A proposal of a method to detect spam from information messages

Michal Rickwood[1], and V. Oujezský[2]

[1]Brno University of Technology, The Czech Republic

[2]Brno University of Technology, The Czech Republic

E-mail: 221568@vut.cz, oujezsky@vut.cz

Abstract—This paper presents a spam detection algorithm that uses solely traffic flow logs in the form of Netflow messages. Internet service providers must detect spam in order for their entire subnets not to be marked as spamming stations. The algorithm was drafted based on an analysis of various datasets containing Netflow records. These datasets consist of valid e-mails, spam and common non e-mail related traffic. The algorithm uses domain name system blacklist verification as the first step of identifying a spamming station. Furthermore, theoretical models of valid clients and spammers have been laid out. In continuation of this work, the dataset will be studied to find correlation with the models. Included in the tracked parameters one can find the number of incoming and outcoming messages, timestamps amongst others.

Keywords—Detection, e-mail, flow, security, spam

1. INTRODUCTION

This paper presents a topic that is encountered by every e-mail user, which in today's society corresponds to approximately 50 % of the world's population. This topic is spam. Spam is a method of sending unwanted e-mail messages to a large number of recipients. Although the ratio of spam to legitimate mail has shown a downward trend in recent years, approximately 45 % of traffic falls into the first category [1].

Many systems have been designed and implemented for detection. Most of them require scanning the content of the messages sent. This approach appears to be relatively simple and highly effective, since artificial intelligences can identify spam mail essentially flawlessly from learned datasets. However, the law side of the equation enters the picture and the whole situation becomes significantly more complicated. Filtering by email content is an obvious and serious invasion of privacy. The problem arises for the ISPs that provide the email server. If a client uses that provider's services for spamming purposes and is flagged as a spammer in the UCEPROTECT project, other users in the subnet are affected. The provider is of course then subject to legal and mostly financial consequences. For this reason, the provider must monitor outgoing traffic from its server in a way other than by scanning the content of the messages in order to be able to detect a possible spamming station earlier.

One widely used detection method is using NetFlow messages. These records contain only metadata about the communication and the users, not the content of the communication itself. This information, which is included in the packet headers, includes such things as Internet Protocol sender and receiver addresses, port numbers, and more. An approach based on monitoring network traffic from NetFlow messages is also addressed in this paper. In order to identify and model the behavior of spamming devices, we use the already created NetFlow message dataset [2]. As a first step, the algorithm looks up the IP address in the domain name system blacklist. It then compares the metadata about the station with the previously created models of the spammer and legitimate user based on the datasets.

2. THE CURRENT METHODS USED

Many of today's spam detection algorithms still work by reading the content of messages, even if only the subject field, for example, and not the text itself. Providers that interfere with messages include leaders such as Google, Yahoo and Outlook [3]. They use various machine learning methods such as neural networks, k-nearest neighbor algorithm and others for detection. The methods are then learned on large sets of spam and valid emails. In addition to using already established rules, these learning-based algorithms can create their own rules. Google's spam detection algorithm has advanced to the stage where it can detect up to 99.9 % of spam. Conventional machine learning algorithms compare each message

against valid and spam data before setting spam detection rules. However, there are also design systems that have been inspired by artificial immune systems and use special features to generate detectors to cover the spam space [4].

Providers use methods that have a very high success rate, but the problem still lies in the exploitation of the content. However, there are many algorithms or method designs that detect spam without reading the messages. These methods observe and collect data about traffic on the network, which they then aggregate by source address [5]. For example, the following table I might result, where an observer has observed the number of incoming connections, outgoing connections on port 25, and the number of distinct addresses in both directions. The table I is sorted by the number of outgoing connections.

IP	dist out	out	dist in	in
1	1980	334356	36354	675381
2	3227	247588	36354	17645
3	11459	11459	36354	745408
4	**39460**	**244117**	**0**	**0**
5	11280	240733	153275	675632
6	3512	238665	788	27738
7	7943	195573	132616	539297
8	**2**	**184698**	**0**	**0**
9	2252	136847	10	187
10	**24213**	**116898**	**1**	**2**
11	7774	115746	8	24972
12	8376	68413	24	172464
13	**17532**	**64685**	**0**	**0**
14	341	57251	66237	901280
15	443	54212	10	578

Table I: Example of the result of traffic data aggregation by IP addresses [5].

In the table I we can see that most IP (Internet Protocol) addresses that have a high number of outgoing connections also have a similar number of incoming connections. In this case, author Vliek presents the assumption that these are active but valid stations. Highlighted, on the other hand, are stations that have a high number of outgoing connections but few or no incoming connections. According to Vliek's assumption, these stations are candidates for spam clients. However, he also states that there are legitimate stations that are only used for messaging. This category includes, for example, accounts set up by banks to send account information, online newsletters that send e-mail in some cases even several times a day. However, most of these cases will have a relatively small number of different IP addresses to which they send mail.

3. THE PROPOSED METHOD

The development of the detection algorithm can be carried out in several different environments with different software and hardware resources. The main hardware requirement for the environment is a relatively high computational capacity. The most practical solution turned out to be the use of the cloud. One variant of this solution is from Kaggle [6].

The detection algorithm is applied to the dataset. It consists of data items of the same type, in our case a packet that was sent over the monitored network. The parameters tracked include the following – send time, protocol type, source IP address and port, destination IP address and port and other.

We use already created datasets. Extensive datasets are available captured from CTU in Prague [2].
Figure 1 shows a schematic of the proposed detection algorithm. The first step is to convert from .pcap

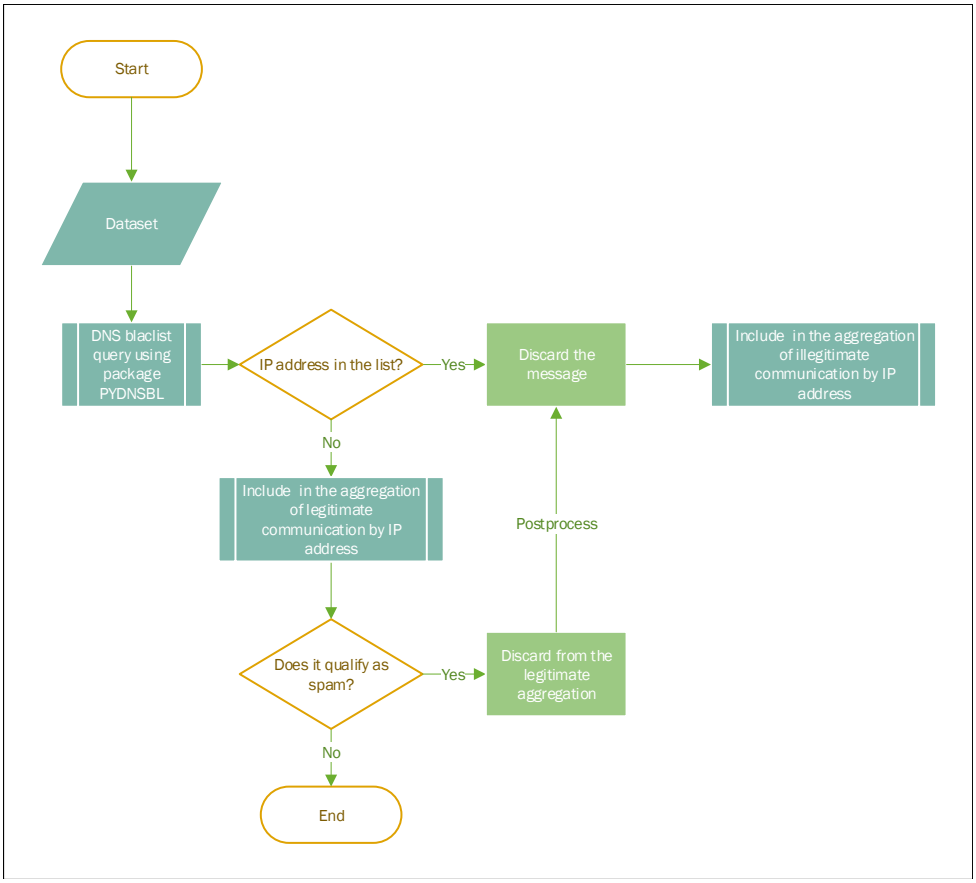

Figure 1: The basic schema of the algorithm.

format to .csv format and then retrieve it using the pandas package [7]. The next step is to query the DNS (Domain Name System) blacklists using the pydnsbl package [8]. If the IP addresses are listed, the message would be dropped by the provider. In our case, it just adds the data from the netflow message to the aggregation of illegitimate traffic. If the response from the DNS blacklist query is negative, the information from the netflow record is added to the aggregation of legitimate communication. It is then checked that the IP address from which the communication is sent still falls into the category of a legitimate station and has not crossed any of the specified thresholds to further verify that it is not a spam station. If no threshold is crossed, the process is terminated.

The main indicator of suspicious behavior is the threshold of a large amount of activity in terms of outgoing connections and on the contrary a small amount of incoming traffic. The second threshold is the traffic behaviour in time computed from the send time NetFlow value. We are looking for a specific repetitive behaviour in terms of time / repetitions. The main value of the threshold is weighted centrality measurement of the average duration. However, this information is definitely not enough. Another important indicator is the ratio of active to idle time. If all communication is sent at one time every day or periodically, for example, every 20 minutes, this station has a higher chance of being a spamming station. The algorithm itself is implemented in Python in the Kaggle web environment.

4. CONCLUSION

The aim of the research was to design an algorithm that can detect end stations in the network generating " spam" traffic from the traffic log in the form of Netflow messages. The primary reason for filtering using logs without interfering with the messages themselves is to protect privacy and personal information.

As part of the design, an analysis of the current spam filtering solution with and without interference to the content of the communication was first performed. The algorithm in its current version retrieves the underlying data, can check the IP address against a list and produce basic aggregated traffic data from the perspective of individual stations. Due to time constraints, not all IP addresses in the dataset used

have been scanned yet, but only the first 10,000 records. Of those addresses, 27,66 % were listed on at least one list. On average, then, an address was listed on 1,9378 lists in the case of a positive response.

Several IP addresses with suspicious behavior were found in the statistics. The main indicator of such suspicious behavior was a large amount of activity in terms of outgoing connections and on the contrary a small amount of incoming traffic. For these stations, there was also a very low number of stations being sent to. At the moment, it is not possible to firmly establish whether these are end stations generating spam traffic. It is possible that these are automated informational e-mail boxes.

Based on the analyses, parameters will be determined which will be monitored in terms of IP addresses and by which it will be decided whether the station is a legitimate user or a spammer. At this stage, any reasonable statistics of the proposed algorithm are not available. Currently, two parameters are used that still need to be tuned. Depending on the accuracy, additional parameters can be added. One of the approaches that will be developed in the continuation of this work is traffic monitoring according to the author Vliek.

REFERENCES

[1] J. Johnson, "Global spam volume as percentage of total e-mail traffic from january 2014 to march 2021, by month." [Online]. Available: https://www.statista.com/statistics/420391/spam-email-traffic-share/

[2] S. Garcia, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods." [Online]. Available: http://dx.doi.org/10.1016/j.cose.2014.05.011

[3] E. G. Dada, J. S. Bassi, H. Chiroma, A. O. Adetunmbi, O. E. Ajibuwa *et al.*, "Machine learning for email spam filtering: review, approaches and open research problems," *Heliyon*, vol. 5, no. 6, p. e01802, 2019.

[4] I. Idris and A. Selamat, "Improved email spam detection model with negative selection algorithm and particle swarm optimization," *Applied Soft Computing*, vol. 22, pp. 11–27, 2014.

[5] G. Vliek, "Detecting spam machines, a netflow-data based approach," University of Twente, The Netherlands, 2009.

[6] Kaggle. [Online]. Available: https://www.kaggle.com/

[7] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[8] dmippolitov, "Pydnsbl," Sep. 2017. [Online]. Available: https://github.com/dmippolitov/pydnsbl/