A COMBINED METHOD FOR DETECTING SPAM MACHINES ON A TARGET NETWORK

Tala Tafazzoli† and Seyed Hadi Sadjadi††,

†,††Faculty members of ICT security department of Iran Telecommunication Research Center

tafazoli@itrc.ac.ir H.sadjadi@itrc.ac.ir

ABSTRACT

The HITS and PageRank algorithms and K-Means clustering algorithm are two main methods for detecting spam machines. In PageRank algorithm, it is proposed to calculate weights based on different factors. Correct selection of weights has important role in the accuracy of the algorithm. In this paper, we propose a good method for convenient selection of weights. We first executed the K-Means algorithm on the traffic of a big target network and divided IP addresses to two parties, normal and anomalous, and assigned a weight to the IP addresses of anomalous party which is used in calculating the energy rank of the second method. With executing the second algorithm, we found a larger set of IP addresses of spam machines and found that we have increased the accuracy of the algorithms perceptibly.

KEYWORDS

spam, clustering method, K-Means clustering algorithm, HITS algorithm, anomalous behavior.

1. INTRODUCTION

Spam is a side effect of free email service and has become a serious problem that threats every Internet user. According to MessageLabs report [1], 60% of email traffic is spam. Although different methods for combating spam have been proposed, Spam messages are still sent to users' mailboxes. This happens because lots of spam detection methods use filtering.

There are different methods for preventing spam. Most organizations and Internet Service Providers (ISPs) use spam filters which are installed on mail servers. These filters extract keywords and other signatures and use statistical and heuristic methods to determine that an email is spam. But spam senders use complicated methods for combining contents intelligently to mislead content based filters. Thus content based filters do not have high performance. [2]

Most spam researches, concentrate on post-send methods which detect spam after sending, but most of the damage caused by spam is before the usage of these detection methods. These methods are not able to reduce overhead, bandwidth, processing power, time and memory used by spam.

In this paper, we identify machines that are sending spam or machines that are compromised and are distributing spam. This work is done in two parts. First, with clustering algorithm [3], machines are separated to normal and anomalous clusters. The extracted features are based on the volume of traffic the machines are sending (num. of packets, bytes, flows). Then based on ranking and link analysis methods [4] and with the weight extracted from the first section, we detect spam machines. Analysis is done on one day of netflow traffic of a large scale ISP.

In section II, we review the related work. In section III, We outline the structure of our approach. Our approach has two parts. In section 3-1 we describe our network data. In section 3-2, we discuss K-Menas clustering algorithm and in section 3-3, we describe the email servers' behavior formation method. Section 5, concludes our paper.

2. Related work

The increasing trend of spam in recent years has attracted the attention of research community. Recent trends show that most spam methods use botnets instead of direct spa sending. [11][12] Traditional research on spam concentrated on receiver oriented spam detection such as mail filters and blacklists. Email address filters, heuristic filters, distributed blacklists and challenge-response techniques [6] are examples of those researches. [8][9][12] Numerous spam mitigation techniques try to understand spammer's behavior. Several studies have used email sinkholes or honeypots to study spammer properties. In these methods, large volumes of spam are collected in sinkholes and are then processed. Many studies are done on these approaches [12] [13] and different aspects of spammer behavior have been collected. One of these researches is presented by Anirudh Ramachandran and Feamester[12]. In his research, email servers are sinkholes that do not have legitimate email addresses. Thus every received email is a spam. The data is extracted from different email sinkholes of different domains and various properties of network level behavior of spammers were extracted. In [15], data was extracted from a limited sinkhole in a domain and the structural characteristics of scam were studied. But the traces received by these methods are limited to an organizational domain. To extract a broader view of spam problem, Open relay sinkholes were proposed in [11]. The idea of this method is to setup open relays in such a way that it can be easily detected by spammers but doesn't send any spam. In this way, information about the source and destination of spam is extracted.

In [9], another method was proposed by Nick Feamester et al. They propose a method that does not detect spam based on IP address or content filtering but detects spam with behavioral analysis. They used the logs of an organization which had 115 domains and analyzed spam in multiple domains. To classify spam, they clustered IP addresses based on similar behaviors. The idea of their clustering algorithm is "bots of a botnet have similar behavior and send small number of messages to a large amount of servers".

There are other approaches that analyze machines' behavior at network level [4] [10]. These methods analyze netflow traffic. In these researches, a large repository of netflow data has been studied to find behavior that differentiates spam machines from normal email servers. In [4], this analysis is done based on HITS algorithm. In [10], the detection is done in two phases. In the first phase, machines displaying suspicious behavior are extracted. To distinguish these machines, statistics such as the ratio between incoming and outgoing SMTP connections, the number of distinct destinations and the number of outgoing connections are used. In the second phase, only processing suspicious machines according to the first criteria, spam machines are detected with probabilistic calculations such as, number of incoming connections, number of distinct destinations, idle time, standard deviation and the peak behavior are used.

3. Our approach

Our approach combines two methods, K-Means clustering algorithm and HITS and PageRank algorithms for constructing graphs of email servers' behavior. In the first section, we use K-Means clustering algorithm [16] and divide the training dataset into two (normal and anomalous) clusters. The centroids of the resulting clusters are then used to detect anomalous behavior of the monitoring data. [3] Our experimental dataset is the netflow traffic of a large scale ISP. We choose K-Means clustering algorithm, because it groups objects based on their feature values into K disjoint clusters. [3] We apply

the algorithm with k=2 on network traffic data and choose three features as number of packets, number of bytes and number of flows. So the algorithm clusters the monitoring data to normal and anomalous IP addresses based on the volume of traffic exchanged. After detection of an anomalous IP address, a weight is assigned to it which is used in rank evaluation. In section 3-3, graphs of email servers' behavior are constructed and the distinction between email servers and spam sending machines is detected. Graph of machine's behavior is constructed in specified time intervals. [4] As it is defined in the PageRank algorithm[4], the weight used in energy calculation, can be assigned based on different factors. In [4], this weight is based on a pre-used value PScore. We use the weight calculated by the clustering algorithm. Using K-Means clustering algorithm for detecting spam, combination of the two methods with each other and determining IP weights K-Means clustering algorithm and using it in the second method are the contributions of this paper. The combinational method is exerted on the sample traffic and spam sending machines have been detected.

3.1. Network traffic

A flow is a summary of traffic traveling in a session. Each flow contains basic information about connection such as IP, source/destination port, number of packets/bytes transferred, protocol used, connection time and TCP flags. Flow record does not contain payload information. Email service connection uses SMTP protocol and its destination port is 25. Thus the analysis is done on TCP traffic with destination port 25. Because netflow traffic information is at medium level and does not contain the payload information of a packet, this method does not have problems of methods that use payload data.

3.2. K-Means clustering method

K-Means clustering algorithm, groups data based on their feature values into K clusters. Objects in a cluster have similar feature values. K is a positive true number that determines the number of clusters and is determined at the beginning of the execution of the algorithm. Now we define steps of K-Means clustering algorithm.

1) Define the number of clusters.
2) Define K different centroids for each cluster. This work is done by arbitrarily dividing objects into K clusters, determining their centroids, and evaluating whether these centroids are different from each other. Alternatively, the centroids can be initialized to K arbitrarily chosen, different objects.
3) Iterate over all objects to determine the distance of each object to the centroid of that cluster. Each object is assigned to the cluster of the nearest centroid.
4) Realculate the centroids of new clusters.
5) Repeat step 3 until centroids doesn't change anymore.

The distance function, which is used in this algorithm to calculate the distance between 2 objects, is the Euclidean distance which is defined in formula (1).

$$d(x, y) = \sqrt[2]{\sum_{i=1}^{m}(x_i - y_i)^2} \qquad (1)$$

Where $x=(x_1,x_2,\ldots,x_m)$ and $y=(y_1,y_2,\ldots,y_m)$ and m is the number of features. In this paper, features are number of packets, number of bytes, number of flows and K is 2. We used the K-Means clustering algorithm on the training dataset, half an hour of the ISP traffic, which contains normal and anomalous information.

Clustering algorithm, divides training dataset into K clusters. In the clustering algorithm, it is important to define the number of clusters correctly. We choose K=2, with this assumption that normal and anomalous traffic forms two different clusters.

K-Means clustering algorithm calculates centroids for normal and anomalous clusters and these centroids are used for detecting anomalous behavior in the network monitoring traffic. New flow records are preprocessed and transformed and their feature values are extracted. To detect anomalous behavior, two distance-based methods could be deployed. These methods are classification and outlier detection which is combined in this paper.

Classification method: In this method, the distances to the centroids of clusters and the new traffic are calculated using Euclidean distance function. The new traffic is classified as normal if it is closer to the centroid of the normal cluster than the centroid of the anomalous one. This distance based classification allows detecting that kind of abnormal traffic and is similar to the characteristics of the training dataset.

Outlier detection method: An outlier is an object which is different from other objects significantly. Thus it can be recognized as anomaly. For outlier detection, only the distance to the centroid of normal traffic is calculated. If the distance between the object and centroid is larger than a predefined threshold, d_{max}, the object is known as an anomaly.

Combined classification and outlier detection method: The classification and outlier detection are used in combined way to reduce the limitations of each method. If the two methods are used simultaneously, an object is known as anomaly if it is closer to the centroid of abnormal cluster or its distance to the centroid of normal cluster is larger than a predefined threshold.

The combination of classification and outlier detection is used in this paper.

3.3. Email servers' behavior formation method

Email servers receive/send emails from/to other email servers. Thus email servers form a community due to interactions with each other and they form a bipartite graph. We use the email servers' behavior to distinguish between normal and anomalous traffic. The bipartite graph is used in other domains such as the web.

3.3.1. Hubs and Authorities

Bipartite graph has been used for web mining. A bipartite core (i,j) is a bipartite subgraph with i nodes of one set of nodes to j nodes of another set of nodes.

With reference to the graph concept, i pages that have communications with other pages are referred to as hubs and j pages that are referenced are the authorities. For a set of pages related to a topic, a bipartite core which includes hubs and authorities is determined using HITS algorithm. [18] Hubs and authorities are important because they serve as good sources of information for that topic. In the domain of email traffic flow, hubs are equivalent to machines that send emails and authorities are machines that receive emails and together they form a bipartite core. Email servers are good hubs and good authorities. Thus the bipartite graph captures the behavior of machines that are email servers. We now describe HITS algorithm. [18] We associate to each email server an authority weight a_p and a hub weight h_p. The reciprocal relationship between hubs and authorities is as follows. If p points to many servers with large x values, then it should receive a large y value and if p is pointed by servers with large y values, then p should receive a large x value. Now we can define two I and O operations. I is defined in formula 2.

$$a_p \leftarrow \sum_{q:(q,p)\in E} h_q \qquad (2)$$

O updates y weights and is defined in formula 3.

$$h_p \leftarrow \sum_{q:(q,p)\in E} a_q \qquad (3)$$

I and O operations strengthen hubs and authorities.

Let A be an adjacency matrix. If there exists at least one connection from machine i to machine j then $A_{i,j}=1$ else $A_{i,j}=0$. The HITS algorithm is as follows. This is a recursive algorithm that assigns to each node a hub and an authority score.

Let *a* be the vector of authority scores and *h* is the vector of hub scores

A=[1,1,…1], *h*=[1,1,…1];
do
    $a=A^T h;$
    $h=Aa;$
    Normalize *a* and *h*;
while *a* and *h* do not converge (reach a convergence threshold)
return a,h;

3.3.2. Detecting spam senders

In order to detect spam senders, we have to differentiate their behavior from email servers. They both have high outgoing traffic. However email servers send email to other email servers whereas spam machines send emails to all machines. We use this aspect to detect spam senders.

Execute the following steps:

1- Preprocess netflow data and construct the graph of email connections.
2- Execute the HITS algorithm on this graph.
3- Eliminate the k% edges between hubs and authorities. These connections showed normal email traffic between normal email servers.
4- Then execute the HITS algorithm on the resultant graph.
5- The new ranks are the spam sending scores.

This algorithm is a two phase algorithm. First it identifies the connections between regular email servers. These connections form a bipartite graph between servers and assigning them hub and authority scores. Then all the connections that contribute normal traffic between email servers are then eliminated. In this stage only edges are removed and not the nodes. This removes the normal email servers' behavior. The second step identifies machines that behave like servers and have high volume of outgoing traffic that are not related to regular email connections. These machines are probably spam machines because they send emails to lots of machines that do not participate in normal email connections.

3.3.3. Rank evaluation

For each node, based on email sender score, a rank is determined and it is called the spam sending rank. [4] Another metric is then calculated based on email sending metric and is called email sending height (PHeight). For the ith node at time t, its height can be determined by formula (4).

$$PHeight_{it}=\log_2(1+1/PR) \qquad (4)$$

For a node with high rank, PR=1 and PHeight=1 and a node with infinite rank, PR=∞ and PHeight=0. Then rate of changes in the rank of a node is calculated over time. Changes for the time period Δt, is calculated in formula (5).

$$v=\Delta PHeight/\Delta t \qquad (5)$$

Since we are interested in changes and not in a positive or a negative change, we take the square of v for our analysis. We also assign a weight to each node based on the results of the K-Means clustering algorithm. This is the result of the combinational method and is the contribution of this paper. As it is said in [4], the node could be weighed based on different factors. In [4], weights are chosen based on PR but we choose weights based on K-Means algorithm which increases the accuracy of rank energy. K-Means is a clustering algorithm and with (K=2) divides IP addressed to two normal and anomalous clusters. The anomalous IP addresses are assigned a weight which is used in rank evaluation. The energy rank of each node is measured as in formula (6).

$$\text{Rank Energy} = \text{Weight} * v^2 \quad (6)$$

Results of the PageRank method [4] and the combinational method are shown in section 4-1. The rank energy is a good indicator of rapid changes of network behavior of nodes. Rapid changes are important for the system analyst because they indicate machines that send spam suddenly or are email servers going down.

4. Results evaluation

Experiments were done in three phases. These experiments were executed on one day of netflow traffic of a big ISP. First the K-Means clustering algorithm was exerted on half an hour of netflow traffic and information was divided to normal and anomalous clusters. The composed method was exerted on 24 hours of data, every 15 minutes of each hour. First K-Means clustering algorithm was applied and if the machine belonged to the anomalous cluster, a weight was assigned to it. The algorithm defined in section 3-3-2, was executed on netflow traffic and IP addresses sending spam were determined. Then the rank of IP addresses based on the weight calculated in clustering section was calculated. The combined method was implemented in Visual C#.

4.1. The results of the application of the combined method

First, half an hour of netflow traffic was used by K-Means clustering algorithm. The data based on three feature values - number of bytes, number of flows and number of packets- was divided to two clusters : normal and anomalous. Then the analysis was done on 24 hours of traffic. In every 24 hours, 15 minutes of every hour were extracted and the K-Means clustering algorithm was exerted on it. The Euclidean distance of each machine to the centroids of normal and anomalous clusters was calculated. If the IP belonged to anomalous cluster, a weight was assigned to it. Then we applied the HITS algorithm, and calculated hub and authority scores for each machine. The relations between email servers with top hub and authority scores were removed and the HITS algorithm was executed again. In this way, the machines with high hub rank were known as spam senders. Then the energy rank of the internal IP addresses of the ISP was calculated two times. Once it was calculated based on the weight defined in [4] and the second time it was calculated based on the weight assigned by K-Means clustering algorithm defined in section 2-3. IP addresses with high hub scores, gained high ranks. The results are shown in table 1. IP address X.133.201.23 has high hub score in two hours of the day. The energy calculated for this machine with the method proposed in [4], as shown in the table, reports no abnormal behavior. The IP address X.133.203.167, has high hub rank in 6 hours of the day. The energy calculated with the combinational method is high in 3rd hour of the day, but is not high in other hours because there is no change in the situation of the system. The method proposed in [4], doesn't show high energy ranks for some of these times. The IP address, X.133.206.80, has normal behavior.

5. Conclusion

In this paper, a combined method for detecting spam machines was proposed. The combined method is based on two algorithms proposed in [3] and [4]. A weight was assigned to the machine that was

known anomalous or abnormal. This weight was used for calculating spam machine ranks in the second method. This work is limited to modeling in single node level. Further research can be done for modeling in multiple node level.

Table 1. The results of the combinational method and the simple method on the sample dataset

IP	Hub Score	Energy of Combination method	Energy of HITS algorithm	Out-Degree
X.133.201.23	0.99142	44.61385	9.0045	250
	0.01574	17.53087	2.11135	106
	0.16238	25.61234	1.04472	179
	0.02157	7.12391	0.03303	125
	0.0985	14.05943	2.01935	209
	0.09335	0.02072	2E-05	190
	0.01743	4.15987	0.02386	154
	0.0151	0.02792	0.00018	120
	0.00422	0.62389	0.01479	144
	0.0124	1.31297	0.01059	154
	0.01733	0.1747	0.00101	147
	0.02573	0.35674	0.00139	172
	0.34131	16.47826	6.04965	202
	0.00261	5.0652	0.19438	105
	0.01619	4.92556	0.03042	160
	0	0	0	51
	0.07496	28.07264	3.71352	152
	0.01431	3.37999	2.02362	186
	0.00974	0.13038	0.00134	145
	0.00517	0.18886	0.00365	132
	0	0	0	0
	0	0	0	0
	0	0	0	0
X.133.203.167	0.04192	4.00323	0.0955	49
	0.23139	5.08506	0.02198	228
	0.98668	86.22088	7.00874	215
	0.99772	0.00288	0	278
	0.99514	0.00016	0	284
	0.9953	0	0	243
	0.99985	0.00048	0	239
	0.68061	2.79503	0.00041	445
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0
X.133.206.80	0 0.00016 0 0 0 0 0 0 0.00991 0.00059 0.00078 0.0061 0.01083 2E-05 0.0008 0.04364 0.00069 0.00532 0.00179 0.00341 0.00198 1E-05 0 0	0 3.53885 0 0 0 0 0 0 16.21924 0.04315 0.00055 0.23842 0.03238 0.00661 0.10863 6.32478 0.10747 0.20433 0.01951 0.01314 0.00539 0.00197 0 0	0 0.9296 0 0 0 0 0 0 5.29992 0.07314 0.0007 0.03906 0.00299 0.38248 0.13595 0.14494 0.15599 0.03844 0.01092 0.00385 0.00272 0.30233 0 0	0 3 0 0 0 0 0 0 12 4 6 11 8 5 12 16 6 8 4 7 5 11 1 2

References

[1] http://www.messagelabs.com/.

[2] Ho-Yu Lam, Dit-Yan Yeung, “A learning approach to spam detection based on social network”, Hong Kong university of science and technology, 2007, www.ceas.cc/2007/papers/paper-81.pdf.

[3] Gerhard Munz, Sa Li, Georg Carle, ″Traffic anomaly detection using K-Means clustring″, Hong Kong university of science and technology, 2007.

[4] Prasanna Desikan, Jaideep Srivastava, "Analyzing network traffic to detect E-Mail spamming machines", Department of computer science,University of Minnesota, 2004.

[5] Wilfried N. Gansterer, Helmut Hlavacs, Micheal Ilger, Peter Lechner, Jurgen Straub, “Token Buckets for outgoing spam prevention”, Institute of distributed and multimedia systems, university of Vienna, 2006.

[6] Mengjun Xie, Heng Yin, Haining Wang, "An effective defense against email spam laundering", *ACM CCS'06*, 2006.

[7] W. Gansterer, M. Ilger, P. Lechner, R. Neumayer, J. Straub, "Anti-spam methods – state-of-the-art", University of Vienna, 2005.

[8] S. Gaeeiss, M. Kaminsky, M. J. Freedman, B. Karp, D. Mazieres, and H. Yu. Re: Reliable email. In *Proc USENIX NSDI 2006*, San Jose, CA, MAY 2006.

[9] Anirudh Ramachandran, Nick Feamster and Santosh Vempala, Filtering spam with behavioral blacklisting, *Proc. ACM Conference on Computer and Communications Security (CCS)*, 2007.

[10] Gert Vliek, Detecting spam machines, a netflow-data based approach, University of twente, 2009.

[11] Abhinav Pathak et al., Peeking into spammer behavior from a unique vantage point, LEET '08, April 2008.

[12] Anirduh Ramachandran et al., Understanding network-level behavior of spammers, Nanog 37, Sept 2006.

[13] L. H. Gomes, C. Cazita, J. M. Almeida and J. Wagner Meira, Workload models of spam and legitimate emails, *Perform Eval.*, 64(7-8): 690-714, 2007.

[14] S. Venkataraman, S. Sen, O. Spatscheck, P. Haffner and D. Song, Exploiting network structure for proactive spam mitigation, In *Proc. Of Usenix Security*, 2007.

[15] D. S. Anderson, C. Fleizach, S. Savage and G.M. Voelleer, Spamscatter:Characterizing internet scam hosting infrastructure, In *Usenix Security*, 2007.

[16] J. MacQueen, "Some methods for classification and analysis of multivariate observations", in *Proceedings of 5-th Berkeley Symposium on Mathematical statistics and probability*, University of California, 1967, pp. 281-297.

[17] Enrico Blanzieri and Anto Bryl, " A survey of learning-based techniques of email spam filtering", Technical Reprot, University of Trento, 2008.

[18] J.M.Kleinberg, "Authoritative sources in hyperlink environments", 9th annual ACM-SIAM symposium on discrete algorithms, pages 668-667, 1998.

Authors

Manager of Information society security group in ITRC

Faculty member of Information society security group in ITRC