Special issue of the 3rd International Conference on Computational and Experimental Science and Engineering (ICCESEN 2016)

Design of a Machine Learning Based Predictive Analytics System for Spam Problem

A.S. Yüksel[a,*], Ş.F. Çankaya[a] and İ.S. Üncü[b]

[a]Süleyman Demirel University, Faculty of Engineering, Department of Computer Engineering, Isparta, Turkey
[b]Süleyman Demirel University, Faculty of Technology, Department of Electrical and Electronic Engineering, Isparta, Turkey

Spamming is the act of abusing an electronic messaging system by sending unsolicited bulk messages. Filtering of these messages is merely another line of defence and does not prevent spam messages from circulating in email systems. This problem causes users to distrust email systems, suspect even legitimate emails and leads to substantial investment in technologies to counter the spam problem. Spammers threaten users by abusing the lack of accountability and verification features of communicating entities. To contribute to the fight against spamming, a cloud-based system that analyses the email server logs and uses predictive analytics with machine learning to build trust identities that model the email messaging behavior of spamming and legitimate servers has been designed. The system constructs trust models for servers, updating them regularly to tune the models. This study proposed that this approach will not only minimize the circulation of spam in email messaging systems, but will also be a novel step in the direction of trust identities and accountability in email infrastructure.

DOI: 10.12693/APhysPolA.132.500

PACS/topics: spam, predictive analytics, machine learning, trust identities

1. Introduction

Before the growth, popularity, and widespread use of the internet, the telephone was the main medium for communication. The internet has changed the way that people communicate with each other and has led to the development of new communication services, such as electronic mail (email). Now it has become an integral part of the communications structure of many organizations and vendors. However, there is a downside, as malicious people abuse this "free" mail infrastructure by sending unsolicited bulk messages gains profit, or steals personal information or identities, causing damage to users. Such people have benefited from the lack of security and trusted identities built into the current electronic mail communication infrastructure that uses simple mail transfer protocol (SMTP), which does not have the ability to verify the origin of emails at the user or mail server levels.

The current SMTP system is open for abuse, since any sender can falsify their identity and send any number of emails, containing any content they desire, to any recipient. This misuse of electronic messaging systems to randomly send unsolicited emails is called "spamming". At present, it is common for email users to find a high percentage of spam emails from unknown senders in their mailbox daily. Spamming has also introduced cyber fraud on the internet, through social engineering, most of which starts from an email from an untrusted source containing a URL that, when opened, compromises one's personal information. Spamming remains economically viable because spammers can manage their mailing lists at a low cost. Due to the minimal investment required by the spamming business, the number of spammers and spam emails has increased. This has resulted in a system in which every email has become a suspect, leading to substantial investment in counter measures, such as the development of spam filtering software, anti-spam software, the creation of domain name server black lists (DNSBL) and white lists, and analysis of spammer activities.

There has been extensive research on spam activities and its infrastructure. Reference [1] presented a study and analysis of global behavior of spammers using open mail relay sinkholes, and classified spammers into high-volume spammers (HVS) — corresponding to direct spammers — and low-volume spammers (LVS) — corresponding to distributed bots (compromised computers) in some botnets that send low-volume spam. In paper [2] the authors did an analysis on the economics and profitability of email spam marketing using botnet infrastructure, pointing out the high turnover on minimal investment in the spam business. References [3] and [4] revealed the existence of spammer network infrastructure–botnets, and how the network was expanded and kept in service. In Ref. [5] the authors investigated the spamming problem and utilized the distributed characteristics of botnet-based spam campaigns to generate email spam signatures for use in the fight against spam. In Ref. [6] the authors worked on email traffic patterns through a single mail server, including message sizes, senders and receivers that could be used to develop an email workload or benchmark for a mailing system. Reference [7] researched SMTP path analysis and presented a learning algorithm for estimating the reputation of email domains

[*]corresponding author; e-mail: asimyuksel@sdu.edu.tr

and associated IP addresses. In Ref. [8] they proposed a support vector machine (SVM) classifier model and introduced a machine learning-based web spam classification approach. In Ref. [9] the authors studied the problem of detecting review spammers by using contextual social relationships that are available in several online review systems. They developed a trust-based rating predication using social relationships, such as friendships and complements relationships, to compute the overall trustworthiness score for every user in the system. Their analysis showed that their trust-based prediction algorithm had a high accuracy and that there was a strong correlation between social relationships and the computed trustworthiness scores. In Ref. [10] the authors designed and implemented a system called TruSMS to control SMS spam. They evaluated the system performance in reaction to a variety of intrusions and attacks, demonstrating that their system was effective in terms of accuracy, efficiency and robustness. In Ref. [11] they proposed an online machine learning-based malicious spam email detection system. They used a term-weighting scheme to represent each spam email and created feature vectors as the input of the classifier. The learning was performed periodically and the classifier was updated. Their results showed that the spam detection system was efficient and accurate in identifying malicious spam emails.

Our goal for developing a solution to the spam problem is different from these previous studies. This research combines predictive analytics and machine learning techniques to build a cloud-based approach that analyses server logs and produces trust models to identify whether or not servers are trustworthy. Therefore, it relies heavily on email server logs that contain valuable information regarding email sending patterns, spam probabilities, IP blacklisting statuses, and virus statuses, which indicate strong trust relationships and improve prediction accuracy.

2. Methodology

The current "work-around" for the spamming problem is the application of spam filters on the mail server and/or client sides. Spam filtering used different techniques, such as white listing, black listing, and content-based filtering, or a combination thereof. Black listing is applied at the mail servers and is based on the IP domain published in the DNS blacklists, while white listing is mostly on the client side and is based on email accounts to which a user has given permission for receiving email. Spammers are always evolving and finding new techniques to continue their spamming business. They are always striving to stay ahead of anti-spam techniques, and so a shift in the research focus is required. This study designs a system that works through the cloud and combines predictive analytics and machine learning techniques. A prototype of the system, which runs on Microsoft Azure platform [12] and make uses of Azure machine learning to create and deploy behavior models of email servers for predictive analysis, has been developed.

2.1. Predictive analytics phase

Predictive analytics is the method of processing large amounts of data into a summary of the information that can be easily understood by humans. This method has been applied in many fields for predicting the outcomes of various events [13–15]. It applies advanced statistical techniques such as generalized linear models [16] or Monte Carlo simulation [17] and attempts to answer the question: "What might happen in the future?" In this case, the question becomes: "Which email may be spam in the future?" This approach has three basic elements:

1. The data: Historical data
2. The statistics: The set of mathematical techniques.
3. The assumptions: What is true/spam? What is false/not spam?

As the first step, an email data set (6 months of logs) has been acquired from our department mail exchange repository. This has then been filtered using DNSBL and anti-spam filtering applications, creating two datasets. The first is the DNSBL list that contains "<timestamp> <relay IP address> <OK|REJECT>". "OK" status means that the IP address has passed the DNS blacklist (DNSBL) check and is used for further spam analysis. "REJECT" status means the email did not pass the DNSBL check.

The second dataset, the anti-spam filter data log, contains "<timestamp> <relay IP address> <relay hostname> <autonomous system number> <probability|VIRUS>". Each email that passes the DNSBL check is passed through SpamAssassin [18], an open-source anti-spam filter tool at the mail server that assigns probabilities between 0 and 1 based on the filtering rules. "VIRUS" status is assigned if the email contains a virus. The logs were manually analyzed and, by cross-checking with our dataset, the spam emails and "good" emails were separated. The filtered data was used to train and test our model. Finally, the "feature vector" was defined, which provides identifying characteristics for each email server. This study's feature vector is composed of features such as spam probability, DNSBL status, number of good/spam emails, number of virus emails, ASN, number of spams from host, number of other spammers in the same ASN, number of spams from host over all spams in the same ASN, life time of host, and burstiness of SPAM emails. A spam score is generated using this feature vector, indicating whether the email is spam or not. These features provide identifying characteristics for each email.

Using the underlying Predictive Analytics architecture, the outcome is the starting point. In this case, the outcomes are the emails that are previously known to be spam. The computers (Azure Cloud) are then taught to automatically uncover the factors that are driving this particular outcome (spam) and the predictive models are run in order to forecast future behaviors, outcomes, and trends of spammers. The result is a far more accurate predictive model that can automatically adjust itself and improve over time.

2.2. Machine learning phase

Microsoft Azure platform provides tools for machine learning. In these experiments, the two class boosted decision tree and the two class support vector machine (SVM) were used as spam classifiers. The decision tree is commonly used in data mining. It has the ability to create a model that predicts the value of a target variable based on several input variables. The SVM is a supervised learning model that has learning algorithms and the ability to analyze data for classification. Given a set of training examples, SVM can decide whether an email belongs to the “spam” or “good” email category.

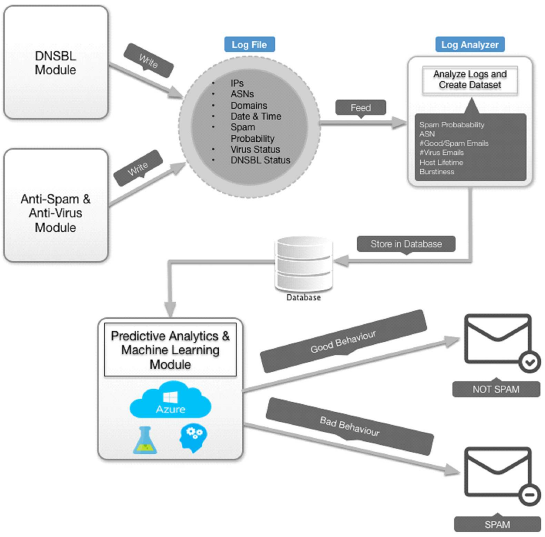

Fig. 1. System architecture.

Separate datasets were generated to train and test the models. First, the data was split into training and test data. Then, the models were trained and evaluated. By using the Azure machine learning studio, we were able to try decision tree and SVM and compare our results. This type of experimentation assisted in finding the best solution to the study problem. The test data that resulted was used to score the trained models. The results of the models were then compared to discover which performed better. Figure 1 shows the architectural diagram of the system.

As seen in Fig. 1, the log analyzer module receives log files as input and analyzes intervals of emails. The interval information is the difference between the time when the email address first received good or spam email and the time when the next good or spam emails are received. By looking at the intervals, the analyzer can formulate sending patterns for both good and spam emails. As a next step, the analyzer stores this information in the database and regularly updates it. The predictive analytics and machine learning module (PAML-M) runs periodically and receives recent data from the database. It then updates its model and the behavior of the servers, or assigns a new behavior if there is no behavior information from the new server. As a final step, the PAML-M identifies good and spam emails based on behavior information. If the mail is spam, it is not delivered and the behavior of the sender is again updated. If the mail is good, it is delivered, and again, the behavior information is updated.

3. Results

The sending patterns of each mail server have been analyzed on a daily basis and on month granularity. The results for a six-month period are presented in Table I. The first 5 hosts are known to be spamming servers. Therefore, our system identifies these servers as spammers and the emails are not delivered. The remaining servers are known to be sending non-spam emails. Therefore, our system identifies those servers as benign and delivers the emails.

TABLE I

Host based analysis of some spammer and benign servers.

Hostname	Lifetime [days]	# # of spamming days	Average spam score	# of total emails	# of total spam emails	# of total benign emails	Spammer?
relayn.netpilot.net	162	162	0.921	1710	1710	0	yes
s2.directhorizon.com	90	90	0.993	1248	1248	0	yes
srv02.ihouseu.com	114	114	0.886	804	804	0	yes
mail.aku.sk	48	48	1.000	702	702	0	yes
email.renci.org	162	162	0.757	14304	1837	547	yes
rv-out-0506.google.com	162	3	0.092	7698	24	7674	no
mailfw2.dd24.net	162	3	0.108	3906	12	3894	no
yw-out-2122.google.com	162	3	0.090	2070	6	2064	no
lyris.media3.net	26	2	0.085	2184	6	2178	no
el-out-1112.google.com	27	2	0.092	3030	6	3024	no

Additionally, the behaviors of mail servers were analyzed based on the frequency (interval) of received email messages. The aim was to investigate any correlation in the sending patterns of the servers; and show if the system is correctly modeling the behavior of servers. To achieve this, the Pearson correlation test was applied. This approach was used to find the correlation between the frequencies of email messages received from mail servers. Table II shows the results of the statistical correlation test for the top 7 servers that sent the highest count of spam emails. The ratio of (number of positive correlations)/(number of total correlations) shows that the servers have 76% similarity in their sending patterns. As shown in Table II, most of the servers have positive correlations, meaning that they have the same behavior.

TABLE II

Statistical analysis of top 7 spamming servers.

Hostname Host #	1.	2.	3.	4.	5.	6.	7.
1. acm26-3.acm.org	1						
2. acm26-4.acm.org	0.2	1					
3. email.renci.org	-0.25	0.26	1				
4. mail-hub-1.cs.cornell.edu	-0.06	0.2	0.13	1			
5. mail-hub-2.cs.cornell.edu	0.03	0.33	0.24	0.33	1		
6. adsl-dynamic-pool-xxx.fpt.vn	-0.14	0.01	0.2	0.46	0.16	1	
7. mx.rinet.ru	-0.34	-0.01	0.23	0.27	0.4	0.1	1

TABLE III

SVM method.

Experiment-1	# of spam	# of benign	False positive	Accuracy
training set	560	4300	2.33%	97.6%
testing set	120	232		

TABLE IV

Decision tree method

Experiment-1	# of spam	# of benign	False positive	Accuracy
training set	560	4300	17.3%	82.6 %
testing set	120	232		

The false positive rate and the accuracy are the most two important parameters to measure the performance of spam classification. As can be seen from the results, the SVM method performed better than the decision tree method. Using more features may improve this performance. However, doing so would create a more complex network that would utilize more space and time.

4. Conclusion

This study proposes a new system that combines predictive analytics and machine learning techniques to overcome the spam problem. A prototype of the system has been developed on the Azure platform and the behavior of email servers has been analyzed. The results showed that spam volumes increase with the number of received emails and there is not a single domain that sends only benign emails. This suggests that spammer activity is distributed across domains. Even if the spamming domain is rejected, spammers keep on spamming, most probably through relay servers. Most spam goes through the DNSBL filter undetected, possibly because spammers are using dynamic IP addresses. This may also suggest that individual machines (bots) are compromised so that they easily pass the DNSBL filtering. The undertaken in this study analysis reveals that current methods to prevent and filter spam through DNSBL, white lists, and anti-spam filtering are not sufficient. The foundations have been laid for follow-up work on containment of spam through the introduction of different trust identities that will transform the current email systems into a more secure email infrastructure. Most of the spamming in the email infrastructure is coordinated out of botnet, proven by this studies' correlation of the burstiness of each of the spamming servers.

References

[1] A. Pathak, Y.C. Hu, Z.M. Mao, in: *Proc. 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, Berkeley (USA)*, 2008, p. 3.

[2] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G.M. Voelker, V. Paxsonf, S. Savage, *Spamalytics: An Empirical Analysis of Spam Marketing Conversion*, ACM 2008.

[3] A. Ramachandran, N. Feamster, D. Dagon, in: *Proc. 2nd Conf. on Steps to Reducing Unwanted Traffic on the Internet, Berkeley (USA)*, Vol. 2, 2006, p. 8.

[4] E. Passerini, R. Paleari, L. Martignoni, D. Bruschi, in: *Detection of Intrusions and Malware, and Vulnerability Assessment*, Ed. D. Zamboni, Springer, Berlin 2008, p. 186.

[5] Y. Xie, F. Yu, R. Panigrahy, *Spamming Botnet: Signatures and Characteristics*, Microsoft Res., 2008.

[6] S. Shah, B.D. Noble, *Softw. Pr. Exper.* **37**, 1515 (2007).

[7] B. Leiba, J. Ossher, V.T. Rajan, R. Segal, M.N. Wegman, in: *CEAS 2005 — 2nd Conf. on Email and Anti-Spam*, 2005, Stanford University, California 2005.

[8] S. Kumar, X. Gao, I. Welch, M. Mansoori, in: *2016 IEEE 30th Int. Conf. on Advanced Information Networking and Applications (AINA)*, 2016, p. 973.

[9] H. Xue, F. Li, H. Seo, R. Pluretti, in: *2015 IEEE Trustcom/BigDataSE/ISPA*, 2015, Vol. 1, p. 726.

[10] L. Chen, Z. Yan, W. Zhang, R. Kantola, *Future Gener. Comput. Syst.* **49**, 77 (2015).

[11] Y. Dai, S. Tada, T. Ban, J. Nakazato, J. Shimamura, S. Ozawa, in: *Neural Information Processing*, Eds. C.K. Loo, K.S. Yap, K.W. Wong, A.T.B. Jin, K. Huang, Springer Int. Publ. 2014, p. 365.

[12] *Machine Learning Documentation Microsoft Azure*.

[13] D. Edla, V. Reddy, V. Gondlekar, V. Gauns, *Acta Phys. Pol. A* **130**, 78 (2016).

[14] E. Kanca, F. Çavdar, M.M. Erşen, *Acta Phys. Pol. A* **130**, 365 (2016).

[15] M. Cevri, D. Üstündağ, *Acta Phys. Pol. A* **130**, 45 (2016).

[16] N. İyit, H. Yonar, A. Genç, *Acta Phys. Pol. A* **130**, 397 (2016).

[17] M. Günay, V.H. Sanchez Espinoza, A. Travleev, *Acta Phys. Pol. A* **128**, B-110 (2015).

[18] Apache, *SpamAssassin*.