Classifying DNS Servers based on Response Message Matrix using Machine Learning

CSCI-ISCW / Poster Paper

1st Keiichi Shima
IIJ Innovation Institute, Inc.
Tokyo, Japan
keiichi@iijlab.net

2nd Ryo Nakamura
The University of Tokyo
Tokyo, Japan
upa@nc.u-tokyo.ac.jp

3rd Kazuya Okada
The University of Tokyo
Tokyo, Japan
okada@ecc.u-tokyo.ac.jp

4th Tomohiro Ishihara
The University of Tokyo
Tokyo, Japan
sho@c.u-tokyo.ac.jp

5th Daisuke Miyamoto
The University of Tokyo
Tokyo, Japan
daisu-mi@nc.u-tokyo.ac.jp

6th Yuji Sekiya
The University of Tokyo
Tokyo, Japan
sekiya@nc.u-tokyo.ac.jp

***Abstract*—Improperly configured Domain Name System (DNS) servers are sometimes used as packet reflectors as part of a DoS or DDoS attack. Detecting packets created as a result of this activity is logically possible by monitoring the DNS request and response traffic. Any response that does not have a corresponding request can be considered a reflected message; checking and tracking every DNS packet, however, is a non-trivial operation. In this paper, we propose a detection mechanism for DNS servers used as reflectors by using a *DNS server feature matrix* built from a small number of packets and a machine learning algorithm. The F1 score of bad DNS server detection was over 0.9 when the test and training data are generated within the same day.**

***Keywords*-DNS, reflection, DoS/DDoS detection, machine learning**

I. Introduction

Domain Name System (DNS) is one of the most important technologies of the Internet. We can convert a domain name into an IP address using DNS. Without this service, the Internet would not be deployed as widely as it is now. DNS messages are normally built on top of UDP packets. Unlike in TCP, it is easy to forge the source address of UDP packets. As a result, DNS requests with a fake source address can easily be sent to a DNS server. In theory, any DNS server can answer any domain name resolution request; there are no protocol requirements that limit or filter request messages from client nodes. When DNS was invented, malicious activity utilizing DNS servers as packet reflectors was not extensive; however, as the Internet grew, attackers started to use this open operating policy to send traffic to victim nodes by forging DNS message source addresses. To prevent this activity, recent DNS servers have been configured to answer requests originating only from specific client nodes, typically filtered by source IP address. Unfortunately, there are more than a few improperly configured DNS servers in the wild; these are called open resolvers[1]. In this paper, we propose a method of classifying a DNS server, according to whether or not it is used as a reflector, by monitoring the incoming DNS messages. We collect a series of DNS packets sent from a DNS server and build a feature matrix of the server, assuming that a reflector may have a different packet sequence pattern than that found with a normal DNS server. The preliminary result shows that our method can classify reflectors with an F1 score greater than 0.9 when the test and training data are generated within the same day.

This work was supported by JST CREST Grant Number JPMJCR1783, Japan.

II. DNS Server Feature Matrix

The basic idea behind this proposal originates from [1]. [1] was invented to detect malicious nodes by investigating a series of TCP SYN packets sent from these nodes. TCP SYN packets are collected based on the source IP addresses of the TCP streams and a feature matrix as an image is generated. In the aforementioned study, it was assumed that the images have different shapes that are dependent on the activities of a malicious host, for example, scanning or DoS. The images generated from SYN packets were used as training data of a deep learning network using a CNN algorithm.

We follow a similar process in our proposal. The difference is that we use DNS response packets received from servers as an input for building the feature matrix.

To apply our method, we first create training data. To split the DNS messages into good messages and suspicious messages, we used the mechanism proposed in [2]. We monitor DNS messages at the boundary of an organization's network and check all request and response messages. If there is a DNS server being used as a reflector, and it is sending unintended

[1]DNS Scanning Project: https://dnsscan.shadowserver.org

response messages, we will not see any matching request messages sent from within the organization.

The values we used to generate a feature matrix are shown in TABLE I.

TABLE I
VALUES OF DNS MESSAGE USED TO BUILD A FEATURE MATRIX

Type	Description
Timestamp	Timestamp of a packet
Port #	Source port # of a packet
Size	Size of a DNS message
OPCODE field	Indicating the DNS message type
AA field	Indicating Authoritative Answer or not
TC field	Indicating if a packet is truncated
RD field	Indicating if recursive query is desired
RA field	Indicating if recursive query is available
Z field	Reserved field and should be 0
RCODE field	Indicating result code
QDCOUNT	# of query items
ARCOUNT	# of answer records
NSCOUNT	# of name servers information
AACOUNT	# of additional records

The captured messages are grouped by source IP address (in this case the DNS server IP address), sorted by timestamp, and divided into groups of 100 packets. Fig. 1 shows an example of a DNS server feature matrix.

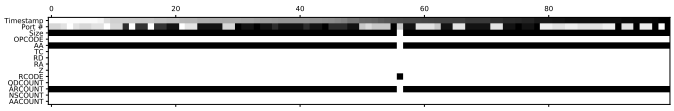

Fig. 1. Example of a visualized DNS server feature matrix

The order of rows is the same as the order presented in Table I. The values are normalized per row. Each column indicates one DNS response message. Because feature matrix is created every 100 packets, the size of columns is 100.

III. LEARNING WITH SVM

The feature matrix image shown in Fig. 1 is based on messages sent from a suspicious DNS server. This particular server kept sending unsolicited DNS response messages; we can guess the behavior by observing the image. A smoothly changing timestamp row means that messages are being sent periodically. Most packets have the same shape except for source port number. Rows that are almost white or black signify, in most cases, the same values.

Fig. 2 shows a feature matrix of a good DNS server. Different from the case shown in Fig. 1, the fields indicating

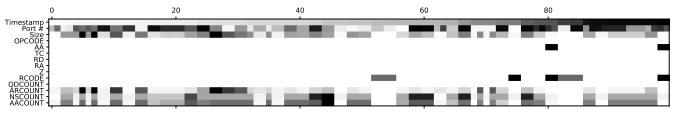

Fig. 2. Example of a feature matrix of a good DNS server

the number of resource records (such as ARCOUNT) in each response packet have several different values. This is plausible because the contents of DNS request messages sent to a specific DNS server vary according to client; responses may also vary, depending on the request messages.

The datasets used with SVM are a single day data of a certain research network captured between 24th August 2019 and 25th August 2019. The sizes of the datasets are listed in TABLE II.

TABLE II
PACKET AND MATRIX COUNTS OF DATASETS

Date	# of Good / Bad DNS pkts	# of Good / Bad matrices
24th Aug.	33,824,531 / 2,863,321	323,269 / 28,291
25th Aug.	30,238,481 / 1,148,935	291,730 / 6,105

The selected hyper-parameters were penalty = 10, gamma = 0.01, and kernel = rbf, using grid search. The model was trained and tested with 20,000 randomly selected good matrices, and 80% of the bad matrices for each day (training/test ratio was 8:2). The table below presents the classification results of bad DNS servers for each day.

Date	Precision	Recall	F1-score	Support
24th Aug.	1.00	1.00	1.00	4,540
25th Aug.	0.98	1.00	0.99	984

The detection ratios were acceptable as long as we use training and test data selected from the same day. Based on these results, we tried to evaluate the rest of the data in the datasets not used in the training phase for each day. However, we have observed unstable results so far. For example, the F1 score of 24th was 0.92 and that of 25th was 0.74. The precision values were largely degraded especially (0.85 and 0.54 respectively). This implies that the produced models generate many false positive results.

IV. CONCLUSION

We attempted to classify DNS servers according to whether or not they were being used as reflectors by capturing a small number of DNS response messages sent from them. We used a method similar to the one proposed in [1] to build a DNS server feature matrix. The preliminary results of classification using SVM show sufficient precision as long as training and test data from the same day is used. It was found that the results can sometimes be dependent on the input training data. Therefore, in the future, we plan on making the results more stable by investigating data and matrix generation approaches (e.g. what values to use to build a matrix) and also by investigating classification algorithms (including deep learning technologies) to achieve superior performance.

REFERENCES

[1] R. Nakamura, Y. Sekiya, D. Miyamoto, K. Okada, and T. Ishihara, "Malicious host detection by imaging SYN packets and a neural network," in *Proceedings of IEEE International Symposium on Networks, Computers and Commnications (ISNCC2018)*, 2018.

[2] G. Kambourakis, T. Moschos, D. Geneiatakis, and S. Gritzalis, "Detecting DNS amplification attacks," in *International Workshop on Critical Information Infrastructures Security*. Springer, 2007, pp. 185–196.