# The Role of Machine Learning in Botnet Detection

**Conference Paper** · December 2016

DOI: 10.1109/ICITST.2016.7856730



**2 authors:**

Sean Miller
The University of the West Indies at Mona
**4** PUBLICATIONS **32** CITATIONS
SEE PROFILE

Curtis C.R. Busby-Earle
The University of the West Indies at Mona
**16** PUBLICATIONS **45** CITATIONS
SEE PROFILE

# The Role of Machine Learning in Botnet Detection

Sean Miller
Department of Computing
The University of the West Indies Mona
Kingston Jamaica
sean.miller@mymona.uwi.edu

Curtis Busby-Earle
Department of Computing
The University of the West Indies Mona
Kingston Jamaica
curtis.busbyearle@uwimona.edu.jm

***Abstract*—Over the past ten to fifteen years botnets have gained the attention of researchers worldwide. A great deal of effort has been given to developing systems that would efficiently and effectively detect the presence of a botnet. This unique problem saw researchers applying machine learning (ML) to solve this problem. In this paper we provide a brief overview the different machine learning (ML) methods and the part they play in botnet detection. The main aim of this paper is to clearly define the role different ML methods play in Botnet detection. A clear understanding of these roles are critical for developing effective and efficient real-time online detection approaches and more robust models.**

***Keywords*—machine learning; botnet detection; cyber-security; supervised learning; unsupervised learning**

## I. Introduction

As individuals and businesses become more dependent on internet services and information systems to efficiently and effectively carry out their everyday tasks the need for protecting the confidentiality, integrity and availability of these systems has never been greater. Botmasters can use the aggregated power of many bots to exponentially raise the impact of malicious activities. A single bot might not be a danger for the Internet, but a network of bots will definitely be able to create a huge disturbance [18]. We have reviewed several different proposed ML-Based botnet detection systems. Taking into consideration the ML Method, we explore the role of each.

## II. RELATED WORK

Other authors have written reviews of botnet detection approaches and detection techniques. Maryam Feily [2] did a survey of botnets and bot detection, explaining how bots operate. Examining different botnet detection approaches placing botnets in one of three classes, namely anomaly-based, DNS based or Mining based detection. This paper surveyed botnets and botnet detection. Its aim was to explain the botnet phenomena and explore different botnet detection techniques. This paper classified botnet detection into four classes, namely: anomaly-based, signature-based, DNS-based, and mining- based. Along with the summarization of each class, detection techniques are compared.

Thomas Hyslip et al [3] surveyed botnet detection techniques based on their command and control infrastructure. With a history on bot detection technique this review examined various detection techniques and their impact on different botnet architectures.

Michael Bailey et al surveyed bot technology and defenses [4]. This survey looked at different detection methods in light of the data sources which provide the bases for detection, such as DNS, net flow, packet tap etc. They also examined different techniques used by detection methods such as detection based on group behavior and detection based on signature. This survey paper also look at existing botnet research, the evolution and future of botnets.

Stevanovic [1] surveyed machine learning based botnet detection approaches. This paper presented a review of current machine learning based botnet detection methods for identifying botnet-related traffic by providing an overview on this area of study by summarizing the recent scientific efforts. This paper also examined each method and their susceptibility to different resilience techniques that may be employed by botmasters along with the result from each technique, the algorithm used and features chosen. Similar to this paper our focus is on ML-based botnet detection methods. The difference is that our paper explores how different researcher used ML-Methods to overcome specific challenges.

The main difference in this paper is a focus on how different ML-based approaches introduced by different researchers, used different ML methods to detect botnet activity. The main aim of this study is to determine what combination of ML method features and technique is best suited for approaching the botnet detection problem.

## III. THE BOTNET PHENOMENON

A botnet is a network of infected computers, (referred to as zombies or bots), enslaved by an attacker to carry out their bidding. The users whose computers are bots in a botnet are usually ignorant to the fact that their systems have been compromised and are possibly taking part in malicious activities.

The resources of the infected computer - (bot), are under the control of the attacker - (botmaster), who uses these resources for his own agenda. Commands are given and received through communication between the enslaved computers and their botmaster via what is known as a command and control server,

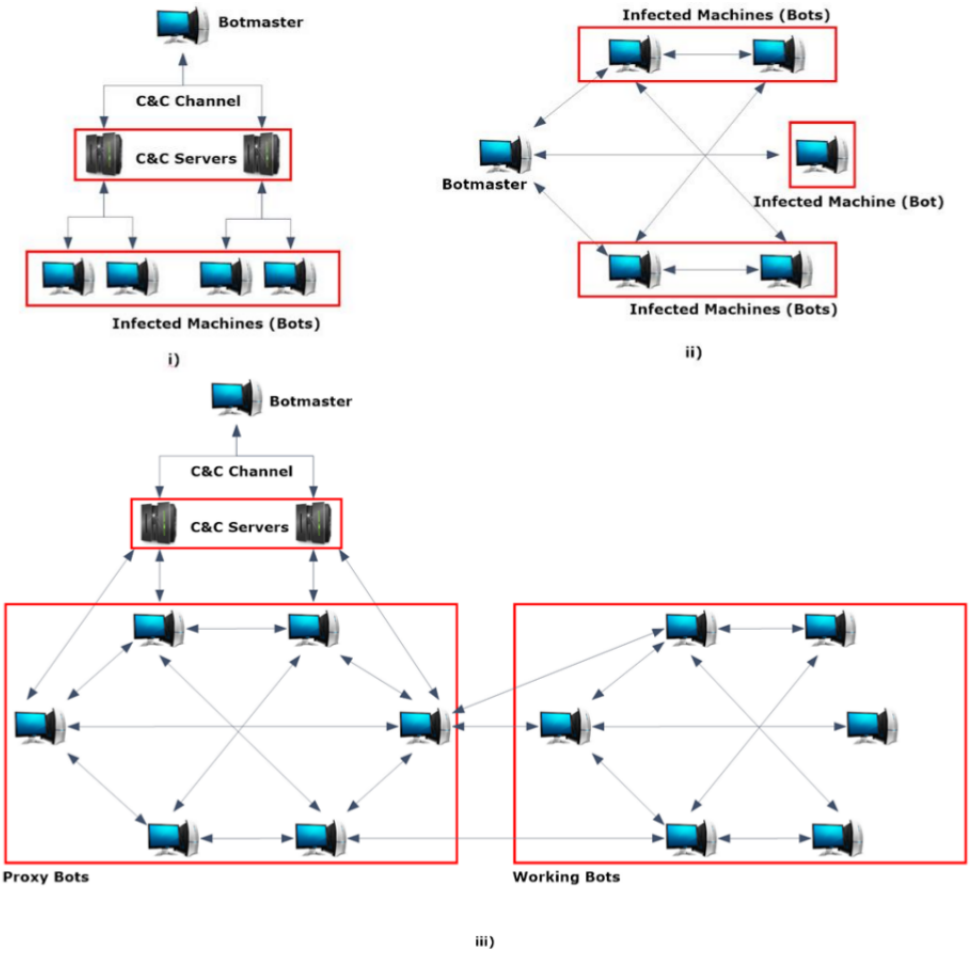


Figure 1: Botnet architectures: i) Centralized ii) Decentralized iii) Hybrid

(C&C). The botnet life-cycle is described in four stages by Leonard et al. [19]

1) Formation, also called infection
2) Command and Control environment
3) Attack
4) Post Attack

At the formation stage where the bot is installed on the users machine by exploiting vulnerabilities, an aspect of this malicious code is responsible for connecting the bot to its command and control server (C&C). After establishing C&C connection, the botmaster is now able to send command to the newly added bot. This transitions into the attack phase, where, upon instructions from the botmaster to the bots in the botnet through the C&C channel, attack commands are issued. After an attack, bots may become exposed and cured, that is, the vulnerabilities they exploited may be patched. The goal then of the botmaster is to recruit more bots - (post attack), and the cycle continues.

### *A. Command and Control architectures*

The command and control channel (C&C) is the backbone of a botnet. The C&C channel is that link between the botmaster and the bot. Unique to bot malware, the C&C channel is its defining characteristic [1]. The C&C channel is used to send instructions to bots and receive information from bots. C&C channels are usually one of three architectures, centralized, decentralized or hybrid (see fig 1). In a centralized botnet all the bots in the network connect to the same C&C server(s) controlled by a single botmaster. Compared to other architectures this has the lowest latency of communication. Decentralized C&C architectures are designed with resilience in mind. Void of a central point of failure, like that of centralized C&C architectures, these botnets possesses multiple paths for sending instructions. The decentralized C&C makes use of peer to peer (P2P) communication protocols as the means of connecting with the infected machines. Hybrid approaches seek to combine the principles of the other two architectures, by using hybrid P2P protocols alongside the low network delay found in centralized architectures.

## IV. BOTNET DETECTION

As botnets become more threatening, researchers and security experts employ different approaches and techniques to solve the problem. The detection approach defines how the solution operates, such as detection by behavior [13] or signature [6]. Different techniques are based on different approaches. ML-based detection techniques are able to use bothapproach. Other techniques used in bot detection include anomaly and DNS.

### A. Detection Approaches

Signature based approaches require detailed knowledge of what a bot or bot related characteristics, (e.g. traffic), may look like. This approach is used to target specific characteristics such as a particular protocol [26] or service. This type of approach tends to be very precise and specific, anything outside the specified scope will go undetected. This approach is very effective against known botnets, but not very useful for unknown bots and are more susceptible to evasion techniques.

Detection based on bot behavior involves describing a model for how botnets generally operate. The generality of this approach makes it possible to capture new or unseen bots, however, too general and the false positive rate may become high. In behavioral approaches researchers make assumptions based on observations about core behaviors of botnets. For botnet detection, the main assumption across approaches is that bots operate in a cooperative manner, engaging in some form of group activity at varying stages of the botnet life cycle [7, 8, 13]. Where specific knowledge of a particular bot drives signature based approaches, a clear definition of bots behavior is at the core of behavioral approaches.

### B. Detection techniques

Anomaly based detection techniques aims to detect bots based on abnormal network activities, such as abnormally high traffic, high latency and unusual port activities. Since bots try to use normal protocols for C&C communication, this is a limitation for anomaly based techniques. Anomaly based techniques takes a behavioral approach to bot detection, thus, it is able to pick up abnormal activities or behavior for unknown bots.

DNS-based detection techniques operate based on DNS information produced by botnets [8]. C&C communication channels are unique to bot malware, bots interact with C&C servers through these channels. To gain access to these servers, bots perform DNS queries. The aim of DNS-based approaches is to capture unusual DNS traffic to identify bots.

ML-based detection techniques have been considered to be the most effective at detecting botnets [1, 2, 3]. Its effectiveness lies within its ability to identify bot related traffic within normal traffic [2]. This is a challenge for other techniques as bots utilize normal protocols to mask C&C communication. However, ML-detection requires a sufficient amount of training examples and well defined features to be effective. As the focus of this paper, machine learning will be discussed in more details in the following section.

### C. Scope of detection

Also, among the botnet detection approaches observed in the literature, each had one goal as it relates to scope of detection, group activity or individual host. The relationship between the scope of detection and ML chosen, demonstrate a high affinity for using unsupervised learning methods when the goal is group detection and supervised when targeting individual hosts.

Detecting bots based on group activity assumes coordinated activities by bots in the same botnet. Distinguished by similar traffic patterns, the aim is to identify all the bots in the network based on their collective action rather that their individual operations. Unlike Group based detection, individual hosts are classified based on their individual actions and characteristics in spite of the activity of the group they might be a part of.

## V. MACHINE LEARNING

Machine learning (ML) is a branch of artificial intelligence that aims to develop systems with the ability to learn from past experience. In machine learning, data, (past experience), is given as input to a ML algorithm to derive patterns that may exist in order to create a model that represents the data. The main concern in this field is, How do we develop computer programs that automatically improve with experience? At the core of ML are statistical and computational principles derived from concepts that exist in many disciplines such as artificial intelligence, philosophy, information theory, biology, cognitive science, computational complexity and control theory [16].

The aim of ML is to create a model based on the data given. This model describes the patterns that exist in the data which should be able to make informed decisions given new (unseen) data.

### A. Machine learning Features

For any machine learning task, the two most important decisions to make will be deciding what features to use and which ML-Method (supervised, unsupervised) to select. The features selected will shape the type of model that is formed. Features are able to represent behaviors or target specific characteristics. The ML-method chosen will impact how the model behaves, one method may create a model whose main concern is how different bots interact with each other while another model concerns itself with how individual bots operate.

Feature selection is the process of extracting the best subset of variables from all possible variables that most accurately represents the data. In botnet detection, the aim of the feature selection process is to select a subset of features that will best describe the behavior of bots or the specific bot being targeted. Features selected will depend on the type of data being used. Number of query lookup may be a feature from DNS data[8], Source and destination IP [15] for net flow data, checksum are features from packet top data. For ML-based detection most researchers chose net flow, (Traffic Flow) data. Traffic flow is an artificial logical equivalent to a call or connection as a sequence of packets sent from a particular

Table I - Role of ML-Method in Botnet detection Systems

| Detection System | ML Method | Scope of detection | Detection approach | True Positive Rate (TPR) | False Positive Rate (FPR) |
|---|---|---|---|---|---|
| David Zhao et al [15] | Supervised | Individual | Signature (P2P) | 98% | 2.3% |
| Carl Livadas et al [6] | Supervised | Individual | Signature (IRC) | NA | 10% — 20% |
| Leyla Bilge et al [11] | Supervised | Individual | Signature ( C&C Server) | 87% | 20% |
| F Sanchez et al [9] | Supervised | Individual | Signature | 91% | 0.56% |
| W. T Strayer et al [5] | Supervised | Individual | Signature (IRC) | NA | 30% |
| Guofei Gu et al [7] | Unsupervised | Group | Behavior | 99% | 1% |
| Yu et al [13] | Unsupervised | Group | Behavior | 100% | 20% |
| Hyunsang Choi et al [8] | Unsupervised | Group | Behavior | 95% | 4% |
| Lei Zhang et al [14] | Unsupervised | Group | Signature | 100% | 0.2% |
| Wei Lu et al [12] | Unsupervised | Group | Signature | 95% | NA |

unicast or multicast destination [20]. From this, flow-level features are derived which describes how each node on a particular network interact with other nodes. Examples of flow-level features are: flow duration, average byte per packet per flow, who indicated the connection (client or server). The features selected will support a particular approach. Flow-level features will support a behavioral approach, packet level features that capture specific characteristics will support a signature-based approach.

The underlying hypothesis for ML-based botnet detection is that, bots produce unique patterns hidden in network traffic or client machine activities. Thus, implementing some form of ML method, one may be able to uncover these patterns to successfully detect malicious activity.

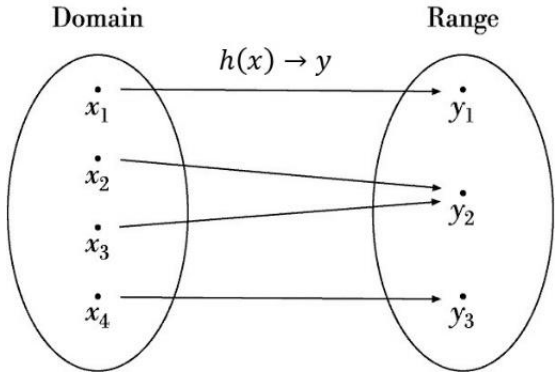

Figure 2: Showing relationship between ML domain x and targets y

## VI. MACHINE LEARNING METHODS USED IN BOTNET DETECTION

In this section we will evaluate the role of each technique in bot detection based on how each has been used. The evaluation of each technique will be separated as follows: first an overview of the technique, second a brief look at how the technique has been used in the literature.

### *A. Supervised Learning (SL)*

In supervised ML, models are built from labeled training data. The aim is to create a model (function h) that represents the data, described by a function (h) that maps input variables x to their appropriate target y (fig 2), this function is sometimes referred to as the hypothesis h(x).

There is also a distinction among supervised learning methods based on how the data is labeled. Supervised learning problems may be categorized as regression or classification. For regression problems, the labeled target values represent a range of values.

For a classification problem, input variables are assigned to classes based on patterns represented in the data. Classification algorithms are concerned with the relationship between class label and input variables. Botnet detection is an example of such a problem, where we are trying to determine what class a packet or sequence of packet may be assigned to, i.e. Botnet or Not botnet traffic.

### *B. The Role of Supervised Learning (SL)*

In 2006 Livadas et al presented a network-based botnet detection approach based on supervised machine learning techniques. The authors did an evaluation of three different machine learning methods for identifying IRC Botnets. Detection was carried out in two phases, the first phase classifies traffic based on IRC traffic. The second phase classifies IRC chat flows as bot-net or real chat flows.

Strayer et al introduced an approach that targets IRC bots. This approach is broken into four stages. In the first stage, flows that are most likely to not have C&C data are filtered out based on knowledge of IRC bots, behavioral patterns and characteristics in flow. The second stage uses supervised learning to identify suspicious traffic flows. The third stage groups flows based on similar predefined characteristics. The groups are then passed to the fourth stage that uses topological analysis to determine flows with the same controller. The flows with the same controller are then examined to see if they are a part of a botnet or not.

Liao et al proposed a method that uses supervised learning techniques to identify P2P bots. The first stage of this two stage approach involves feature extraction. In this stage, specific features that may be used to characterize P2P bots are extracted from the traffic flows. The features of these flows are passed to the second stage where supervised learning algorithm are used to classify each flow.

Shin et al introduced a bot detection system that classi-

fies bots based on activities both on the network and the client's computer. This method has five modules (M1-M5) that correlate bot-related activities on the network and individual clients. The first module M1 is the human-process-network correlation analysis module. This module detects malicious process by monitoring human process on the host relating to the keyboard and mouse and correlating them with network activity. The system checks the time difference between a process producing a mouse click or keyboard event, the source of the event also checked whether or not the process is running in the foreground at that time is also taken into consideration. A small time difference may indicate that the process was generated by a human otherwise this process will be marked as suspicious and forwarded to the M2, M3 and M4. M2 and M3 uses supervised learning to classify queried domains names as malicious or benign and classify malicious behavior on host computers respectively. M4 monitors traffic generated by the suspicious process on the hosts network interface. Incoming packets and exchange rate between process and remote site are compared. If the exchange ratio is smaller than a predefined value, bot behavior is suspected. Finally, after each module makes its decision, the correlation engine - M5, combines the results to make the final decision using a weighted voting scheme.

### C. Unsupervised Learning (UL)

Unsupervised Learning is the area of machine learning concerned with developing systems that can learn how to represent patterns in a data set based solely on input variables. The main aim of such a learner is to establish a function to describe hidden patterns in unlabeled data. The absence of target values (y), or external environmental evaluation, is what distinguishes unsupervised learning from supervised and reinforcement learning [17]. The most common form of unsupervised learning is called clustering. This is an unsupervised learning technique used to fined similarity in unlabeled data by grouping them in sections called clusters.

Seeing that all data points looks the same (unlabeled), the aim of a clustering algorithm is to understand the relationship between each data points and group them accordingly. In the same way, as it relates to botnet detection, clustering algorithms have been used to group traffic of similar characteristics in an effort to single out and identify traffic with malicious intent. The Botminer detection system [7] clusters similar communication traffic and similar malicious traffic and performs cross cluster correlation to identify the hosts that share both similar communication patterns and similar malicious activity patterns.

### D. The Role of Unsupervised Learning (UL)

Yu et al proposed a method for online detection using the k-means clustering algorithm to group bot related traffic. The approach uses network flow features [20] in predefined time windows. The aim is to group traffic based on similarity. The cluster with similarity greater than a predefined threshold will be classed as suspicious thus the host related to these flow will be flagged.

Chioi et al proposed a method for detecting bots based on how different host use DNS services. Bots use DNS to look up C&C servers and victims. The assumption by the researchers is that, bots apart of the same botnet will use DNS services similarly. This method uses the X-means clustering algorithm to group domains that may be related to a botnet.

Zhang et al introduced a system for detecting botnets that identifies P2P botnets in spite of the botnet being currently engaged in malicious activity. The emphasis of this method is to detect P2P bots by identifying C&C communication patterns that characterize P2P bots. The system first identifies P2P hosts then P2P bots among those hosts. This approach uses flow level features, the system presumes that P2P nodes create many failed outgoing flows. For each cluster of flows their destination IP is checked and for each IP their BGP prefix are checked. If the number of distinct BGP prefixes are smaller than a predefined amount are ignored. To differentiate legitimate P2P traffic from bot P2P connections, the authors assume that bots of the same botnet uses similar P2P protocol and network. Also they assume that pairs connect by two bots that have longer overlaps than that of legitimate P2P traffic.

Gu et al like most, assumes bot exhibit similar patterns in their traffic flows. Using the X-means clustering algorithm, the authors group flow with similar communication patterns. This method has five components with three levels. The first level has the A and C-Plain monitors that monitors outgoing and internal traffic flows respectively. The second level is made up of the A and C-Plain clustering that clusters traffic, filtered by their respective monitors of the previous level. The results from these clusters are then passed to the third level, the cross-plain correlator, which makes the final decision about hosts that may be a part of a botnet. By combining the results from the A and C plain clusters.

W.Lu et al proposed a method that clusters flows based on similarities in payload. This method is split up into three sections, the first stage analyses feature, the second, clusters flows and the third, botnet decision. In the first stage, features are extracted from the flow payload in the time intervals as a 256-dimentional vector. In the second stage, flows are clustered using k-means and x-means clustering algorithm. These clusters are then passed to the third phase where the cluster with the lowest standard deviation is marked as botnet.

## VII. Conclusion

As bots became more threatening, research efforts in the area intensified, producing various methods of detecting and defending against botnets. To date, ML-based detection methods have proven to be quite effective, though not without their limitations. Timely detection, real-time monitoring and adaptability to new threats are issues still to be solved. The different ML methods have different strengths and weaknesses as seen in the role they play in bot detection. The statistical foundation of SL methods (i.e. the hypothesis representation), concerns its self with the relationship between the features (x)

and target (y). In order to accurately represent the behavior of bots using SL, this must be defined by the features selected and thus assumes some detailed knowledge about what this behavior looks like.

Based on the characteristics of SL researchers in this field has made use of the precision of SL methods to accurately identify bots based on some known and specific characteristics (features used in SL). The precision of SL can be quite effective against bot traffic that seek to camouflage itself among legitimate traffic, given some specific characteristics of the malicious traffic. In our survey of the SL techniques we have observed a common trend. Apart from specific insights about bot traffic revealed in the feature space, SL methods perform very poorly. SL methods may overcome the camouflaged nature of bots. As seen in Table I, supervised learning methods are employed for cases where some specific characteristic is known.

Unsupervised learning methods are mainly used to target behavioral patterns not unique to any type of bot. The aim of research that used unsupervised methods is to capture group activity by bots in a botnet. Unsupervised learning in contrast to supervised learning has as its main concern, the relationship between samples. For this reason, it is able to recognize patterns that appears. Being so concerned with the similarity between samples, may cause a high rate of false positives as bots try to camouflage their activities. This issue however has been dealt with by some researchers [7], by representing specific characteristics in the features space that shape how groups are formed.

## REFERENCES


[1] Stevanovic, Matija, and Jens Myrup Pedersen. "Machine learning for identifying botnet network traffic." (2013).

[2] Maryam Feily , Alireza Shahrestani , Sureswaran Ramadass, A Survey of Botnet and Botnet Detection, Proceedings of the 2009 Third International Conference on Emerging Security Information, Systems and Technologies, p.268-273, June 18-23, 2009

[3] Thomas S. Hyslip,Jason M. Pittman,A Survey of Botnet Detection Techniques by Command and Control Infrastructure. 2015.

[4] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, A Survey of Botnet Technology and Defenses, Cybersecurity Applications & Technology Conference for Homeland Security, IEEE Computer Society, Los Alamitos, CA, 2009, pp. 299-304.

[5] W. T. Strayer, D. Lapsely, R. Walsh, C. Livadas, Botnet detection basedon network behaviour, in: W. Lee, C. Wang, D. Dagon (Eds.), Botnet Detection, Vol. 36 of Advances in Information Security, Springer, 2008, pp. 124.

[6] C. Livadas, R. Walsh, D. Lapsley, W. Strayer, Usilng machine learning technliques to identify botnet traffic, in: Local Computer Networks, Proceedings 2006 31st IEEE Conference on, 2006, pp. 967 974. doi:10.1109/LCN.2006.322210.

[7] G. Gu, R. Perdisci, J. Zhang, W. Lee, Botminer: Clustering analysis of network traffic for protocol- and structure-independent botnet detection,in: Proceedings of the 17th conference on Security symposium, 2008, pp. 139154.

[8] H. Choi, H. Lee, Identifying botnets by capturing group activities in DNS traffic, Journal of Computer Networks 56 (2011) 2033.

[9] F. Sanchez, Z. Duan, Y. Dong, Blocking spam by separating end-user machines from legitimate mail server machines, in: Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference, CEAS 11, ACM, New York, NY, USA, 2011, pp. 116124. doi:10.1145/2030376.2030390.

[10] W. Strayer, R. Walsh, C. Livadas, D. Lapsley, Detecting botnets with tight command and control, in: Local Computer Networks, Proceedings 2006 31st IEEE Conference on, 2006, pp. 195 202. doi:10.1109/LCN.2006.322100.

[11] L. Bilge, D. Balzarotti, W. Robertson, E. Kirda, C. Kruegel, Disclosure: detecting botnet command and control servers through large-scale netflow analysis, in: Proceedings of the 28th Annual Computer Security Applications Conference, ACSAC 12, ACM, New York, NY, USA, 2012, pp. 129138. doi:10.1145/2420950.2420969

[12] W. Lu, G. Rammidi, A. A. Ghorbani, Clustering botnet communication traffic based on n-gram feature selection, Computer Communications 34 (2011) 502514.

[13] X. Yu, X. Dong, G. Yu, Y. Qin, D. Yue, Data-adaptive clustering analysis for online botnet detection, in: Computational Science and Optimization (CSO), 2010 Third International Joint Conference on, Vol. 1, 2010, pp. 456 460. doi:10.1109/CSO.2010.214.

[14] J. Zhang, R. Perdisci, W. Lee, U. Sarfraz, X. Luo, Detecting stealthy P2P botnets using statistical traffic fingerprints, in: 2011 IEEE/IFIP 41st International Conference on Dependable Systems and Networks (DSN), Hong Kong, IEEE/IFIP, 2011, pp. 121132.

[15] Zhao, David, Issa Traore, Ali Ghorbani, Bassam Sayed, Sherif Saad, and Wei Lu. "Peer to peer botnet detection based on flow intervals." In Information Security and Privacy Research, pp. 87-102. Springer Berlin Heidelberg, 2012.

[16] T. M. Mitchell, Machine Learning, 1st Edition, McGraw-Hill, Inc., New York, NY, USA, 1997.

[17] Dayan, Peter. "Unsupervised learning." The MIT encyclopedia of the cognitive sciences (1999).

[18] Vania, Jignesh, Arvind Meniya, and H. B. Jethva. "A review on botnet and detection technique." Int J Comput Trends Technol 4, no. 1 (2013): 23-29.

[19] Leonard, Justin, Shouhuai Xu, and Ravi Sandhu. "A framework for understanding botnets." In Availability, Reliability and Security, 2009. ARES'09. International Conference on, pp. 917-922. IEEE, 2009.

[20] Brownlee, N., C. Mills, and G. Ruth. "RFC2722." Traffic Flow Measurement: Architecture (1999).