Poster: Detecting Command and Control Servers of Botnet With Randomized Traffic

Di Wu[1,2], Binxing Fang[3], Fangjiao Zhang[1,2]
1 (Institute of Information Engineering, Chinese Academy of Sciences)
2 (School of Cyber Security, University of Chinese Academy of Sciences)
3 (Institute of Electronic and Information Engineering in Dongguan UESTC)
zhangfangjiao@iie.ac.cn

***Abstract*—Botnet continue to be a significant threat to Internet. Accordingly, the present research of botnet traffic detection mainly based on the assumption that communication or attack flows between a botnet tend to have space-time similarities. However, in order to bypass existing detection systems, attackers begin to add some randomness to the process of botnet propagation and control to make the feature matching or aggregating difficult. For example, randomly changing the communication contents or letting bot randomly visit benign domains. In this paper, we address this issue and propose a botnet command and control (C&C) servers detection system to against the randomization attack. The system, combined features of host-side and server-side, successively employs the clustering inference and supervised learning based on feedback mechanism. The two-step structure and two dimensions of features assure that the botnet can be fully detected with lower false positive rate.**

I. INTRODUCTION

A botnet consists of a network of compromised computers controlled by botmasters via C&C channels[1]. Botnet represent a persistent threat to Internet security, and researchers found that bots always work in a coordinated way, which will lead to time-space similarities in communication content and patterns. Based on this assumption, there have been a few studies about botnet detection, which take advantage of clustering methods. Gu et al. [2] perform cross cluster correlation to identify hosts that share both similar communication patterns and malicious activities. Zhang et al. [3] mine the relationships between all servers from multiple dimensions. Lu [4] et al. identify network traffic into known applications and find botnet behaviors based on the n-gram features.

For this kind of detection system, attackers try to create time-space deviation by injecting randomized traffic into the botnet. Cui[5] et al. point out that botmaster can randomize its C&C communication contents to eliminate space similarity(e.g., injecting packet and flow-level noise) and add a random delay to eliminate time similarity. In addition, it may be effective to avoid server clustering by letting bots randomly visit many specific benign domains with the same URI file. How to confront such attacks has become an urgent need.

So traditional detection systems based on time-space similarities may have two problems in the face of randomization attack: cannot bringing together entire botnet traffic and mistakenly clustering benign traffic with malicious traffic.

Due to this issue, we propose a novel detection system based on machine learning to detect C&C servers of botnet with randomized traffic and make two main contributions. First, we design a two-step detection structure. The system uses unsupervised clustering to mine suspicious traffic and filtrates benign traffic by supervised classification model, which utilized time-space similarity and known malicious behavior, respectively. Thereinto, the prior module is used for reducing false negative rate and another module is used to reduce false positive rate. Second, we present a clustering and correlation model with two-dimensional features: host-side and server-side. Suspicious hosts and servers generated from each dimension itself might be unable to dispose the randomized noise interference, so associated results will be correlated for further processing.

II. DESIGN

The primary goal of our system is finding out botnet C&C servers from large-scale network traffic. The structure of our proposed system consists of two main components, showed in Figure 1: (1) the clustering correlation module, (2) the classification learning module. We will introduce how these components work in the following.

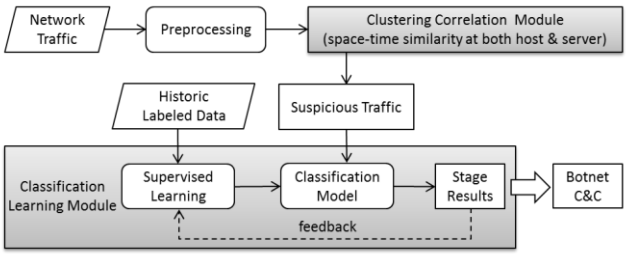

Fig. 1. The architecture of the detection system

Prior to detecting, in order to improve efficiency and reduce the traffic workload, we should preprocess the network traffic dataset for filtering out irrelevant traffic flows.

There are mainly two kinds of traffic that we need to deal with: flows related to well known as legitimate servers(e.g., Google) and communications between internal hosts.

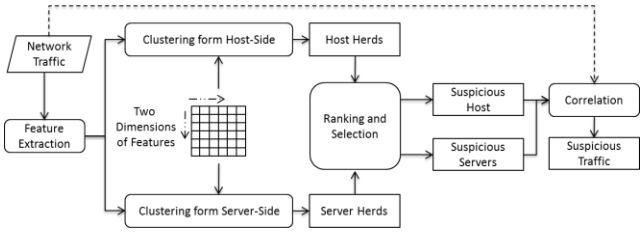

Fig. 2. The process of clustering correlation module

A. *Clustering Correlation Module*

Since bots or servers in the same botnet generally share similar behavior as we discussed before, we firstly cluster the filtered data. However, as randomized traffic could easily interfere the clustering model which from just one angle of characteristic, our system combines features of host-side and server-side, utilizing them to find suspicious traffic, as showed in Figure 2.

On the host side, we identify the host that share both similar communication patterns and malicious activities. In this way, communication flows are analyzed at transport layer, and extracted features from both time and space grounds. Malicious activities monitor is built based on some intrusion detection systems. On the server side, because servers involved in the same malicious campaign usually have similar malware client groups, so we cluster servers which have similar set of clients. Moreover, server attributes are also considered, such as URI files in the server and its whois information. The partial specific selected features is shown in TABLE I.

Once we obtain the host herds and server herds from unsupervised clustering, we perform ranking and selection to choose suspicious samples. In order to do this, we compute a botnet score for each host and server, and filter them by certain detection thresholds. In general, if a host occurs in several different dimensional host herds, it will likely be deemed suspicious with high score, so do servers. Due to the influence of randomized traffic, selected herds from individual dimension may miss some malicious samples and contain a few benign samples. So we extract all traffic which related to suspicious hosts or servers from initial flows. This process ensures that all botnet traffic can be found furthest, i.e., the false negative rate of the system will be reduced.

B. *Classification Learning Module*

In order to further filtering out benign traffic in results extracted from the prior module, we utilize historic labeled data to train a supervised random forest classifier. There are three aspects of features that we selected: flow size, client access patterns and temporal characteristic. The first class of features is depended on the observation that the flow size distributions of C&C servers is necessarily different from benign servers. And the latter two classes are related to the behavior patterns of client connections, for example, normal traffic is usually produced during the day and its access time interval is irregular. However, the malicious traffic is opposite. Through the trained model, the C&C servers of botnet will be discovered and chosen.

For enhancing the robustness of the model, there is a feedback mechanism in place. We label the initial traffic through the stage result produced by the classifier, and combine it with historic data to train the model again. This self-learning process with proper feedback frequency can make the model more applicable to target samples.

TABLE I. SELECTED FEATURES

Module	Feature Property	Partial Specific Features
Clustering (Host-side)	flow similarity	the number of flows per hour, the number of bytes per second
	activity similarity	types of hosts' activities
Clustering (Server-side)	client similarity	shared client groups
	server similarity	URI files, ip address set, whois
Classi-fication	flow size	average packet length, total number of packets
	client access patterns	the time interval between adjacent flows
	temporal features	the connection time period

III. CONCLUSION

To detect botnet C&C servers from large-scale network traffic with randomized traffic, in this paper, we propose a two-step detection structure, which combines cluster analysis based on two dimensional features and supervised classification. These two modules are able to ensure higher completeness and accuracy.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful comments for improving this paper. This work is supported by the Industry-University-Research Cooperation Project of Guangdong Province "Academician Workstation of Healthcare Cloud Security in Guangdong Province" (No. 2016B090921001) and the Ministry of Science and Technology of China (No. 2016QY08D1602).

REFERENCES

[1] Silva S S C, Silva R M P, Pinto R C G, et al. Botnets: A survey[J]. Computer Networks, 2013, 57(2): 378-403.

[2] Gu G, Perdisci R, Zhang J, et al. BotMiner: Clustering Analysis of Network Traffic for Protocol-and Structure-Independent Botnet Detection[C]//USENIX Security Symposium. 2008, 5(2): 139-154.

[3] Zhang J, Saha S, Gu G, et al. Systematic mining of associated server herds for malware campaign discovery[C]//Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on. IEEE, 2015: 630-641.

[4] Lu W, Rammidi G, Ghorbani A A. Clustering botnet communication traffic based on n-gram feature selection[J]. Computer Communications, 2011, 34(3): 502-514.

[5] Xiang C, Binxing F, Lihua Y, et al. Andbot: towards advanced mobile botnets[C]//Proceedings of the 4th USENIX conference on Large-scale exploits and emergent threats. USENIX Association, 2011: 11-11.