e-mail addresses are published. They should certainly never appear as a list on a Web site, and you should encourage users to be similarly cautious.