Detecting Spam at the Network Level

Anna Sperotto, Gert Vliek, Ramin Sadre, and Aiko Pras

University of Twente
Centre for Telematics and Information Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
P.O. Box 217, 7500 AE Enschede, The Netherlands
{a.sperotto,r.sadre,a.pras}@utwente.nl,
g.vliek@student.utwente.nl

Abstract. Spam is increasingly a core problem affecting network security and performance. Indeed, it has been estimated that 80% of all email messages are spam. Content-based filters are a commonly deployed countermeasure, but the current research focus is now moving towards the early detection of spamming hosts. This paper investigates if spammers can be detected at the network level, based on just flow data. This problem is challenging, since no information about the content of the email message is available. In this paper we propose a spam detection algorithm, which is able to discriminate between benign and malicious hosts with 92% accuracy.

1 Introduction

Spam is a problem that all Internet users experience in their everyday lives. Symantec Corporation estimates that over 80% of all emails sent in 2008 were spam, a trend that, with a touch of irony, the company considers to be "normal" [1]. The reason we are constantly flooded with unsolicited messages is that spam is profitable. As such, spam detection is likely to remain an "open battlefield" in the coming years.

Nowadays, the most common countermeasures against spam are spam filters. Mail servers usually host the core of spam filtering operations: tools such as Spamassassin [2] reject or accept email messages based on their content. Moreover, many mail clients also locally scan the user's inbox. However, spam messages are designed to look similar to legitimate emails: examples are "phishing" emails that ask you to provide your bank details. Such camouflaging behavior reduces the effectiveness of content-based methods.

We propose a spam detection approach that does not rely on content information. More specifically, our contribution is based on network flows, defined as "a set of IP packets passing an observation point in the network during a certain time interval and having a set of common properties" [3]. These common properties typically include source/destination addresses/ports and protocol type, and they unequivocally define a flow. Flows have recently received great attention in the research community [4], since they allow scalable network monitoring of large infrastructures. Flows typically only report information about the amount of packets and bytes exchanged during a connection, but nothing about the content of the communication.

M. Oliver and S. Sallent (Eds.): EUNICE 2009, LNCS 5733, pp. 208–216, 2009.

In this context, spam detection is a challenge. This paper aims to address the following question: *Is it possible to detect hosts from which spam originates by using just flow data?* More specifically, we want to investigate (a) if spam differs from legitimate SMTP traffic at the flow level and (b) how to detect spam at the flow level. The paper summarizes the results of the MSc thesis of Gert Vliek. More details about the approach can be found in [5].

The general assumption in the research community is that a spammer host will behave differently from a legitimate mail server [6,7,8]. Capturing this behavior at the network level can lead to the development of powerful tools for early spam detection, easing both the server-side load and the filtering in the client. One contribution in this field is the work of Desikan et al. [9], in which the analysis of time-evolving SMTP connection graphs helps distinguish between mail servers and spammers. A different approach is that taken by Ramachandran et al. [6]. The authors' assumption is that the network behavioral patterns of a spamming host are far less variable than the spam content itself. They therefore propose a spam detection approach based on automatic clustering and classification of sender IP addresses that show a similar behavior over a short observation time. More attention to flow approaches has been given in the works of Schatzmann et al. [7,8] and Cheng et al. [10]. In [7,8], the authors suggest that the average number of bytes, packets and bytes/packets of failed, rejected and accepted connections are flow properties suitable for the classification of spam flows. The authors rely on server logs for flow classification. On the other hand, in [10], the authors propose an alternative definition of flows that allows the stateful analysis of spam traffic. Finally, Žádník et al. [11] propose the use of classification trees for spam identification based on flow characteristic.

Compared to the previously mentioned contributions, we propose a spam detection algorithm that relies on Netflow compatible flow data and allows the detection of spamming hosts based on just network characteristics.

This paper is organized as follows. Section 2 describes SMTP traffic from a flow perspective, highlighting the differences between a normal and a suspicious host. In Section 3 we present our spam detection algorithm, followed by a validation of our approach in Section 4. Conclusions are drawn in Section 5.

2 SMTP Traffic at the Flow Level

It is a common assumption that a spamming host's behavior will differ from legitimate SMTP servers. Yet it is interesting to see if this assumption holds in real traffic.

The University of Twente, for example, relies on a system of five load-balanced mail servers, all of them having a similar behavior. Figure 1(a) shows the outgoing SMTP traffic time-series of one of them. Each time slot on the x-axis corresponds to a 5 minutes interval, for a total of five days of observation. There is one main aspect in the mail server behavior. The mail server presents a rather constant activity baseline at around 100 connections per time slot that rarely rises above 250 connections per time slot. This aspect is very significant in our case since it shows that a legitimate mail server is characterized by a steady level of usage. Figure 1(b), on the other hand, shows the outgoing SMTP traffic time series for a host known to have sent spam. Its

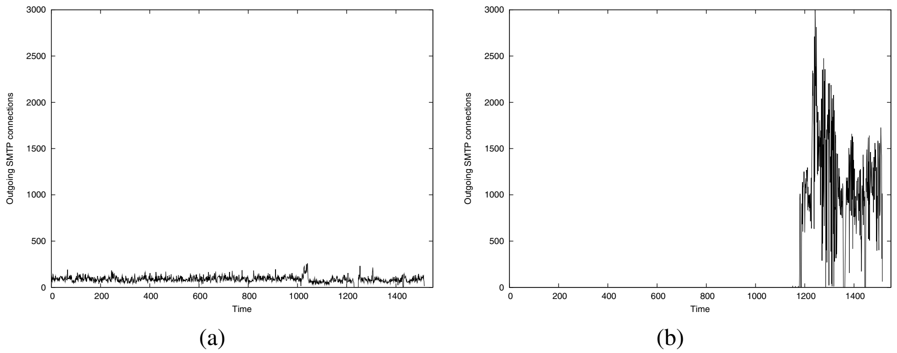

Fig. 1. Flow level behavior for a university mail server (a) and a suspicious machine (b)

network behavior is totally different from the one in Figure 1(a): the time series is characterized by sudden and prolonged activity peaks and a long period in which there is no traffic. Moreover, no usage baseline is present, at the contrary of the mail server. A deeper analysis of the spammer host behavior also reveals that there is no incoming traffic, suggesting that the host has no real traffic exchanges. This behavior is commonly observed in other hosts that have sent spam.

This example suggests that the behavior of suspicious hosts differs substantially from that of legitimate mail servers. Parameters such the incoming and outgoing traffic, as well as the widely variable level of usage can be useful in defining an algorithm for automatic spam detection.

3 An Algorithm for Spam Detection

In the previous section, we showed qualitatively that the network behavior of suspicious and legitimate hosts could be very different. We now propose an algorithm that will detect, based on just flow information, hosts that are most likely to be spammers. The algorithm consists of two main phases and a post-processing step. In the first phase, hosts that do not satisfy three basic selection criteria are filtered out. This phase aims to reduce the amount of data to be analyzed and to improve the overall performance of the algorithm. The hosts selected in the first phase are then ranked in the second phase by means of five ordering criteria according to their likelihood of being spammers. Finally, ranked hosts are once again filtered according to a post-processing criterion. The algorithm analyzes the SMTP traffic sent and received from the network that is monitored. Of course, this means spam traffic generated by a spammer outside the monitored network and targeting a different network cannot be considered for the analysis. However, the results show that it is not necessary to have a complete overview of all the traffic generated by a spammer to achieve a good detection level.

The selection and ordering criteria are explained in Sections 3.1 and 3.2, respectively. The criteria that we propose are based on the analysis of a data set of seven days of SMTP traffic captured at the University of Twente. We describe the resulting algorithm in Section 3.3.

3.1 Selection Criteria

The selection criteria allow us to concentrate, in the second phase, on a smaller subset of hosts. Therefore, in order to be further analyzed, a host has to satisfy all the selection criteria. The selection criteria aim to filter out at an early stage the majority the not-malicious clients. These criteria are defined as follows:

SC$_1$ **Number of outgoing connections:** We only select hosts that exhibit a certain level of activity:

$$\text{number of outgoing SMTP connections} > \theta_1 \tag{1}$$

SC$_2$ **Connection ratio:** A host is suspicious if it sends far more than it receives. The connection ratio criterion is defined as:

$$\frac{\text{number incoming SMTP connections}}{\text{number of outgoing SMTP connections}} < \theta_2 \tag{2}$$

SC$_3$ **Number of distinct destinations:** Criterion **SC**$_1$ could also flag as suspicious a host that relies on SMTP as logging mechanism (as a printer, for instance). Such a host would probably not receive any message. Nevertheless, such host would usually report to only a limited number of destinations, while a spammer would typically diversify its destinations. A threshold for the minimum number of distinct destinations is used for discriminating these cases:

$$\text{number of distinct destinations} > \theta_3 \tag{3}$$

3.2 Ordering Criteria

Once the suspicious hosts have been selected by applying the selection criteria, we apply the ordering criteria to rank them according to their likelihood of being spammers. While the selection criteria are combined into a binary decision, the ordering criteria yield values from a to e that are later combined into a total score for each host.

OC$_1$ **Number of incoming connections:** This criterion is a refinement of **SC**$_2$. We assume that spammers are not interested in receiving SMTP connections. Therefore, a host that does not have any incoming connection is more likely to be a spammer than one that has incoming SMTP traffic. The score is calculated as follows:

$$a = \begin{cases} 1 \text{ if number of incoming SMTP connections} = 0 \\ 0 \text{ otherwise} \end{cases} \tag{4}$$

OC$_2$ **Number of distinct destination:** This criterion is a refinement of **SC**$_3$. We assume that a spammer would try to diversify its destinations. Therefore, hosts with a high number of distinct destinations are suspicious. We define the score b as:

$$b = \begin{cases} 1 \text{ if number of distinct destination servers} > \theta_4 \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

OC$_3$ Percentage of idle time: We assume that hosts with long idle periods are more suspicious than hosts that communicate more regularly over time. We define the score c as:

$$c = \text{percentage of idle time} \tag{6}$$

OC$_4$ Irregularity in activity: Our studies suggest that a suspicious host tends to have a highly irregular transmission pattern. We assume that a host that has a high standard deviation σ of the number of outgoing SMTP flows per 5 minute time slot is more suspicious than one with a low one. We define the score d as:

$$d = \begin{cases} 1 \text{ if } \sigma > \theta_5 \\ 0 \text{ otherwise} \end{cases} \tag{7}$$

OC$_5$ Number of peaks: We assume a suspicious host to show sudden traffic peaks. We define peaks as time slots where the number of outgoing connections is higher than $(\mu + k \cdot \sigma)$. μ and σ are respectively the mean and the standard deviation of the number of outgoing connections per 5 minute time slot for the host, and k is a parameter that influences the sensitivity of the measure. Hosts with a high number of peaks are therefore more suspicious. We define the score e as:

$$e = \begin{cases} 1 \text{ if } |\{\text{slots where connection rate} > (\mu + k \cdot \sigma)\}| > \theta_6 \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

3.3 The Detection Algorithm

Algorithm 1 presents the pseudocode for the detection procedure. As explained earlier, the first phase filters hosts according to the three selection criteria (lines 4 through 6). However, in order to keep the algorithm efficient, we only consider the n most active hosts, in terms of outgoing connections, that satisfy the criteria (lines 3 and 7).

In the second phase, the hosts are scored and ordered according to the ordering criteria (lines 11 through 17). For the overall score v, we calculate the average of the single scores a through e. While ranking the hosts, the algorithm also selects a subset of them that, in conjunction with the ranking, are most likely to be spammers. More specifically, only hosts that are not involved in any traffic exchange for the majority of the observation time γ are considered (line 13). This filtering permits the discrimination between hosts that have a fairly constant behavior and hosts that only transmit in bursts, as for example the hosts in Figure 1.

Finally, the algorithm only reports the m top ranked hosts (line 18). The parameter m allows tuning of the output according to the desired security level.

4 Experimental Results and Validation

Section 4.1 will describe our approach to the validation of our results. Next, in Section 4.2 we will describe our experimental setup and the results we obtained. Finally, Section 4.3 presents a study on the impact of each criterion on the performance of the algorithm.

Algorithm 1. Spam detection procedure

```
procedure SpamDetection($Q$ : host set)
$S_1 = \emptyset$; $S_2 = \emptyset$;
for all $x \in Q$ ordered by decreasing number of outgoing connections do
    if $x$ satisfies $\mathbf{SC_1} \wedge \mathbf{SC_2} \wedge \mathbf{SC_3}$ then
        $S_1 := S_1 \cup \{x\}$;
    end if
    if $|S_1| = n$ then
        break;
    end if
end for
for all $y \in S_1$ do
    Compute $v := \frac{1}{5} \cdot (a + b + c + d + e)$;
    if $c > \gamma$ then
        $S_2 := S_2 \cup \{y\}$;
    end if
end for
Order elements in $S_2$ by decreasing value of $v$;
return top $m$ elements in $S_2$;
```

4.1 Validation Approach

Since we based our algorithm on flows, no information about the content of the SMTP connections is available. We therefore need to rely on external services in order to evaluate our results. DNS blacklists (DNSBL) are Internet services that publish lists of offending IP addresses: in our context, IPs that have been involved in spamming activities. Spam DNSBL are repositories which content is likely to rapidly change over time: indeed, a blacklisted host can be rehabilitated if it is no longer involved in spamming activities for a sufficiently long period. Iverson [12] periodically monitors the most commonly used DNSBL and reports on their reliability.

We selected five DNSBL as trusted sources for validation: `zen.spamhaus.org`, `bl.spamcop.net`, `safe.dnsbl.sorbs.net`, `psbl.surriel.com` and `dnsbl.njabl.org`. We chose this set of DNSBL because they clearly indicate under which conditions a host is going to be added and removed from the list. We define a host to be *positively validated* if it has been blacklisted in at least one of the five DNSBL we are considering.

4.2 Experimental Settings and Results

We evaluate our algorithm over three data sets collected at the University of Twente: a reference data set used to developed the algorithm and two newly collected data sets referred as *Set 1* and *Set 2* in the following. Each data set spans over a period of seven days, with an average of ∼15M flows. The time windows over which the data sets span are not overlapping. The implementation of our approach uses SQL scripts and can process a data set in a period of 5 hours.

In our experiments, we measure the *accuracy* of the method, defined as:

$$accuracy = \frac{|\{\text{positively validated}\}|}{m} \quad (9)$$

where m is the number of hosts reported as output by the algorithm and it can be set according to the desired security level. We decided not to compute the false positive and false negative rate since it is not possible to establish a ground truth. For the hosts that we report as suspicious and that are not listed in any DNSBL, indeed, we are unable to say if they are (a) spammers that are not yet listed (true positive) or (b) normal hosts (false positive).

Table 1. Criteria parameter settings chosen for the experiments

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
θ_1	200	θ_2	0.005	θ_3	5	θ_4	10
θ_5	1	θ_6	50	k	5	γ	80%

Table 1 shows which parameter values have been used in the experiments. The parameters have been manually tuned based on the statistical properties of the reference data set. We measured that only 5% of the hosts we analyzed have more than 200 connections ($\theta_1 = 200$). In a similar way, only 1% of the hosts have more than 10 distinct destinations ($\theta_4 = 10$). Less than 20% of the hosts present a standard deviation $\sigma > 1$ of the number of outgoing connections per time slot ($\theta_5 = 1$). Moreover, only 1% of the hosts have more that 50 peaks (for $k = 5$, $\theta_6 = 50$). The remaining parameters are specific to the network we are analyzing. In particular, as said in Section 2, the University of Twente relies on 5 mail servers. Therefore, we set $\theta_3 = 5$. In our network, high volume SMTP sources might receive a limited amount of incoming connections ($\theta_2 = 0.005$) and, for an observation window of seven days, spammers have shown to be idle for at least 80% of the time ($\gamma = 80\%$). Finally, the algorithm selects n = 20,000 hosts that satisfy the selection criteria and outputs the top m = 100 hosts according to the ordering criteria.

Our experimental results show that, on average, the accuracy of the system is 92%. Table 2 presents the detection accuracy for each of the considered data sets. We observe that our algorithm reaches an overall accuracy of 99% in the reference data set, while the accuracy slowly decreases in the newly collected data sets. This phenomenon suggests that the parameters chosen for our experiments might need to be periodically re-tuned according to spam flow characteristics.

4.3 Criteria Impact

In Section 3 we introduced the criteria we used in our detection algorithm. We now evaluate the impact of each single criterion to the overall detection accuracy of the algorithm. We start evaluating the impact of the only selection criteria **SC**$_1$ and incrementally add one criterion at each run. We measure the overall accuracy on the data set

Table 2. Detection accuracy for the considered data sets

Data set	Time window	Accuracy
Reference set	18 – 24 November 2008	99%
Set 1	2–7 April 2009	96%
Set 2	8–14 April 2009	81%

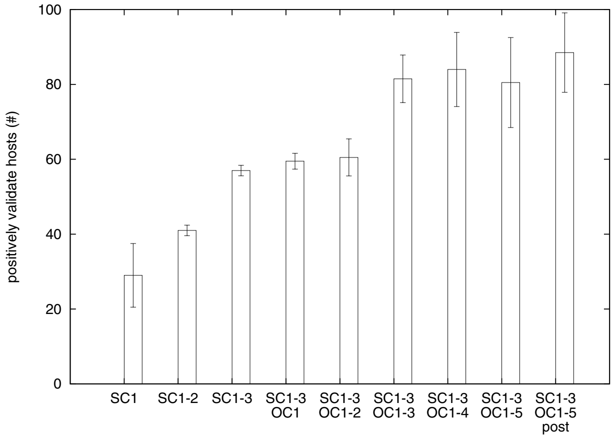

Fig. 2. Impact of the selection and ordering criteria on the overall accuracy

Set 1 and *Set 2* presented in Table 2. Figure 2 shows the average trend of the accuracy curve with respect to the number of applied criteria. The error bars indicate the standard deviation of the number of validated hosts w.r.t. *Set 1* and *Set 2*. Selection criteria $\mathbf{SC}_1$ and ordering criteria $\mathbf{OC}_3$ have the most impact on the accuracy measure, meaning that a high number of connections in a short period of time (bursts) is a key characteristic of a spamming host. Moreover, the accuracy measure presents an increasing trend, meaning that each criterion is beneficial to the detection process. The only exception is $\mathbf{OC}_5$: in data set *Set 1*, indeed, the criterion forces a decrease of the accuracy, suggesting that under certain condition this criterion may report false positives (legitimate hosts flagged as spammers).

5 Conclusions

This paper investigates if it is possible to detect spammers at the flow level, without relying on email content. Our findings show that the network behavior of suspicious hosts differs substantially from that of a legitimate mail server, both in activity level and incoming/outgoing traffic patterns. Based on these observations, we propose a detection algorithm that makes use of just flow information. Our algorithm has been validated using trusted blacklisting services. The results show that we can detect spamming machines with a 92% accuracy for the traces on which we validate our approach, meaning that the algorithm has a low probability to report false positives (host that are not spammers, but they are flagged as such).

Our work is a first step in flow-based spam detection. In the future, we are interested in assessing the *completeness* of our system, in terms of undetected spamming hosts (false negatives). Moreover, we plan to study how our approach behaves in the presence of very peculiar services, for example a server that is only used for mailing lists. It might happen, indeed, that such systems rank high according to our algorithm, suggesting that other metrics can be added to filter them out. We are also interested in extending our approach to different scenarios, for example Botnet detection.

Acknowledgments. This research has been supported by the EC IST-EMANICS Network of Excellence (#26854).

References

1. Symantec Enterprise Security: The state of spam, a monthly report (February 2009)
2. Spamassassin (March 2009), http://spamassassin.apache.org
3. Quittek, J., Zseby, T., Claise, B., Zander, S.: Requirements for IP Flow Information Export (IPFIX). RFC 3917 (Informational)
4. Sperotto, A., Schaffrath, G., Sadre, R., Morariu, C., Pras, A., Stiller, B.: An Overview of IP Flow-based Intrusion Detection. IEEE Surverys & Tutorials (to appear, 2009)
5. Vliek, G.: Detecting spam machines, a Netflow–data based approach. Master's thesis (Feburary 2009), http://essay.utwente.nl/58583/1/scriptie_G_Vliek.pdf
6. Ramachandran, A., Feamster, N., Vempala, S.: Filtering spam with behavioral blacklisting. In: Proc. of the 14th ACM conference on Computer and Communications Security, CCS 2007 (2007)
7. Schatzmann, D., Burkhart, M., Spyropoulos, T.: Flow-level Characteristics of Spam and Ham. Technical Report TIK Report Nr. 291, Computer Engineering and Networks Laboratory, ETH, Zurich (August 2008)
8. Schatzmann, D., Burkhart, M., Spyropoulos, T.: Inferring Spammers in the Network Core. In: Proc. of 10th International Conference on Passive and Active Network Measurement, PAM 2009 (2009)
9. Desikan, P., Srivastava, J.: Analyzing Network Traffic to Detect E–Mail Spamming Machines. In: Proc. of the 2004 ICDM Workshop on Privacy and Security Aspects of Data Mining, PSDM 2004 (2004)
10. Cheng, B.-C., Chen, M.-J., Chu, Y.-S., Chen, A., Yap, S., Fan, K.-P.: SIPS: A stateful and flow-based intrusion prevention system for email applications. In: Li, K., Jesshope, C., Jin, H., Gaudiot, J.-L. (eds.) NPC 2007. LNCS, vol. 4672, pp. 334–343. Springer, Heidelberg (2007)
11. Žádník, M., Michlovský, Z.: Is spam visible in flow-level statistic? Technical report, CESNET 6/2008 (2008)
12. Iverson, A.: Blacklist statistic center (March 2009), http://stats.dnsbl.com/