The Tricks of the Trade: What Makes Spam Campaigns Successful?

Jane Iedemska, Gianluca Stringhini, Richard Kemmerer, Christopher Kruegel, and Giovanni Vigna
University of California, Santa Barbara
{*7_am, gianluca, kemm, chris, vigna*}*@cs.ucsb.edu*

***Abstract*—Spam is a profitable business for cybercriminals, with the revenue of a spam campaign that can be in the order of millions of dollars. For this reason, a wealth of research has been performed on understanding how spamming botnets operate, as well as what the economic model behind spam looks like.**

Running a spamming botnet is a complex task: the spammer needs to manage the infected machines, the spam content being sent, and the email addresses to be targeted, among the rest. In this paper, we try to understand which factors influence the spam delivery process and what characteristics make a spam campaign successful. To this end, we analyzed the data stored on a number of command and control servers of a large spamming botnet, together with the guidelines and suggestions that the botnet creators provide to spammers to improve the performance of their botnet.

I. Introduction

Email spam is one of the main engines that drive the underground economy on the Internet, with large campaigns that are able to earn between $400,000 and $1,000,000 per year [8], [9]. For example, spam sustains a large fraction of the infamous illegal-drug online commerce, and involves many intermediate (dishonest or not) parties, such as payment processors and banks [13], [14].

Nowadays, more than 85% of worldwide spam is sent by *botnets* [22]. Botnets are networks of compromised computers that act under the control of a single entity, known as the botmaster. Cybercriminals typically use more or less sophisticated methods to infect victim machines [17], and they then rent their botnet to spammers, who use it to promote their illicit goods [20]. Spam is such a large burden to the Internet, email servers, and pharmaceutical companies that several take-down operations have been performed to shut down the most aggressive spamming botnets [10], [20]. Unfortunately, due to the many parties involved, the problem is so complex that it is very hard to develop a policy that prevents cybercriminals from getting back in business.

An important aspect of a spamming operation is the performance of the whole botnet. The more emails a botnet is able to send – and to have evade spam filters– the more "customers" (i.e., victims) will receive the illicit advertisements and, potentially, be lured into purchasing the goods. However, setting up a functional botnet is not an easy task. Spammers need to have a good overview of how all the parts that are involved in the email spam process are performing: the command and control channels (C&C), the bots, and the filters on the receiving end. To the best of our knowledge, no research has been conducted on which aspects make a spam campaign successful. However, this information is important for security researchers, because it allows to develop countermeasures that are able to cripple the performance of a botnet.

In this paper, we study what distinguishes a successful spam campaign from a failed one. We analyzed the data contained on 24 command and control server instances from the *Cutwail* spamming botnet. These servers contain detailed statistics about the activity of 17 spammers, for the period between 2007 and 2010. This data allowed us to understand what challenges spammers are facing and how they overcome them. We studied how the Cutwail botmasters help their customers set up a successful campaign, which command and control settings successful spammers use, and what impact the geographic distribution of the botnet has on its spamming performance.

In summary, this paper makes the following contributions:

- The botnet developers provide their customers with a user manual. We translated the manual from Russian and analyzed it, discussing in particular the guidelines that should help the spammer set up a successful spam campaign. Surprisingly, we found that many of the guidelines are mathematically incorrect and provide no help in practice. Apparently, a successful spammer has to rely on experience, more than on hints.
- We study the command and control settings that make a spam campaign successful, such as how the number of bots involved in it influences the campaign's performance.
- We analyze how the geographic location of the bots influences the success of a campaign. Anecdotal evidence shows that bots located in North America are considered capable of sending spam faster, and they are sold for a higher price on the black market [2]. Surprisingly, we found that the most successful spammers bought most of their bots in countries that are supposed to have poorer performance, and, therefore, are cheaper, such as India.

II. Related Work

A wealth of research has been conducted on the underground economy surrounding email spam. Previous research falls in two main categories: *studying the botnet infrastructure* and *studying the spam conversion process*.

Studying the botnet infrastructure. In the past, researchers have infiltrated botnets by reverse engineering the command and control protocol, and writing their own programs that were able to connect and record traffic [3], [5], [11], [12], [18]. This provides very interesting insights, such as the type of spam emails that these botnets sent and how large these botnets were. John et al. presented Botlab, a system able to run and track the activity of spamming bots [6]. Nunnery et al. [15] cooperated with an Internet Service Provider and obtained access to two C&C servers for the *Waledac* botnet, and they studied the information contained on these servers. A similar study was performed by Stone-Gross et al. on the Cutwail botnet [20]. Pathak et al. studied the spam distribution by analyzing several sinkholes based on open relays [16]. Other studies focused on understanding how machines get infected, and how they interact with their botmaster [2], [4], [18], [19]. In this paper, we are interested in how successful spammers operate, regardless of the botnet they use.

Studying the conversion of spam. A wealth of work has been performed on trying to understand how much money spammers make, and how many people purchase the advertised goods [8], [9]. Levchenko et al. studied the whole pipeline of spam purchases, and identified the agents that are involved in the process [13]. In this paper, we are not interested in the conversion of spam, but look at what makes a spam operation successful from the botnet point of view: by having more spam emails delivered, the spammers will have a better chance of generating purchases.

III. The Cutwail Botnet

The Cutwail botnet has been one of the largest spamming botnets over the last years. The structure and operation of this botnet was described in detail in our previous work [20]. The botnet, known as `0bulk Psyche Evolution` on the underground market, can be rented by spammers, who can use it to deliver their malicious or illegal content. The spammer gets access to a web interface on the C&C server, from which he can administer his bots and set up the content of the emails to be sent. The infected machines are typically purchased separately, and the spammer has to instruct them to connect to the C&C server that he rents.

The spammer can specify several different settings that determine how his botnet should operate. In particular, he can set up the email templates that the bots should send out. The emails built from a template advertise the same content, but they include variable fields, to avoid easy detection. We define all the emails built from the same template as a *spam campaign*. A spam operation, which instructs a certain number of bots to send a certain number of emails to a set of victim email addresses is called a *bulk*. A spam campaign is typically composed of several bulks. For each bulk, the bots report to the C&C server detailed information on the errors that they encountered during the email delivery process (such as non-existing email addresses).

In 2010, we obtained access to 24 ready-to-use C&C servers [20]. The servers stored data about each bulk performed by their bots in a database. Therefore, it was possible for us to extract the information about the past spammers' activity. The C&C databases contained records of all bulks for the time period between 2007 and 2010, for 17 different spammers. Each database contained a summary on the bots and bulks, as well as on the detailed settings that the spammer set for each of the bulks. For each bot, the database stored their identifier, IP address, country, the first and the last time they connected to the server, and statistics on the sent messages. For each bulk, the database kept statistics for the delivered and not delivered messages, the number of active bots, and details on the botnet settings used by the spammer. This information allowed us to analyze the importance of the settings and of the geography of the bots used, which we explain in Section V-A.

IV. The Cutwail Manual

The botnet developers provide spammers with a manual to become familiar with the system. Since the botnet developers and most of their customers are located in Russia [20], the manual is written in Russian. We translated the manual into English, and studied the guidelines that are contained in it.

The manual contains a detailed description of the administration web interface and introduces the general principles of how the system works. Since there is a large number of settings that a spammer can customize, the final chapter of the manual provides general guidelines and advice on how to tune the botnet. Some of the claims made in the guidelines are supported by mathematical formulas.

A. Guidelines for Spam Campaigns

The guidelines fall into three categories: those that apply to the message contained in the emails, those that refer to the email database management, and those that apply to the technical settings. In the following, we describe these guidelines in detail.

Email text guidelines. Since the final goal of the spam campaign is to lure victims into believing whatever is written in the spam email, one of the goals of a spammer is to make the email content as convincing as possible. The manual claims that the most effective content for an email would be one that looks like a friendly personal message from one person to another.

Depending on the type of product or service being advertised, the manual suggests using different combinations of plain text and HTML. The manual presents advantages and disadvantages of both HTML and plain text emails and their possible usage. For example, plain text emails would have a small size and would successfully be displayed by any email client. A plain text email will not trigger picture-filter alerts, but for the same reason it would not be as promotional as it could be with pictures. On the other hand, HTML emails could help in passing content filters, but they have a relatively big size and might not be displayed by some email clients (such as on mobile phones).

Email address database guidelines. According to the manual, the first common issue that a spammer could face is that many of the targeted email addresses might not be valid. Previous research noted that this is a big problem for spammers [20], [21]. For a successful campaign, it is important to have as many valid email addresses as possible. However, since most of the email addresses are harvested from the web, there is no guarantee that they will actually be active and reachable.

Thus, it is important to check the database for non-existing email addresses.

The second issue is the distribution of email addresses among domains. If the amount of email addresses per single domain is too small, then resources are wasted, because the bots will have to resolve more domains and set up more connections to the victim servers. On the other hand, if the amount of email addresses per domain is too big, the probability of being detected and blacklisted increases. Spammers are advised to set up their email address lists so that they contain a variety of domains.

The final advice that the manual gives to spammers is to aim to get more out of paid email accounts. This type of email address is considered to be the most valuable, since these accounts are very likely to be in use and their owner should check them with higher probability than any other account. However, the database is not automatically examined for the presence of such email addresses.

Technical guidelines. A problem spammers have to face is how to get their spambots installed on the victims' machines in the first place. Spammers are free to install bots on their own, or to buy infected machines from a third party. However, spammers are strongly encouraged to use the loader provided by the Cutwail developers (Pushdo [20]). The botnet creators claim that other malware installers tend to download different malware alongside the Cutwail bots, and that this affects the bandwidth available to the bot and, therefore, the bulk quality, while the Pushdo loader will only download one bot per machine. We have no information to verify the claim and it could be made only for marketing purposes.

As a second guideline, the manual claims that limiting the number of bots that are sending emails at the same time increases the botnet's performance. The creators of Cutwail have estimated that having 1,000 bots online at the same time generates a good throughput of emails, and that for the best delivery performance the number should be between 2,000 and 3,000 online bots.

The third guideline is about the duration of a bulk. The manual says that the less time the bulk lasts, the more outcome it will have, since spam filters would have less time to update patterns, and less messages will be detected as spam.

B. Mathematical model for the email delivery process

The Cutwail manual includes an extensive mathematical analysis of the botnet operation and the spam delivery process, and it provides spammers with guidelines on how to dimension their botnet and their bulks to obtain optimal delivery results. This mathematical analysis looks very interesting. However, after studying it we found out that the model is invalid. It is possible that it was included to make the work of the botmaster look more professional and trustworthy. It is also possible that it was only meant to give the spammer a qualitative idea of what the important parameters involved in the process are. In the following, we analyze in detail the mathematical description from the manual.

The ratio of the number of email addresses in the database to the duration of the bulk expresses the average rate of a bulk V, which is defined as

$$V = \frac{Q}{T},$$

where Q is the number of email addresses in the database, and T is the duration of the bulk. The current average speed of the bulk V_b is defined as

$$V_b = \frac{Q_b}{T_b},$$

where Q_b is the number of already processed email addresses, and T_b is the time that has passed since the start of the bulk. The average speed of the bulk V_t in the time interval t is defined as

$$V_t = \frac{q(t_0,t_1)}{(t_0-t_1)},$$

where t_0 and t_1 are the start and the end of the time interval, and $q(t_0, t_1)$ is the number of email addresses processed in the time interval (t_0, t_1). The number of emails that are sent by each bot per second v is defined as

$$v = \frac{kBGP}{WL},$$

where B is the average rate of the bulk, G is the fraction of email addresses in the database that actually exist, P is the specific bot rate or number of emails sent per time unit using 100 connections, k is a coefficient that represents the combined influence of all the other settings on the bulk speed, W is the *size* of the email message in bytes, and L is the *time* it takes for a bot to generate the email (from the provided template).

The botnet creators also assume that sending a message error causes a delay in the bulk and, thus, B depends on G. Additionally, they assume that P depends on the quality of the network connection that the average bot has available. Another assumption that the botnet developers make is that B depends on how the email addresses in the database are sorted. Having the email addresses sorted by domain would save DNS lookups to the bots, which would allow them to connect to each server once and send all the spam destined to that domain. On the other hand, having too many emails sent to the same domain by each bot might result in the bots being quickly blacklisted.

From the definition of the average bulk, the authors estimate the rate V, which is given above. This rate equals to vO, where O is the number of bots online. Thus,

$$\frac{Q}{T} = vO, \text{ or } T = \frac{Q}{vO}.$$

From this formula it follows that if $O \to \infty$, then $T \to 0$, which means that the more bots that are online the faster the bulk will be finished. However, the manual claims that there are additional limitations that prevent decreasing the bulk duration by increasing the number of bots. The first limitation includes the overload of the network channel and of the C&C's processing power — the channel has limited bandwidth and the C&C server might be too busy, causing the bots to wait. The second limitation has to deal with the number of bots that the spammer has available and with the number of email addresses that need to be reached. In particular, if N is the number of email addresses in the bulk and n is the number of email addresses assigned to each bot, the number of bots required to complete the task is

$$o = \frac{N}{n}.$$

If $O > o$, this means there are idle bots. This situation is called *online oversaturation*, and it leads to financial losses, because the spammer purchased more bots than he needed. On the other hand, if $O < o$ then there are not enough bots to process all the available tasks. The manual refers to this situation as *online deficit*.

The botnet developers claim that in practice, the number of bots that are needed is

$$o = \frac{N(1+0.2)}{n},$$

where the 20% additional idle bots are required to replace the active ones that get blacklisted, shut down, or are needed to resend emails to those email addresses that reported errors.

The manual concludes that most of the parameters of the system are provided based on a balance between the quantity and the quality of delivery. However, the developers leave the final choice to the spammer.

C. Analysis of the mathematical model

The mathematical model in the manual and its conclusions are not supported by any type of numeric calculations. Therefore, we decided to verify the model. In this section, we present our analysis.

First, we verify the formula for o, and compare it to the values that were actually used by the spammers who rented the C&C servers. Let v be the average rate of emails delivered per bot. As we said:

$$\frac{Q}{T} = vO,$$

where Q is the number of email addresses in the database, T is the duration of the bulk, and O is the number of online bots. Assuming that each bot processes email addresses with average speed v, then each bot processes each portion of email addresses with time

$$t_n = \frac{n}{v},$$

where n is, as before, the number of email addresses assigned to each bot.

In the case that each bulk does not cover the entire email address database, but just a part of it, there are going to be $\frac{Q}{N}$ address portions that will be used in each bulk. If $O \geq \frac{N}{n}$ it means that the number of online bots is larger than needed, and only one sub-portion will be given to each bot. From this it follows that the total time for processing all the email addresses will be:

$$T = \frac{Qn}{Nv} = \frac{Q}{v}\left(\frac{n}{N}\right).$$

Note that the time in this formula does not depend on the number of online bots anymore and reflects the fact that only a fixed number of bots ($\frac{N}{n}$) process the bulk, while other bots remain idle. In the other case, when $O < \frac{N}{n}$, there will be several sub-portions of email addresses given to each bot: $\frac{N}{On}$, since the C&C server gives n email addresses to each bot per bulk. The total time in this case will be:

$$T = \frac{Q}{N}\frac{N}{On}\frac{n}{v} = \frac{Q}{vO}.$$

In this case, the formula does not depend on N or n and it shows that the task is simply divided between the online bots. Therefore, we can generalize the formula as follows

$$T = \frac{Q}{v\min(O, \frac{N}{n})}.$$

The analysis performed in the previous paragraphs shows how the spamming process (number of email addresses, number of bots, etc.) should be dimensioned to achieve good results, according to the botnet manual. We looked at the actual values for these parameters that were used by the spammers who rented the Cutwail C&C servers, to see if these values make sense. The manual suggests that the number of online bots should be $O = 1.2\frac{N}{n}$. The average number of online bots used by the actual Cutwail customers was 2,500. This is in line with the guidelines from the manual, which suggest a number of online bots between 1,000 and 3,000. We then looked at the default settings for N and n, which are $5,000,000$ and $1,000$ respectively. By plugging these numbers into the previous formula, the optimal number of bots O would be 6,000. This shows that the mathematical formulas in the manual are in direct contradiction with both what the C&C software is programmed to do by default and what the spammers ended up doing in the wild.

Let's look at another example of the incorrectness of the mathematical model. We consider $O < \frac{N}{n}$, together with the bulk rate formula.

In the manual, nothing is said about how the bots generate the email message. It is unclear whether the bot generates each message from a template and sends it, or whether it sends the message while generating the next one. Therefore, we have considered both cases. For the first case, suppose that each bot generates a template, connects to the mail server, and uploads the email to the server. The time it will take to generate and send the set of n email addresses is

$$t_n = t_{gen} + t_{upload} + t_{pr},$$

where t_{gen} is the time to generate the emails for their templates, t_{upload} is the time to upload them to the victim mail servers, and t_{pr} is the additional time needed to process them. Let's say that the size of the message is W bytes and a bot generates and uploads v_{gen} and v_{upload} bytes per second. A bot can also process responses from the mail server at a rate of v_{pr} emails per second. In total, a bot has an average speed of processing of v_{av} emails per second. If we have n emails, from the previous equation it follows that:

$$\frac{n}{v_{av}} = \frac{nW}{v_{gen}} + \frac{nW}{v_{upload}} + \frac{n}{v_{pr}}, \ \frac{1}{v_{av}} = \frac{W}{v_{gen}} + \frac{W}{v_{upload}} + \frac{1}{v_{pr}}.$$

From this equation we see that:

$$W \to 0 \Rightarrow v_{av} \to v_{pr}.$$

We can get the same result for the other case, when a bot generates spam emails and sends them in parallel. In this case, the time for processing tasks would be:

$$t_n = \max(t_{gen}; t_{upload} + t_{pr}),$$

and the average speed for the n emails of W bytes correspondingly would be:

$$\frac{n}{v_{av}} = \max(\frac{nW}{v_{gen}}; \frac{nW}{v_{upload}} + \frac{n}{v_{pr}}),$$
$$\frac{1}{v_{av}} = \max(\frac{W}{v_{gen}}; \frac{W}{v_{upload}} + \frac{1}{v_{pr}}).$$

Again,

$$W \to 0 \Rightarrow v_{av} \to v_{pr}.$$

However, in the original formula it was:

$$W \to 0 \Rightarrow v_{av} \to 0.$$

This limit case shows that there is a contradiction with the intuitive expectations on how the model works.

Let L be the time needed by the bot to generate one message. The other contradiction is related to the proper relationship between v_{av}, W , and L. From our equations, it follows that:

$$v_{av} = \frac{1}{\alpha W + const} = \frac{1}{L + const},$$

where W is the size of the email message sent (in bytes) and L is the time needed by a bot to generate the email. However, in the original formula, we have:

$$v_{av} \sim \frac{1}{WL} \text{ or } v_{av} \sim \frac{1}{W^2}.$$

This is the second contradiction. In this formula, the variables are not related correctly, and, thus, the mathematical description is invalid. Therefore, we conclude that the mathematical model presented in the manual is partially invalid and does not give correct predictions. This means that the developers tried to make their work look more solid than it actually is, and successful spammers set the botnet parameters based on their experience, rather than on this mathematical model.

V. Successful Spam Campaigns

We have shown that the Cutwail manual does not provide accurate information on the importance of the settings that a spammer can tune to enhance the performance of his botnet.

First of all, a large number of technical parameters, despite being listed, have neither an explanation of their influence on the quality of a bulk, nor of what effect they cause. Moreover, the botnet developers ask spammers not to try to tune anything without a clear understanding of the parameters functionality. However, there still remains a question on whether any of the settings is important. To investigate this aspect, we performed our own study of the data logged by the C&C servers.

A. Analysis of the Spam Settings

We believe that if researchers gained a good understanding of what determines the success of a spam campaign, they could develop novel techniques to mitigate the activity of spamming botnets. For these reasons, we analyzed a number of different campaigns that were performed by the Cutwail customers to understand which settings play an important role in the performance of a bulk.

For our analysis, we selected two classes of bulks. We define a successful bulk as follows: a bulk is successful if the spammer was able to send a large part of the emails without receiving errors in return. This is because previous work showed that due to very small conversion rates, spammers must target as many victims as they can [8].

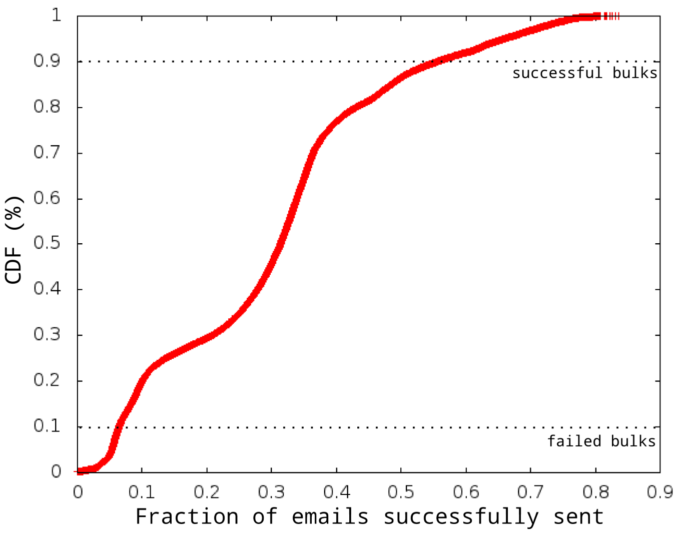

Figure 1: Cumulative distribution function (CDF) of successfully sent emails for bulks.

	Successful bulks	Failed bulks
Avg # of online bots	3,458	5,145
Sent emails	25,111,335,857	5,627,237,918
Avg # of addresses in the database	11,297,676	47,819,013

Table I: Aggregated statistics for the successful and failed bulks.

Based on these observations, we select failed and successful bulks based on the fraction of emails that they successfully sent. Of course, there is no guarantee that, after an email gets delivered, it will actually go through the post-delivery anti-spam mechanisms that the victim might have in place (e.g., SpamAssassin [1]). Since we are studying the email delivery capability of the spam operation, and not the conversion, we consider these assumptions to be good for our purpose.

Figure 1 shows the cumulative frequency distribution (CDF) of the ratio of successfully sent emails for the bulks in our dataset. For our analysis, we selected the top 10% bulks as successful, and the bottom 10% as failed.

The aggregated statistics for the two classes are presented in Table I. One can see that successful spammers managed to send five times more emails than the unsuccessful ones. They also used email databases almost five times smaller and almost twice less bots than the spammers that failed. These numbers are probably due to how the botnet works: pruning the email list of non-existing email addresses is likely to generate more compact lists, while the botnet can effectively handle a certain number of bots, and adding more bots results in wasted resources. Intuitively, a spammer that clearly has these two concepts in mind is more likely to succeed in his operations than one who does not.

Number of emails
Number of online bots
Duration of the bulk
Tries before giving up
Retry in case of error (12 features)
Block bot in case of error
Various settings (48 features)

Table II: Different settings that a spammer can tune for a bulk.

In our analysis, we have excluded the number of invalid email addresses from the total number of email addresses in the database. This allows us to evaluate how many messages were delivered in proportion to the number of valid email addresses and to exclude the influence of email databases with poor quality.

Table II shows the different parameters that a spammer can set for his bulks. Many of the parameters deal with whether the bot should retry sending an email in case it received a specific error (and if yes, how many times), and with various settings of the TCP connections opened by the bots (for example, after how much time a bot should give up in waiting for a response). In addition, as we said in Section IV-B, the botnet operators recommend specific values for the number of emails per bulk and the number of online bots, but they advise the spammers not to change any of the other settings. However, as it turns out, some of the spammers modified these settings anyway, and the outcome of their bulks varied a lot. Therefore, we performed an analysis on these settings, to determine which ones can influence the performance of a bulk.

For our analysis we used the Weka framework and its Sequential Minimal Optimization (SMO) machine learning algorithm. This tool helped us create a classification model and to establish the importance of the different parameters. The output of Weka is represented by a weight vector for each bulk. In the case of the linear classifier, the weights can be seen as the measure of influence and how important certain parameters are for the bulk to be in one of the two classes.

The classification model lets us observe which parameters were used in successful bulks. Interestingly, the setting that most successful campaigns had tuned are the number of times a bot should try sending an email after receiving a network error or a server timeout. The reason for this might be that bot-infected machines often times have bad Internet connections and experience more network errors than typical email clients. This has already been noted by previous research, and has been leveraged for spam detection [7]. Thus, if bots are instructed to try again upon receiving an error, instead of giving up right away, this increases the chance of an email being eventually sent.

Surprisingly, the duration of the bulk, or the number of online bots do not seem to influence the final outcome of a bulk. In fact, both successful and unsuccessful spammers used a variety of configurations for these parameters. The fact that the botnet developers give guidelines on how many bots and emails should be sent at the same time might just be to make their customers rent more C&C servers, instead of giving them actual suggestions on how to set up successful bulks.

Country	**% of the bots**
India	24.9
Russian Federation	4.5
Australia	3.8
Ukraine	2.2
Turkey	1.9
Brazil	1.7
Korea, Republic of	1.5
Romania	1.3
Philippines	1.3

Table III: Geographic distribution of the bots in successful bulks.

Country	**% of the bots**
India	15.8
Brazil	9.0
United States	8.6
Mexico	4.0
Australia	3.0
Russian Federation	2.7
United Kingdom	1.9
Colombia	1.7
Korea, Republic of	1.5

Table IV: Geographic distribution of the bots in failed bulks.

B. Bot Geographic Distribution

Previous research reported that bots located in certain countries are sold for a higher price on the black market [20]. The idea is that bots located in developed countries will have a better Internet connection, and therefore be able to send more spam in the same amount of time. To check whether the physical location of a bot actually makes a difference in its spamming capability, we made an aggregated geographic distribution for each of the successful and failed bulk classes, to see whether there are any preferences in the bot location. Intuitively, if successful spammers picked bots from those countries that are considered more expensive, we would have evidence that the geographic location of bots might actually matter.

Statistics for the two classes are presented in Table III and Table IV. The statistics show that successful bulks have most of the bots from countries with relatively low bot prices. On the other hand, spammers who launched failed bulks were using a big percentage of bots from the United States and the United Kingdom, which have the most expensive bots. We have found that 30% of the spammers who sent failed bulks use more than 3% of their bots from the United States, and 25% of them use more than 1% of their bots from the United Kingdom. On the other hand, only 2.7% of the spammers who performed successful campaigns used more than 3% of their bots from the United States and 2.7% of these spammers used more than 1% of their bots from the United Kingdom.

Our results show that the geographic location of the bots does not play a big role in the quality of a bulk. This is in contradiction with the market price, which is set by customer demand. It might be that having a bot in a richer country has advantages in some cases, such as for an information-stealing botnet. However, in the case of a spamming botnet, it does not seem to play a relevant role.

VI. Discussion

The results discussed in this paper are somewhat surprising, and give us new insights into the underground economy and the dynamics of spam operations. The most interesting result is that the location of the bots does not influence the success of a spam campaign. A consequence of this is that the prices of malware infections that are offered in the underground market are inflated, since there is no advantage for a cybercriminal to purchase the most expensive bots and have them sending spam.

Other elements that we discovered could be leveraged for spam mitigation. For example, given that successful spammers will have their bots retrying multiple times after receiving an error, one could leverage previous work to identify a spambot and keep sending them errors until they give up [7]. This would decrease the performance of a bulk, because bots would keep connecting to a certain server instead of sending emails to other victims.

VII. Conclusions

In this paper, we have analyzed the parameters that make a spam campaign successful. We have shown that botmasters provide their customers with detailed mathematical models for the botnet operation, but that these models are of little if any help for spammers. Instead, experience seems to be what matters most for a spammer, and by manually tuning a botnet parameters a cybercriminal can dramatically increase the outcome of his spam operations. We have also shown that parameters that are commonly believed to influence the outcome of a spam campaign, such as the physical location of the bots or the number of bots online, actually do not matter much in the result of a spam operation. Future work will focus on studying ways to leverage the observations made in this paper for spam mitigation.

VIII. Acknowledgments

This work was supported by the Office of Naval Research (ONR) under grant N000140911042, the Army Research Office (ARO) under grant W911NF0910553, and Secure Business Austria.

References

[1] Apache Foundation. Spamassassin. http://spamassassin.apache.org.

[2] J. Caballero, C. Grier, C. Kreibich, and V. Paxson. Measuring Pay-per-Install: The Commoditization of Malware Distribution. In *USENIX Security Symposium*, 2011.

[3] J. Caballero, P. Poosankam, C. Kreibich, and D. Song. Dispatcher: Enabling Active Botnet Infiltration Using Automatic Protocol Reverse-engineering. In *ACM Conference on Computer and Communications Security (CCS)*, 2009.

[4] K. Chiang and L. Lloyd. A Case Study of the Rustock Rootkit and Spam Bot. In *USENIX Workshop on Hot Topics in Understanding Botnets (HOTBOTS)*, 2007.

[5] C. Cho, J. Caballero, C. Grier, V. Paxson, and D. Song. Insights from the Inside: A View of Botnet Management from Infiltration. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2010.

[6] J. P. John, A. Moshchuk, S. D. Gribble, and A. Krishnamurthy. Studying Spamming Botnets Using Botlab. In *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2009.

[7] G. Kakavelakis, R. Beverly, and Y. J. Auto-learning of SMTP TCP Transport-Layer Features for Spam and Abusive Message Detection. In *USENIX Large Installation System Administration Conference (LISA)*, 2011.

[8] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. Voelker, V. Paxson, and S. Savage. Spamalytics: An Empirical Analysis of Spam Marketing Conversion. In *ACM Conference on Computer and Communications Security (CCS)*, 2008.

[9] C. Kanich, N. Weaver, D. McCoy, T. Halvorson, C. Kreibich, K. Levchenko, V. Paxson, G. Voelker, and S. Savage. Show Me the Money: Characterizing Spam-advertised Revenue. *USENIX Security Symposium*, 2011.

[10] B. Krebs. Taking Stock of Rustock. http://krebsonsecurity.com/2011/01/taking-stock-of-rustock/, 2011.

[11] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. On the Spam Campaign Trail. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2008.

[12] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamcraft: An Inside Look at Spam Campaign Orchestration. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2009.

[13] K. Levchenko, A. Pitsillidis, N. Chachra, B. Enright, M. Félegyházi, C. Grier, T. Halvorson, C. Kanich, C. Kreibich, H. Liu, et al. Click trajectories: End-to-end analysis of the spam value chain. In *IEEE Symposium on Security and Privacy*, 2011.

[14] D. McCoy, P. A., G. Jordan, N. Weaver, C. Kreibich, B. Krebs, J. Voelker, S. Savage, and K. Levchenko. PharmaLeaks: Understanding the Business of Online Pharmaceutical Affiliate Programs. In *USENIX Security Symposium*, 2012.

[15] C. Nunnery, G. Sinclair, and B. B. Kang. Tumbling Down the Rabbit Hole: Exploring the Idiosyncrasies of Botmaster Systems in a Multi-Tier Botnet Infrastructure. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2010.

[16] A. Pathak, Y. C. Hu, and Z. M. Mao. Peeking into spammer behavior from a unique vantage point. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2008.

[17] N. Provos, P. Mavrommatis, M. A. Rajab, and F. Monrose. All your iFRAMEs point to Us. In *USENIX Security Symposium*, 2008.

[18] B. Stock, J. Gobel, M. Engelberth, F. Freiling, and T. Holz. Walowdac Analysis of a Peer-to-Peer Botnet. In *European Conference on Computer Network Defense (EC2ND)*, 2009.

[19] B. Stone-Gross, M. Cova, C. Kruegel, and G. Vigna. Peering Through the iFrame. In *IEEE Conference on Computer Communications (INFOCOM)*, 2011.

[20] B. Stone-Gross, T. Holz, G. Stringhini, and G. Vigna. The Underground Economy of Spam: A Botmaster's Perspective of Coordinating Large-Scale Spam Campaigns. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2011.

[21] G. Stringhini, M. Egele, A. Zarras, T. Holz, C. Kruegel, and G. Vigna. B@BEL: Leveraging Email Delivery for Spam Mitigation. In *USENIX Security Symposium*, 2012.

[22] Symantec Corp. State of spam & phishing report. http://www.symantec.com/business/theme.jsp?themeid=state_of_spam, 2010.