End-users’ resource consumption of spam and a 3D anti-spam evaluation framework

Minh Tran
Centre for Advanced Internet Architectures
Swinburne University of Technology
Melbourne, Australia
mtran@swin.edu.au

Grenville Armitage
Centre for Advanced Internet Architectures
Swinburne University of Technology
Melbourne, Australia
garmitage@swin.edu.au

Abstract—Our work is motivated by the challenge of coupling excessive network consumption and control of the consumption; the importance of cost models to quantify and characterize the consumption; and the crucial requirements of controlling techniques to limit the theft of network resources. In this paper, we particularly examine the network consumption of spam. We quantify the end-user's network resource consumption caused by spam as a function of different email retrieving mechanisms, and estimate the financial costs of spam for both residential and business users. The paper also describes our anti-spam rate-limiting tool, MT Proxy, and introduces a general three-dimensional framework to evaluate anti-spam techniques. We examine and compare both traditional and novel anti-spam solutions and show a simple comparison using our model.

Index Terms—Resource consumption, spam, anti-spam, 3D framework, technical cost, financial cost, social cost.

I. INTRODUCTION

One of the most significant challenges facing the Internet today is the weak binding between an arbitrary host's ability to consume resources (service consumption) and a network operator's ability to control the consumption and charge back notional fees related to the scale of the consumption (cost to use). There are a number of areas where the 'service-cost disconnection' relationship is yet to be satisfactorily resolved. A topical example is email spam (‘mass unsolicited electronic mail’ [1]). The current Internet email system is predicated on the assumption that people use what they need. Spammers have leveraged this openness to great effect, 'borrowing' transmission capacity from unwitting mail servers and virus-infected home machines.

Due to its uncontrolled consumption of network and end-user resources, the characteristics of spam are worth close study. In this particular study, we attempt to quantify the end-user cost of spam by measuring the amount of time and network traffic wasted downloading and deleting spam using a number of common email retrieval mechanisms.

As the swift increase in spam volume, effective anti-spam solutions are now crucial. Many anti-spam techniques (at either end-user or corporate levels) have been proposed and implemented yet there has not been a general framework to compare between these tools. We introduce our comparison model with three cost-based factors to evaluate the effectiveness (ability to protect users from spam attacks [1]) and cost-efficiency of anti-spam solutions (including our own rate-limiting anti-spam tool, MT Proxy). Our visual 3D model provides a general evaluation framework where technical, social and financial costs are used as decision-making factors.

II. QUANTIFYING END-USER COST OF SPAM

A. Introduction and a brief overview of methodology

Spam is ‘costly’ in terms of network resources, users’ time, social trust, false positives (when legitimate emails are filtered as spam) and deployment of anti-spam solutions. There have been several commercial studies on the financial cost of spam. Research by Ferris Inc. estimates spam will cost about $50 billion (USD) globally in 2005, of which $17 billion is lost by US businesses alone [2].

We are particularly interested in quantifying the network consumption cost of spam at end-users by using different common email retrieval mechanisms. We aim to calculate minimum costs from recognizing and deleting spam emails (we do not model a user reading the entire message before making decision of deleting it).

Our sample spam emails are taken from a collection of 6955 spam emails logged by our University's IT department over a 24 hour period in early 2004. Their sizes range from 1 KByte to 11 KBytes with a mean of 4.64 KBytes. We retrieved and deleted 100 spam emails for each trial and repeated 100 trials for every email retrieving method.

Our experiment is done on an asymmetric link of 1.5Mbps downstream (mail server to mail client) and 128Kbps upstream (mail client to mail server).

Tcpdump and Ethereal are run at the mail client to capture the traffic and provide traffic statistics. Matlab 6.5 is used with our own software to plot the amount of byte transferred versus time from tcpdump files' data.

B. Comparison of three basic email retrieving models

A recipient’s mail client (Mail User Agent – MUA) uses certain email retrieving methods to access emails stored at the mail server (Mail Transfer Agent – MTA). There are generally three basic email retrieval models, namely Offline, Online and Disconnected. Fig. 1 shows the three basic email retrieval models and their associated protocols (POP3 and IMAP4).

The Offline model allows emails to be downloaded and processed locally within the MUA. In the Online model, emails are stored at the mail server and are remotely manipulated by the mail client. With the Disconnected model, the MUA downloads copies of emails so they can be processed locally. When the MUA reconnects to the server, they synchronize with each other.
We use POP3 for the Offline model and IMAP4 for the Online and Disconnected models in our experiment. We use two methods of reading and deleting emails in the Online model: delete each individual email after reading that email and delete all emails after reading all of them.

Fig. 2 compares the three models of reading/deleting spam emails. We find that the Online model accounts for only 10% of network traffic consumed by the Offline model or the Disconnected model as the Online model allows users to read email header without incurring the cost of downloading the whole email content. The Disconnected model takes longer than the Offline model to download emails, due to the larger number of control message exchanges.

C. Comparison of ten email retrieving mechanisms

We implemented ten methods of accessing emails to represent end-users’ behaviours, as listed below:

- Offline POP3: download all spam emails from the server, store, read, delete locally.
- Disconnected-IMAP4: download all spam emails from the server, store, read, delete locally and synchronize with the server after deleting spam.
- IMAP4-delete-all: read all spam email headers and delete all at the same time.
- IMAP4-delete-each: read each spam email header and delete each of them.
- Yahoo-empty-bulk: log into the Yahoo mailbox and empty spam email bulk folder.
- Yahoo-delete-all: log into the Yahoo mailbox, read all spam email headers on one page and delete all of them at the same time.
- Yahoo-delete-each: log into the Yahoo mailbox, read each spam email header and delete each of them.
- Hotmail-empty-bulk, Hotmail-delete-all and Hotmail-delete-each: equivalent to similarly named Yahoo cases, but using Hotmail email service.

We chose Yahoo and Hotmail due to their being the two biggest Web front-end mail systems in Australia. We believe these ten email retrieving methods reflect most typical email user’s behaviour.

Fig. 3 shows Yahoo-delete-each method (565±7% seconds) leads the group for time taken to delete spam emails, followed by Hotmail-delete-each (493±5% seconds). These two methods are about four times longer compared to POP3 and IMAP4 methods. Yahoo-delete-each and Hotmail-delete-each represent extreme cases where users have to go through each page to look for spam among with their legitimate emails and have to delete each spam email individually.

Fig. 4 presents an interesting result on the amount of network traffic consumed for different retrieval methods. Hotmail-delete-each causes the most data transferred...
server. Table I is useful when the SMTP server resides outside the company’s network (POP3 or IMAP4 methods) or when employees use personal emails at work (for example, Yahoo or Hotmail). The total cost will be the cost of downloading emails from immediate MTA to company’s SMTP server plus extra cost due to employees’ personal spam emails.

### Table I – Cost of spam per month for business use

<table>
<thead>
<tr>
<th>Method</th>
<th>Within quota limit</th>
<th>Over quota limit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(to nearest MByte)</td>
<td>(to nearest (dollar)</td>
</tr>
<tr>
<td>Offline POP3</td>
<td>82</td>
<td>1</td>
</tr>
<tr>
<td>Offline IMAP4 - delete all</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Online IMAP4 - delete each</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>Disconnected IMAP4 - delete all</td>
<td>85</td>
<td>17</td>
</tr>
<tr>
<td>Yahoo - empty bulk</td>
<td>81</td>
<td>16</td>
</tr>
<tr>
<td>Yahoo - delete all</td>
<td>113</td>
<td>23</td>
</tr>
<tr>
<td>Yahoo - delete each</td>
<td>799</td>
<td>160</td>
</tr>
<tr>
<td>Hotmail - empty bulk</td>
<td>105</td>
<td>21</td>
</tr>
<tr>
<td>Hotmail - delete all</td>
<td>156</td>
<td>31</td>
</tr>
<tr>
<td>Hotmail - delete each</td>
<td>1420</td>
<td>284</td>
</tr>
</tbody>
</table>

#### III. Three-Dimensional Evaluation Framework

### A. Our rate-limiting anti-spam tool

We proposed an IP-based anti-spam approach called ‘MT Proxy’ [12]. We have two objectives: to reduce the negative cost of false-positives (by allowing all emails to get through but slowing down spam emails significantly); and to shift back the costs to spammers (by increasing the effort in time and resources to which a spammer must go through in order to send their spam emails).

MT Proxy (Fig. 7) uses local and DNS black listing method and analyzes content of the incoming SMTP packages. The spam result is used to rate-limit spammer’s traffic by adding additional IP-level latency and bandwidth restrictions. Since our first prototype of MT Proxy [12], we

![Figure 5. Comparison of data transferred (Quartile view)](image)

![Figure 6. Cost of spam per month for residential use](image)

![Figure 7. Architectural design of MT Proxy](image)
have added a ‘short-term’ blacklist so that MT Proxy can work more efficiently for small emails and emails where where signs of spam do not occur early. MT Proxy can learn which hosts have attempted to send spam and blacklist these hosts.

Fig. 8 shows test-bed results with our MT Proxy implementation. We sent 100 emails of average size 5KBytes to three types of users (who receive all non-spam emails; all spam emails; half non-spam emails + half spam emails respectively). Spam received at end-users is reduced by an amount of 9.9 Kbytes per second or approximately 2 spam emails per second. This reduces the financial cost on users (per the relationship shown previously in Fig. 8).

B. An overview of anti-spam methods

The simplest anti-spam techniques involve black-listing (block emails coming from black-listed addresses) and white-listing (only allow emails from white-listed addresses). As local blacklists are hard to keep up-to-date, remote blacklists have sprung up that use the DNS protocol to enable queries and replies about the status of particular IP addresses. Examples of these Domain Name System Blacklists (DNSBLs) include SpamHauss [3] and Open Relay Database (ORDB) [4]. Challenge-Response (CR) mechanisms have been added to update the status of local white lists - unknown senders are challenged by ‘request emails’, and white-listed only if they send a correct reply to the challenge.

Early content filtering methods used text characterisation (TC), where email contents were analysed for manually-constructed spam-patterns (words or sentences). Automated solutions (also called statistical methods) are more attractive and have undergone intensive research efforts [5]. Statistical filtering is implemented by applying machine learning algorithms and pre-classified documents to a spam classifier during its training stage. The filter can then apply its learnt knowledge to detect spam during its classification stage. Examples of typical statistical filtering methods are Naïve Bayes, k-Nearest Neighbour (k-NN) and Neural Networks [6].

Significant anti-spam efforts try to make spammers ‘pay’ and reduce the rate at which they can send spam emails. Microsoft’s ‘stamp of approval’ adds additional delays at the senders’ end by requiring them to solve a cryptographic puzzle set by the receivers [7]. Economics-based schemes simulate a ‘mail postage system’, making the senders pay refundable costs of sending emails (bankable postage) [8] or buying stamps (e-postage) [9] or an improved payment method with differentiated surcharge [10].

Costs can also be shifted back to spammers by rate-limiting their email connection. The Anti-Spam Router (ASR) of TurnTide [11] performs TCP shaping on incoming SMTP traffic according to QoS (based on spam analysis) allocated to the traffic. Our own work on MTProxy (as described in section III.A) involves rate limiting the IP packets of individual SMTP connections. We assign rate limits to connections whose source IP addresses are blacklisted or who appear to be carrying spam.

Authentication against spam forgery and phishing have been objectives of ‘Sender ID Framework’, merged by ‘Caller ID’ of Microsoft and ‘Sender Policy Framework’ of Pobox [13]. As opposed to the IP-based approach ‘Sender ID Framework’, signature-based approach is also suggested. An example is DomainKeys Identified Mail (DKIM), merged by Cisco’s ‘Identified Internet Mail’ and Yahoo’s ‘Domain Keys’ [14].

Other techniques include ‘Collaborative Filtering’ [15] (a group of receivers’ spam filter share knowledge of spam signatures and are able to thwart spam based on the shared information), ‘Disposable Email Address’ (DEA) with Rolling Email Address Protocols (REAP) [16] (manage addresses associated with providers and can shut down an address if it is exploit by spammers).

C. A cost-based three-dimensional evaluation framework

Inspired by a colleague’s work on visualising the relationships between Internet pricing schemes [17], we propose a visual 3D framework for evaluating anti-spam techniques. There are three ‘cost’ categories: technical cost, financial cost and social cost (X, Y, Z respectively) (Table II). Within each cost category, each measurement is also assigned a weight of Xs, Ys, Zs and a numerical score ai, bi, ci or N/A (if the criteria ‘does not apply’). The weight represents the importance of each measurement towards the total cost. Total Cost (TC) is calculated as in Fig. 9, where NsX, NsY, NsZ are the numbers of measurement criteria for each cost category.

Technical costs measure the capability of an anti-spam technique in detecting different types of spam ‘trick’. These include techniques used by spammers to conceal their identity or trick anti-spam filters from detecting spam content of the

<table>
<thead>
<tr>
<th>Table II–Cost factors in the evaluation framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical Cost</td>
</tr>
<tr>
<td>Types of spam (X1)</td>
</tr>
<tr>
<td>Numerical Score (0-10) or N/A</td>
</tr>
</tbody>
</table>

Authorized licensed use limited to: SWINBURNE UNIV OF TECHNOLOGY. Downloaded on January 7, 2010 at 20:41 from IEEE Xplore. Restrictions apply.
message. Other measurement criteria of technical costs are the feasibility of deploying an anti-spam solution without requiring radical changes in the network infrastructures, the requirement of monitoring and reporting and the maintenance of up-to-date spam sources. Higher cost means lower effectiveness in spam detection and solution implementation.

Financial costs involve 'real costs' imposed on users and ISPs, such as: cost of deployment, cost due to loosing legitimate emails (false positives), cost in time and resources of letting spam through (false negatives). The criterion 'spam reduction' ranks an anti-spam solution according to its capability of reducing spam in a long run. Economics-based and rate-limiting solutions impose less costs in terms of 'spam reduction' since they can slow down and reduce the amount of spam emails sent by spammers.

Social costs deal with spam types perceived by ISPs/users as 'important'. An anti-spam solution has high social cost if it cannot detect or filter spam within that cost criterion. Domain Keys of Yahoo and Sender ID Framework of Microsoft/Pobox are designed for authentication against spam phishing whilst MailGate of Tumbleweed [18] is most effective to throttle spam in combined with email Denial of Service (eDoS) attack and Directory Harvest Attack (DNA); and thus they will have very low costs in these categories.

Fig. 10 compares the total cost of different anti-spam methods, where technical, financial and social costs are equally weighted. MailGate of Tumbleweed appears to be most attractive with its high blocking rate, low false positive (utilizing a Personal Quarantine Manager), filtering 'dark traffic' [18] (eDoS and DNA) and combined advantages of implementing listing, filtering and DNS authentication. ISPs/users might assign different weights of cost categories and cost measurement criteria according to their perception; which can result in different 'best' anti-spam solutions.

IV. RELATED WORKS AND OUR CONTRIBUTION

A. Quantifying and characterising spam

Gomes, Cazia, Almeida, Almeida, Meira [19] presented their approach of identifying the quantitative and qualitative characteristics (email arrival process, email size, numbers of recipients per email and popularity of the emails) that significantly distinguish spam from non-spam traffic and assessing the impact of spam on the aggregate traffic by evaluating how the latter deviates from the non-spam traffic.

Bertolotti and Calzarossa [20][21] characterised the workloads of e-mail servers (email inter-arrival times, e-mail sizes, and number of recipients per e-mail). They also looked at the characteristics (inter-access times, number of messages per users' mailbox, mailbox sizes, size of deleted e-mails) of email retrieving process (using POP3 protocol) and proposed models of user behaviours.

Cranor and LaMacchia [22] examined the distribution of spam arrivals and spam content at selected sites from the AT&T and Lucent sub-domains. They also identified factors contributing to the spam problem, which are "the low price of bulk email" and "cheap pseudonyms."

Twinning, Williamson, Mowbray, Rahmouni [23] characterized server workload to investigate the effectiveness of anti-spam mechanisms. They provided statistical information on the number of servers that send only good emails, only spam emails and a mixture, together with the number of messages sent.

Our work quantified the minimal network cost of spam imposed on end-users in compared to other characteristics (arrival times, e-mail sizes, et al. of spam [19][22], email traffic [20][21] and sending email servers [23]) in related works. We examined the impact of different email retrieval techniques whilst authors in [20][21] investigated the users’ behaviours of email retrieving process with only POP3 protocol. There are also several commercial studies on financial cost of spam [2]. However, our work particularly examined and compared the minimal network and financial cost at end-users with common email retrieval mechanisms.

B. Evaluating anti-spam solution

Cranor and LaMacchia [22] examined several anti-spam methods: ones with minimal requirements in changing ISP’s network infrastructures/billing-systems (automated/semi-automated filtering, counter-attack) and ones with some radical changes (opt-out lists, channels, payments, fee restructuring). Although they looked at the strengths and weaknesses of each anti-spam method, they did not directly compare the methods.

Jung and Sit [24] examined the effectiveness and correlation of seven popular DNS blacklists (spam, open relay sources). They suggested factors to consider when combining...
these blacklists. Zang, Zu and Yao [5] evaluated five supervised learning methods in the context of statistical spam filtering. The paper compared the effectiveness between these classifiers and impact of different feature pruning methods on them. In compared to our work, their work ([24] and [5]) solely compared anti-spam methods within particular areas.

Twining, Williamson, Mowbray, Rahmouni [23] investigated the efficiency of existing anti-spam techniques. The authors classified them into pre-acceptance (black-listing, rate-limiting, temp-failing) and post-acceptance (filtering) according to whether these solutions are applied before or after emails are accepted for delivery by recipient mail servers. The paper showed how these methods are limited in handling spam problems, without directly comparing the methods.

Our work examined a broader picture of existing anti-spam techniques and proposed a general framework that allows for direct comparison between them. We introduced three costs factors in our evaluation framework, which we believe are important in assessing the effectiveness and efficiency of anti-spam solutions.

V. CONCLUSION AND FUTURE WORK

The Internet's architecture embodies a broad disconnect between the ability to consume network resources and viable techniques of controlling the excessive/unauthorized consumption. Email spam is a typical example of the misuse of the Internet. So, its consumption characteristics and controlling techniques are worth close study.

In this paper, we investigated the quantitative minimum network consumption of spam imposed on end-users. We showed how our rate-limiting anti-spam tool MT Proxy could be used to mitigate the consumption. We examined traditional, novel anti-spam solutions and proposed a general framework of evaluating these solutions. We introduced three cost-based factors and several cost criteria in our evaluation framework. Our 3D evaluation framework can be extended by developing algorithms for the automated selection of a best set of anti-spam techniques according to pre-defined criteria.

We wish to further investigate the casual relationship between service consumption and service cost/control techniques across different areas of the Internet. One similar example to spam is the Denial-of-Service (DOS) attacks, which also have their roots in IP network architectures failing to impose costs for sending traffic. Another example is in the area of ad-hoc networks, where clusters of mobile nodes form loose-knit transient network topologies, forwarding each other's packets between various points in the ad-hoc mesh. Unfortunately, most mobile nodes are battery powered, and forwarding a stranger's IP packets has very real costs in terms of battery consumption. There are also problems in the presence of malicious nodes, who consume the network resources for their own benefits.

REFERENCES

[10] Rui Dai and Kang Li, "Shall we stop all unsolicited email messages?" Proc. of the First Conference on Email and Anti-Spam, Mountain View, CA, USA, July 2004