On the Use of Machine Learning for Identifying Botnet Network Traffic

Stevanovic, Matija; Pedersen, Jens Myrup

Published in:
Journal of Cyber Security and Mobility

DOI (link to publication from Publisher):
10.13052/jcsm2245-1439.421

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

On the Use of Machine Learning for Identifying Botnet Network Traffic

Matija Stevanovic and Jens Myrup Pedersen

Wireless Communication Networks Section, Department of Electronic Systems
Aalborg University, Aalborg, Denmark
Email: {mst; jens}@es.aau.dk

Received 31 August 2015; Accepted 20 November 2015;
Publication 22 January 2016

Abstract

During the last decade significant scientific efforts have been invested in the development of methods that could provide efficient and effective botnet detection. As a result, an array of detection methods based on diverse technical principles and targeting various aspects of botnet phenomena have been defined. As botnets rely on the Internet for both communicating with the attacker as well as for implementing different attack campaigns, network traffic analysis is one of the main means of identifying their existence. In addition to relying on traffic analysis for botnet detection, many contemporary approaches use machine learning techniques for identifying malicious traffic. This paper presents a survey of contemporary botnet detection methods that rely on machine learning for identifying botnet network traffic. The paper provides a comprehensive overview on the existing scientific work thus contributing to the better understanding of capabilities, limitations and opportunities of using machine learning for identifying botnet traffic. Furthermore, the paper outlines possibilities for the future development of machine learning-based botnet detection systems.

Keywords: Botnet detection, State of the art, Comparative analysis, Traffic analysis, Machine learning.

Journal of Cyber Security, Vol. 4, 1–32.
doi: 10.13052/jcsm2245-1439.421

1 Introduction

Botnets represent networks of computers compromised with sophisticated bot malware that puts them under the control of a remote attacker [1]. Bot malware provides the attacker with the ability to remotely control behavior of the compromised computers through specially deployed Command and Control (C&C) communication channels. Computers compromised by the bot malware are popularly referred to as bots or zombies, while the attacker is referred to as the botmaster. Controlled and coordinated by the botmaster, botnets represent a collaborative and highly distributed platform for the implementation of a wide range of malicious and illegal activities, such as sending SMAP e-mails, DDoS (Distributed Denial of Service) attacks, Information theft, etc. Due to their malicious potential botnets are often regarded as one of the biggest security threats today [1, 2].

Over the course of the last decade, many botnet detection approaches have been reported in the literature, with various goals, based on diverse technical principles and varying assumptions about bot behavior and the characteristics of botnet network activity [2–4]. As botnets rely on the Internet for both communicating with the attacker as well as for implementing different attack campaigns network traffic analysis is one of the main means of identifying existence of botnets. One of the latest trends in network-based botnet detection is the use of machine learning algorithms (MLAs) for identifying patterns of malicious traffic. The main assumption of machine learning-based methods is that botnets create distinguishable patterns within the network traffic and that these patterns could be efficiently detected using MLAs. This class of detection approaches promises automated detection that is able to generalize knowledge about malicious network traffic from the available observations, thus avoiding pitfalls of signature-based detection approaches that are only able to detect known traffic anomalies. Various detection methods have been developed using an array of MLAs deployed in diverse setups [5–24]. These methods employ diverse principles of traffic analysis targeting various characteristics of botnet network activity. Furthermore, contemporary detection methods have been evaluated using different evaluation methodologies and data sets. The great number of diverse detection solutions introduces the need for a comprehensive approach to summarizing and comparing existing scientific efforts, with a goal of understanding the challenges of this class of detection methods and pinpointing opportunities for the future work.

A number of authors have tried to summarize the field of botnet protection through series of survey papers [1, 2, 25, 26]. Although providing

a thorough overview of the field, they only briefly address contemporary detection approaches. In parallel, several authors, such as Feily et al. [25], Bailey et al. [26], Garcia et al. [3], Hyslip et al. [27] and Karim et al. [4] have summarized scientific efforts on botnet detection by proposing novel taxonomies of detection methods and presenting some of the most prominent methods. The authors have acknowledged the potential of machine learning-based approaches in providing efficient and effective detection. Garcia et al. [3] and Karim et al. [4] have provided some of the most comprehensive surveys on existing network-based botnet detection approaches indicating the crucial place of approaches that use MLAs for identifying botnet network activity. Garcia et al. [3] have compared 14 contemporary anomaly-based detection approaches from which 8 were based on MLAs, while Karim et al. [4] analyzed only 3 approaches based on MLAs in more details. Masud et al. [28] and Dua et al. [29] have analyzed the general role of machine learning within modern cyber-security. The authors have outlined the benefits of using machine learning for discovering the existence of the malware on both network and client levels. However, the authors have not provided an overview of the state of the art on botnet detection, leaving the question of current trends within the field of botnet detection unanswered. Finally, several authors have addressed the challenges of using MLAs for network-based detection [30, 31]. Sommer et al. [30] have pointed out some of the challenges and pitfalls of using MLAs for intrusion detection. Although relevant to the realm of botnet detection, claims regarding the usefulness of MLA should be re-evaluated for the botnet detection context. Aviv et al. [31] have presented some of the challenges in experimenting with botnet detection methods that are highly relevant to the use of MLAs for botnet detection.

To the best of our knowledge this paper is the first to provide a comprehensive overview of contemporary detection methods that rely on MLAs for identifying botnet network traffic. The paper has the goal of contributing to a better understanding of capabilities, limitations and opportunities of using machine learning for identifying botnet traffic. The contribution of the paper is three-fold. First, the paper provides a detailed insight on the field by summarizing current scientific efforts. The paper analyzes 20 contemporary detection methods by investigating the principles of traffic analysis used by the detection approaches and how different machine learning techniques are adapted in order to recognize botnet-related traffic patterns. Second, the paper compares the detection methods by outlining their capabilities, limitations and detection performance. Special attention is placed on the practice of experimenting with existing detection approaches and the methodologies of performance

evaluation. Third, the paper indicates challenges and limitations of the use of machine learning for identifying botnet traffic and outlines possibilities for future development of machine learning-based botnet detection systems.

The rest of the paper is organized as follows. Section 2 presents the background on botnet detection. The section places special emphasis on botnet detection based on traffic analysis and the use of machine learning for identifying botnet-related traffic. Section 3 presents the analysis principles used in order to evaluate existing detection methods. Section 4 presents the comparative analysis of the state of the art on botnet detection based on machine learning. This section present the most prominent detection approaches by analyzing their characteristics, capabilities and limitations. The discussion of the presented scientific efforts and possibilities for future improvements is presented in Section 5. Finally, Section 6 concludes the paper.

2 Botnet Detection

This section presents the background on botnet detection with the focus on network-based detection. Furthermore, this section presents the principles of using MLAs for identifying botnet network traffic.

Depending of the point of deployment detection approaches can generally be classified as *client-based* or *network-based*. Client-based detection approaches are deployed at the client computer targeting bot malware operating at the compromised machine [28, 32–37]. These methods commonly detect the presence of bot malware by examining different client level forensics, such as: API calls, file changes, application and system logs, active processes, key-logs, usage of the resources, etc. In addition, some client-based detection methods also include the analysis of traffic visible on the computer's network interfaces [22–24]. Network-based detection, on the other hand, provides botnet detection by analyzing network traffic at different points in the network. This class of methods identifies botnets by identifying network traffic produced in different botnet operational phases such as C&C communication, attack phase and propagation [2].

There are several conceptual differences between client- and network-based detection because of which network-based detection is often seen as a more promising solution. Network-based detection is targeting the essential aspects of botnet functioning, i.e. network traffic produced as the result of botnet operation. Network-based approaches assume that in order to implement its malicious functions botnets have to exhibit certain network activity. Botnets could make their operation more stealthy by limiting the intensity of attack

campaigns (sending SPAM, launching DDoS attacks, scanning for vulnerabilities, etc.) and by tainting and obfuscating C&C communication. However, this often contradicts the goal of providing the most prompt, powerful and efficient implementation of malicious campaigns. On the other side, attackers invest great efforts in making the presence of bot malware undetectable at compromised machines through a number of client level resilience techniques such as rootkit ability and code obfuscation [38–40]. The attackers also try to deploy a number of network based resilience techniques such as Fast-flux, Domain-flux and encryption but these techniques often introduce additional botnet traits that can be used for detection [17, 41]. Furthermore, as network-based detection is primarily based on passive analysis of network traffic it is more stealthy in its operation and even undetectable to botnet operators in comparison to the client-based detection which could be detected by the malware operating at the compromised machine. Finally, depending of the point of traffic monitoring network-based detection can have a wider scope then the client-level detection systems. When deployed in core and ISP networks network-based detection approaches are able to capture traffic from a larger number of client machines. This provides the ability of capturing additional aspects of botnet phenomena, for instance, group behavior of bots within the same botnet [8, 42], time regularities of bots activity and diurnal propagation characteristics of botnets [43].

2.1 Network-Based Detection

Network-based detection is based on the analysis of network traffic in order to identify presence of compromised computers. Network-based detection is commonly classified based on the principles of functioning as *signature-* or *anomaly-based* methods. Furthermore, methods can be classified as *passive* or *active* depending on the stealthiness of their operation.

The passive detection approaches operate based on observation only thus they do not interfere with botnet operation which makes them stealthy in their operation and undetectable to the attacker. Active detection methods represent more invasive methods that actively disturb botnet operation by interfering with malicious activities or the C&C communication of the bots. Additionally, these techniques often target specific heuristics of the C&C communication or attack campaigns, arguably providing higher accuracy of detection. The majority of contemporary botnet detection approaches are passive while only a few such as BotProbe [44] are active.

Signature-based methods are based on recognizing botnet specific characteristics of traffic, also known as "*signatures*" [45–48]. The signature-based methods rely on a set of predefined rules regarding anomalous traffic and packet level signatures. These approaches commonly performs packet level analysis by using deep packet inspection (DPI) in order to match signatures of malicious payloads. This class of detection techniques covers all three phases of botnet life-cycle and it is able to detect known botnets with bounded number of false positives (FP). The main drawback of signature-based approaches is that they are only able to detect known threats, and that efficient use of these approaches requires constant update of botnet traffic signatures. Additionally, these techniques are liable of various evasion techniques that change signatures of botnet traffic and malicious activities of bots, such as encryption and obfuscation of C&C channel, Fast-flux and DGA techniques, etc.

Anomaly-based detection is a class of detection methods that is devoted to the detection of traffic anomalies associated with botnet operation [5–24, 42, 49–53]. The traffic anomalies that could be used for detection vary from easily detectable as changes in traffic rate and latency, to more finite anomalies in flow patterns. Some of the most prominent anomaly-based approaches detect anomalies in packet payloads [45, 49], DNS traffic [17, 51, 52], botnet group behaviour [8, 42, 43], etc. The anomaly-based detection can be realized using different algorithms ranging from the statistical approaches, machine learning techniques, graph analysis, etc. In contrast to the signature-based detection, the anomaly-based detection is generally able to detect new forms of malicious activity that exhibits anomalous botnet related characteristics. However, one of the main challenges of using anomaly-based detection is the fact that in contrast to signature-based detection these approaches result in false positives. One of the latest and the most promising sub-class of anomaly-based methods are detection methods that rely on machine learning for detection of bot-related traffic patterns. The machine learning is used because it offers the possibility of automated recognition of bot-related traffic patterns. Additionally, machine learning provides the ability of recognizing the patterns of malicious traffic without a prior knowledge about the malicious traffic characteristics, but by inferring knowledge from the available botnet traffic traces.

2.2 Machine Learning for Botnet Detection

The basic assumption behind machine learning-based methods is that botnets produce distinguishable patterns of network activity and that these patterns could be detected by employing some of the MLAs [28, 29].

Machine Learning (ML), is a branch of artificial intelligence, that has the goal of construction and studying of systems that can learn from data [54]. Learning in this context implies ability to recognize complex patterns and make qualified decisions based on previously seen data. The main challenge of machine learning is how to provide generalization of knowledge derived from the limited set of previous experiences, in order to produce a useful decision for new, previously unseen, events. To tackle this problem the field of Machine Learning develops an array of algorithms that discover knowledge from specific data and experience, based on sound statistical and computational principles. Machine learning algorithms can be coarsely classified based on the desired outcome of the algorithm as *supervised* MLAs and *unsupervised* MLAs.

Supervised learning [55] is the class of well-defined machine learning algorithms that generate a function (i.e., model) that maps inputs to desired outputs. These algorithms are trained by examples of inputs and their corresponding outputs, and then they are used to predict output for some future inputs. The supervised MLAs are used for classifying input data into some defined class and for regression that predict continuous valued output. In the context of botnet detection, supervised MLAs are commonly used for implementing network traffic classifiers that are able to classify malicious from non-malicious traffic or identify traffic belonging to different botnets. Some of the most popular supervised MLA used for botnet detection are: SVM (Support Vector Machines), ANN (Artificial Neural Networks), Decision tree classifiers and Bayesian classifier.

Unsupervised learning [56] is the class of machine learning algorithms where training data consists of a set of inputs without any corresponding target output values. The goal of unsupervised learning may be to discover groups of similar examples within the input data, referred to as clustering, to determine the distribution of data within the input space, known as density estimation, or to project the data from a high-dimensional space down to two or three dimensions for the purpose of visualization. In the context of botnet detection, un-supervised MLAs are commonly used for the clustering of bot-related observations. The main characteristic of unsupervised MLAs is that they do not need to be trained beforehand. The most popular unsupervised learning approaches used for botnet detection are: K-means, X-means and Hierarchical clustering.

In both learning scenarios traffic is analyzed from a certain analysis perspective that entails how do traffic instances, that will be classified or clustered by MLAs, look like. For each of traffic instances a set of features

is extracted and used within the MLAs to represent them. Choosing the right features representation is one of the most challenging task of practical deployment of MLAs. The chosen features should capture targeted botnet traffic characteristics and pose balanced requirements in terms of feature extraction and selection.

In parallel with the two learning problems outlined here modern machine learning-based approaches commonly implement detection through several phases, using the combination of different MLAs or by deploying MLAs in an adaptive manner. This way more fine grained, flexible, and adaptable detection can be achieved. More details on contemporary detection approaches based on machine learning can be found in Section 4.

3 Principles of the Analysis

This section presents the principles of analyzing contemporary botnet detection approaches that rely on MLAs for identifying botnet network activity. The main goal of the analysis is to provide a review of characteristics and performance of the existing detection methods in order to assess if they can provide accurate, real-time detection that is robust to evasion techniques. The analysis is done by analyzing the characteristics of detection methods, their performance and evasion techniques methods are vulnerable to. The details on the principle of the analysis are presented in the following.

3.1 Characteristics of Detection Methods

The characteristics of detection methods are investigated in order to get the understanding of capabilities and limitations of contemporary detection methods. The analysis of the characteristics is realized by analyzing the following:

- **Point of traffic monitoring** – Point of traffic monitoring infers different capabilities of detection methods. Existing detection approaches monitor traffic at compromised clients, local networks, campus/enterprise networks and in core and ISP networks. The main difference between the points at which methods are implemented is the visible network scope. For instance, a detection system implemented in the core network has the potential of having more comprehensive outlook on the behavior of bots within a certain botnet, than the detection system implemented at a gateway connecting a local network to the Internet. By the same token, the client-based techniques are only able to capture network traffic produced

by individual bots. In this analysis we outline points of traffic monitoring assumed by the detection methods.

- **Detection target** – Detection methods commonly detect botnets by identifying bots or C&C infrastructure. However, in this study we also address methods that do not directly detect botnets but provide detection of different network traffic anomalies that commonly characterize botnet operation such as Fast-flux and Domain-flux. These methods are DNS-based detection methods that commonly discover malicious domain names. The findings from DNS-based methods could be used for discovering C&C infrastructure, as well as for discovering potentially compromised clients that try to resolve malicious domain names. In the analysis we outline detection targets of the contemporary detection approaches. Furthermore, we identify all possible detection targets for the DNS-based detection approaches.
- **Botnet type** – Botnets are often coarsely classified based on the employed C&C communication protocol as IRC, HTTP and P2P botnets. Detection methods can cover specific types of botnets or be able to detect botnets independently from the used C&C protocol. Detection approaches that target specific types of botnets are often more efficient than more generic methods. However, these detection techniques are at the same time less flexible to the changes in the communication technology used by botnets. In this analysis we outline the type of botnets targeted by the detection methods.
- **Operational phase** – Detection methods can target different botnet operational phases i.e., the propagation phase, the C&C communication phase or the attack phase [3]. Detection approaches that cover the C&C communication phase can be directed at various communication protocols (IRC, HTTP, P2P), while detection approaches that cover the attack phase can target different attack campaigns (SPAM, DDoS, etc.). Similarly as for the previous characteristic, detection methods that target a specific operational phase can be more efficient than the ones that target more operational phases. Again, these detection techniques are less flexible to the changing nature of the botnet phenomenon. In this analysis we outline operational phases targeted by the detection methods.
- **Communication protocol** – Detection methods analyze traffic at different communication protocols in order to achieve detection. The targeted communication protocols to a large degree depend on the type of botnets and operational phases targeted by the detection approach. Most commonly, detection approaches analyze traffic at transport layer by

targeting TCP and UDP traffic and at application layer targeting HTTP, IRC and DNS traffic. In this analysis we outline communication protocols analyzed by the detection methods.

- **MLAs** – The contemporary detection methods rely on different MLAs employed in diverse setups in order to perform detection of malicious network traffic. Both supervised and unsupervised MLAs are used offering different levels of automation and resulting in different types of findings. Furthermore, traffic is analyzed by the MLAs from different perspectives depending on the targeted communication protocols. Finally, in order to capture anomalous characteristics of botnet traffic detection approaches rely on a diverse sets of traffic features extracted for each traffic instance analyzed by the MLAs. In this analysis we outline MLAs and traffic analysis perspective used by the detection methods. Furthermore, we briefly elaborate on the employed feature representation.
- **Real-time operation** – Timely detection is the preferred characteristic of detection systems defined as the ability of operating efficiently and producing the detection results in a "reasonable" time. The timely detection entails a need for a detection method to operate in real-time fashion, thus being capable of processing large quantities of data efficiently. However, it should be noted that the requirements of realtime operation vary depending on the used traffic analysis principles and the goal of detection. In this analysis we outline methods that are advertised as providing real-time detection and detection approaches that have potential of being used in real-time based on their operational efficiency.

3.2 Performance Evaluation

Performance evaluation is realized by analyzing evaluation practices used for experimenting with detection methods, quantitative and qualitative aspects of evaluation data and obtained detection performance.

One of the main prerequisites of reliable performance evaluation is the quality of traffic data sets used for training and testing of the proposed detection approaches [3, 31]. The evaluation data sets should include a substantial amount of traffic for which the "*ground truth*" is known i.e. traffic consisting of elements labeled as malicious and non-malicious. Correctly labeled data sets are one of the main requirements of deterministic evaluation of detection performance. The malicious traffic represent traffic produced by botnets, while non-malicious traffic, often refereed to as the "*background*" traffic originates from benign applications running on "*clean*" computers. The labeled data set

is formed either by labeling previously recorded traffic trace or by combining malicious and non malicious data sets. Typical scenarios of obtaining labeled data sets are outlined bellow:

- **Scenario 1** – Labeled traffic is obtained by performing labeling of a network trace using a variety of different labeling practices. The labeling of the traffic can be done by using some of the existing IDS systems [46, 47, 57] and signature-based botnet detection systems [45, 48] or by relying on domain name and IP address blacklists.
- **Scenario 2** – Labeled traffic is obtained by merging a non-malicious traffic trace with a malicious traffic trace captured by Honeypots [58] deployed by researchers themselves or by some third party.
- **Scenario 3** – Labeled traffic is obtained by merging a non-malicious traffic trace with a malicious traffic trace generated within fully controllable testing environments, where researchers have total control on both C&C servers and compromised computers. This scenario requires bot malware source code to be available. Having the source code, experiments can be realized in safe and totally controlled fashion.
- **Scenario 4** – Labeled traffic is obtained by merging a non-malicious traffic trace with a malicious traffic trace generated in semi-controlled testing environments, by purposely infecting computers with a specific bot malware. Compromised computers are allowed to contact the C&C servers in order for bot-related traffic to be recorded. In order to limit any unwanted damage to the third parties on the Internet the traffic produced by infected machines is filtered using different rate and connection limiting techniques as well as matching of the malicious signatures of bot traffic [58].

Non-malicious traffic traces could be obtained in various ways: from self generated traffic using statistical traffic generators to the network traces recorded on LAN, campus/enterprise and in some cases even core and ISP networks. However, it should be noted that for the process of obtaining background traffic the primary concern is to make sure that the traffic traces are benign and "representable" for the particular point of traffic monitoring. While easily achieved on the controlled LAN networks, making sure that traffic obtained from other real-world networks is benign is a much more challenging task. Furthermore, as traffic from one network to another vary, background traffic should match the malicious trace in terms of the point of traffic monitoring and the type of network.

Besides the way evaluation data sets are obtained, the number of distinct bot malware samples used for the evaluation of botnet detection methods is also very important for assessing the generality of the obtained detection performance. Evaluating the detection method using traffic traces from different types of botnets could indicate the ability of the method to cope with new threats. Furthermore, using traffic traces form different botnets for training and testing could give a good indication if a method can generalize well or not.

Understanding performance metrics used for characterizing contemporary detection methods is crucial for assessing the capabilities of approaches. Some of the most frequently used performance metrics are the following:

True positives rate i.e. recall: $TPR = recall = \dfrac{TP}{TP + FN}$

True negative rate: $TNR = \dfrac{TN}{TN + FP}$

False positive rate: $FPR = \dfrac{FP}{FP + TN}$

False negative rate: $FNR = \dfrac{FN}{FN + TP}$

Accuracy: $accuracy = \dfrac{TP + TN}{TP + FP + TN + FN}$

Error rate: $error = \dfrac{FP + FN}{TP + FP + TN + FN}$

Precision: $precision = \dfrac{TP}{TP + FP}$

True positive (*TP*) is the number of positive samples classified as positive, true negative (*TN*) is the number of negative samples classified as negative, false positive (*FP*) is the number of negative samples classified as positive, and false negative (*FN*) is the number of positive samples classified as negative. However, it should be noted that the performance of detection approaches are commonly expressed using only a subset of the presented performance metrics, most commonly TPR and FPR.

3.3 Evasion Tactics

Detection methods should be robust on evasion techniques in such a way that for detection to be evaded botnets should severely limit the efficiency of implementing their malicious agenda. The vulnerability of detection approaches to evasion techniques highly depend on the principles of traffic analysis and the characteristics of botnet traffic targeted by the detection method. Targeting easily changeable botnet characteristics can lead to evasion, which would consequently limit the effectiveness of the detection approach. Following Stinson et al. [59] framework for systematic evaluation of the robustness of botnet detection methods we evaluate the contemporary detection methods against following evasion tactics (ET):

- **ET1** – Evasion of client based detection. Evasion tactic that evades botnet detection at the client machine. This category includes a wide range of techniques, such as evasion by attacking process monitor and evasion by tainting bot malware behavior at the client computer.
- **ET2** – Evasion by traffic encryption. Tactic that performs encryption of the traffic used within the C&C channel.
- **ET3** – Time-based evasion. Evasion techniques that try to avoid bot activity in specific time windows in which detection methods operate, thus restricting the detection methods from catching the right observations.
- **ET4** – Evasion by flow perturbation. This class of evasion techniques changes the patterns of traffic by changing the flow statistics.
- **ET5** – Evasion by performing only a subset of available attacks, thus limiting the available observation for the methods that are targeting the attack phase of botnet operational life-cycle.
- **ET6** – Evasion by restricting the number of attack targets, by targeting clients at the same internal network, thus evading the methods that monitor traffic at network boundaries.
- **ET7** – Evasion of cross-host clustering by employing sophisticated schemes avoiding the group activities of bots within the same administrative domain.
- **ET8** – Evasion by out-of-band coordination of bots, by using Fast-flux and DGA algorithms as a mean of communicating, thus providing a level of privacy and resilience to malicious C&C servers.

The majority of the existing detection methods could be evaded by deploying some of the presented evasion techniques. However, evasion techniques are characterized with implementation costs and performance loss that vary from low to very high [59], thus often causing severe damage to the utility of botnets.

Therefore, the fact that detection system could be evaded does not necessarily mean that the cost of evasion will be justified.

4 State of the Art: The Analysis Outlook

This section analyzes contemporary machine learning-based botnet detection approaches, on the basis of the principles presented in Section 3. The section evaluates 20 contemporary detection methods [5–24]. The majority of the evaluated methods are purely network-based [5–21] while the study also covers several client-level detection approaches that strongly rely on network traffic analysis [22–24].

4.1 Capabilities and Limitations

The characteristics of the analyzed detection approaches are summarized by Table 1 and Table 2, where Table 1 provides an overview of the characteristics of detection methods, while Table 2 summarizes the principles of traffic analysis and MLAs used by the approaches.

Depending on the point of traffic monitoring the majority of detection approaches addressed by this survey monitor traffic at local [9, 10, 12, 21] and possibly campus/enterprise networks [5–8, 11, 20], while others can be implemented in core and ISP networks [8, 13–19]. Finally, some of the client-based techniques that strongly rely on network traffic analysis are also addressed in this survey [22–24].

The detection methods typically contribute to the identification of bots [5–12, 19–21, 23, 24] or malicious C&C servers [13, 18, 22]. DNS-based detection methods [14–17] provide identification of malicious domains that can contribute to the detection of both bots and C&C servers. Based on the identified malicious domains it is possible to identify both bots that try to resolve them as well as the C&C infrastructure behind them.

The majority of the analyzed detection approaches target C&C communication as the main characteristics of botnet operation, while some also include the ability to capture botnet attack campaigns as well [7, 10, 22–24]. The propagation phases is covered by only one detection method [7], most likely as the propagation could be effectively tackled by existing IDS/IPS systems.

Roughly a half of the analyzed detection methods are independent of C&C communication [7, 8, 13–18, 23, 24], while other methods target specific types of botnets, such as IRC-based [5, 6, 12, 22], HTTP-based [21] and P2P-based [9–11, 19, 20] botnets by relying on specific traits of IRC,

Table 1 The characteristics of detection methods

Detection Method	Traffic Monitoring Point	Detection Target	Botnet Type	Operational Phase	Communication Protocol	Real-Time Operation
Livadas et al. [5]	LAN, Campus	Bots	IRC	C&C	TCP, IRC	–
Strayer et al. [6]	LAN, Campus	Bots	IRC	C&C	TCP, IRC	potentially
Gu et al. [7]	LAN, Campus	Bots	Generic	Propagation, C&C, Attack	TCP, UDP	potentially
Choi et al. [8]	Campus, ISP	Bots	Generic	C&C	DNS	advertised
Saad et al. [9]	LAN	Bots	P2P	C&C	TCP, UDP	–
Zhao et al. [10]	LAN	Bots	P2P	C&C, Attack	TCP, UDP	potentially
Zhang et al. [11]	LAN, Campus	Bots	P2P	C&C	TCP, UDP	potentially
Lu et al. [12]	LAN	Bots	IRC	C&C	TCP, UDP	potentially
Bilge et al. [13]	ISP	C&C Servers	Generic	C&C	TCP, UDP	advertised
Bilge et al. [14]	ISP	Bots, C&C Servers	Generic	C&C	DNS	advertised
Antonakakis et al. [15]	ISP	Bots, C&C Servers	Generic	C&C	DNS	potentially
Antonakakis et al. [16]	ISP	Bots, C&C Servers	Generic	C&C	DNS	potentially
Perdisci et al. [17]	ISP	Bots, C&C Servers	Generic	C&C	DNS	potentially
Tegeler et al. [18]	LAN, ISP	C&C servers	Generic	C&C	TCP, UDP	advertised
Zhao et al. [19]	ISP	Bots	P2P	C&C	DNS	–
Zhang et al. [20]	LAN, Campus	Bots	P2P	C&C	TCP, UDP	potentially
Haddadi et al. [21]	LAN	Bots	HTTP	C&C	HTTP	–
Masud et al. [22]	Client	C&C Servers	IRC	C&C, Attack	TCP	–
Shin et al. [23]	Client	Bots	Generic	C&C, Attack	TCP, UDP, DNS	–
Zeng et al. [24]	Client, LAN, Campus	Bots	Generic	C&C, Attack	TCP, UDP	–

Table 2 Traffic analysis perspective and machine-learning algorithms

Detection Method	Analysis Perspective	Supervised/ Unsupervised	MLAs
Livadas et al. [5]	Flow	S	C4.5 Tree, Naive Bayes and Bayesian Network classifiers
Strayer et al. [6]	Flow	S	C4.5 Tree, Naive Bayes and Bayesian Network classifiers
Gu et al. [7]	Client	U	Two level clustering using X-means clustering
Choi et al. [8]	DNS query/response	U	X-means clustering
Saad et al. [9]	Flow	S	SVM, ANN, Nearest Neighbours, Gaussian, and Naive Bayes classifiers
Zhao et al. [10]	Flow	S	Naive Bayes and REPTree (Reduced Error Pruning) Decision Tree
Zhang et al. [11]	Flow	U	Two level clustering using BIRCH algorithm and Hierarchical clustering
Lu et al. [12]	Flow	U	K-means, Un-merged X-means, Merged X-means clustering
Bilge et al. [13]	Flow	S	C4.5, SVM, and Random Forest classifiers
Bilge et al. [14]	DNS query/response	S	C4.5 classifier
Antonakakis et al. [15]	DNS query/response	S, U	X-Means clustering and Decision Tree using Logit-Boost strategy (LAD)
Antonakakis et al. [16]	DNS query/response	S	Random Forest classifier
Perdisci et al. [17]	Clusters of domain names	S	C4.5 classifier
Tegeler et al. [18]	Flow	U	CLUES (CLUstEring based on local Shrinking) algorithm
Zhao et al. [19]	DNS query/response	S	REPTree (Reduced Error Pruning) Decision Tree
Zhang et al. [20]	Flow	U	Two level clustering using K-means algorithm and Hierarchical clustering
Haddadi et al. [21]	Flow	S	C4.5 classifier
Masud et al. [22]	Flow	S	SVM, C4.5, Naive Bayes, Bayes Network, and Boosted decision tree classifiers
Shin et al. [23]	Flow	S	Correlation of the findings of two MLAs: SVM and One Class SVM (OCSVM)
Zeng et al. [24]	Flow	S, U	Correlation of the findings of two MLAs: Hierarchical clustering and SVM

HTTP and P2P C&C channels, respectively. It should be noted that we assume that DNS-based detection methods [14–17] can contribute to the detection of botnets independent of the used C&C communication technology.

The methods analyze different communication protocols in order to perform botnet detection. Based on the analysis TCP, UDP and DNS protocols are the most widely targeted which is reasonable as these traffic protocols cover the majority on botnet network activity. The majority of detection approaches relies on the analysis of TCP and UDP traffic while some more specifically cover IRC [5, 6] and HTTP [21] protocols as they are targeting IRC and HTTP botnets. One approach analyzes all three protocols in order to capture the majority of the botnet network activities [23].

The real-time operation is promised by only a handful of approaches [8, 13, 14, 18]. Some of the contemporary detection approaches show the potential of providing real-time detection as they operate in a time window and they could be periodically re-trained using the new training set or by periodically updating the clusters of the observation [6, 7, 10–12, 15–17, 20]. Finally, some methods such as [14] have proved their ability of real-time operation through a real-world operational deployment.

As illustrated by Table 2, the existing techniques use a variety of machine learning algorithms deployed in diverse setups. In total 15 different MLAs were considered by the analyzed approaches. Supervised and unsupervised MLAs are evenly represented in the analyzed methods. Some of the authors experimented with more than one MLA providing the good insight on how the assumed heuristics hold in different learning scenarios as well as what are the performance of different MLAs [5, 6, 9, 10, 12, 13, 22]. Additionally, some authors used MLAs in more advanced setups, where clustering of observation is realized through two level clustering schemes [7, 11, 20] or where the findings of independent MLAs were correlated in order to pinpoint the malicious traffic pattern [7, 15, 23, 24]. Several authors used the same MLAs within their detection systems [5, 6, 9, 10, 13, 14, 16, 17, 22] indicating some of the well performing clustering and classification algorithms, such as Decision Tree based classifiers and X-means clustering.

The existing methods use several perspectives of traffic analysis. The approaches that analyze TCP and UDP traffic generally analyze it from the perspective of traffic "*flows*". It should be noted that definition of a flow varies from the approach to the approach so some use NetFlow flows [13, 18, 24] while others use a conventional definition of traffic flows where a flow is defined as traffic on a certain 5-tuple i.e. $<ip_{src}$, $port_{src}$, ip_{dst}, $port_{dst}$, $protocol>$. Furthermore, some approaches consider bi-directional

flows in order to capture the differences in incoming and outgoing traffic [10]. DNS-based detection approaches commonly analyze DNS traffic from the perspective of DNS query responses (i.e. domain names and their resolving IPs) [8, 14–16, 19], while some analyze it from the perspective of domain clusters [17].

Traffic instances are represented as sets of traffic features in MLAs. As already indicated, feature selection is a challenging task as the feature set should capture targeted characteristics of malicious traffic. The analyzed detection approaches greatly vary in employed feature representation. The TCP/UDP based approaches addressed by the survey use features that are are generally independent from the payload content, relying on the information that can be gathered from packets headers as well as different traffic statistics. Several techniques [7, 12, 21, 22] rely on the content of payloads thus being easily defeated by the encryption or the obfuscation of the packet payload. Furthermore, some approaches rely on IP addresses as features [9, 10] opening the possibility of introducing bias in the evaluation of the detection performance. In the case of DNS analysis approaches typically rely on information extracted from the DNS query responses, such as: lexical domain name features, IP-based features, geo-location features, etc.

4.2 Detection Performance

The analysis of the performance of the methods is illustrated in Table 3, by providing a brief overview of evaluation practice and data sets used within the approaches as well as reported performance for analyzed detection methods. However, it should noted that the results presented in the table represent the bottom range of reported detection performance and that some of the approaches are able to provide better results for specific botnet samples, traffic trace, etc. Additionally, the methods should not be directly compared based on the reported performance alone, as they used different evaluation practices and testing data sets. However, the presented performance can still indicate the overall capabilities of the particular approach in identifying botnet traffic or bots.

Due to the challenges of obtaining training and testing data, the evaluation of the proposed botnet detection systems is one of the most challenging tasks within the development of detection methods [31]. As illustrated in Table 3 the labeled data sets used for development and evaluation of the approaches were obtained through all four scenarios, presented in Section 3. The background data is obtained at the point in the network

Table 3 Evaluation methodology and achieved detection performance

Detection Method	Evaluation Data Sets	Background Data Sets	Number of Botnet Families/ Samples	Detection Performance
Livadas et al. [5]	Scenario 3	Campus	1/1	Flow classification: FPR (10–20%), FNR (30–40%)
Strayer et al. [6]	Scenario 3	Campus	1/1	Flow classification: FPR (<30%), FNR (>2.17%) Bot detection: TPR (90%)
Gu et al. [7]	Scenarios 2, 3, 4	Campus	7/8	Bot detection: TPR (75%–100%), FPR (<1%)
Choi et al. [8]	Scenario 1	Campus, ISP	*NA*	Domain classification: TPR (>95.4%), FPR (<0.32%)
Saad et al. [9]	Scenario 2	LAN	2/2	Flow classification: TPR (>89%), error (<20%)
Zhao et al. [10]	Scenario 2	LAN	2/2	Flow classification: TPR (>98.1%), FPR (<2.1%)
Zhang et al. [11]	Scenario 1, 4	LAN, Campus	2/2	Bot detection: TPR (100%), FPR (<0.2%)
Lu et al. [12]	Scenarios 2, 4	ISP	2/2	Flow classification: TPR (>95%)
Bilge et al. [13]	Scenario 1	Campus, ISP	*NA*	C&C server classification: TPR (>64.3%), FPR (<1%)
Bilge et al. [14]	Scenario 1	ISP	*NA*	Domain classification: TPR (>98.4%), FPR (<1.1%)
Antonakakis et al. [15]	Scenario 1, 4	ISP	*NA*	Domain classification: TPR (>96.8%), FPR (<0.38%)
Antonakakis et al. [16]	Scenario 1	ISP	*NA*	Domain classification: TPR (>98.1%), FPR (<1.1%)
Perdisci et al. [17]	Scenario 1	ISP	*NA*	Domain clusters classification: TPR (>99.3%), FPR (<0.15%)
Tegeler et al. [18]	Scenario 4	LAN, ISP	6/188	Bot detection: TPR (49%–100%)
Zhao et al. [19]	Scenario 1	ISP	2/2	Domain classification: TPR (100%), FPR (0.5%)
Zhang et al. [20]	Scenarios 1, 4	LAN, Campus	2/2	Bot detection: TPR (100%), FPR (<0.4%)
Haddadi et al. [21]	Scenarios 1, 3	LAN	3/6	Flow classification: TPR (>84%), FPR (<10%)
Masud et al. [22]	Scenario 3	LAN	2/2	Flow classification: accuracy (>95.2%), FPR (<3.2%)
Shin et al. [23]	Scenario 4	LAN	15/15	Bot detection: TPR (100%), FPR (<0.68%)
Zeng et al. [24]	Scenario 3, 4	LAN, Campus	5/6	Bot detection: FPR (<0.16%), FNR (<12.5%)

Comment: NA – not available values

corresponding to the monitoring point the methods are developed for, most commonly on campus or LAN networks. A number of approaches obtained evaluation data sets by relying on Scenario 1 [8, 11, 13–17, 19–21]. The majority of these approaches perform DNS traffic analysis so they used domain/IP blacklists and whitelists of popular domains [8, 14–17, 19, 21] for performing labeling while others rely on commercial IDS for doing the labeling [11, 13, 20]. The rest of the approaches relied on other three scenarios of obtaining the evaluation data, where Scenario 2 was used by only 4 approaches, indicating that the researcher needed to run either malware code (Scenario 3) or malware binaries (Scenario 4) in order to obtain malicious network traces.

Furthermore, the malicious traffic samples are usually recorded for a limited number of bot samples. For instance, the performance of only five detection approaches were evaluated on the traffic traces produced by more than 5 bot samples [7, 18, 21, 23, 24], while the maximal number of samples used for evaluation was 188 in case of [18]. The rest of the methods were tested with less than 4 bot malware samples. Finally, the diversity of the used malware samples is poor as the majority of the analyzed approaches rely on less than 3 distinct families of botnets. It should be noted that for DNS-based approaches the number of botnet families that contributed to DNS traffic contained in evaluation data sets is commonly unknown.

The performance reported by the analyzed detection methods indicate a great perspective in identifying botnet traffic and bots using MLAs. Several detection methods indicate TPR of 100% and overall low FPR [11, 19, 20]. Furthermore, a number of approaches is characterized with a FPR less than 1%. These results indicate the possibility of using some of the approaches in real-world operational networks.

4.3 Vulnerability to Evasion Techniques

Table 4 illustrates how different approaches cope against evasion techniques presented in Section 3, by indicating the strength of the indication (SF – strong factor and WF – weak factor) of methods being evaded by them. However, it should be noted that the indications given by Table 4 should be used more as guidelines than as precise measures.

As illustrated in Table 4, the proposed approaches are more or less vulnerable on different evasion techniques. Generally, the majority of the analyzed methods are resistant to evasion by encryption of botnet traffic. Only four approaches [7, 12, 21, 22] that rely on features extracted from packet payload are vulnerable on this evasion strategy. However, the majority

Table 4 An overview of evasion tactics for detection methods

	Evasion Tactics							
Detection Method	ET1	ET2	ET3	ET4	ET5	ET6	ET7	ET8
Livadas et al. [5]	–	–	SF	SF	–	–	–	–
Strayer et al. [6]	–	–	SF	SF	–	–	SF	–
Gu et al. [7]	–	WF	WF	SF	SF	WF	SF	–
Choi et al. [8]	–	–	WF	–	–	–	SF	SF
Saad et al. [9]	–	–	SF	SF	–	–	–	SF
Zhao et al. [10]	–	–	SF	SF	SF	WF	–	SF
Zhang et al. [11]	–	–	SF	SF	–	WF	–	SF
Lu et al. [12]	–	SF	SF	SF	–	–	–	SF
Bilge et al. [13]	–	–	SF	SF	–	–	SF	SF
Bilge et al. [14]	–	–	SF	–	–	–	SF	–
Antonakakis et al. [15]	–	–	WF	–	–	–	SF	–
Antonakakis et al. [16]	–	–	WF	–	–	–	SF	–
Perdisci et al. [17]	–	–	SF	–	–	–	SF	–
Tegeler et al. [18]	–	–	SF	SF	–	WF	SF	WF
Zhao et al. [19]	–	–	SF	–	–	–	SF	WF
Zhang et al. [20]	–	–	SF	SF	–	–	–	SF
Haddadi et al. [21]	–	SF	SF	SF	–	–	–	SF
Masud et al. [22]	SF	SF	SF	SF	–	–	–	–
Shin et al. [23]	SF	–	SF	SF	WF	–	–	WF
Zeng et al. [24]	SF	–	SF	SF	–	WF	SF	–

Comment: SF – strong factor, WF – weak factor

of the techniques are vulnerable on evasion by flow perturbation, due to the fact that they analyze traffic on the flow level. Furthermore, all the techniques are more or less vulnerable on time-based evasion, especially the ones that promise the real-time operation. Finally, the analyzed client-based [22–24] techniques are vulnerable on evasion techniques that target the monitoring of the internals of a host computer.

This study does not address the complexities of evasion techniques and their effect on the overall utility of the botnet. The future work could be directed at more thorough analysis of evasion techniques and vulnerability of modern detection systems to them, covering effect of evasion techniques to both detection systems and overall utility of the botnet.

5 Discussion

In this section we elaborate on the findings of the analysis presented in the previous section outlining several challenges of using MLAs for identifying botnet network activity.

5.1 Principles of Traffic Analysis

As shown in Section 3 detection methods target botnets from different monitoring points where the majority of the approaches are implemented at local and campus/enterprise networks. This can be explained due to several reasons. First, MLAs are data-driven and they depend on available evaluation data sets that are mostly formed by capturing botnet network activity at local networks. Second, the analysis of traffic in core and ISP networks assumes the ability of processing a substantial amount of traffic which can be a challenge for some of the approaches. However, it should be noted that depending on the monitoring point different portions of a botnet can be seen and consequently different malicious traffic characteristics can be targeted. Therefore, detection methods can often be seen as complementary solutions to the botnet detection problem rather than competing solutions.

The analyzed detection methods can either target all botnets types or a specific type based on their C&C communication technology. Furthermore, the approaches could capture different botnet operational phases. The idea of targeting specific type of botnets is to achieve better detection performances which has proved to be true by a number of studies [11, 20]. However, these more specific detection approaches should be used in combination with other detection approaches in order to provide effective and hard to evade detection.

The analyzed detection approaches most commonly target TCP, UDP and DNS protocols using different traffic detection perspectives. We would like to stress the importance of the analysis perspective as choosing the analysis perspective is important to the practical use of machine-learning based approaches. The used analysis perspective should encompass the nature of the targeted phenomena and should carry the context that would be understandable to the operator of the system. A suitable example would be DNS traffic analysis, where DNS traffic could be analyzed from the perspective of either domain names or more complex domains-to-IPs mappings. The former implies that for each domain name a number of features is extracted and MLAs is used to identify if domain is malicious or not. The latter extracts domains-to-IPs mappings i.e. mappings between queried domain names and resolving IP addresses and determines if the mappings are malicious or not. The difference between the two scenarios is significant. Analyzing DNS from the perspective of domain names would segment traffic into much larger sets of instances than in the case of domains-to-IPs mappings. Furthermore, in the first case the result of detection would bring only limited information to the operator, i.e. only

detection result. Analyzing DNS traffic from the perspectives of domains-to-IPs mappings would, on the other hand, yield more descriptive detection results as the identified mappings bring more information to the system operator then the simple classification of domain names. Furthermore, domains-to-IPs perspective encompasses the characteristics of IP- and Domain-flux strategies that are often used by the cyber criminals.

As presented in Table 2 different MLAs have been used as a tool for identifying botnets. Some of the most popular MLAs are decision tree classifiers (C4.5, Random Forests, REPTree) for classification and Hierarchical clustering and X-means clustering for grouping traffic observations. This does not come as surprise as these algorithms have showed their capabilities in network traffic analysis and classification of Internet traffic over the last decade [60]. Furthermore, as indicated in the previous section detection approaches use diverse traffic features for representing traffic instances within their detection algorithms. However, using features that would introduce bias in the detection such as IP addresses should be avoided. Also the use of features based on packet payloads should be avoided due to possible evasion by encryption and the violation of the privacy of end-users.

5.2 Evaluation Challenge

The main challenge in evaluating the analyzed detection methods is obtaining reliable evaluation data sets that would successfully capture a substantial amount of both malicious and non-malicious traffic instances. As illustrated in the previous section, existing studies have used different strategies for obtaining the ground truth on botnet network activity, using all scenarios outlined in the Section 3. However, each of the scenarios comes with drawbacks that should be thoroughly understood. The main drawback of Scenario 1 is inherited from the imperfections of data set labeling techniques. Labeling by relying on domain and IP blacklists and signature-based IDSs has its drawbacks that could lead to unreliable ground truth [61–64]. Although many authors [65] have tried to solve this problem proposing different strategies of eliminating false positives in the process of labeling, the problem remains largely unsolved. The scenarios that rely on merging the malicious and non-malicious traffic traces suffer from the pitfall of artificially merging diverse traces. The technique of merging should not introduce additional traffic anomalies that would lead to a biased detection. Furthermore, the non-malicious background traffic used for forming the evaluation data sets should

be obtained at the point of traffic monitoring corresponding to the monitoring point at which the botnet trace was recorded.

Based on the analysis the majority of the detection approaches were evaluated using traffic from a modest number of botnets. This could be explained due to many legal, ethical and practical limitations of obtaining the botnet traffic traces. However, the small number of used bot samples indicates the need for more thorough testing where more comprehensive set of malware samples would be used, in order to prove that the detection system is able to generalize inferred botnet knowledge to previously unseen botnets.

5.3 Cost of Errors

The evaluated botnet detection methods report overall promising performances of identifying malicious traffic. However, the majority of the approaches are characterized with a substantial number of FP and FN. We would like to elaborate on the cost of these errors by following Sommer et al. [30] analysis of the cost of errors on intrusion detection use case. As in the case of any other anomaly detection system botnet detection systems are sensitive to number of FP. The high number of FP positives can easily deem the detection method unusable from the perspective of the use in operational networks. Furthermore, depending on the use case botnet detection can have a high cost of FN as any compromised machine within the operational network could cause a lot of technical and financial damage. Therefore, an optimal detection approach would have FP close to zero and minimized number of FN. However, based on the analysis presented in Section 4 only a few detection methods could potentially fulfill this requirement opening the space for further improvements.

5.4 Opportunities for Future Work

Future work should be devoted to the development of detection methods that would find their use within operational networks. These methods should rely on the principles of network traffic analysis that would encompass targeted botnet network characteristics and would carry the context that is understandable to the operator of the system. Furthermore, one of the important goals of future detection systems is to operate in real-time thus facilitating timely detection. The future methods should be evaluated using an extensive set of network traces originating from different types of botnets. Finally, special attention should be placed on minimizing the number of errors in identifying

botnet network traffic so the proposed methods would performance-wise be suitable for being used in operational networks.

6 Conclusion

The use of machine learning algorithms (MLAs) for identifying botnet network traffic has been the subject of interest within the research community during the last decade resulting in numerous detection methods. The contemporary detection methods are based on different principles of traffic analysis, they target diverse traits of botnet network activity using a variety of machine learning algorithms and they consequently provide varying performance of detection. This paper outlines the opportunities and challenges of using MLAs for identifying botnet network activity and presents the review of the most prominent contemporary botnet detection methods based on MLAs. The presented study covers 20 detection methods, proposed over the last decade. The methods have been analyzed by investigating principles of their operation, the used evaluation procedures, obtained performance and vulnerabilities on evasion techniques. The analysis indicates a great potential of this class of approaches to be used for identifying botnet network traffic. However, the study also indicates some of the challenges of using MLAs in the context of network-based botnet detection that should be thoroughly understood in order for this class of detection methods to be effectively used.

References

[1] Hogben, G. (ed.), "Botnets: Detection, measurement, disinfection and defence," ENISA, Tech. Rep., 2011.

[2] S. S. Silva, R. M. Silva, R. C. Pinto, and R. M. Salles, "Botnets: A survey," *Computer Networks*, vol. 1, no. 0, pp. –, 2012.

[3] S. García, A. Zunino, and M. Campo, "Survey on network-based botnet detection methods," *Security and Communication Networks*, vol. 7, no. 5, pp. 878–903, 2014.

[4] A. Karim, R. B. Salleh, M. Shiraz, S. A. A. Shah, I. Awan, and N. B. Anuar, "Botnet detection techniques: Review, future trends, and issues," *Journal of Zhejiang University SCIENCE C*, vol. 15, no. 11, pp. 943–983, 2014.

[5] C. Livadas, R. Walsh, D. Lapsley, and W. T. Strayer, "Using machine learning techniques to identify botnet traffic," in *Proceedings of*

2006 31st IEEE Conference on Local Computer Networks, Nov. 2006, pp. 967–974.

[6] W. T. Strayer, D. Lapsely, R. Walsh, and C. Livadas, "Botnet detection based on network behaviour," in *Botnet Detection*, ser. Advances in Information Security. Springer, 2008, vol. 36, pp. 1–24.

[7] G. Gu, R. Perdisci, J. Zhang, and 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. 139–154.

[8] H. Choi and H. Lee, "Identifying botnets by capturing group activities in dns traffic," *Computer Networks*, vol. 56, no. 1, pp. 20–33, 2012.

[9] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, W. Lu, J. Felix, and P. Hakimian, "Detecting p2p botnets through network behavior analysis and machine learning," in *2011 Ninth Annual International Conference on Privacy, Security and Trust (PST)*, July 2011, pp. 174–180.

[10] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani, and D. Garant, "Botnet detection based on traffic behavior analysis and flow intervals," *Computers & Security*, vol. 39, pp. 2–16, 2013.

[11] J. Zhang, R. Perdisci, W. Lee, U. Sarfraz, and 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, Jun. 2011, pp. 121–132.

[12] W. Lu, G. Rammidi, and A. A. Ghorbani, "Clustering botnet communication traffic based on n-gram feature selection," *Computer Communications*, vol. 34, pp. 502–514, 2011.

[13] L. Bilge, D. Balzarotti, W. Robertson, E. Kirda, and C. Kruegel, "Disclosure: Detecting botnet command and control servers through large-scale netflow analysis," in *Proceedings of the 28th Annual Computer Security Applications Conference*, ser. ACSAC '12. ACM, 2012, pp. 129–138.

[14] L. Bilge, S. Sen, D. Balzarotti, E. Kirda, and C. Kruegel, "Exposure: A passive dns analysis service to detect and report malicious domains," *ACM Transactions on Information and System Security (TISSEC)*, vol. 16, no. 4, p. 14, 2014.

[15] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster, "Building a dynamic reputation system for dns," in *Proceedings of the 19th USENIX conference on Security*, ser. USENIX Security'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 18–18.

[16] M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou II, and D. Dagon, "Detecting malware domains at the upper dns hierarchy." in *USENIX Security Symposium*, 2011, p. 16.

[17] R. Perdisci, I. Corona, and G. Giacinto, "Early detection of malicious flux networks via large-scale passive dns traffic analysis," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 5, pp. 714–726, 2012.

[18] F. Tegeler, X. Fu, G. Vigna, and C. Kruegel, "Botfinder: Finding bots in network traffic without deep packet inspection," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. ACM, 2012, pp. 349–360.

[19] D. Zhao and I. Traore, "P2p botnet detection through malicious fast flux network identification," in *2012 IEEE Seventh International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 2012, pp. 170–175.

[20] J. Zhang, R. Perdisci, W. Lee, X. Luo, and U. Sarfraz, "Building a scalable system for stealthy p2p-botnet detection," *IEEE Transactions on Information Forensics and Security,* vol. 9, no. 1, pp. 27–38, 2014.

[21] F. Haddadi, D. Runkel, A. N. Zincir-Heywood, and M. I. Heywood, "On botnet behaviour analysis using gp and c4.5," in *Proceedings of the 2014 conference companion on Genetic and evolutionary computation companion*. ACM, 2014, pp. 1253–1260.

[22] M. Masud, T. Al-khateeb, L. Khan, B. Thuraisingham, and K. Hamlen, "Flow-based identification of botnet traffic by mining multiple log files," in *First International Conference on Distributed Framework and Applications, 2008. DFmA 2008*, Oct. 2008, pp. 200–206.

[23] S. Shin, Z. Xu, and G. Gu, "EFFORT: Efficient and Effective Bot Malware Detection," in *Proceedings of the 31th Annual IEEE Conference on Computer Communications (INFOCOM'12) Mini-Conference*, March 2012, pp. 71–80.

[24] Y. Zeng, X. Hu, and K. Shin, "Detection of botnets using combined host- and network-level information," in *2010 IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 28 2010-July 1 2010, pp. 291–300.

[25] M. Feily and Shahrestani, "A survey of botnet and botnet detection," *Third International Conference on Emerging Security Information, Systems and Technologies, 2009. SECURWARE '09*, pp. 268–273, 2009.

[26] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, "A survey of botnet technology and defenses," in *Conference For Homeland Security,*

2009. CATCH '09. Cybersecurity Applications Technology, March 2009, pp. 299–304.

[27] T. Hyslip and J. Pittman, "A survey of botnet detection techniques by command and control infrastructure," *Journal of Digital Forensics, Security and Law*, vol. 10, no. 1, pp. 7–26, 2015.

[28] M. Masud, L. Khan, and B. Thuraisingham, *Data Mining Tools for Malware Detection*. Taylor & Francis Group, 2011.

[29] S. Dua and X. Du, *Data mining and machine learning in cybersecurity*. Boca Raton, FL: CRC Press. xxii, 234 p. $ 89.95, 2011.

[30] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *2010 IEEE Symposium on Security and Privacy (SP)*, IEEE, 2010, pp. 305–316.

[31] A. J. Aviv and A. Haeberlen, "Challenges in experimenting with botnet detection systems," in *Proceedings of the 4th conference on Cyber security experimentation and test*, ser. CSET'11. Berkeley, CA, USA: USENIX Association, 2011, pp. 6–6.

[32] M. Bailey, J. Oberheide, J. Andersen, Z. M. Mao, F. Jahanian, and J. Nazario, "Automated classification and analysis of internet malware." in *RAID*, ser. Lecture Notes in Computer Science, C. KrÃijgel, R. Lippmann, and A. Clark, Eds., vol. 4637. Springer, 2007, pp. 178–197.

[33] E. Stinson and J. C. Mitchell, "Characterizing bots' remote control behavior," in *Botnet Detection*, ser. Advances in Information Security, W. Lee, C. Wang, and D. Dagon, Eds. Springer, 2008, vol. 36, pp. 45–64.

[34] L. Liu, S. Chen, G. Yan, and Z. Zhang, "Bottracer: Execution-based bot-like malware detection," in *Proceedings of the 11th international conference on Information Security*, ser. ISC '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 97–113.

[35] C. Kolbitsch, P. M. Comparetti, C. Kruegel, E. Kirda, X. Zhou, and X. Wang, "Effective and efficient malware detection at the end host," in *Proceedings of the 18th conference on USENIX security symposium*, ser. SSYM'09. Berkeley, CA, USA: USENIX Association, 2009, pp. 351–366.

[36] U. Bayer, P. M. Comparetti, C. Hlauschek, C. KrÃijgel, and E. Kirda, "Scalable, behavior-based malware clustering." in *NDSS*. The Internet Society, 2009, pp. 5–5.

[37] Y. Park, Q. Zhang, D. Reeves, and V. Mulukutla, "Antibot: Clustering common semantic patterns for bot detection," in *2010 IEEE 34th*

Annual Proceedings on Computer Software and Applications Conference (COMPSAC), July 2010, pp. 262–272.

[38] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic malware-analysis techniques and tools," *ACM Comput. Surv.*, vol. 44, no. 2, pp. 6:1–6:42, Mar. 2008.

[39] I. You and K. Yim, "Malware obfuscation techniques: A brief survey," in *2010 International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA)*, Nov. 2010, pp. 297–300.

[40] J. Marpaung, M. Sain, and H.-J. Lee, "Survey on malware evasion techniques: State of the art and challenges," in *2012 14th International Conference on Advanced Communication Technology (ICACT)*, Feb. 2012, pp. 744–749.

[41] Damballa, "A new iteration of the tdss/tdl4 malware using dga-based command-and-control," Damballa, Tech. Rep., 2012.

[42] A. Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale botnet detection and characterization," in *Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, ser. HotBots'07. Berkeley, CA, USA: USENIX Association, 2007, pp. 7–7.

[43] D. Dagon, C. Zou, and W. Lee, "Modeling botnet propagation using time zones," in *Proceedings of the 13 th Network and Distributed System Security Symposium NDSS*, 2006, pp. 7–7.

[44] G. Gu, V. Yegneswaran, P. Porras, J. Stoll, and W. Lee, "Active botnet probing to identify obscure command and control channels," in *Proceedings of the 2009 Annual Computer Security Applications Conference*, ser. ACSAC '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 241–253.

[45] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "BotHunter: Detecting malware infection through IDS-driven dialog correlation," in *Proceedings of the 16th USENIX Security Symposium, San Jose, California.* USENIX Association, Jul. 2007, pp. 167–182.

[46] V. Paxson, "Bro: A system for detecting network intruders in real-time," *Computer Networks*, vol. 31, no. 23Ãćâ Ćňâ ĂIJ24, pp. 2435–2463, 1999.

[47] M. Roesch, "Snort - lightweight intrusion detection for networks," in *Proceedings of the 13th USENIX conference on System administration*, ser. LISA '99. Berkeley, CA, USA: USENIX Association, 1999, pp. 229–238.

[48] J. Goebel and T. Holz, "Rishi: Identify bot contaminated hosts by irc nickname evaluation," in *Proceedings of the first conference on First*

Workshop on Hot Topics in Understanding Botnets, ser. HotBots'07. Berkeley, CA, USA: USENIX Association, 2007, pp. 8–8.

[49] G. Gu, J. Zhang, and W. Lee, "BotSniffer: Detecting botnet command and control channels in network traffic," in *Proceedings of the 15th Annual Network and Distributed System Security Symposium (NDSS '08)*, February 2008, pp. 1–1.

[50] A. Ramachandran, N. Feamster, and D. Dagon, "Revealing botnet membership using dnsbl counter-intelligence," in *Proceedings of the 2nd conference on Steps to Reducing Unwanted Traffic on the Internet - Volume 2*, ser. SRUTI'06. Berkeley, CA, USA: USENIX Association, 2006, pp. 8–8.

[51] R. Villamarin-Salomon and J. Brustoloni, "Identifying botnets using anomaly detection techniques applied to DNS traffic," in *Proceedings of 5th IEEE Consumer Communications and Networking Conference (CCNC 2008)*, 2008, pp. 476–481.

[52] R. Villamarín-Salomón and J. C. Brustoloni, "Bayesian bot detection based on dns traffic similarity," in *Proceedings of the 2009 ACM symposium on Applied Computing*, ser. SAC '09. New York, NY, USA: ACM, 2009, pp. 2035–2041.

[53] X. Yu, X. Dong, G. Yu, Y. Qin, D. Yue, and Y. Zhao, "Online botnet detection based on incremental discrete fourier transform," *JNW*, vol. 5, no. 5, pp. 568–576, 2010.

[54] T. M. Mitchell, *Machine Learning*, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 1997.

[55] S. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Frontiers in Artificial Intelligence and Applications*, vol. 160, p. 3, 2007.

[56] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, Sep. 1999.

[57] Suricata, IDS, "open-source ids/ips/nsm engine," 2015.

[58] N. Provos and T. Holz, *Virtual honeypots: From botnet tracking to intrusion detection*, 2nd ed. Addison-Wesley Professional, 2009.

[59] E. Stinson and J. C. Mitchell, "Towards systematic evaluation of the evadability of bot/botnet detection methods," in *Proceedings of the 2nd conference on USENIX Workshop on offensive technologies*, ser. WOOT'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 5:1–5:9.

[60] T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

[61] M. Kührer, C. Rossow, and T. Holz, "Paint it black: Evaluating the effectiveness of malware blacklists," in *Research in Attacks, Intrusions and Defenses*. Springer, 2014, pp. 1–21.

[62] C. J. Dietrich and C. Rossow, "Empirical research of ip blacklists," in *ISSE 2008 Securing Electronic Business Processes*. Springer, 2009, pp. 163–171.

[63] S. Sinha, M. Bailey, and F. Jahanian, "Shades of grey: On the effectiveness of reputation-based âĂIJblacklistsâĂİ," in *3rd IEEE International Conference on Malicious and Unwanted Software, MALWARE 2008.* 2008, pp. 57–64.

[64] S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang, "An empirical analysis of phishing blacklists," in *Sixth Conference on Email and Anti-Spam (CEAS)*, 2009.

[65] N. Kheir, F. Tran, P. Caron, and N. Deschamps, "Mentor: Positive dns reputation to skim-off benign domains in botnet c&c blacklists," in *ICT Systems Security and Privacy Protection.* Springer, 2014, pp. 1–14.

Biographies

M. Stevanovic received the M.Sc. in Electrical Engineering in 2011, from the Faculty of Electrical Engineering, Belgrade University, specializing in system engineering. He is currently a Ph.D. Student in the Wireless Communication Section, Department of Electronic Systems, Aalborg University. His research interests include network security, traffic anomaly detection and malware detection based on network traffic analysis.

J. M. Pedersen received the M.Sc. in Mathematics and Computer Science in 2002, and the Ph.D. in Electrical Engineering in 2005 from Aalborg University, Denmark. He is currently Associate Professor at the Wireless Communication Section, Department of Electronic Systems, Aalborg University. His research interests include network planning, traffic monitoring, and network security. He is author/co-author of more than 70 publications in international conferences and journals, and has participated in Danish, Nordic and European funded research projects. He is also board member of a number of companies within technology and innovation.