A Software Framework for Spam Campaign Detection and Analysis

Prepared by:
Dr. Mourad Debbabi
Mr. Francis Fortin
Son Dinh
Taher Azab

Scientific Authority:
Rodney Howes
DRDC Centre for Security Science

Defence Research and Development Canada
Contract Report
DRDC-RDDC 2014-C52
April 2014

IMPORTANT INFORMATIVE STATEMENTS

This project Algorithm Development for Large Volume Spam Detection CSSP-2012-CD-1027 was supported by the Canadian Safety and Security Program (CSSP) which is led by Defence Research and Development Canada's Centre for Security Science, in partnership with Public Safety Canada. Partners in the project include CRTC, Industry Canada, National Cyber Forensics Training Alliance (NCFTA), and RCMP.

The CSSP is a federally-funded program to strengthen Canada's ability to anticipate, prevent/mitigate, prepare for, respond to, and recover from natural disasters, serious accidents, crime and terrorism through the convergence of science and technology with policy, operations and intelligence.

National Cyber-Forensics and Training Alliance (NCFTA) Canada

In Collaboration with

Canadian Radio-television and Telecommunications Commission (CRTC)

Technical Report

A Software Framework for Spam Campaign Detection and Analysis

Supervised by: Dr. Mourad Debbabi
Mr. Francis Fortin

Executed by: Son Dinh
Taher Azab

Contents

List of Figures

Acknowledgments

The research reported in this document is the result of a fruitful collaboration between National Cyber-Forensics & Training Alliance (NCFTA) Canada and Canadian Radio-television and Telecommunications Commission (CRTC).

1 Introduction

Every infrastructure is susceptible to abuse, email systems are not an exception. Spam, defined by Spamhaus [2] as "unsolicited bulk email" (UBE) [3], has become one of the biggest sources of email systems exploitation. Studies have shown that the number of spam emails is astonishing. For instance, according to Symantec Intelligence Report, the global ratio of spam in email traffic is 64.1% or 1 in 1.56 emails is spam [4]. Spam has affected almost all users around the world and wasted lots of storage space and network bandwidth. In addition, it has become a major tool for criminals to conduct illegal activities on the internet, such as stealing sensitive information, selling counterfeit goods, distributing malware and child pornography, etc.

Despite the damages caused by spam, it is difficult for investigators and law enforcement agencies to track spammers and stop their malicious activities. This is mainly due to the astronomical amount of spam data, which makes its analysis by humans almost impossible. In addition, the strategies that are used by spammers to obfuscate the content of spam messages are various and in constant evolution, which make the investigation task even harder. Thus, it is absolutely necessary to perform an analysis of the information available on spam to determine its relative value for investigation.

A vast range of studies have been conducted by security researchers on spam to mitigate its effect. However, it is still a cat-and-mouse game, where spammers continue to discover new techniques to evade anti-spam methods. Moreover, most existing work on spam has focused on detecting spam from large corpus email messages. Advanced forensic analysis of spam for the purpose of cyber crime investigation is still missing in the literature. In this report, we propose a new methodology and tool for forensic spam analysis that can be used by investigators to enforce Canada's Anti-Spam Legislation [5].

1.1 Project Objectives

The primary objectives of this research are to:

- Consolidate spam messages into campaigns using different machine learning techniques.
- Rank spam campaigns with a focus on campaigns that are executed by Canadian spammers or abuse Canadian information system infrastructures including servers in Canada and .ca domain.
- Cluster spam campaigns into different categories to help investigators concentrate on more severe campaigns, e.g., child porn, malware distribution and phishing.

1.2 Contributions

In this research, we provide a methodology and tool for detecting and characterizing spam campaigns. More precisely:

- We consolidate spam messages into campaigns based on relevant features extracted from spam data.
- We correlate the campaigns using extra information such as geo-location, malware, WHOIS information, and passive DNS.

- We rank and score the campaigns.
- We label and categorize the campaigns.

The identification of spam campaigns is a crucial step for analyzing spammers' strategies for the following reasons. First, the amount of spam data is astronomical, and analyzing all spam messages is costly and almost impossible. Hence, clustering spam data into campaigns reduces significantly the amount of data to be analyzed, while maintaining their key characteristics. Second, because of the characteristics of spam, spam messages are usually sent in bulk with specific purposes. Hence, by clustering spam messages into campaigns, we can extract relevant insights that can help investigators understand how spammers obfuscate and disseminate their messages.

Geo-locating the campaigns helps investigators prioritize them based on their local locations. For example, Canadian law enforcers can focus on spam campaigns that are originated from Canadian spammers or compromised hosts in Canada. In addition, scoring and ranking along with categorizing and labeling the campaigns significantly increase the effectiveness of an investigation. An investigator can specify which attributes are important to his/her investigation and assign larger score to those attributes. Thus, campaigns with high score and special characteristics will attract more attention from the investigator. Categorizing and labeling help the investigator to further understand the background behind a campaign. Therefore he/she can focus on spam campaigns that cause more damage (e.g. malware distribution, phishing).

1.3 Report Organization

The remainder of this report is structured as follow. Section 2 provides the background information that is related to the topics addressed in this research. In Section 3, we present an overview of the state-of-the-art, methods and tools that are related to this research. The overall architecture of our approach for detecting and characterizing spam campaigns is presented in Section 4, followed by a detailed methodology in Section 5. Some statistics and experimental results are provided in Section 6. Finally, we conclude the report in Section 7.

2 Background

This section provides the fundamental concepts and definitions that are related to spam and anti-spam topics. It also elaborates some machine learning methods that are commonly used for filtering spam.

2.1 Spam

In the context of Email, the word "spam" means "unsolicited bulk email (UBE)" [3]. Technically, "an electronic message is spam if: (A) the recipient's personal identity and context are irrelevant because the message is equally applicable to many other potential recipients, and (B) the recipient has not verifiably granted deliberate, explicit, and still-revocable permission for it to be sent [3]." In addition, ACMA [6] defines spam as "any message that doesn't meet three conditions: (A) Consent—the message must be sent with your consent, (B) Identify—the message must contain accurate information about the person or the organization that authorized the sending of the message, and (C) Unsubscribe—the message

must contain a functional 'unsubscribe' facility to allow you to opt out of receiving messages from that source [7]." According to ACMA, "messages do not have to be sent out in bulk to be considered spam. Under Australian law, a single electronic message can also be considered spam [7]." Canada's Anti-Spam Legislation [5] provides a more general definition of spam—"Spam generally refers to the use of electronic messaging systems to send unsolicited and bulk messages. Spam messages may contain deceptive content, support illegal activities and may also be used to deliver electronic threats such as spyware and viruses [8]."

2.2 Spamtrap

A spamtrap is an email address that is created not for the purpose of sending or receiving emails but to collect spam. Spamtraps do not belong to real users and are never published where a human can find them. Spamtraps are obtained by spammers through the use of automated harvesters, dictionary attackers or buying lists from 'black markets' [9]. The spam messages that are captured by spamtraps can be used to filter spam by automated anti-spam systems. Moreover, the source IP addresses of the messages received by spamtraps can be added to blacklists to block further emails originated from those IP addresses. However, if a spamtrap is revealed by spammers, the spammers can exploit it to have some control over the process of the automated anti-spam system. A spamtrap, once exposed, can also be used as a backscattered email address to send spam.

2.3 Spam Campaigns

A spam campaign is a group of messages, which are sent to achieve a specific goal such as stealing sensitive information, advertising for a specific product, etc. Spam campaigns are generally identified by clustering spam messages hierarchically, where messages that share similar features such as subject, URL, layout, are grouped together. Usually, spam messages that belong to the same campaign are generated using the same obfuscation mechanisms, which comprise both content obfuscation and network exploitation strategies [10]. Therefore, the identification of spam campaigns is an important step in the analysis of spam data as it significantly reduces the amount of data that should be analyzed by investigators. In addition, it reveals important information about spammers' strategies that would not be possible by analyzing each single message separately.

2.4 Clustering

Clustering is a data mining technique used to partition a set of data (or objects) into groups called clusters. The purpose of clustering is to group similar or related objects into the same group to be able to treat them collectively. Clustering can be achieved using different methods, such as techniques based on the analysis of frequent patterns (e.g., Frequent Pattern Tree (or FP-Tree) [11]), techniques that consider user-specified or application-specific constraints, and link-based techniques.

2.5 Classification

Classification is a data mining technique that is used to predict the unknown class of a variable based on known classes of other variables. The input data, also called the training set, consists of several records

(e.g., observations, measurements), each having multiple attributes or features. Each record is assigned a label indicating the class of the observations. The objective of classification is to analyze the training set and build an accurate model for each class using the features present in the data. Afterwards, this model will be used to classify new data, for which the class descriptions are not known. Several classification methods are proposed in the literature. Examples of such methods include decision trees, rule-based methods, neural networks, bayesian classification, support vector machines, etc.

2.6 Content-based Filtering

Content-based filtering is one of the mainstream classification techniques used to filter spam messages. The idea is to filter spam messages by checking some features and the information available in the content of the messages (e.g., keywords, regular expressions). For example, to filter spam messages advertising "Viagra" products, the administrator might place this word in the filter configuration. The mail server would then reject any message containing this word. Content-based filtering techniques can be classified into rule-based filtering and statistical-based filtering.

2.6.1 Rule-based filtering

Rule-based filtering relies on the specification of a set of heuristic rules for detecting spam. These rules identify patterns such as specific words and phrases, special characters, malformed headers, dates in the future or in the past, etc. Based on these rules, a score, which represents the possibility of being spam, is generated for each message. A message is categorized as legitimate or spam based on a specific score threshold. Although this technique can be quite effective, it has many disadvantages. The most important one is that rule-based filters tend to have high false positive rates. For example, a system administrator who attempts to reject spam messages advertising a product may inadvertently block legitimate emails on the same subject. In addition, since these rules are static, spammers constantly change the content of their messages to avoid being detected. Advanced spammers can even test their emails on popular rule-based filters before sending them. As a consequence, system administrators need to specify a large number of rules and maintain them regularly.

2.6.2 Statistical-based filtering

Statistical-based filtering detects spam messages in terms of the likelihood of certain events. This filtering technique requires no administrative maintenance; instead, users mark email messages as legitimate or spam and the spam filters decide which messages to filter based on these judgements. Typical statistical-based filters use single words to decide if a message should be classified as spam or not. However, a more powerful calculation can be made using groups of two or more words taken together. An advantage of statistical-based filtering compared to rule-based one is that it can quickly handle the changes in spam content, without administrative intervention, as long as users consistently designate false negative messages as spam when received in their email. Statistical-based filters can also look at message headers, thereby considering not just the content but also peculiarities of the transport mechanism of the email. Using statistical-based filtering, a biochemist who is researching "Viagra" won't have messages containing the word "Viagra" automatically flagged as spam, because "Viagra" will show up often in his or her legitimate messages. But still, spam emails containing the word "Viagra" do get

filtered because the content of the rest of the spam messages differs significantly from the content of legitimate messages [12].

2.7 Association Rule Learning

Association rule learning is a data mining technique that is used for identifying relationships among a set of items in a database based on the co-occurrence of the data items [13, 14]. Association rules are useful in many application domains including marketing, Web usage mining, intrusion detection, and bioinformatics. For example, in the case of marketing, association rules allow to estimate the probability that customers will buy a product A given a list of other products they have purchased. Association rules are usually required to satisfy a user-specified minimum support and a user-specified minimum confidence at the same time. Association rule generation is usually split up into two separate steps: First, minimum support is applied to find all frequent itemsets in a database. Second, these frequent itemsets and the minimum confidence constraint are used to form the rules. Many algorithms have been proposed for generating association rules. The most important ones are Apriori [14], Eclat [15], and FP-Growth [16].

2.8 FP-Growth Algorithm

FP-Growth is a data-mining algorithm that is used for discovering association rules between items in large datasets. It includes two main steps: FP-Tree Construction and Frequent Itemset Generation. In the first step, the algorithm builds a compact data structure, called FP-Tree, using two passes over the dataset. In the first pass, the algorithm scans the dataset and counts the number of occurrences of each item in the dataset. In the second pass, the FP-Tree structure is constructed by inserting instances from the dataset. Items in each instance are sorted in a decreasing order based on their frequency in the dataset, while infrequent items are discarded so that the tree can be processed quickly. In the second step, the FP-Growth algorithm extracts frequent items from the FP-Tree. It starts from the bottom of the tree by finding all instances matching a given condition. Then, each prefix path sub-tree is processed recursively to extract the frequent itemsets. This construction method allows item sets that have several common features to arise naturally within the FP-Tree, instead of generating candidate items and testing them against the entire dataset. Indeed, using this technique, items that have most features in common share the same path in the tree. The root of the tree is the only empty node, separating items that have no features in common. The FP-Tree usually has a smaller size than the uncompressed dataset since items that share similar features are grouped together. Hence, this compressed version reduces significantly the amount of data that should be analyzed, while maintaining the characteristics of the items.

2.9 Frequent-Pattern Tree

A Frequent-Pattern Tree (or FP-Tree in short) is a tree structure that is defined as follows [17]:

- It consists of one root labeled as “null”, a set of item-prefix sub-trees as the children of the root, and a frequent-item-header table.

- Each node in the item-prefix sub-tree consists of three fields: (1) *item-name*, (2) *count*, and (3) *node-link*. The field *item-name* registers the item that this node represents. The field *count* registers the number of transactions represented by the portion of the path reaching this node. The field *node-link* links to the next node in the FP-Tree carrying the same *item-name*, or *null* if there is none.
- Each entry in the frequent-item-header table consists of two fields, (1) *item-name* and (2) *head of node-link* (a pointer pointing to the first node in the FP-Tree carrying the item-name).

3 Literature Review

A vast majority of approaches have been proposed in the literature for grouping spam messages into campaigns. These approaches have adopted different techniques, such as techniques that consider URL information [18, 19, 20, 21], signature-based approaches [22, 23], techniques that compute similarities between spam images [24, 25], and techniques based on FP-Tree [10]. However, most of these approaches are for the purpose of detecting spam messages and blocking them. In the following, we provide an overview of the main approaches.

3.1 An Empirical Study of Clustering Behaviour of Spammers

F. Li *et al.* [19] have proposed an approach for clustering spam emails based on URLs. The authors believe that spam emails with identical URLs are highly clusterable and mostly sent in burst. The authors also consider the money amount mentioned inside the body of emails as another factor for clustering spam emails. However, the small size of these clusters shows that money alone is not an appropriate feature for clustering and is better to be considered as a complementary criteria with URLs. In this approach, if the same URL exists in the spam emails from a source A and a source B, and each has a unique IP address, then they will be connected with an edge to each other. The key observation is that if a spammer is associated with multiple groups, it is more probable that he/she will send more spam emails in the future. From empirical results, 2% of active spammers send near 20% of total spams. In addition, it seems sensible to score them higher in order to block these "highly active" spammers who are associated to multiple groups.

3.2 FP-Tree for Clustering Spam Emails

The basic hypothesis of the FP-Tree method for clustering spam emails is that some parts of spam messages are static and can be used for recognizing a spam campaign. H. Calais *et al.* [10] have focused on four groups of criteria that could be analyzed after identifying spam campaigns: Source and destination IPs, type of abuse (proxies if they are HTTP, SOCKS or relays), and the content obfuscation strategy. The campaign identification is organized into two main steps. In the first step, relevant features (e.g., language, layout, message type, URL, and subject) are extracted from each spam message. Then, in the second step, these features are used to build a frequent pattern tree [11], where each node of the tree represents a feature that is shared by all the sub-trees beneath. Based on that, a spam campaign is obtained by grouping all spam messages that have several frequent features in common and differ only in infrequent features. From the FP-Tree, it is inferred that the *language* feature is the first criteria

that is selected for clustering campaigns, after that comes *message type* and *message format*. Also, not surprisingly, there is a strong correlation between the language of the spam message and the location of the sender. Regarding campaign characterization, this approach uses data mining techniques, such as clustering and association rule mining, to analyze spammers' strategies both in terms of content obfuscation and exploitation of network resources. In particular, the authors found strong relations between the origin of the spam and how it abused the network, and also between operating systems and types of abuse.

3.3 Spamscatter: Characterizing Internet Scam Hosting Infrastructure

Anderson *et al.* [25] have introduced a new technique, named spamscatter, which automatically clusters destination Web sites extracted from URLs in spam emails, with the use of image shingling. The latter is a technique that uses the same idea as text shingling, i.e., it divides images into blocks and then hashes the blocks. Two images are considered similar if 70% of the hashed blocks are the same. In the analysis stage, scams (i.e., Web sites marketing with the use of spam) are identified across servers and domains. In addition, distributed and shared infrastructure, lifetime, stability and location are also reported. Characterizing the scam infrastructure helps to reveal the dynamics and the business pressures exerted on spammers and also to identify means to reduce unwanted sites and spam. Moreover, the geographical location seems to be more important for scammers rather than spammers, and is mostly located in U.S. (about 57%). The authors traced domains and concluded the infrequency of IP addresses, that does not seem correct, since spammers improve their hosting infrastructure for protecting the servers. They also mentioned that most host scams are short-lived campaigns. However a spammer may change DNS entries and so point to the Web site by another IP address; so ending a host or domain name may not mean termination of a campaign work [26].

3.4 O-means Clustering Method for Analyzing Spam Based Attacks

J. Song *et al.* [20] have focused on clustering spam emails based on IP addresses resolved from URLs inside the body of these emails. Two emails are regarded as the same cluster if their IP address sets, resolved from URLs, are exactly the same. By examining spam gathered on their SMTP server during three weeks, the authors have concluded that this method outperforms comparing to clustering techniques based on domain names and URLs. They have justified their claim on the basis that domain names associated with scam change frequently. In addition, the period that a URL is active is too short for investigation, and mostly, URLs used in spam emails are unique.

In another work [27], the authors mentioned that considering only IP addresses resolved from URLs is insufficient for clustering, since Web servers contain lots of Web sites with the same IP address. Each IP cluster in [20] consists of a large amount of spam emails sent by different controlling entities. The basic idea of O-mean method for clustering spam emails is based on K-means clustering method. The idea is that k instances from a data set are randomly chosen and considered as initial centres. Then, each instance is assigned to the cluster that the distance between the center point of the cluster and the instance is the least. Afterwards, the center of each cluster is replaced with the mean value of its

members, and this process is continued until there is no changes in the clusters. To use this method for clustering spam emails, some parameters need to be redefined, mainly, how to define the distance between two emails and what would be k initial centers. For defining the distance, J. Song *et al.* focused on 12 features in the body of an email, which are expressed by numbers and used Aucludien meter for measuring the distance between two emails.

After clustering spam emails according to O-means method, the authors, in another work [28], found that the 10 largest clusters had sent about 90% of all spam emails in their data set. The authors have investigated these 10 clusters to implement heuristic analysis for selecting significant features among the 12 features used in [27]. As a result, they have selectd four most important features, which could effectively separate these 10 clusters from each other. These features are "Size of emails", "Number of lines", "Length of URLs", and "Number of dots". They have also mentioned that it is not the best method for selecting the most significant features, since it was based on analysis of the top 10 clusters; But the most important is that it results almost in the same accuracy of clustering of the previous method, which used 12 features.

3.5 Spamcraft: An Inside Look at Spam Campaign Orchestration

Kreibich *et al.* [29] have proposed an approach for detecting spam campaigns based on keywords that stand specifically for specific types of campaigns. First, campaigns are found manually based on keywords, and then, some interesting results are extracted from groups of campaigns. For identifying types of campaigns, the templates are manually and iteratively analyzed to check if the string characteristic for a specific campaign exists or not. The evasive techniques that are used by spammers have also been analyzed. Afterwards, strategies of campaigns containing the patterns used for spamvertized domains, harvested email addresses, target list maintenance, and target group selection have been studied. It has been discovered that 65% of instances last less than 2 hours. The longest running ones are pharmaceutical and are available for months, and crucial self propagation that run for 12 days. This interesting result suggests that it may be better to focus from the starting point of clustering on headers to identify these three campaigns and then continue to identify other campaigns with the use of other techniques.

3.6 Detecting and Characterizing Social Spam Campaign

In [21], although the authors focus on analysis of spam URLs in Facebook, their study of URLs and clustering spam messages is similar to our goal. Since a wall post without a URL can not help spammers to reach their goal, all wall posts without URLs are removed from this study. Each post is considered as a node and there is an edge between nodes if they refer to the same URL or they contain similar text content. So, the problem of identifying spam campaigns changes to the problem of distinguishing connected sub-graphs that is not difficult since it is just needed to select arbitrary nodes and find their transitive closures. Also, factors like bursty activity and distributed communication are considered. First, all wall posts that share the same URLs are clustered together. Then, the description of wall posts are analyzed and if two wall posts have the same description, their clusters are merged and the continuing comparison between these two clusters will stop. After spam messages are clustered by identifying the similarity between URLs and context, the clusters are categorized by their URLs, based on their format and domain name, and the prevalence of each category.

3.7 A Case for Unsupervised-learning-based Spam Filtering

Qian *et al.* [22] have introduced a case for spam filtering based on unsupervised machine learning algorithm. According to the authors, the use of unsupervised algorithms is not labor-intensive and costly in comparison with supervised ones. A framework, namely SpamAssassin, has been elaborated to extract textual signatures from spam emails. After the segmentation of emails, the proposed framework preprocesses emails by performing term reduction to remove low terms injected by spammers. The Porter Stemming Algorithm is used to reduce English words to their basic forms. The latent semantic analysis is used to boost the semantics of invariants. The framework clusters emails by separating low-entropy emails. For each cluster, the framework generates frequent term sequences as signatures. Afterwards, the framework purifies clusters by removing legitimate emails, clusters that have limited number of spam origins, as well as small and legitimate clusters. The framework has been evaluated by exposing different HTML, URL and textual signatures. In addition, the accuracy of the unsupervised detection method together with the spam filtering throughput have been also assessed. In comparison with Judo [30], this framework is easier to deploy since it does not need bot monitoring infrastructure. Judo sticks mainly to templates that are generated from spam produced during the analysis of malware in sandboxes, whereas SpamAssassin considers campaign signatures from raw traces of mixed spams and non spams. Moreover, SpamAssassin can adapt easily to the emergence of new templates or campaigns.

3.8 Mining Spam Email to Identify Common Origins for Forensic Application

Wei *et al.* [31] have proposed an approach that uses data mining techniques to study spam emails with the focus on law enforcement forensic analysis. The approach starts by extracting useful attributes from spam emails. Some attributes are retrieved directly from the email (e.g., sender email, sender IP address, subject), while other attributes are derived from the inherent ones using additional sources (e.g., WHOIS data and screenshots of Web pages of domains). These extracted features are then used by clustering algorithms to identify relationships between messages. Two main clustering algorithms are used by this approach. First, the agglomerative hierarchical algorithm [32] is used for the clustering of the entire dataset based on common values of email attributes. Next, the resulting clusters are refined into smaller clusters using the connected components with weighted edges model [33]. The second clustering method is used mainly to eliminate the false positives that result from the first clustering method. The results of this approach have been validated by comparison of graphical images of Web site fetches, WHOIS data, and the IP address of the computer hosting the advertised sites. This approach targets primarily spam messages that are sent to advertise a product or service. However, spam used for spreading viruses through attachments, or spam that sends visitors to hacked Web sites for purposes of phishing or other fraud are not supported by this approach.

4 Architecture

This section presents an overview of our approach for spam campaign analysis. The approach architecture is depicted in Figure 1. The main components of the proposed architecture are the following:

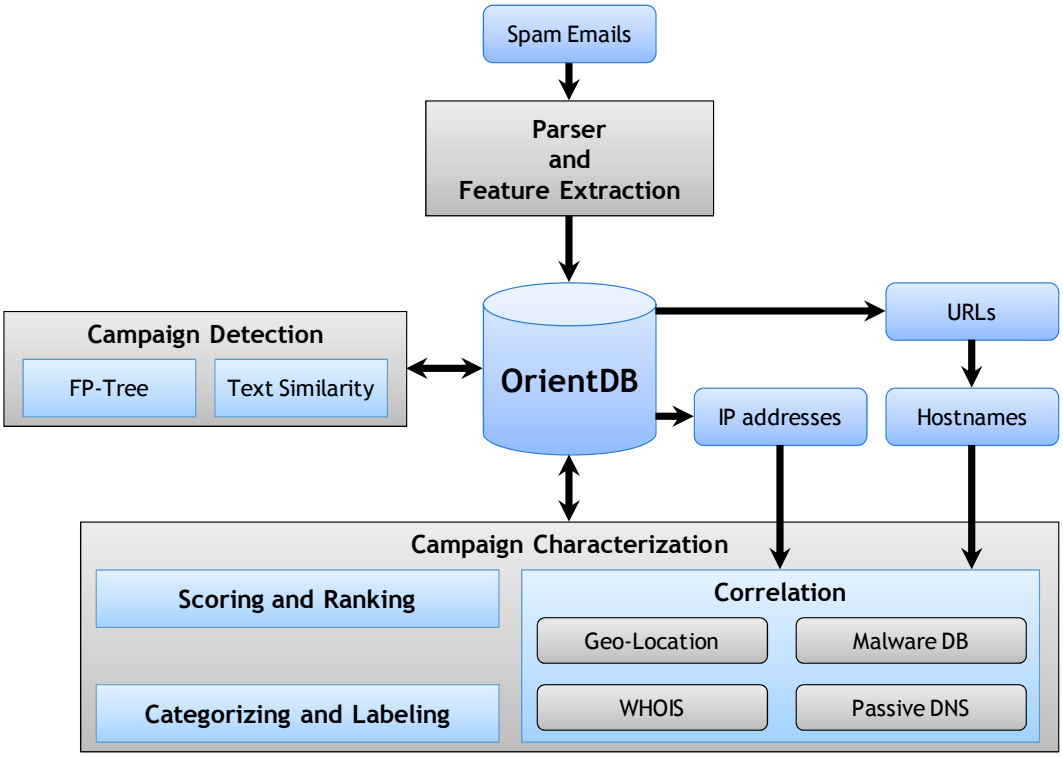

Figure 1: Overall Architecture

1. *Parser and Features Extraction:* This component is used to parse spam emails and extract important features such as header fields, URLs, subjects, etc. The extracted features are then stored in a database and analyzed. More details about this component are given is Section 5.1.

2. *Campaign Detection:* The extracted features are used to cluster spam emails into campaigns (or groups). Spam campaigns reduce significantly the amount of data that should be analyzed. In addition, they provide important insights about spammers that can be used by investigators. More details about these methods are presented in Section 5.2.

3. *Campaign Characterization:* Each identified campaign is assigned a number of attributes using different information and techniques. Extra information such as geo-location, malware, WHOIS and passive DNS can be utilized to correlate with spam campaigns. Moreover, scoring and ranking along with categorizing and labeling the campaigns significantly increase the effectiveness of an investigation. An investigator can specify which attributes are important to his/her investigation and assign larger score to those attributes. Thus, campaigns with high score and special characteristics will attract more attention from the investigator. Categorizing and labeling help the investigator to further understand the background behind a campaign. Therefore he/she can focus on spam campaigns that cause more damage (e.g. malware distribution, phishing). More details about this component are provided in Section 5.3.

5 Methodology

This section presents our methodology for spam campaign analysis. As shown in Figure 1, our approach comprises three main components, namely parser and features extraction, campaign detection, and campaign characterization. In the following, we provide more details about the design and the implementation of each component.

5.1 Parser and Features Extraction

This component is used to parse spam emails and extract important features from them. It takes, as input, spam emails with full header fields, content and attachment(s). As shown in Figure 2, an average of 1917 spam emails per day have been analyzed during January 2013.

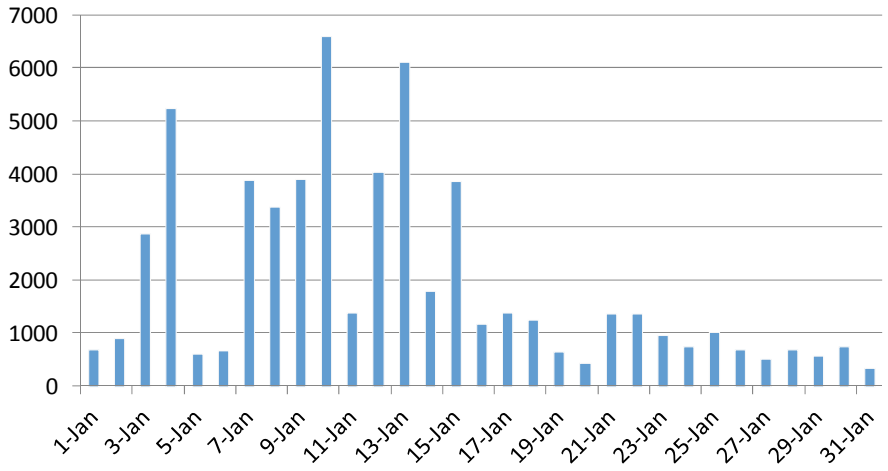

Figure 2: Number of Processed Spam Emails in January 2013

From each spam email, we extract relevant features that are needed for analysis. In particular, we extract important header fields based on RFC 5322 [34]. In addition, URLs inside the body and the attachments to the emails are also extracted. Moreover, the feature vector of each spam email is also extracted during this process.

All of the extracted features, along with the associated spam emails, are stored in a database. In addition, the relationships between spam emails and other important information such as the originating IP addresses, hostnames, and passive DNS are also stored in the database.

5.2 Campaign Detection

In this section, we present our methodology for consolidating spam emails into campaigns. Spam campaigns are groups of messages that are sent to achieve the same goal (e.g., advertising a specific product). Usually, spam campaigns are generated using the same obfuscation strategies. Therefore, clustering spam into campaigns is a necessary step in the analysis of spam emails since it can reveal important information about spammers' exploitation strategies. In our approach, two main methods have been adopted for consolidating spam emails into campaigns, namely Frequent Pattern Tree (FP-Tree) [11] and Text Similarity [35]. In the following, we provide details about each method.

5.2.1 Frequent Pattern Tree (FP-Tree)

Frequent Pattern Tree (FP-Tree) [11] is the first step of the FP-Growth algorithm [16]. The latter is a well-known data-mining technique used for discovering association rules between items in large datasets (See Section 2.8). The main idea of using FP-Tree is that the more frequent a feature is, the more it is shared among the spam emails. Less frequent features correspond to the parts that are obfuscated by the spammers. Thus, representing spam emails in an FP-Tree helps us categorize them into campaigns and extract important information about spammers and their strategies.

Each node of the FP-Tree represents a feature that is extracted from the spam emails, such as subject, URL, layout, etc. Each path in the tree represents sets of features that co-occur in spam messages, in a decreasing order of frequency of occurrences. Thus, two messages that have several frequent features in common and are different only on infrequent features are more likely to be grouped into the same campaign. The root is the only empty node, separating sub-trees, which have nothing in common. The constructed FP-Tree is used to consolidate spam emails into campaigns. If a node is identified as a campaign, all the leaves that share the same node are the spam emails of this campaign.

The FP-Tree technique has the advantage of not only identifying spam campaigns, but also describes how the spam messages were constructed and obfuscated [10]. However, this algorithm works only with a static set of data. Nowadays, spam emails are automatically generated and sent using sophisticated techniques. In addition, some spam campaigns have a very short life-time. For example, phishing campaigns can last only for a few hours. Therefore, an online and incremental algorithm is necessary, as the FP-Tree can get updated and the spam campaigns can be identified dynamically [36].

5.2.2 Text Similarity

Text similarity is a technique that is used to find the similarity between two texts. In our approach, we use this technique to find similarities between spam emails so that similar ones can be grouped into the same campaign. The idea is to calculate a number indicating the average of similarities between spam emails inside a campaign. The most important methods that are used in text similarity are: W-Shingling, Locality-Sensitive Hashing, and Fuzzy Hashing. In the following, we provide an overview of each of these methods.

W-Shingling W-Shingling is a text similarity technique based on the so-called w-shingles. A w-shingle is a contiguous subsequence of w words that can be used to find the similarity between two documents. The "w" denotes the number of words in each shingle. The w-shingling of a document D, S(D, w), is the set of unique w-shingles of D. For example, given two texts A and B, where:

A contains m words (A_1, A_2, ..., A_m), and

B contains n words (B_1, B_2, ..., B_n).

If the size of a shingle is considered equal to w, then the sets: $(A_1A_2...A_w, A_{w+1}A_{w+2}...A_{2w}$,...) and $(B_1B_2..B_w, B_{w+1}B_{w+2}...B_{2w}$, ...) are shingles from A and B.

The similarity between A and B can be expressed as the ratio of the magnitudes of their shinglings' intersection and union, i.e.,

$$\text{r(A,B)} = \frac{|S(A) \bigcap S(B)|}{|S(A) \bigcup S(B)|}$$

where S(A) denotes the w-shingling of a document A and |A| denotes the size of A. Notice that the resemblance of two documents is a number in the range of [0,1], where 1 indicates that the two documents are identical.

Locality-Sensitive Hashing and Fuzzy Hashing Locality-Sensitive Hashing (LSH) is a technique for hashing messages so that similar ones are mapped to the same buckets with high probability. Unlike normal hashing algorithms that are used to detect the slightest change of data, local-sensitive hashing gives the degree of similarity between two data objects. To better illustrate this technique, let us consider a spam message, shown in Figure 3 [1]. Figure 4 shows the same spam message after the spacing and the spelling were cleaned up [1]. The locality-sensitive hashing codes of the two messages are:

`773e2df0a02a319ec34a0b71d54029111da90838cbc20ecd3d2d4e18c25a3025` for the spam message of Figure 3, and

`47182cf0802a11dec24a3b75d5042d310ca90838c9d20ecc3d610e98560a3645` for the spam message of Figure 4.

The similarity of these two codes is 92 on a scale of -128 to +128, which means that 36 bits are different and 220 bits are the same. This indicates that the two messages are very similar and not independently generated [1].

Dear Sir / Madam,
It is a great honour to have the chance to introduce our company(Guangdong Kestar Electronic Co.Ltd.) and our main products.
Guangdong Kestar Electronic Co.Ltd. is located in Guangdong province mainland China.It is a professional manufcturer on Varistors ,equipped with a complete set of advanced machinery,possessed high abilities of development & exploitation.We have 10 years experience in managing large size production.
Our main seven series varistors are WYG current varistor, MYL lihtningproof varistor,MYE high load varistor,MYA minitype varistor,MYP squareness varistor,MYR highly reliable&varistor,low-pressure power supply lightning arrester.. The Varistors conform to ISO9002 .The products have passed several international authentication.
if you are interested in our products, please contact us without hesitate.

yours truely Lin Cang.

Guangdong Kestar Electronic Co.Ltd

Figure 3: Example of Spam [1]

Dear Sir / Madam,
It is a great honour to have the chance to introduce our company (Guangdong Kestar Electronic Co. Ltd.) and our main products.
Guangdong Kestar Electronic Co.Ltd. is located in Guangdong province, mainland China. It is a professional manufacturer of varistors, equipped with a complete set of advanced machinery, possessing high abilities of development & exploitation. We have 10 years experience in managing large size production.
Our main seven series varistors are WYG current varistor, MYL lightningproof varistor, MYE high load varistor, MYA minitype varistor, MYP squareness varistor, MYR highly reliable varistor, low-pressure power supply lightning arrester. The Varistors conform to ISO9002. The products have passed several international authentications.
if you are interested in our products, please contact us without hesitation.

yours truly, Lin Cang.

Guangdong Kestar Electronic Co. Ltd

Figure 4: The Same Spam After Cleaning [1]

Text similarity methods can work well on some datasets, but they have some limitations. Recently, spammers start to use templates to automatically generate spam emails. Although all spam emails in one campaign serve one specific purpose, their content can be very different and the similarities between them are very low. In addition, the computation cost for these methods can be very high. Moreover, these methods are considered inefficient for spam emails with a short content. However, text similarity methods can be used as an additional technique to verify the campaigns that are detected by the FP-Tree algorithms. Other methods such as clustering are also considered for campaign identification. However, the drawback of these methods is the difficulty of measuring the distance between two spam emails.

5.3 Campaign Characterization

During this step, we analyze the characteristics of the identified campaigns to extract useful information about spammers and their strategies in order to increase the effectiveness of an investigation. In addition, we rank and score the campaigns so that investigators can focus on the most important ones. Furthermore, we categorize and label the campaigns to help investigators understand the background behind a campaign. In the following, we present each of these techniques in more detail.

5.3.1 Correlation

The purpose of this step is to find relationships between spam campaigns. Extra information, such as geo-location, malware information, WHOIS information, and passive DNS, can be used to correlate with spam campaigns.

Geo-Location The purpose of this step is to identify geo-location information about spam emails. Examples of such information include the location (i.e., city, region, country) of spammers, information about their organizations and ISPs, etc. Geo-location can help investigators prioritize spam campaigns based on their local locations. For example, Canadian law enforcers are more interested in spam campaigns that are originated from Canadian spammers or compromised hosts in Canada.

Malware Information The extracted information from spam emails can also be correlated with malware information such as malicious IPs, which a malware tries to connect to. The purpose of doing this is to check wether the IP addresses or the hostnames of spam emails are associated with malware (usually a spam botnet).

WHOIS Information Second-level domains are extracted from the URLs inside spam emails and used to query WHOIS servers. The latter reveal crucial information about a domain including the owner, status, creation and expiry dates, name servers, registration information, etc.

Passive DNS Passive DNS stores historical DNS data. It can provide important information about a domain, including IP address(es) of that domain during a specific duration in the past and the number of requests to resolve that domain to the IP address(es). A source of passive DNS information can be used to correlate with the information extracted from spam emails.

5.3.2 Scoring and Ranking

Spam campaigns are scored and ranked so that highly scored ones get more attention from the investigators. We rank spam campaign according to criteria such as:

- Number of spam emails inside a campaign
- Has IP(s) in Canada
- Has domain(s) resolved to IP(s) in Canada

- Has Canadian city name(s) inside the content
- Has "Canada" inside the content
- Etc.

5.3.3 Categorizing and Labeling

Labeling is intended to assign a label to each campaign, based on the topic that this campaign is revolving around. The labeling process starts by extracting frequent words from the whole set of emails belonging to the same campaign. These words are considered our top topic terms. Afterwards, we train two classifiers. The first classifier is meant for the terms that might have more than one meaning, so that we only get the relevant articles to the context of the words that we have. The second classifier is trained to score the links between articles, based on their relatedness, and eliminate the unrelated article topics.

6 Results

This section presents the results that we obtained from our experiments. Table 1 presents statistics about the spam messages that we analyzed during one week experiment (from January 01, 2013 to January 07, 2013). More precisely, it presents the number of spam messages, IPs, hostnames, campaigns, and attachments. Furthermore, Figure 5 through Figure 8 present statistics about the identified campaigns as follows. Figure 5 illustrates the number of spam campaigns that are detected using different types of features. Figure 6 presents the distribution of spam campaigns according to the languages used to write the spam emails. Figure 7 depicts the distribution of the top 20 campaigns by score. Finally, Figure 8 presents the top 8 topics of the identified campaigns.

Table 1: Statistics

Messages	14865
IPs	3384
Hostnames	6237
Campaigns	119
Attachment	17

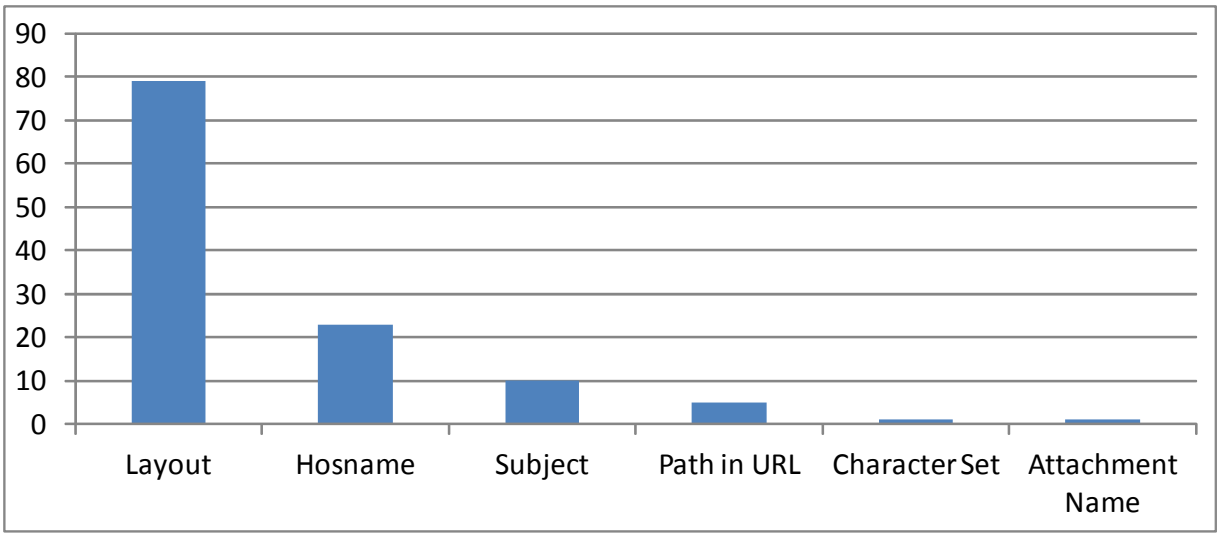

Figure 5: Number of Campaigns Detected using Different Types of Features

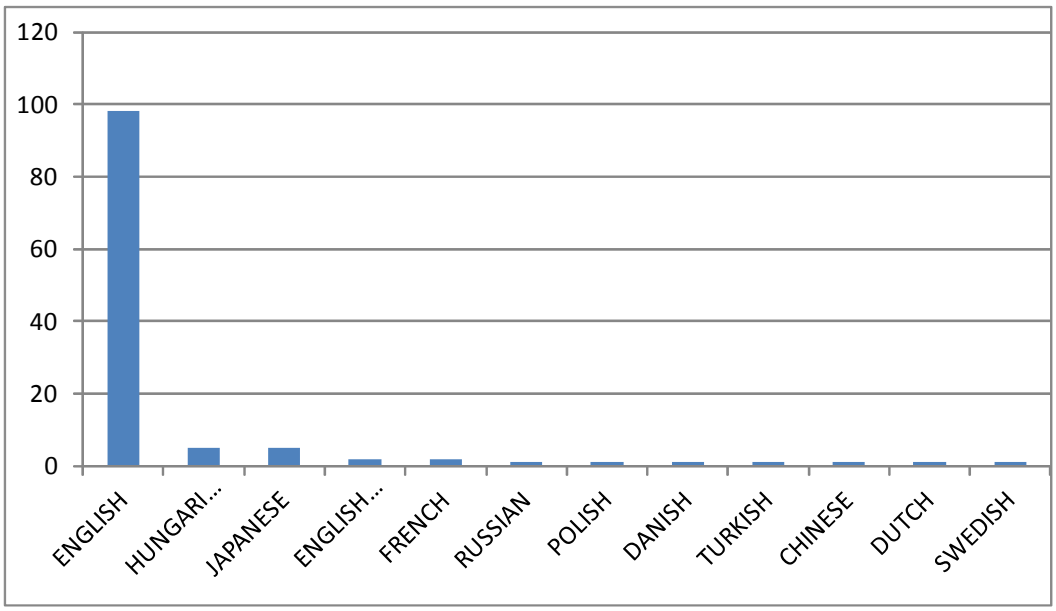

Figure 6: Distribution of Spam Campaigns According to Languages

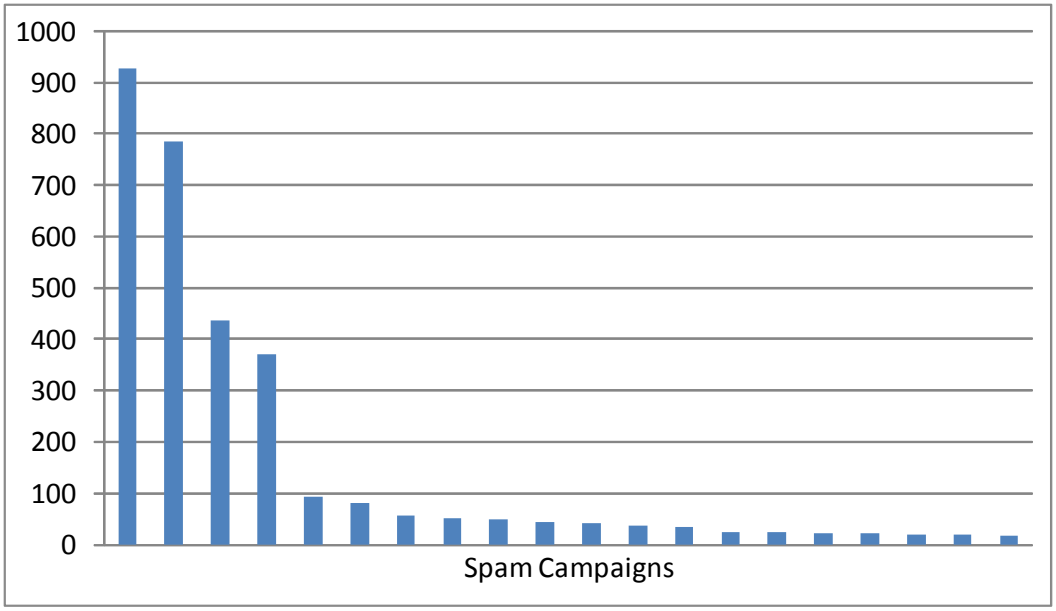

Figure 7: Score of Spam Campaigns

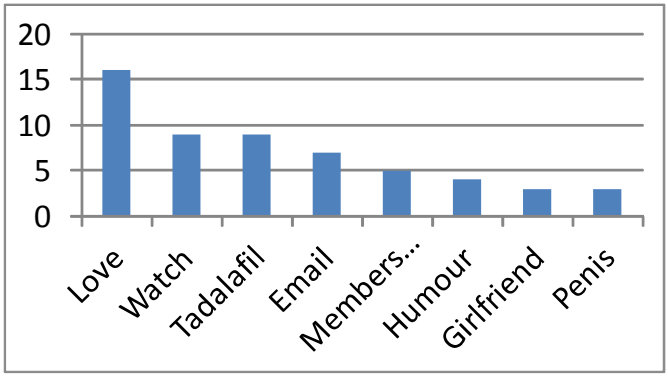

Figure 8: Top 8 Topics of Spam Campaigns

Figure 9 through Figure 12 present screenshots of the identified campaigns. More precisely, Figure 9 depicts an overview of a spam campaign containing 201 spam messages. Figure 10 presents an overview of some of the largest campaigns that we have identified with the number of messages in each campaign. Figure 11 presents another spam campaign containing 2226 spam messages. It also shows the subjects of messages that belong to this campaign. Finally, Figure 12 shows an overview of these messages after zooming the campaign. As shown in this figure, all the messages that belong to this campaign have the same subjects.

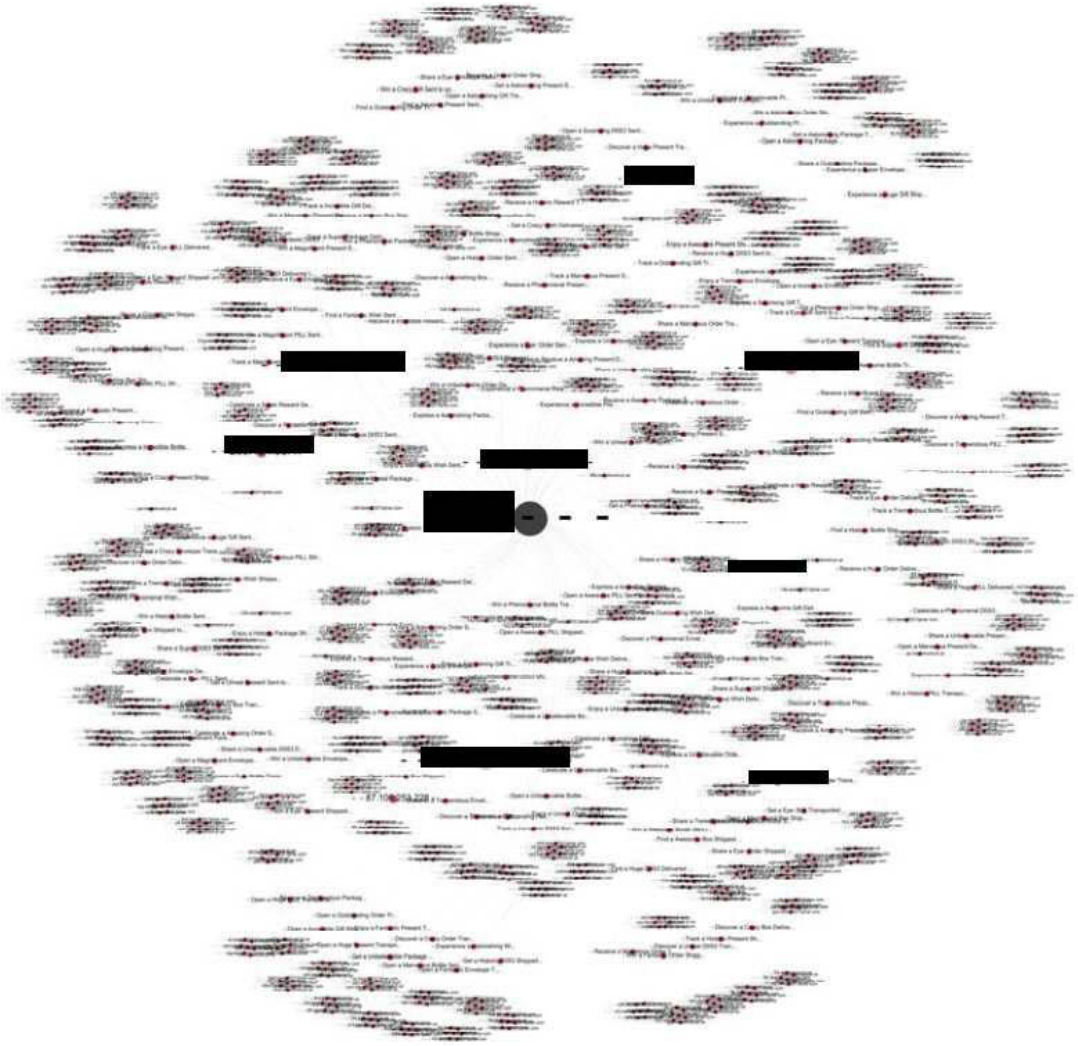

Figure 9: A Spam Campaign

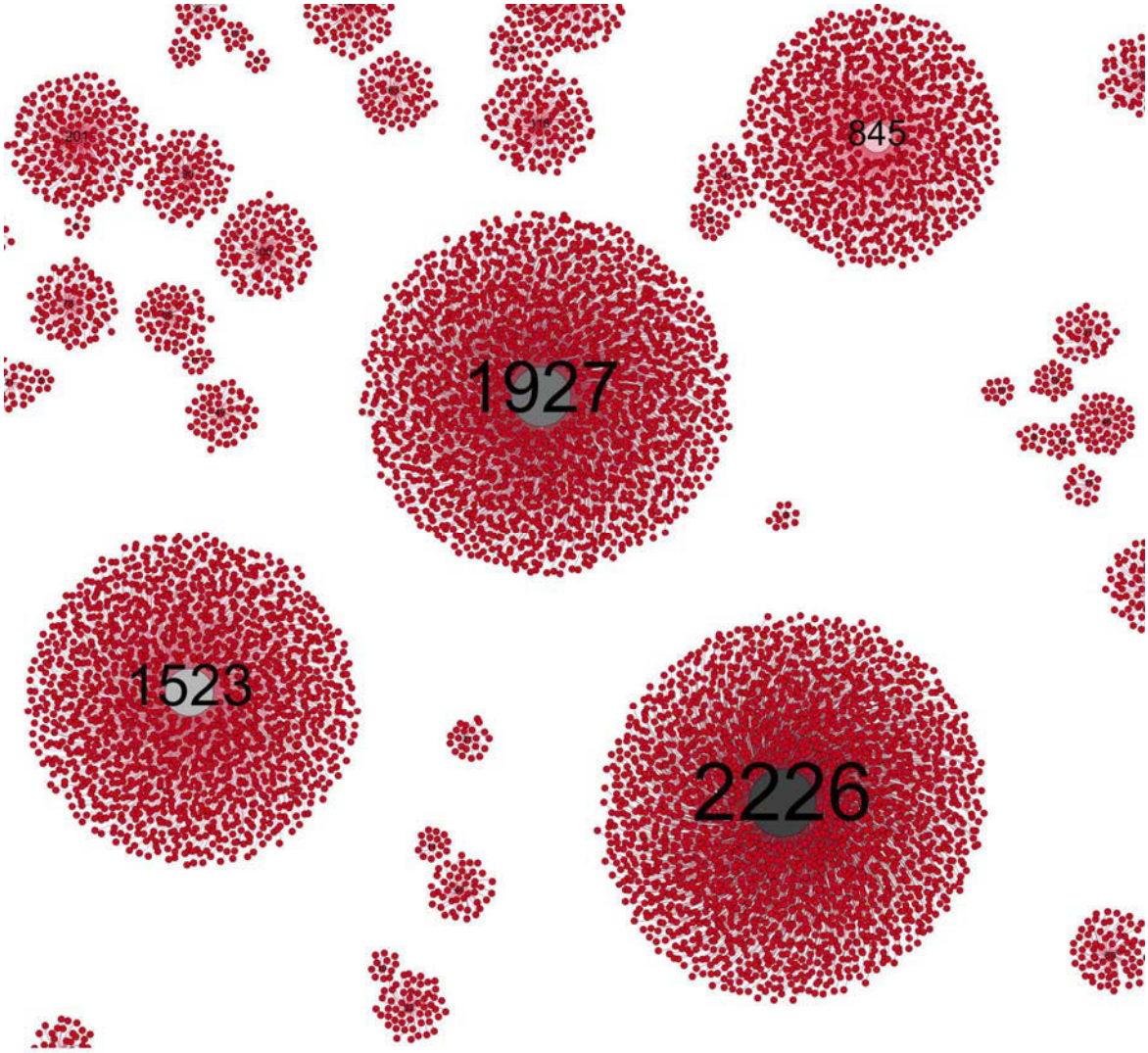

Figure 10: Spam Campaigns

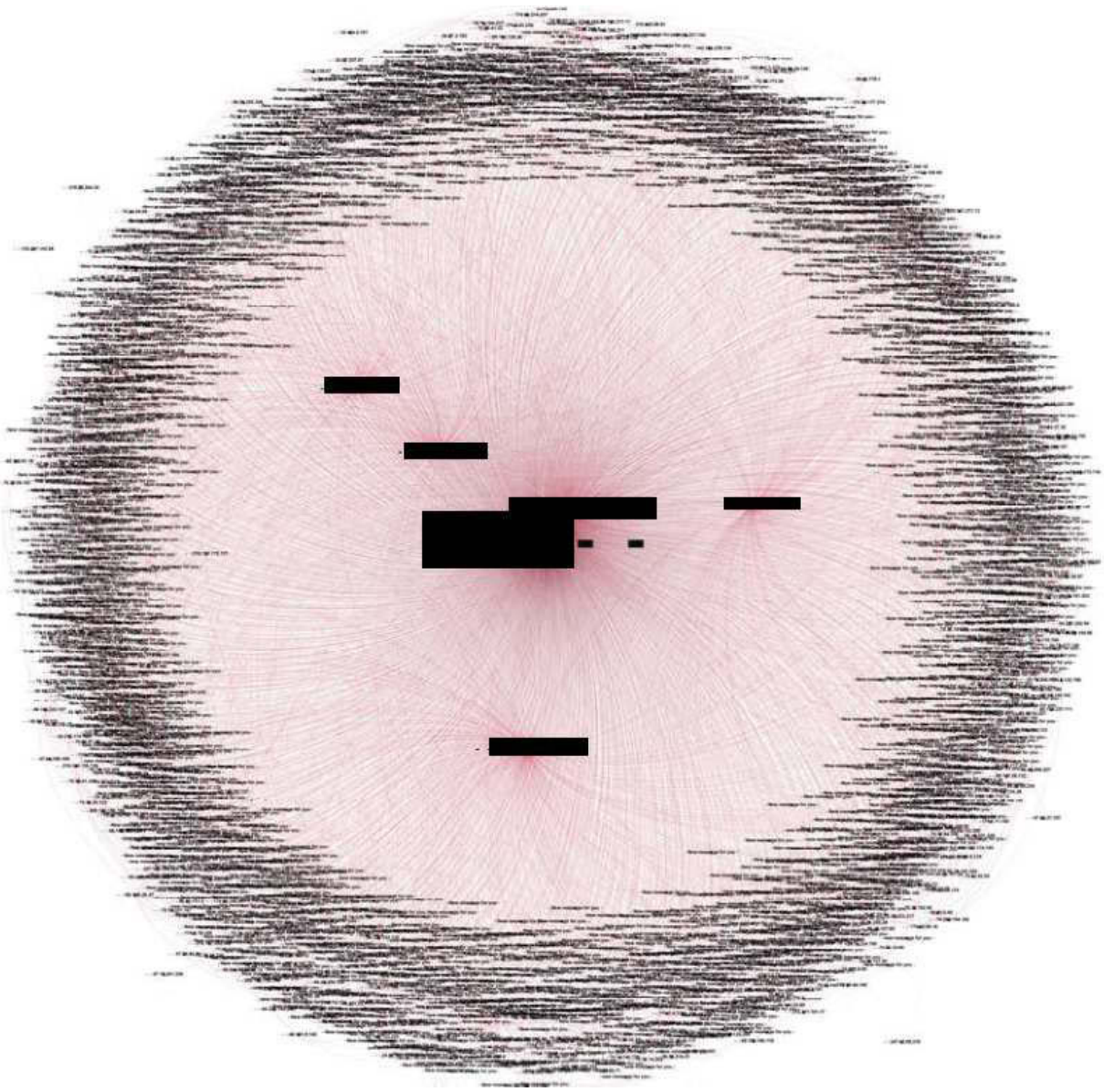

Figure 11: A Spam Campaign with IPs and Subjects

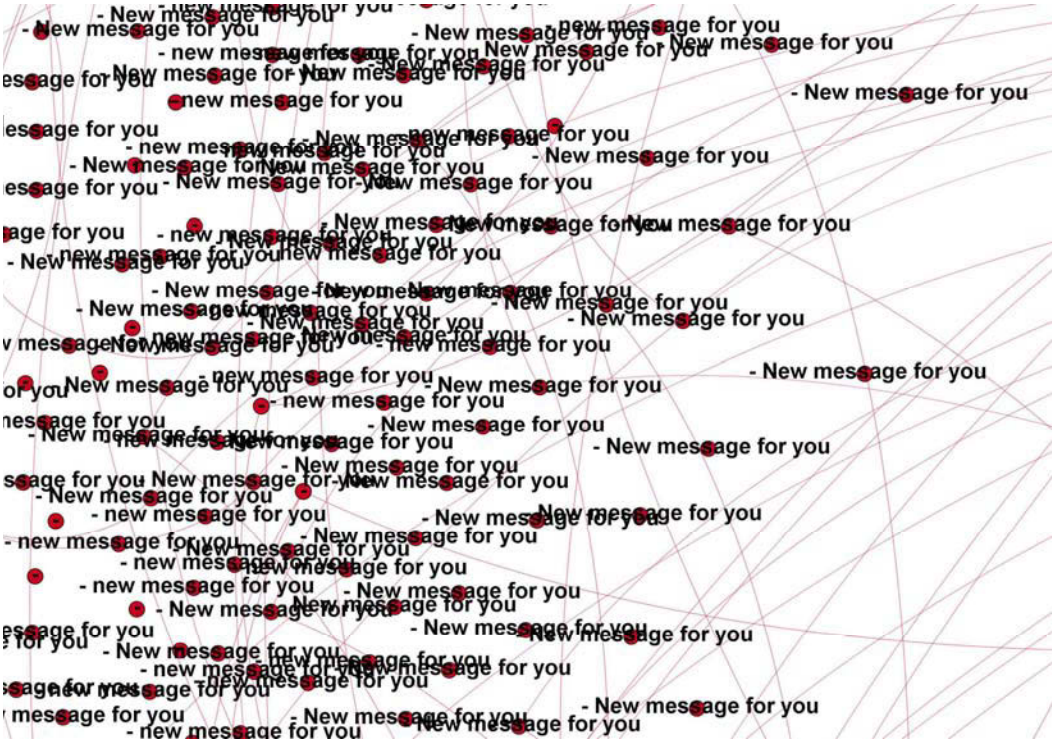

Figure 12: A Spam Campaign with Subjects

7 Conclusion

In this report, we have presented our approach for identifying and analyzing spam campaigns. Spam campaigns are very useful for email forensic analysis since they significantly reduce the amount of data that should be analyzed. In addition, they provide important insights about spammers that can be used by investigators. From spam emails, we start by extracting relevant features that we use to detect spam campaigns. Our approach for campaigns identification is mainly based on two methods that are Frequent Pattern Tree and text similarity. These methods are widely used in data-mining and have the advantage of not only identifying spam campaigns, but also help us understand how the spam messages were generated. After detecting spam campaigns, we use additional data such as geo-location, malware, WHOIS information, and passive DNS to correlate with spam campaigns. In addition, we rank and score the campaigns so that investigators can focus on the most important ones. Furthermore, we categorize and label the campaigns to help investigators understand the background behind a campaign.

References

[1] Locality-Sensitive Hashing. http://www.mit.edu/~andoni/LSH/.

[2] The Spamhaus Project. http://www.spamhaus.org/, last accessed on February 2013.

[3] The Spamhaus Project - The Definition of Spam. http://www.spamhaus.org/consumer/definition/, last accessed on February 2013.

[4] Symantec Intelligence Report: January 2013. http://www.symantec.com/content/en/us/enterprise\discretionary-/other_resources/b-intelligence_report_01-2013.en-us.pdf, last accessed on February 2013.

[5] Canada's Anti-Spam Legislation. http://fightspam.gc.ca/eic/site/030.nsf/eng/home, last accessed on February 2013.

[6] ACMA - The Australian Communications and Media Authority. http://www.acma.gov.au/WEB/HOMEPAGE/pc=HOME, last accessed on February 2013.

[7] ACMA - Understanding Spam. http://www.acma.gov.au/WEB/STANDARD..PC/pc=PC_310296, last accessed on February 2013.

[8] Glossary - Canada's Anti-Spam Legislation. http://fightspam.gc.ca/eic/site/030.nsf/eng/h_00216.html, last accessed on February 2013.

[9] The Spamhaus Project - Frequently Asked Questions (FAQ). http://www.spamhaus.org/faq/section/Glossary, last accessed on February 2013.

[10] P. H. Calais, D. E. V. Pires, D. O. Guedes, W. Meira, C. Hoepers, and K. Steding-jessen, "A Campaign-based Characterization of Spamming Strategies," in *In the Proceedings of the Fifth Conference on Email and Anti-Spam (CEAS 2008)*, 2008.

[11] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining, (First Edition)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005.

[12] Anti-spam Techniques. http://en.wikipedia.org/wiki/Anti-spam_techniques#Statistical_content_filtering, last accessed on March 2013.

[13] R. Agrawal, T. Imieliński, and A. Swami, "Mining Association Rules Between Sets of Items in Large Databases," *SIGMOD Rec.*, vol. 22, no. 2, pp. 207–216, Jun. 1993. [Online]. Available: http://doi.acm.org/10.1145/170036.170072

[14] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 487–499. [Online]. Available: http://dl.acm.org/citation.cfm?id=645920.672836

[15] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, "New Algorithms for Fast Discovery of Association Rules," Rochester, NY, USA, Tech. Rep., 1997.

[16] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," *SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, May 2000. [Online]. Available: http://doi.acm.org/10.1145/335191.335372

[17] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data mining and knowledge discovery*, vol. 8, no. 1, pp. 53–87, 2004.

[18] C.-C. Yeh and C.-H. Lin, "Near-duplicate Mail Detection based on URL Information for Spam Filtering," in *Proceedings of the 2006 international conference on Information Networking: advances in Data Communications and Wireless Networks*, ser. ICOIN'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 842–851. [Online]. Available: http://dx.doi.org/10.1007/11919568_84

[19] M. H. F.Li, "An Empirical Study of Clustering Behavior of Spammers and Groupbased Anti-Spam Strategies," in *CEAS 2006 Third Conference on Email and AntiSpam*, 2006, pp. 27–28.

[20] J.Song, D.Inque, M.Eto, H.C.Kim, and K.Nakao, "An empirical study of spam: Analyzing spam sending systems and malicious web servers," in *Proceedings of the 2010 10th IEEE/IPSJ International Symposium on Applications and the Internet*, ser. SAINT '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 257–260. [Online]. Available: http://dx.doi.org/10.1109/SAINT.2010.20

[21] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Zhao, "Detecting and characterizing social spam campaigns," in *Proceedings of the 10th annual conference on Internet measurement*. ACM, 2010, pp. 35–47.

[22] F. Qian, A. Pathak, Y. C. Hu, Z. M. Mao, and Y. Xie, "A Case for Unsupervised-learning-based Spam Filtering," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 367–368, Jun. 2010. [Online]. Available: http://doi.acm.org/10.1145/1811099.1811090

[23] A. Kolcz and A. Chowdhury, "Hardening Fingerprinting by Context," in *CEAS 2007 - The Fourth Conference on Email and Anti-Spam, 2-3 August 2007, Mountain View, California, USA*, 2007.

[24] Z. Wang, W. Josephson, Q. Lv, M. Charikar, and K. Li, "Filtering Image Spam with Near-duplicate Detection," in *In Proceedings of the Fourth Conference on Email and AntiSpam, CEAS2007*, 2007.

[25] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, "Spamscatter: characterizing internet scam hosting infrastructure," in *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, ser. SS'07. Berkeley, CA, USA: USENIX Association, 2007, pp. 10:1–10:14. [Online]. Available: http://dl.acm.org/citation.cfm?id=1362903.1362913

[26] G. W. A. S. C. Wei, A. Sprague, "Identifying New Spam Domains by Hosting IPs: Improving Domain Blacklisting," 2010.

[27] J. Song, D. Inoue, M. Eto, H. C. Kim, and K. Nakao, "O-means: An Optimized Clustering Method for Analyzing Spam Based Attacks," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. 94, pp. 245–254, 2011.

[28] J.Song, D.Inque, M.Eto, H.C.Kim, and K.Nakao, "A heuristic-based feature selection method for clustering spam emails," in *Proceedings of the 17th international conference on Neural information processing: theory and algorithms - Volume Part I*, ser. ICONIP'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 290–297. [Online]. Available: http://dl.acm.org/citation.cfm?id=1939659.1939698

[29] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage, "Spamcraft: An inside look at spam campaign orchestration."

[30] A. Pitsillidis, K. Levchenko, C. Kreibich, C. Kanich, G. M. Voelker, V. Paxson, N. Weaver, and S. Savage, "Botnet Judo: Fighting Spam with Itself," in *NDSS*. The Internet Society, 2010.

[31] C. Wei, A. Sprague, G. Warner, and A. Skjellum, "Mining Spam Email to Identify Common Origins for Forensic Application," in *Proceedings of the 2008 ACM symposium on Applied computing*, ser. SAC '08. New York, NY, USA: ACM, 2008, pp. 1433–1437. [Online]. Available: http://doi.acm.org/10.1145/1363686.1364019

[32] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[33] S. Baase and A. V. Gelder, *Computer Algorithms: Introduction to Design and Analysis (3rd Edition)*. Addison-Wesley, 1999.

[34] Internet Message Format. http://tools.ietf.org/html/rfc5322.

[35] A. Huang, "Similarity Measures for Text Document Clustering," pp. 49–56, 2008.

[36] W. Cheung and O. Zaiane, "Incremental Mining of Frequent Patterns without Candidate Generation or Support Constraint," in *Database Engineering and Applications Symposium, 2003. Proceedings. Seventh International*, 2003, pp. 111–116.