ISSN 2278-3091

Volume 9, No.5, September - October 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse229952020.pdf

https://doi.org/10.30534/ijatcse/2020/229952020

Comparative Analysis of Network flow-based Botnet Detection Methods Using Supervised Machine Learning Algorithms

FarhanTariq[1]
Electrical and Computer Engineering
Center for Advanced Studies in Engineering
Islamabad, Pakistan
farhantariq2@gmail.com

Shamim Baig[2]
Head of Department of Computer Science
Muslim Youth University
Islamabad, Pakistan
msbaig@myu.edu.pk

ABSTRACT

The botnet emerges as a top-listed threat to interconnected computer network systems. The increasing number of botnet attacks and rapidly changing evasion techniques demanding more generic long term and resilient botnet detection systems. The signature-based approaches naturally not able to cope up with this rapidly changing footprint. To automate the behavioral-based approaches, researchers start applying machine learning algorithms. These approaches either support online detection mode or offline detection mode. The detection proposals that apply supervised machine learning algorithms show promising results, where the C4.5 algorithm stands on top. In this paper, we present a comparative study of online botnet detection methods that apply C4.5 supervised machine learning algorithms. We have conducted a simulation study to evaluate the performance of two top listed detection methods from traditional IP networks and three of our botnet detection methods from Software Defined Networks. The evaluation is performed using CTU-13 publicly available real botnet dataset. The results show that the detection methods that are designed using a more diverse dataset perform better when a new variant of botnets introduced in the test dataset and has more detection coverage area. The results also conclude that an ensemble of multiple type-specific botnet classifiers not only help to detect botnet type but also perform better than one single generic classifier.

Key words :botnet, detection, comparative analysis, malware, machine learning, NBA, SDN, TSDR, OpenFlow, Opendaylight, flows

I. INTRODUCTION

The use of botnet as an underline tool for malicious activities has increased during the last decade and botnet technology emerges as a top-listed threat to interconnected computer systems. During this time, the researchers also have been put significant research efforts into the advancement of techniques that could give proficient and successful botnet locations. Subsequently, a variety of recognition techniques dependent on different specialized standards and focusing on different parts of botnet life-cycle have been characterized. The powerful feature of the botnet that distinguishes it from other malware is its remote controllability and for this botnet required to do command and control (C2C) communication. This C2C communication introducesthe dependability of the botnet on the internet for communicating back to its master and leaves a network footprint that helps researchers to identify botnet patterns after network traffic analysis. The deterministic steps of network behavior analysis introduced the use of machine learning algorithms in network-based botnet detection methods. The machine learning algorithms are programs that are designed in a way that helps the computer system to learn by studying the data pattern and make a prediction on new data. Machine learning has two main categorizations namely supervised learning and unsupervised learning. Supervised learning has two phases namely learning and testing. During the learning phase, the supervised learning algorithms are provided with a pre-labeled dataset. These labels help underline algorithm to establish a criterion for specific labels. Then for testing the trained model, the dataset is provided without label and the trained model based on its learning predicts labels of provided data sets. The unsupervised learning machine learning algorithms cluster together provided dataset based on different similarities in data. There is no learning phase and no labeled data is provided in the unsupervised learning approach.

The researchers deploy both supervised and unsupervised machine learning algorithms in their botnet detection proposals but mostly focused on supervised machine learning due to its promising results. The botnet detection proposals that deploy supervised machine learning algorithms either support online or off-line detection mode. The online detection methods analyze network flows either for a limited time or small batches of flows whereas offline detection methods are designed to analyze all traffic once at a time. The scope of this paper is online network flow-based botnet detection methods that deploy supervised machine learning algorithms. The five botnet detection methods that support online detection and apply decision tree-based C4.5 supervised machine learning algorithms used in this comparative study. Two of these methods proposed in [1,2] are from the traditional IP network and the other three methods [3,4,5] from SDNs. This comparative study is simulation-based and uses real-world publicly available botnet dataset for training and testing of all four methods. In this study, we contributed with the following two conclusions

The supervised machine learning botnet detection methods that are designed by introducing more diversity in their datasets have the natural ability to detect new botnets.

The multiple botnet type-specific classifiers perform better that one single generic classifier.

The rest of the paper is as follows: Section 2 discussed the background and related work. The detail of botnet detection methods selected for comparative study provided in section 3. The comparative analysis discussed in section 4. Section 5 finally concludes the paper.

II. Related Work

There are several proposals for malware detection and most of these proposals from the last decade apply machine learning in their methods [12][16][18]. The supervised machine learning approaches from these proposals show promising results as compare to anomaly-based approaches. The work of Nguyen Vuong Tuan Hiep et al., [17] proposed a network based botnet detection approach using supervised machine learning algorithms. The work proposed in [10] survey 20 previous detection methods that apply machine learning. The review detection proposals from the aspect of information level and distinguish the proposals into either host-based, network-based, or hybrid. They also analyze if detection proposals offer online detection and weather these proposals are signature independent or not. The work proposed in [11] surveys 14 network-based botnet detection techniques. And identified issues related to reproducibility due to scripted or private dataset selection, lacking documentation of detection method. These issues discourage the comparison of new proposals with previously proposed methods. The work also identifies that only one method provides third-party comparisons out of fourteen analyzed botnet detection approaches. The work proposed in [1] specifically highlighted these issues and the cause of the same. This work also provides a publicly available real-world botnet traffic dataset to address dataset selection challenges that end up with detection methods results that are not reproducible. These researchers also compare two of their methods with third-party detection method in this work and proposed and evaluation methodology.

III. Botnet Detection Methods

This section discussed in detail five network flow-based botnet detection methods that apply the C4.5 decision tree supervised machine learning algorithm. Two of these algorithms are from a traditional IP network and uses NetFlow protocol to collect network session-level traffic. The other three are designed for SDNs and uses OpenFlow protocol to collect network session-level traffic centrally from the SDNs controller.

A. Traditional IP Network

There are several proposals to detect botnet in traditional IP networks using network-level information. Most of these proposals use only network session-level information to address privacy and encrypted command and control sessions. The below sections provide the detail of two top listed flow-based detection methods that apply C4.5 decision tree-based supervised machine learning algorithms. These third-party methods used in the comparative study of this work.

1) Zhao et al:

Zhao et al. [1] Proposed a botnet detection method based on flow intervals. The work uses a decision tree based Reptree algorithm in their approach and evaluate it against four different time intervals and shows that interval of 180s is best for both detection accuracy and detection method performance. The work shows a high detection accuracy of 99 % for a time interval of 300 seconds. This work uses a custom dataset with only two botnet traffic traces.

2) Biglar et al

The work proposes by Biglar in [2] proposed a decision tree based supervised machine learning approach to detect botnet. This proposal studies the effectiveness of network flow-based features for botnet detection. The work started with the most used network features from previous proposals and repeated lyremove features from the pool based on experiments result. This work uses a publicly available real-world botnet traffic traces for experimentation and also include 50% diversity in the test dataset. This work shows a detection rate of 75% against a highly diverse dataset.

B. Software Defined Networks

The brought together centralized visibility and dynamic programmability of SDNs not only give new hope to researchers but also opening new challenges due to quickly developing SDNs impression in the market. The researchers start proposing solutions to address the botnet problem for emerging network technology platform the SDNs. The three of our OpenFlow based botnet detection methods that similarly apply the C4.5 supervised machine learning algorithm are selected for the comparative study of this work. Two of these methods apply a one-class approach and output either the detected flow is botnet or not botnet. The third method appliesa multiclass approach and also detect botnet type with each detected infection. These methods called as method1, method 2, and method 3 in the rest of the paper.

1) Detection method1

The Farhan et al. Method 1 [4] proposed to detect botnet in SDNs using OpenFlow protocol. This work is based on network flow statistics and applies C4.5 decision tree-based supervised MLA to automatically identify botnet behavior characteristics from network session traffic. This work shows that OpenFlow protocol can also be used to detect botnets. This simulation-based work uses the OpenDayLight SDNs controller and collects network flow statistics centrally from the controller. The work uses passive flow collection and commands the controller to send statistics on each flow removed event by setting OFPT_FLOW_REMOVED flag. This detection method achieved 78% detection accuracy with a precision of 86%.

2) Detection Method2

The detection method 2 [5] extends the work proposed in method 1 proposes to use both the last 60 min network flow data with real-time network flows. This work uses flow trace concepts and first try to reassembled flow traces from incoming flow statistics. The flow trace is an order sequence of flows of the same application between two network endpoints. Once a flow trace with 10 or more flows captured during the flow trace reassembling process, the batch of flows of detected flow trace for a unit time forwarded for feature computation process. The feature computation extract feature from this batch flows and fetch the last 60-min flows for the source and destination of processed flow trace. This new batch of flows helps to compute further features from the historical network activity of the source and destination IP of the flow trace. The work uses the Time Series Data repository (TSDR) plugin to fetch OpenFlow statistics. The work shows promising results with a detection accuracy of 94.5 % and a precision of 90 %.

3) Detection Method3

The third detection method proposed by Farhan [3] appliesa multiclass detection approach. This proposal extends the work proposed in method 2 from a single one-class classifier approach to multiple one-class classifiers that are tuned to detect specific types of botnets. The method works the same as method 2 for flow collection and feature computation. The classification process including training and testing is adopted for multiple one-class classifiers. This work applies One-versus-all multiclass decomposition approach and trains three one-class classifiers. The output of all these three classifiers sends to the final decision process where the output with the highest detection confidence is selected as the final decision. This work shows higher detection accuracy and precision of 96.5 % and 97.8 % respectively.

C. Datasets

The choice of the dataset for experimentation and assessment of supervised machine learning algorithms is important.The poor choice of datasets prompts one-sided results and there is a high probability that the same model performs poorly against any new dataset. To keep away from biasedness the dataset forevaluation of supervised machine learning algorithms must have diversity, real-word botnet, and normal traffic traces [2]. To fulfill these conditions, this work selects genuine traffic follows for both botnet and background traffic traces from publicly available sources [1]. The selected botnet dataset comprises of 3 different botnets from each class of IRC, HTTP, and P2P and traces of normal / background traffic.

1) Training Dataset

The training dataset used to train the selected detection methods. To introduce diversity during testing of the detection methods, one botnet from each type namely Murlo from IRC, Soguo from HTTP, and Sality from P2P completely excluded during training dataset generation process. The 60% of the traffic traces as shown in Table used to train the detection methods and the remaining is used during validation and testing.

2) Test Dataset

The test dataset generated in two chunks. The first chunk

Botnet	**Type**
Rbot	IRC
Murlo	IRC
Neris	IRC
Virut	HTTP
Soguo	HTTP
Zeus	HTTP
Weladec	P2P
Sality	P2P
ZeroAccess	P2P
Normal	Background

Table 1: Dataset

Botnet	**% flows**
Rbot	9.7%
Neris	9.9%
Virut	9.1%
Zeus	8.4%
Weladec	8.9%
ZeroAccess	9.2%
Normal	44.8%

Table 2: Training dataset

named as alpha α is the remaining 20 % of traffic traces from Table2. The second chunk named beta β is generated from the

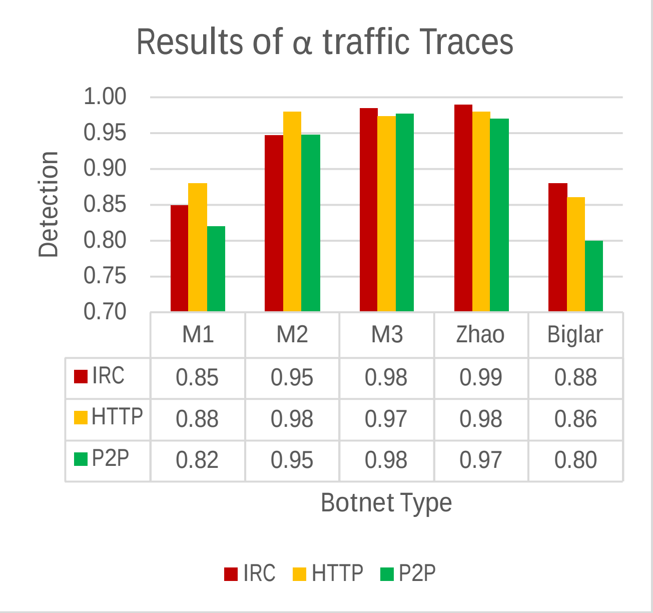
Table 3: Test dataset

Botnet	chunk	% flows
Rbot	α	6.7%
Noris	α	6.9%
Virut	α	8.1%
Zeus	α	7.4%
Weladec	α	7.9%
ZeroAccess	α	6.2%
Murlo	β	6.7 %
Soguo	β	5.9 %
Sality	β	7.1 %
Normal	Background	37.1%

botnets that were not used during the training phase and used to test the novelty detection capability of each detection method. Table3 shows the distribution of traffic traces of the test dataset.

IV. Comparative Evaluation

This section presents the results of all five detection methods and provides comparative investigation. The trial strategy applied in this work attempts to address recognized issues of reproducibility, diversity, and generality as identified in [1] [2]. The experimental evaluation in this study uses a training dataset as shown in the table to train all the methods and then replay the testing dataset as shown in the table to test all these methods. The M1, M2, and M3 denoted in the table for method1, method2, and method3 from SDNs respectively. The results of detection methods parted into two sections. The first section summarized the results against α traffic traces. The 80% of traffic traces of samebotnet used during the training and validation process and the remaining 20% are the α traffic traces. The second section summarized the testing results against β traffic traces. The β traffic traces are generated using Murlo from IRC, Soguo from HTTP, and Sality from P2P botnets. These three botnets intentionally not included in training dataset to introduce diversity and test novelty detection capability of detection methods.

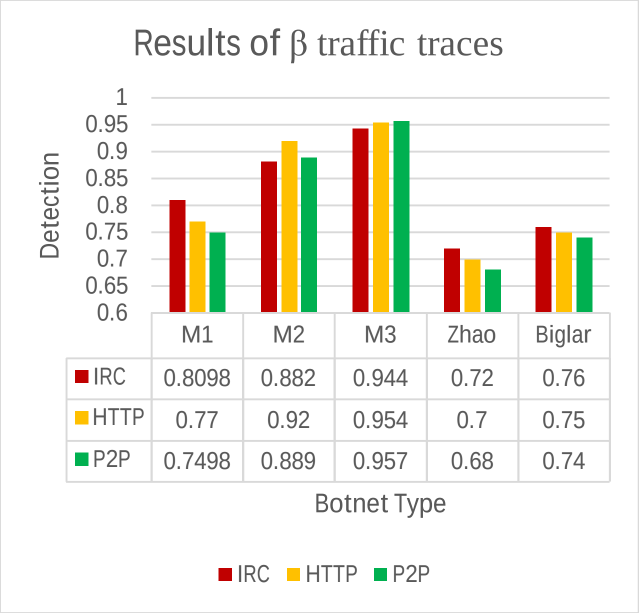


	M1	M2	M3	Zhao	Biglar
🟥 IRC	0.85	0.95	0.98	0.99	0.88
🟨 HTTP	0.88	0.98	0.97	0.98	0.86
🟩 P2P	0.82	0.95	0.98	0.97	0.80

Figure 1: Results against α traffic traces

A. *α Traffic Traces*

The test experiments result against α traffic traces are presented in section 1 of table. These results show that Zhao detection method performs best against IRC botnet traffic traces with detection of 99 %. The both methods M2 and Zhao performs best against HTTP botnet traffic traces with detection of 98%. The detection method M3 performs best against P2P botnet traffic traces with detection rate of 98%. The both M3 and Zhao achieve average detection of 98% against all three botnet types and stands at first position. The



	M1	M2	M3	Zhao	Biglar
🟥 IRC	0.8098	0.882	0.944	0.72	0.76
🟨 HTTP	0.77	0.92	0.954	0.7	0.75
🟩 P2P	0.7498	0.889	0.957	0.68	0.74

Figure 2: Results against β traffic traces

method M2 shows average detection of 96 % and took 2nd position whereas both method M1 and Biglar achieve average detection of 85% and stands at last position. The figure presents the results of all detection methods against α test dataset.

B. β traffic traces

The test experiments result against β traffic traces presented in section 2 of the table. The β traffic traces are generated from botnets that are completely new to all detection methods. The Zhao method not performs well against β traffic traces and falls to the lowest position. Its average detection falls 25% as compared to results against α traffic traces. The detection method M3 and M2 retain their 1st and 2nd position respectively. M3 average detection fall by 3% from 98% to 95%, and M2 average detection fall by 7% from 96% to 89%. The method M1 retain its third position with average detection fall by 7.5% from 85% to 77.6%, whereas Biglar method took 4th position with average detection fall by 10 % from 85% to 75 %. The result shows that Zhao is the most vulnerable method against diversity whereas M3 stands strongest by lowest average detection fall against β traffic traces. Figure (a) shows the result of the detection method against β traffic traces.

V. Concluding Remarks

This goal of this study is to evaluate network-based botnet detection methods. The current trends in network-based botnet detection proposals focusing only on network session-level traffic and using machine learning algorithms to automatically detect botnet behavior patterns. Where supervised machine learning proposals show more promising results as compare to the unsupervised machine learning algorithm. The proposals from literature also evaluate different supervised machine learning algorithms and show that decision tree-based algorithms stand on the top [6]. Hence this comparative study evaluates five network flow-based botnet detection proposals that apply decision tree-based supervised machine learning algorithms. All these five methods train and test against publicly available real-world botnet traffic traces. The results conclude that the detection methods that are designed without introducing enough diversity in their datasets may only focus on the dominant characteristics of botnet use to train the method. Such a method performs well against test data from the same botnets but may perform poorly against a new type of botnets. The results also conclude that multiple type-specific classifiers not only help to detect botnet type but also performs better with more coverage area than one binary classifier.

REFERENCES

[1] Garcia, S., Grill, M., Stiborek, J., & Zunino, A. (2014). An empirical comparison of botnet detection methods. computers & security, 45, 100-123.

[2] Beigi, E. B., Jazi, H. H., Stakhanova, N., & Ghorbani, A. A. (2014, Oct 29-31, 2014). Towards effective feature selection in machine learning-based botnet detection approaches. Paper presented at the 2014 IEEE Conference on Communications and Network Security, San Francisco, CA, USA.

[3] Tariq, F., & Baig, S. (2019, March, 2019). Multiclass Machine Learning Based Botnet Detection in Software Defined Networks. IJCSNS, 19(3), 150-156.

[4] Tariq, F., & Baig, S. (2016, Aug, 2016). Botnet classification using centralized collection of network flow counters in software defined networks. International Journal of Computer Science and Information Security, 14(8), 1075-1080.

[5] Tariq, F., & Baig, S. (2017, Dec, 2017). Machine learning based botnet detection in software defined networks. International Journal of Security and Its Applications, 11(11), 1-11. doi:10.14257/ijsia.2017.11.11.01

[6] Stevanovic, M., & Pedersen, J. M. (2014, Feb 3-6, 2014). An efficient flow-based botnet detection using supervised machine learning. Paper presented at the 2014 international conference on computing, networking and communications (ICNC), Honolulu, HI, USA.

[7] Stevanovic, M., & Pedersen, J. M. (2015, June 8-9, 2015). An analysis of network traffic classification for botnet detection. Paper presented at the 2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), London, UK

[8] Panimalar, P., Rameshkumar, K. A Novel Traffic Analysis Model for Botnet Discovery in Dynamic Network. Arab J Sci Eng 44, 3033–3042 (2019). https://doi.org/10.1007/s13369-018-3319-7.

[9] Wu, W., Alvarez, J., Liu, C., & Sun, H. M. (2018). Bot detection using unsupervised machine learning. Microsystem Technologies, 24(1), 209-217.

[10] Stevanovic, M., & Pedersen, J. M. (2016). On the use of machine learning for identifying botnet network traffic. Journal of Cyber Security and Mobility, 4(2), 1-32.

[11] Garcia, S., Zunino, A., & Campo, M. (2014). Survey on network based botnet detection methods. Security and Communication Networks, 7(5), 878-903. doi:10.1002/sec.800.

[12] Lashkari, A. H., Gil, G. D., Keenan, J. E., Mbah, K. F., & Ghorbani, A. A. (2017, November). A survey leading to a new evaluation framework for network-based botnet detection. In Proceedings of the 2017 the 7th International Conference on Communication and Network Security (pp. 59-66).

[13] Letteri, I., Della Penna, G., & De Gasperis, G. (2018, October 29-31, 2018). Botnet detection in software defined networks by deep learning techniques. Paper presented at the International Symposium on Cyberspace Safety and Security, Amalfi, Italy.

[14] Latah, M., & Toker, L. (2018). Artificial intelligence enabled software-defined networking: a comprehensive overview. IET Networks, 8(2), 79-99.

[15] Letteri, I., Del Rosso, M., Caianiello, P., & Cassioli, D. (2018, February). Performance of Botnet Detection by Neural Networks in Software-Defined Networks. In ITASEC.

[16] Amer, Ahmed and Abdul Aziz, Normaziah (2019) Malware detection through machine learning techniques. International Journal of Advanced Trends in Computer Science and Engineering, 8 (5). pp. 2408-2413. ISSN 2278-3091

[17] Nguyen, Tisenko, Do Minh, Lam,N.A. Tuan (May-June 2020), Detecting Botnet based on Network Traffic. International Journal of Advanced Trends in Computer Science and Engineering. 9 (3)

[18]Manzoor, S. I., & Singla, J. (2019). A comparative analysis of machine learning techniques for spam detection. International Journal of Advanced Trends in Computer Science and Engineering, 8(3), 810-814.