Detecting Bots inside a Host using Network Behavior Analysis

Seshadri Rao Chinta
Anil Neerukonda Institute of Technology and Sciences
Sangivalasa, Visakhapatnam
India

Vinod Babu Polinati
Anil Neerukonda Institute of Technology and Sciences
Sangivalasa, Visakhapatnam
India

P. N. Srinivas
Anil Neerukonda Institute of Technology and Sciences
Sangivalasa, Visakhapatnam
India

ABSTRACT

Being well aware of the drastic changes brought by the Internet to the world there exists an explosion of network traffic. This burst traffic brings in lots of unwanted communication as a side-effect from the infected machines also called victims. Bots are such type of infected machines which work under a super power called botmaster. A botnet is a collection of compromised machines or bots receiving and responding to commands from the Command and Control (C&C) server that serves as a rendezvous mechanism for commands from a human or controller i.e., the bot master. The aim of our work is to detect the presence of the bot in the network traffic. This is accomplished in a two-step process. The work first captures network traffic from the infected host, and second step analyzes the captured traffic and detects the presence of a bot. To meet the goal we experimented on CTU-13 data set, a data set of botnet traffic captured in the CTU University, Czech Republic. Our work uses decision trees, Naïve Bayes, SVM and K Nearest Neighbor to detect the presence of bot. We found that decision trees gives 99.9% positive detection rate compared to other algorithms.

Keywords

Bots, SVM, KNN, Decision tree, bot detection

1. INTRODUCTION

Botnet are the primary means to cyber-criminals to carry out their malicious tasks such as DDOS attacks, cyber fraud, click fraud, sending spam mails, stealing personal data etc. The bot is a compromised machine or victim which operates under the control of the botmaster via C & C server. C&C is a Command and Control server that sends instructions to the machine to the bots on behalf of the botmaster. The survey of taxonomy of the bots and their defenses is widely studied in [1][2][3][4][5][6]. This study of bots prevails to only few subsets of classes and doesn't cover the entire population classes of bots.

Botmaster is a person who creates the bots and infects machines in the internet through propagation via C & C server. The C & C Server can be a proxy which does not reveal the information of the botmaster and act as stepping stones for the botmaster. The propagation happens as a rallying mechanism using IRC, HTTP, P2P communication, etc. The bots residing in the infected machine scan the network to look which other machines in the network can be infected, infects the machine and sends the information of the infected machine to the other peers or C&C server and waits for commands from C&C. Agobot, Spybot, Sdbot are few IRC bots[7], Phatbot [8], Storm [9] and Nugache[10] are example s of p2p bots. The botmasters use bots for information gathering, spreading malware, to do distributed denial-of-service attack (DDOS), cyber fraud. To evade detection bots use different topologies like centralized, star, hierarchical, distributed, p2p and always presents a challenge to detect them in a novel way.

2. RELATED WORK

Several authors contributed their work to detect the presence of bot. New mechanisms are continuously adapted by the botmasters to evade the detection, this presents a challenge to detect the bot in a novel way.

Lu et al.[11] classified the bots using payload signatures and their results show 40% of the traffic goes undetected. BotHunter is an intrusion detection system developed by Gu et al. [12] work on the snort rules and fires an alarm as bot activities are detected. The BotHunter scans the network, captures the payload and does analysis on payload to detect common malware intrusions by correlating the payload traffic with Snort rules and triggers an alarm for any anomaly behavior detected. To evade detections botmasters use encrypted traffic so that the BotHunter cannot detect the bot. Analyzing the payload is too costly as payloads are heavier and contradicts the principle of user privacy.

BotMiner developed by Gu et al. [13] clusters the traffic using two planes C-plane and A-plane. C-plane logs traffic flows and A-plane detects suspicious activities. The clustering information form C-plane and A-plane are correlated using cross-pane correlation to detect the bot infected machine.

G. Miinz et al., [14] used flow-based analysis to analyze the traffic flow to detect bots. The presence of botnet traffic in the traffic flow causes changes in the volume of traffic that differentiates the infected machines from those of legitimate hosts.

Yukiko Sawaya et al., [15] used traffic flow statistics obtained by NetFlow, sFlow to study the characteristics of attackers sending traffic flow to object ports and closed ports without deep packet inspection. They calculated the flow statistics of the obvious attacker targeting a specific port and identify the nuisance attackers based on the similarity of features between hosts sending flow to port P and the samples.

K Shanthi et al., [16] proposed a novel method of classify bots from normal hosts through traffic flow analysis based on time intervals. The authors did not include payload inspection. The network traffic is captured, filtered by removing all IP addresses that are not botnets and C&C servers. After the filtering process, attributes are selected to classify the bots. Naïve Bayes, J48 decision tree are used to classify the bots using the attributes selected for classification. The true positive rate is 78.5% and 86.6% respectively.

Fransisco et al., [17] designed a novel method to detect bots using the features at their Command & Control(C &C) phase. The aim to find feature set based on the connections of botnets at their C&C phase. Genetic algorithm is used to find the features set and C4.5 algorithm to do classification between connections belonging and not to botnet.

3. PROPOSED WORK

We are proposing a novel way to detect to detect the bot inside a host by observing the network behavior of the host i.e., the traffic flow to the host and to other networks and vice-versa. The network traffic is captured using Wireshark tool. It is an open source tool widely used to capture the traffic. To start with, we used CTU-13 dataset for our experiment, a data set of botnet traffic captured in the CTU University, Czech Republic. The experiment is also carried out using Wireshark tool. Our work comprises of five steps.

Step 1: The traffic is captured using Wireshark tool and stored in a file.

Step 2: After traffic is captured, filtering phase begins. In this step attributes like source IP, destination IP, etc., are selected from the captured file and few attributes are derived like number of packets flow, frequency of flow, etc., from the existing attributes. The list of captured attributes are given below

- Source IP address
- Destination IP address
- Protocol
- Timeline
- Length
- Information about the packet

The list of derived attributes from the above attributes is given in Table 1.

Table 1: Derived attributes from the captured file.

BytesSD	Bytes transferred from source to destination
BytesDS	Bytes received from destination to source
NumP	Number of packets flow from Source to destination
FreqP	Frequency of the packet flow
Duration	Duration of the flow
AvgPSize	Average packet size
IntPSent	Time interval between packets sent
IntPReceived	Time intervals between packets received

Step 3: We implemented our work on CTU-13 dataset, a data set of botnet traffic captured in the CTU University, Czech Republic. We applied machine learning algorithms like-Decision Tree, KNN, Naïve Bayes and SVM for analysis the network traffic using the attributes in step2 to detect the presence of bot.

The architecture of the proposed work is shown in Figure 1.

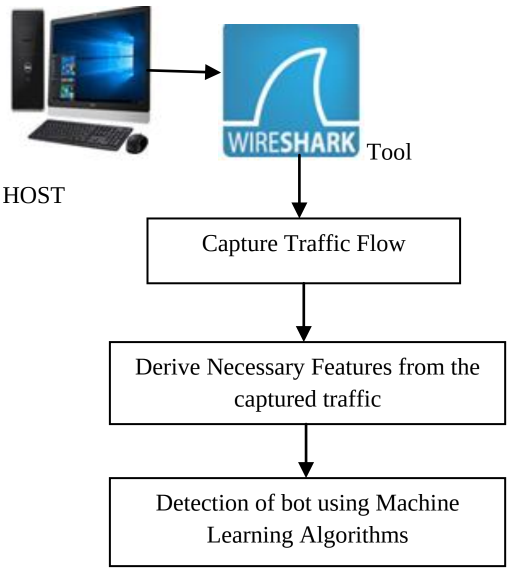
Tool
HOST
WIRESHARK
Capture Traffic Flow
Derive Necessary Features from the captured traffic
Detection of bot using Machine Learning Algorithms

Figure 1: Architecture diagram of the proposed work.

In the first step, the traffic flow is captured using the Wireshark tool. In the second step, the captured traffic is tabularized into a worksheet for analysis. The above mentioned attributes in step 2 are calculated from the captured traffic. In the third step, we first applied Naïve Bayes algorithm to detect the presence of bot in the traffic. To do this we considered the CTU-13 dataset which contains Neris botnet traffic and then performed the experiment. We have calculated the mentioned attributes for the dataset. The detection rate using Decision Tree is 99.90%

The table below shows the count of true positives and false positives for each algorithm applied on the CTU-13 dataset for the detection of Neris Bot. The true positive count is significant for Decision tree algorithm, KNN and SVM being the next having good detection rate.

Algorithm	**True Positives + True Negatives (TP + TN)**	**False Positives + False Negatives (FP+FN)**
Naïve Bayes	7226	2276
Decision Trees	9992	10
KNN	9821	181
SVM	9806	196

Figure 2: Number of labeled records detected (CTU-13 Data Set- Neris Bot).

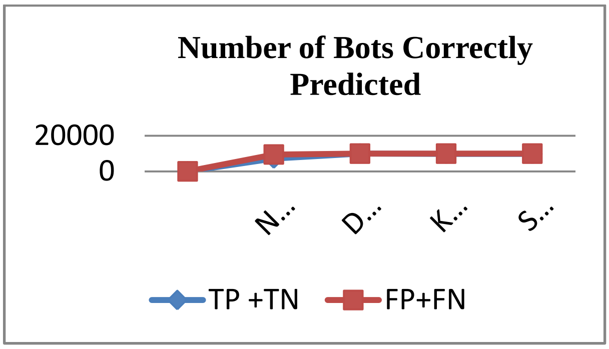

Figure 3: Number of Neris-bot Correctly predicted.

We also conducted the experiment and analyzed the true positives and false positives for other machine algorithms like Decision trees, K-Nearest Neighbor and Support Vector Machine. As seen from the above table, the detection of bots is more accurate using decision trees than Naïve Bayes, KNN, SVM.

4. RESULTS

Our work show that the bots are detected from analyzing the traffic flow by selecting the necessary attributes that contributes to the classification of bots like duration of flow, bytes per flow on each side, frequency of the flow, average packet size, etc. Our results show that decision trees gives better result for labeled data set with detection rate accuracy of 99.90%. The detection rate for each algorithm is shown in Figure 4.

Algorithm	Detection Rate
Naïve Bayes	72.25%
Decision trees	99.90%
K- Nearest Neighbor (KNN)	98.19%
Support Vector Machine (SVM)	98.04%

Figure 4: Detection Accuracy Rate of Neris Bot in CTU-13 Dataset.

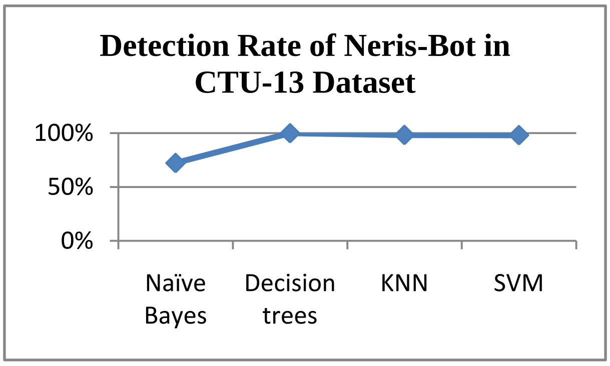

Figure 5: Detection rate of Neris-Bot in CTU-13 Dataset using different machine learning algorithms.

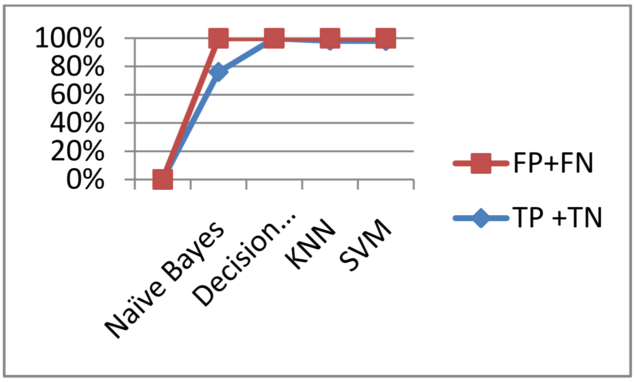

Figure 6: The above graph shows the Percentage of correctly predicted Neris bot in CTU-13 Dataset.

5. CONCLUSION

This work presents the novel approach to detect and classify the bot from the captured traffic or using any captured flow data sets. The bots are detected with high accuracy on CTU-13 data set. Bots pose real threat to the Internet. Traffic flows in the Internet carry lots of unwanted traffic in terms of malware, spam and infect the vulnerable host. There is a need to detect such traffic in novel ways as to protect the host or network from being attacked or being made victim of cyber fraud, click fraud or DDoS attacks.

Our work presents the novel way of detecting the bots with high accuracy by only choosing the attributes from the network flow. However, many botmasters use variety of techniques like encrypting the payload, use HTTP, IRC and P2P protocols to evade detection. Our work can be extended to detect bots using these protocols that are evading detection in addition to our traffic flow characteristics.

6. REFERENCES

[1] Dagon, 2005. Botnet Detection and Response – The network is the infection. OARCW Workshop.

[2] T. Micro, 2006. Taxonomy of Botnet Threats. White Paper.http://www.cs.ucsb.edu/~kemm/courses/cs595G/TM06.pdf

[3] D. Dragon, G. Gu, C.P. Lee and W. Lee, 2007.A Taxonomy of Botnet Structures, ACSAC.

[4] Jose Nazario, 2008, Bot and Botnet Taxonomy. https://www.slideshare.net/digitallibrary/bot-and-botnet-taxonomy.

[5] G. Ollman, Botnet Communication Topologies, 2009. http://technicalinfo.net/papers/PDF/WP_Botnet_Communications_Primer_(2009-06-04).pdf

[6] D. Plohmann, E. Gerhards-Padilla, and F. Leder, Botnets: Detection, 2011. Measurement, Disinfection and Defense, European Network and Information, Security Agency, Tech. rep., 2011.

[7] P. Barford and V. Yegneswaran, 2007. An Inside Look at Botnets, in Malware Detection, ser. Advances in Information Security. Springer US, 2007, vol. 27, ch 8, pp.171-191.

[8] J. Stewart, Phatbot Trojan Analysis, 2004. Retrieved from SecureWorks:http://www.secureworks.com/research/threats/phatbot, 2004.

[9] T. Holz, M. Steiner, F. Dahl, E. Biersack and F. Freilling, 2008. Measurements and Mitigation of peer-to-peer

based botnets-A Case Study on Storm Worm, in Proceedings 1st Usenix Workshop on Large-Scale Exploits and Emergent threats(Leet, Berkely, CA, USA, 2008).

[10] S. Stover, D. Dittrich, J. Hernandez and S. Dietrich, 2007. Analysis of the Storm and Nugache Trojans: P@P is here, in USENIX; login, vol. 32, no. 6, 2007.

[11] W. Lu, M. Tavallaee and A.A. Ghorbani, 2009. Automatic Discovery of botnet communities on large-scale communication Networks in Proc. 4th International Symposium on Information, Computer and Communications Security, ser. ASIACCS'09. New York, USA: ACM, 2009, pp. 1-10.

[12] GuofeiGu, Phillip Porras, Vinod Yegneswaran, Martin Fong and Wenke Lee, 2007. BotHunter: Detecting Malware Infection through IDS-driven dialog correlation, Proc. 16th USENIX security Symposium, pp. 167-182, 2007.

[13] GuofeiGu, Roberto Perdisci, Junjie Zhang and Wenke Lee, 2008. BotMiner: Clustering Analysis of Network traffic for protocol and Structure-independent Botnet Detection, Proc. 17th USENIX security Symposium, pp. 139-154, 2008.

[14] G. Miinz, G. Carle, 2007. Real-time Analysis Flow data for Network Attack Detection, in Proc. of 10th IFIP/ IEEE International Symposium on Integrated Network Management, pp. 100-108, 2007.

[15] Yukiko Sawaya, Ayumu Kubota, Yutaka Miyake, 2011. Detection of Attackers in Servers using Anomalous Host Behavior Based on Traffic Flow Statistics, PSJ International Symposium on Applications and the Internet, pp. 353-359, 2011.

[16] K. Shanthi, D. Sreenivasan, 2015. Detection of Botnet by Analyzing Network Traffic Flow Characteristics using Open Source tool, IEEE Sponsored 9th International Conference on Intelligent Systems and Control (ISCO) 2015.

[17] Francisco Villegas Alejandre, Nareli Cruz Cort'es and Eleazar Aguirre Anaya, 2017. Feature Selection to Detect Botnets using Machine Learning Algorithms, IEEE, and International Conference on Electronics, Communications and Computers (CONIELECOMP) 2017.