Co-Clustering Host-Domain Graphs to Discover Malware Infection

Xin Xie
College of Computer Science and Engineering
UESTC
Chengdu China
sancica@163.com

Weina Niu
College of Cybersecurity
Sichuan University
Chengdu China

XiaoSong Zhang
College of Computer Science and Engineering
UESTC
Chengdu China

Zhongwei Ren
College of Computer Science and Engineering
UESTC
Chengdu China

Yuheng Luo
College of Computer Science and Engineering
UESTC
Chengdu China

Jiangchao Li
College of Computer Science and Engineering
UESTC
Chengdu China

ABSTRACT

Malware is at root of most of cyber-attacks, which has led to billions of dollars in damage every year. Most malware, especially Advanced Persistent Threat (APT) malware make use of Domain Name System (DNS) to control compromised machines and steal sensitive information. Therefore, several security products identified malware infection by combining machine learning technology with DNS data. However, the existing detection approaches cannot simultaneously identify both malicious domain names and infected hosts. To solve the problem, this work proposed a co-clustering based detection approach without labeled data, which integrates active DNS data with graph inference. According to active DNS data, a host-domain graph was generated in the first. Then partial domain nodes were labeled under the aid of blacklist, popular domain list, and Alexa ranking. At last, semi-supervised co-clustering was used to discover potential malicious domains and malware-infected hosts in the monitored network. This work implemented experiments in a network of hundreds of internal hosts that access 145 malware domains. Experimental results showed that the proposed detection approach was able to identify malware domains with up to 97.2% true positives. This work also compared and analyzed the results using different cluster calculating formulas with two different bipartite edge weights. Results showed that clustering with maximum and minimum edge weights has a better tolerance to different distance calculation methods.

AIAM 2019, October 17–19, 2019, Dublin, Ireland
© 2019 Association for Computing Machinery. 978-1-4503-7202-2/19/10...$15.00
https://doi.org/10.1145/3358331.3358380

CCS CONCEPTS

CCS→Security and privacy→Network security→Web protocol security

KEYWORDS

APT, malware-infected hosts, active DNS, semi-supervised, semi-supervised co-clustering, malicious domains

ACM Reference format:

Xin Xie, Weina Niu, XiaoSong Zhang, Zhongwei Ren, Yuheng Luo, and Jiangchao Li. 2019. Co-Clustering Host-Domain Graphs to Discover Malware Infection. In *AIAM' 19: 2019 International Conference on Artificial Intelligence and Advanced Manufacturing, October 17–19, 2019, Dublin, Ireland. ACM, New York, NY, USA, 6 pages.* https://doi.org/10.1145/3358331.3358380

1 INTRODUCTION

Advanced Persistent Threat (APT) has become one of the most critical threats on the internet nowadays. Unfortunately, they are hard to detect due to strong concealment, innovative and sophisticated means [1-2]. Attackers often install APT malware on the infected machine after hacking into the target network, like trojan horse or backdoor. Therefore, attackers can control the compromised hosts to steal sensitive information about users over a long period. However, traditional detection methods based on pattern matching cannot identify APT malwares since attackers exploit unknown vulnerabilities and compromise a limited number of specific hosts to mimic normal behavior [3].

With the aid of Domain Name Server (DNS), attackers' command and control (C&C) servers can communicate with compromised hosts. Traffic analysis is a relatively efficient way to discover unknown malware domains. Since almost malware used DNS to resolve its control server address and DNS traffic is small in all Internet traffic. Therefore, there are many detection approaches based on DNS data have been proposed, like DNS active analysis and DNS passive analysis [4-7]. They can be used to identify malware in a large-scale network with the different C&C protocol. The sensor network environment is also not immune, so several papers [8-11] have studied related security issues. Many mainstream security products extract features based on DNS query/response data and then use machine learning technology to discover malware domains. However, they cannot discover new malware activities without similarity. To solve the problem, some works made use of host-domain or domain-IP graph reasoning to discover malicious domains and host related to malware. Moreover, there are many detection methods combining graph analysis and DNS data analysis to improve accuracy. Unfortunately, the existing approaches cannot simultaneously identify malware domains and infected hosts without training data.

This work is not aimed to identify domains that are surely used by attackers to data exfiltrations. Instead, want to find out thousands of domains accessed by devices of an area - few domains that behave suspicious and malware-infected hosts. In this way, it can help security analysts to be more effective locate C&C servers of APT malware and internal hosts controlled by malware. To achieve this goal, this work propose a co-clustering based detection approach that is able to identify malicious domains and infected hosts at the same time. Blacklist, popular domain list, and Alexa ranking are used to label partial malicious domains and benign domains, respectively. Then, the semi-supervised co-clustering host-IP graph is used to cluster domains and hosts, and it shows that malware infection in different clusters based on the above label rules. The contributions of this paper are threefolds:

- Proposed a hybrid detection method combined machine learning and graph reasoning to identify infected hosts and domain involved in unknown malware.
- Labeled unkown domains using semi-supervised co-clustering based on active DNS data of experimental environment.
- Set edge weights of the host-domain bigraph according to unilateral similarity between host and domain name.

The rest of the paper is organized as follows: section 2 discusses the related works, section 3 describes the hybrid detection approach in detail, section 4 introduces semi-superivised co-clustering, section 5 analyzes the experimental results, conclusions are drawn and future works are highlighted in section 6.

2 RELATED WORK

Scholars proposed many approaches to identify malware C&C based on DNS traffic [12-14], which mainly focus on active, passive DNS data analysis and host-domain, domain-IP graph reasoning.

Several approaches distinguished legitimate and malicious users activities according to group similarity [15−18]. Therefore, researchers made use of similarities of malware infection to calculate domain name score and identify malicious domains or hosts related to malware infection. Antonakakis et al. [19] developed a dynamic, comprehensive DNS reputation system, Notos. The system calculated the reputation scores of a new domain to determine whether the new domain is malicious based on network, zone and evidence features from passive DNS query data. However, the approach cannot able to accurately identify the malicious domains mapped to a new address space each time. To solve the problem, Bilge et al. [20] developed a malware domains detection system based on passive DNS request data, EXPOSURE. The system extracted 15 behavioral features and then used J48 decision tree algorithm to classify domains as malicious or benign. Unfortunately, EXPOSURE is unable to detect malware with less specific features and behavior.

In order to identify malware using network traffic in a network environment with a few infected machines, some works began to convert domain queries and responses to a graph, and then identified malware using graph reasoning [21-24]. Manadhata et al. [25] constructed a host-domain graph and then calculated the reputation scores of domains through belief propagation to detect malicious domains accessed by Internal hosts. However, the approach identified malicious domains based on HTTP proxy data. Lee et al. [26] made use of host-domain graph clustering to detect malicious domains according to sequential correlation. Khalil et al. [27] made use of graph reasoning to discover malicious domain names based on passive DNS data. The approach did not need to trigger malicious activities to capture information comparing to active analysis approaches. The main limitation is that once detection rules are known to the malware user, attackers will find ways to evade detection. To solve the problem, Rahbarinia et al. [28] constructed a machine-domain graph, and then labeled machine and domain nodes and pruned graph, classified domain using behavior-based features.

In order to further improve the recognition accuracy, researchers began to discover malware domain combined DNS data analysis method and graph reasoning. Shi et al. [29] proposed a multi-behavioral-based hybrid learning approach to detect unknown malicious domains. The approach filtered partial traffic according to Alexa ranking of domains and domain whitelist in the first. Second, they selected some domains as seeds according to the results of n-gram analysis on domain names and commercial security products. Third, they made use of Scalable Reputation Propagation (SRP) algorithm to calculate the scores. Fourth, they used Random Forest (RF) to compute a reputation score for each domain. Finally, they ranked domain combined two different rules. But, this work did not identify infected hosts and domain at the same time. Zhao et al. [6] calculated the scores of Internal devices using malicious DNS detection technology and got corresponding suspicious C&C server IP addresses. Then, based on network traffic related to suspicious IP addresses, they used signature-based and anomaly-based detection technology to calculate the scores. Finally, they computed a reputation score for an Internal device combining three different scores to judge whether it is infected or not. However, the approach cannot effectively identify malware that does not rely on domain names and it did not identify infected hosts and domain at the same time.

3 HYBRID MALWARE DETECTION METHOD

The section illustrates the proposed hybrid malware detection method, whose goal is to discover malicious domain and infected hosts without labeled data at the same time. The workflow of the proposed detection method is shown in Figure 1, which includes four components: pre-processing, generating host-domain graph, labeling partial nodes and semi-supervised co-clustering.

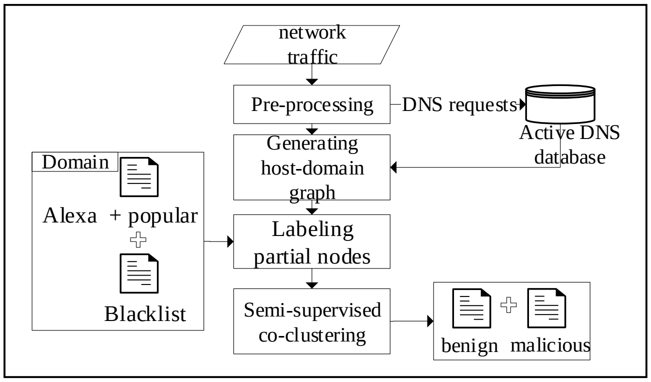

Figure 1: The workflow of the proposed method.

The first component, pre-processing, saves DNS traffic and extracted related information to active DNS database. The second component is used to generate the host-domain graph according to active DNS records. The third component labels benign domain according to Alexa ranking and local top-talking popular domain list. It also labels suspicious domains using blacklist. The output of the component is labels of partial domains and hosts. The third component, semi-supervised co-clustering, discovers all the suspicious domains and infected hosts according to partial labels.

3.1 Pre-processing

Pre-processing component saves the related domain records to an active DNS database according to DNS requests. The related DNS record is represented as dns_data_i (ip_src, port_src, ip_dst, port_dst, qry_name, arr_time). Where, ip_src indicates source IP address of the device that accesses the domain, port_src indicates source port, ip_dst represents destination IP address, port_dst represents destination port, qry_name represents domain, arr_time represents the domain request time when the domain request was initiated. For example, dns_data_1=("223.87.253.100", "53", "10.140.143.137", "64920", "itunes.apple.com.edgekey.net", "Aug19, 2016, 15:11:18.224902000 CST"), which represents that the device in the monitored network whose IP address is 222.87.253.100 accesses itunes.apple.com.edgekey.net at Aug19,2016,15:11:18.224902000 CST.

3.2 Generating host-domain graph

This component is responsible for generating the host-domain graph, which describes who queried what. Select domain and source IP address fields from the domain-related records generated in the first component and remove duplicate domain names. A bipartite graph is expressed as G $= (V_1, V_2, E)$, where $V_1 = (v_{11}, v_{12}, .., v_{1n})$, $V_2 = (v_{21}, v_{22}, .., v_{1m})$indicates the node set, E $= (v_{1i}, v_{2j})$, i $= 1, \dots,$ n; j $= 1, \dots,$ m indicates the edge set. Therefore, the host-domain graph use graph G $= (\{I\}, \{D\}, \{E\})$ to represent, where I indicates the set of hosts, D indicates the set of domains, E $= (I_i, D_j)$ indicates the relationship between host and domain. Make use of adjacency matrix to present connection relationship. For example, there is a host-domain relationship is shown in Figure 2. The corresponding host-domain graph is represented as G=({I1,I2,I3,I4,I5,I6,I7},{D1,D2,D3,D4},{(I1,D1),(I2,D1),(I2,D2) ,(I3,D1),(I3,D2),(I4,D2),(I4,D3),(I5,D3),(I5,D4),(I6,D3),(I7,D4) }).

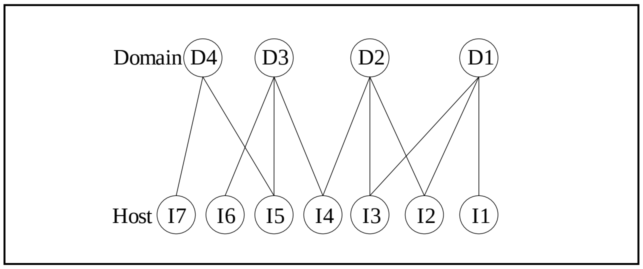

Figure 2: An example of host-domain graph

3.3 Labeling partial nodes

Researcher can label partial domain and host nodes according to blacklist, popular domain list, and Alexa ranking. Since the Alexa value of most benign domains is no more than 10,000. Researcher can label domain whose Alexa value is less than 10,000 as normal. Researcher can also label domain as benign according to the local popular domain list. Moreover, Researcher can set the label of domain or IP as malicious that it appears in the blacklist.

3.4 Semi-supervised co-clustering

First, Researcher split the host-domain graph into different connected components in the first time. Then, generate 2 clusters according to Spectral Clustering. This process is illustrated in section 4.

4 SEMSUPERVISED CO-CLUSTERING

First, extract connected components $G' = \{G_i\}, i = 1, \dots, n$, where $G_i' = (\{D_i'\}, \{I_i'\}, \{E_i'\})$from the host-domain bipartite graph G $= (\{D\}, \{I\}, \{E\})$. Moreover, $\sum_{i=1}^{n}\{D_i'\} = \{D\}$, $\sum_{i=1}^{n}\{I_i'\} = \{I\}$, $\sum_{i=1}^{n}\{E_i'\} = \{E\}$. Then modify the weight matrix using the following rule in equation 1.

$$E_{ij} = \max\left(\frac{E_{ij}}{\sum_i E_{ij}}, \frac{E_{ij}}{\sum_j E_{ij}}\right) \tag{1}$$

Then, generate the matrix A_n through the following equation 2:

$$A_n = D_1^{-1/2} \mathrm{A} D_2^{-1/2} \tag{2}$$

Where, D_1 and D_2 are diagonal matrix, whose diagonal element value indicates the number of domain resolved to the IP, the number of IP resolved by different domain, respectively. The diagonal element value of $D_1^{-1/2}$ equals to the reciprocal of the square root of corresponding value in D_1 .

Then, calculate left singular vectors $U_1 = \{u_1, u_2, \dots, u_l\}$ and right singular vectors $V_1 = \{v_1, v_2, \dots, v_l\}$ through singular value decomposition (SVD). Here, left singular vectors $U_1 = \{u_1, u_2, \dots, u_l\}$ are ranked according to singular-values. Select k number of left singular vectors $U_1 = \{u_1, u_2, \dots, u_l\}$ from 2 to k+1. Similarly and get the same number of right singular vectors $V_1 = \{v_1, v_2, \dots, v_l\}$. The new matrix Z is generated through the following equation:

$$\mathrm{Z} = \begin{bmatrix} \mathrm{D}_1^{-1/2}\mathrm{U} \\ \mathrm{D}_2^{-1/2}\mathrm{V} \end{bmatrix} \tag{3}$$

Finally, cluster domain-IP into two clusters through K-Means algorithm. The top m row represents domains.

Here, the label of some domains and IP are certain. Thus, researcher can label class according to rules illustrated in Section 3. The semi-supervised co-clustering process is shown in Algorithm 1.

ALGORITHM 1 SEMI-SUPERVISED CO-CLUSTERING ALGORITHM

```
Require: $\mathrm{G} = (\{\mathrm{D}\}, \{\mathrm{I}\}, \{\mathrm{E}\})$ : Samples of domain-IP, $S_1 = \bigcup_{i=1}^{k} D_{i,1} + \bigcup_{j=1}^{l} I_{j,1}$: Samples with normal label, $S_2 = \bigcup_{i=1}^{k'} D_{i,2} + \bigcup_{j=1}^{l'} I_{j,2}$: Samples with malicious label, k-the number of clusters.
Ensure: $C_1$: the malware-related domains and IPs, $C_2$: the set of benign domains and IPs.
Extract connect components $G' = \{G_I\}, i = 1, \dots, n$
for each connected component $G'$ do
    Calculate correlation matrix $A_n$
    Generate ranked left and right singular vectors through SVD
    Generate matrix Z
    Calculate cluster center according to labeled samples
    for all $Z_i$ do
        Calculate the distance from different cluster
        if $d_{i,1} > d_{i,2}$ then
            $Z_i$ is in the cluster 1
        else
            $Z_i$ is in the cluster 2
        end if
    end for
end for
```

5 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, evaluate malware detection approach using semi-supervised co-clustering. This experiment results are based on a prototype that deployed on a machine equipped with 3.6 GHz cores, 16 GB of RAM and a 1 TB hard drive.

5.1 Evaluation metrics

In this paper, compared those methods using metrics defined as follows.

Precision = $\mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$

Recall = $\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$

TPR = $\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$

FPR = $\mathrm{FP}/(\mathrm{TN}+\mathrm{FP})$

F1 (F-measure) = 2 $* \mathrm{Precision} * \mathrm{Recall}/(\mathrm{Precision}+\mathrm{Recall})$

True Negative (TN) is the number of domain names that were used in malware is correctly labeled as malicous, False Positive (FP) is the number of domain names that were used in malware is incorrectly classified as belonging the benign, False Negative (FN) is the number of domain names that were not used in malware is falsely classified to be malicious, True Positive (TP) means that the number of normal domain names is precisely classified as benign. F-measure is a comprehensive assessment criteria, it will consider both false negative rate and false positive rate.

In general, the higher the Precision, Recall and F1, the better the recognition effect. On the contrary, the lower the Precision, Recall and F1, the less effective the detection approach.

5.2 Experimental setup

To test this approach, researcher used experimental data consisting of normal domain access records and malware domains access records in this work. This experimental data is generated from Makednslog, which is used to generate DNS log files. DNS access logs with 145 malicious domain names were simulated in the experiment.

5.3 Experiment in the second data

This malware detection approach using semi-supervised co-clustering in a test environment consisting of 1471 normal domain access logs, and 145 malware domain access logs.

There are two ways to process the original data to obtain the weight matrix. The first is based on the number of times the host accesses the domain name, and the other is according to the maximum of A and B, where A equals to the ratio of number of host access domain to total number of times the host accesses different domain names, B equals to the ratio of number of host access domain to total number of times the domain name was visited. Moreover, this experiment adopt three common distance formulas when clustering, like euclidean, chebyshev, and cityblock. The experimental results are shown in Table1.

Table 1: All training and testing biflows used in the experiment

Edge weight	Distance formula	FN	FP	TN	TP	TPR	FPR

edge weight	euclidean	1353	118	28	117	0.98	0.498
edge weight	chebyshev	1328	143	4	141	0.997	0.496
edge weight	cityblock	1353	118	28	117	0.98	0.498
max_min	euclidean	1325	146	4	141	0.997	0.491
max_min	chebyshev	1325	146	4	141	0.997	0.491
max_min	cityblock	1325	146	4	141	0.997	0.491

The proposed co-clustering detection approach using max_min to obtain the weight matrix has higher TPR and lower FPR than that using edge weight to obtain the weight matrix. This indicates that the algorithm with max_min to obtain the weight matrix performs better. The reason is that edge weight from host-domain graph will bring the incomplete information to clustering-based approaches. The TPR and FPR of co-clustering using max_min are 0.997 and 0.491, respectively. The Precision, Recall and F1 of the hybrid malware domain detection method based on semi-supervised co-clustering are shown in Figure 3. Among this results, the recall is significantly higher than the accuracy in all experiments, which reflects that we hope to detect malicious attacks more strictly and reduce the false negative rate as much as possible, so as to achieve a safe environment.

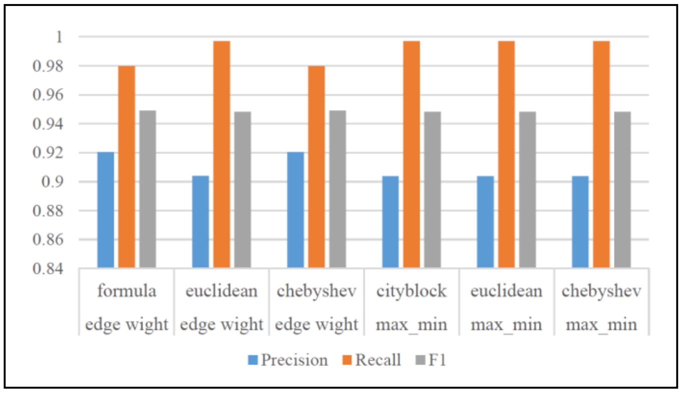

Figure 3: The Precision, Recall, and F1

Experimental results showed that the proposed detection approach was able to identify malware domains with up to 97.2% true positives. As can be seen from Fig. 3, there are gaps in experiment results with different cluster calculating formulas when using edge weights as the bipartite edge weights. However, clustering with maximum and minimum edge weights has a better tolerance to different distance calculation methods.

6 CONCLUSION AND FUTURE WORK

Malware is a long-lived issue in the cybersecurity, especially advanced malware brings new change to analyzers. Mainstream detection methods make use of passive or active DNS data. However, they need training samples that seldom in the monitored network. This work proposes an efficient malware-related domain detection approach combined DNS records with graph analysis. We label potential malicious domains using passive DNS database according to domain list accessed by users in the monitored network. Then, we make use of semi-supervised co-clustering to discover potential malicious domains. The experiments are performed in an off-line network. Experimental results show that our proposed approach has more identification accuracy. Moreover, the approach don’t need training samples.

The future work will be devoted to the co-clustering running in a distributed platform, for example, we can use map-reduce to calculate clustering host-domain graph.

ACKNOWLEDGMENTS

This work is supported in part by National Key R&D Program of China (2017YF-B0802903), National Natural Science Foundation of China (Grant No. 61572115).

REFERENCES

[1] M. Ask , P. Bondarenko, J. E. Rekdal, A. Nordbo, P. Bloemerus, D. Piatkivskyi, Advanced persistent threat (APT) beyond the hype. Project Report in IMT4582 Network Security at Gjovik University College, Springer.

[2] A. K. Sood, R. J. Enbody. Targeted cyberattacks: a superset of advanced persistent threats. IEEE security & privacy 11(1): 54-61.

[3] M. Marchetti, F. Pierazzi, M Colajanni, A. Guido (2016). Analysis of high volumes of network traffic for Advanced Persistent Threat detection. Computer Networks, 109, 127-141.

[4] Gardiner J, Nagaraja S (2016). On the security of machine learning in malware c&c detection: A survey. ACM Computing Surveys (CSUR), 49(3), 59.

[5] S. Xu, S. Li, K. Meng, L. Wu, M. Ding（2017). An Adaptive Malicious Domain Detection Mechanism with dns traffic, in: Proceedings of the 2017 VI International Conference on Network, Communication and Computing, ACM, 86-91.

[6] G. Zhao,K. Xu, L. Xu, B. Wu (2015). Detecting APT malware infections based on malicious DNS and traffic analysis, IEEE Access 3, 1132-1142.

[7] W. Niu, X. Zhang, G. Yang, J. Zhu, Z. Ren (2017). Identifying apt malware domain based on mobile dns logging. Mathematical Problems in Engineering.

[8] X. Du, H.-H. Chen, Security in wireless sensor networks, IEEE Wireless Communications 15 (4).

[9] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, M. Galloway (2007). A survey of key management schemes in wireless sensor networks, Computer communications 30 (11-12), 2314-2341.

[10] Du, M. Guizani, Y. Xiao, H.-H. Chen (2009). A routing-driven elliptic vurve cryptography based key management scheme for heterogeneous sensor networks, IEEE Transactions on Wireless Communications 8(3), 1223-1229.

[11] F. X. Du, Y. Xiao, M. Guizani, H.-H. Chen (2007). An effective key management scheme for heterogeneous sensor networks, Ad Hoc Networks 5(1), 24-34.

[12] J Gardiner, S. Nagaraja (2016). On the security of machine learning in malware c&c detection: A survey, ACM Computing Surveys (CSUR), 49(3), 59.

[13] Neugschwandtner, P. M. Comparetti, C. Platzer, Detecting malware's failover c&c strategies with squeeze (2011). in: Proceedings of the 27th annual computer security applications conference. ACM, 21-30.

[14] K. Xu, P. Butler, S. Saha, D. D. Yao (2013). DNS for massive-scale command and control, IEEE Transactions on Dependable and Secure Computing.

[15] H. Choi, H. Lee (2012). Identifying botnets by capturing group activities in DNS traffic, Computer Networks 56(1), 20-33.

[16] M Thomas, A. Mohaisen (2014). Kindred domains: detecting and clustering botnet domains using dns traffic, in: Proceedings of the 23rd International Conference on World Wide Web. ACM, 707-712.

[17] R. Sharifnya, M. Abadi (2015). Dfbotkiller: domain-flux botnet detection based on the history of group activities and failures in DNS traffic, Digital Investigation, 12, 15-26.

[18] Y. Zhou, Q. Li, Q. Miao, K. Yim (2013). Dga-based botnet detection using dns traffic, J. Internet Serv. Inf. Secur., 3(3/4), 116-123.

[19] . Antomakakis, R. Perdisci, D. Dagon, W. Lee, N. Feamster (2010). Building a dynamic reputation system for dns, in: USENIX security symposium, 273-290.

[20] L. Bilge, E. Kirda, C. Kruegel, M. Balduzzi (2011). Exposure: Finding malicious domains using passive dns analysis, in: Ndss.

[21] N. Jiang, J. Cao, Y. Jin, L. E. Li, Z.-L (2010). Identifying suspicious activities through dns failure graph analysis, in: Network Protocols (ICNP), 2010 18th IEEE International Conference on, IEEE, 144-153.

[22] F. Zou, S. Zhang, W. Rao, P. Yi (2015). Detecting malware based on dns graph mining, International Journal of Distributed Sensor Networks 11(10), 102687.

[23] A. Berger, A. DAlconzo, W. N. Gansterer, A. Pescape (2016). Mining agile dns traffic using graph analysis for cybercrime detection, Computer Networks, 100, 28-44.

[24] P. Camelo, J. Moura, L. Krippahl. Condenser: A graph-based approachfor detecting botnets, arXiv preprint arXiv:1410. 8747

[25] P. K. Manadhata, S. Yadav, P. Rao, W. Horne (2014). Detecting malicious domains via graph inference, in: European Symposium on Research in Computer Security. Springer, 1-18.

[26] J. Lee, H. Lee (2014). Gmad: Graph-based Malware Activity Detection by DNS traffic analysis, Computer Communications, 49, 33-47.

[27] I. Khalil, T. Yu, B. Guan (2016). Discovering malicious domains through passive DNS data graph analysis, in: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security. ACM, 663-674.

[28] B. Rahbarinia, R. Perdisci, M. Antonakakis (2015). Segugio: Efficient behavior-based tracking of malware-control domains in large ISP networks, in: Dependable Systems and Networks (DSN), 2015 45th Annual IEEE/IFIP International Conference on. IEEE, 403-414.

[29] L. Shi, D. Lin, C. V. Fang, Y. Zhai (2015). A hybrid learning from multi-behavior for malicious domain detection on enterprise network, in: Data Mining Workshop (ICDMW), 2015 IEEE International Conference on. IEEE, 987-996.