DNS Traffic Analysis for Botnet Detection

Monika Wielogorska, Darragh O'Brien

School of Computing, Dublin City University, Glasnevin, Dublin 9
{monika.wielogorska2,darragh.obrien}@dcu.ie

Abstract. Botnets pose a major threat to cyber security. Given that firewalls typically prevent unsolicited incoming traffic from reaching hosts internal to the local area network, it is up to each bot to initiate a connection with its remote Command and Control (C&C) server. To perform this task a bot can use either a hardcoded IP address or perform a DNS lookup for a predefined or algorithmically-generated domain name. Modern malware increasingly utilizes DNS to enhance the overall availability and reliability of the C&C communication channel. In this paper we present a prototype botnet detection system that leverages passive DNS traffic analysis to detect a botnet's presence in a local area network. A naive Bayes classifier is trained on features extracted from both benign and malicious DNS traffic traces and its performance is evaluated. Since the proposed method relies on DNS traffic, it permits the early detection of bots on the network. In addition, the method does not depend on the number of bots operating in the local network and is effective when only a small number of infected machines are present.

1 Introduction

A botnet is a distributed network of compromised machines, called bots, that can be remotely controlled by an attacker without their owners' knowledge or consent [6, 15, 24]. The essential mechanism that allows the attacker, commonly referred to as a *botherder* or *botmaster*, to direct the actions of a botnet is the Command and Control (C&C) server. Through the C&C server the botherder issues commands to and receives responses from individual bots. Botnets have been used to support a range of malicious activities including: DDoS attacks, keylogging, propagating spam and implementing ransomware [8, 9, 19]. A recently discovered botnet, Mirai [16], infected over 2.5 million *Internet of Things* devices and demonstrates that botnets continue to constitute a major threat to cyber security.

Given the significant threat posed by botnets, numerous methods for their detection have been proposed. As more effective botnet detection and mitigation approaches are developed, botnet designers respond by employing new evasion techniques. Since effective operation of the botnet depends on the availability of the C&C server and the reliability of the corresponding communication channel, attackers expend significant effort in protecting the identity of C&C server, as well as concealing C&C communication among legitimate traffic.

In this paper we propose and test a classifier to distinguish between benign and malicious DNS traffic streams. The method is employed to detect botnet-infected networks and relies on features extracted from DNS traffic. Malicious traffic captures are generated using from real-world malware samples while benign traffic captures are created through network interaction with popular Internet domains.

The paper is organised as follows. In Section 2 background and related work are presented. In Section 3 the proposed approach is described. Section 4 presents implementation details, describes our data generation approach and reports on an experiment to test the effectiveness of the proposed method. Conclusions and future work are presented in Section 5.

2 Background and related work

The most common botnet architecture, given its straightforward implementation, is *centralized*. Under this design, bots periodically initiate a connection with one or more C&C servers in order to download commands and upload results. Analysis of over 45,000 malware samples in a dynamic analysis environment over a 12-month period revealed that 92% of samples generated DNS traffic [22]. Like other forms of malware, bots typically rely on DNS to discover IP addresses of corresponding C&C servers.

DNS-based evasion strategies employed by modern-day botnets include *Fast-Flux Service Networks* [18] (both *Single* and *Double Flux* varieties) and *Domain Generation Algorithm* (DGA)-based techniques [27, 1]. The primary advantage of DNS-based evasion techniques is their flexibility: they permit dynamic updates to C&C infrastructure thereby often delaying detection and take-down attempts. DNS and related evasion techniques are described below. There follows a brief summary of previously reported approaches to the problem of botnet detection.

2.1 Domain Name System (DNS)

DNS is a hierarchical naming system for computers, services, or any other resource connected to the Internet [17]. A Domain Name Service translates *domain names* into *IP addresses*. DNS makes it possible for Internet users to locate network resources independently of their actual physical location or changing IP address. Since DNS provides a flexible mapping between domain names and IP addresses and is universally available, it is frequently employed by bot creators to hide the location of C&C servers and to prevent or inhibit their take-down.

2.2 Evasion techniques

A Fast-Flux Service Network (FFSN) [20, 26, 13] is an evasion technique providing functionality similar to *Round Robin DNS* (RRDNS) [3]. In essence, an FFSN allows for one domain name to have an unlimited number of IP addresses, having short Time-To-Live (TTL) values and rotating in a round robin fashion.

IP addresses belonging to such a domain consist of members of the same botnet, acting as a proxy for any device attempting a connection with their respective C&C server. The technique increases overall botnet resilience by hiding the real identity and location of the underlying C&C server.

An FFSN variant, the *Double Flux* [23] network provides further protection by introducing an additional layer of redundancy whereby bots serve not only as a web proxy for their respective C&C server, but also as a *name server* for the corresponding domain. As a result, both IP addresses in DNS *A records* and name servers in DNS *NS records* are short-lived and frequently changed.

A further DGA-based [27] technique has also been deployed by malware developers to bolster the resilience of their C&C infrastructure. Specifically, when malware attempts to initiate a connection with its C&C server it uses an in-built randomly-seeded domain generation algorithm to derive a set of possible domain names to connect to. When a botmaster seeks to issue commands to or receive responses from botnet members, the same generation algorithm is used to derive the same set of domain names. Only one is registered however and it allows bots to establish a connection with the C&C server. Generated domains may be short-lived in order to minimise exposure of C&C servers [10].

2.3 Botnet detection and mitigation

DNS traffic analysis is an appealing approach to network-based botnet detection [25, 4, 28] for the following reasons: DNS is used by the majority of malware to map domain names to IP addresses of corresponding C&C servers; DNS lookups are performed prior to initiating a connection with a C&C server, allowing for botnet detection in its *earliest stages* and potentially before significant damage occurs. Below we survey several DNS-based botnet detection approaches that have been reported in the literature.

Anomaly-based traffic analysis at the ISP level: EXPOSURE [2] is a detection system operating at the Internet Service Provider (ISP) level and capable of large-scale passive DNS traffic analysis with the aim of detecting malicious domains. Based on several months of analysis of both benign and malicious domains, the authors identified four feature sets, containing 15 distinctive features that could signal a domain with malicious intent [7]:

- *Time-based features*: Short life, daily similarity (in terms of number of requests per time of day), repeating patterns and access ratio (popular versus idle domain).
- *DNS answer-based features*: Number of distinct IP addresses, number of distinct countries, number of domains sharing the IP address, reverse DNS query results.
- *TTL-based features*: Average TTL, TTL standard deviation, number of distinct TTL values, number of TTL changes, percentage usage of specific TTL values.

– *Domain name-based features*: Percentage of numerical characters in a domain name, percentage of the length of LMS.

The main advantage of *EXPOSURE* is its operational level: Being deployed at the ISP level it has access to large volumes of DNS traffic originating from multiple locations, allowing it to inspect a substantial volume of requests for the same domain. Given its location it can analyse features that would be otherwise unavailable such as *daily similarity* and *access ratio*. One limitation of *EXPOSURE*, however, is its inability to identify specific infected hosts on the network from which the suspicious DNS requests originate.

Machine learning at the local area network level: The *BotGAD* system [5] employs machine learning techniques to identify malicious domains. BotGAD is an anomaly-based, light-weight botnet detection technique that relies on DNS traffic analysis. The designers base their approach on an inherent characteristics of botnets: *group activity*. The underlying assumption is that bots can be detected through analysis of their actions as a co-ordinated group. These actions may include simultaneously resolving a domain name or initiating a TCP connection to the same IP address.

BotGAD runs at the local network level meaning it can pinpoint the IP addresses of infected hosts. However this detection strategy has certain limitations. It requires a given number of hosts to be infected in order to be effective. If the number of infected hosts is low, analysis of their DNS traffic may not trigger any alerts.

Fast-flux service network detection: Some researchers have focused their efforts on detecting FFSNs. *FluXOR* [20] is a system designed to detect and track FFSNs. In contrast to the proposed approach described in Section 3, FluXOR uses *active* probing techniques to detect whether a given domain is part of an FFSN. Each suspect domain is monitored over a period of time, during which it is actively queried. FluxOR's aim is not only to discover FFSN domains, but also to detect the number and identity of associated proxy servers in order to prevent their reuse in a future FFSN. In [21] the authors propose a passive approach for detecting and monitoring FFSNs. The proposed technique is based on an extensive analysis of DNS traffic collected over a 45-day period at a two large ISP networks. The benefit of this passive detection approach is that the attacker is unaware of any ongoing monitoring activity and therefore does not take preventative steps in order to conceal normal botnet operation.

DGA detection: In [1] the authors propose a DGA-based botnet detection system named *Pleiades*. The proposed system however, operates at ISP level whereas our system aims to discover bots present at the enterprise or local area network level. Similar to our detection approach, Pleiades makes use of DNS features including *NXDomain* responses in order to identify DGA-based botnets.

3 Proposed approach

We propose a passive DNS analysis approach whose aim is to detect and alert network administrators to an early-stage botnet presence on a local area network. In contrast to blacklist- or signature-based detection methods the proposed approach is capable of detecting networks infected with hitherto unseen malware samples. The approach makes use of features related to the ASN (Autonomous System Number) associated with the IP address(es) returned by a DNS query. These features are described below and we explore their efficacy in the classification task in Section 5.

3.1 Autonomous systems

An *Autonomous System* (AS) is a connected group of one or more IP prefixes, blocks of class A, B or C networks, under the control of a network operator, or multiple operators sharing a routing policy [12]. Each AS is assigned a unique *Autonomous System Number* (ASN). For example, both *Dublin City University* and *NUI Galway* networks are part of *ASN 1213 HEAnet Limited. ASN 1213* consists of several network prefixes e.g. 136.206.0.0/16 (DCU), 140.203.0.0/16 (NUIG) and 134.226.0.0/16 (TCD). An AS is not restricted to a single country and may contain geographically dispersed locations.

3.2 Analysis of benign domains

In order to train a classifier to distinguish between malicious and benign domains it is imperative to identify pertinent features to aid in their differentiation. To this end we initiated an analysis of benign domain DNS behaviour by assembling a set of domains obtained from the *Alexa Top 500*[1]. The latter lists the most popular worldwide domains on a monthly basis and we assume them non-malicious. Also, as they are in high demand they require the capability to reliably serve a large number of users and as such often employ legitimate techniques for load distribution, load balancing and fault tolerance such as Content Delivery Networks [7] and Round Robin DNS [3]. Analysis of such domains will help to determine whether it is possible to distinguish domains utilizing such legitimate techniques from those using FFSN for malicious purposes.

Sample benign DNS data was generated by querying the top 500 domains over a 24-hour period and collecting all DNS responses. (The experiment was repeated several weeks later in order to confirm findings remained valid.) Each domain was queried several times over a 24-hour period in July 2017 and all returned DNS data was collected. From each DNS response a number of features were extracted. Of specific interest for our experiment was the ASN associated with each IP address returned in each DNS response.

Analysis revealed that 457 domains returned one or more IP addresses that were associated with a single ASN. Only 38 domains returned IP addresses that

[1] https://www.alexa.com/topsites

were associated with more than one ASN. (Five domains returned no IP address.) A similar analysis of FFSN domains is presented below for comparison. Since a botnet creator's intention is to infect as many machines as possible he/she often exercises little to zero control over their location. We therefore expect bots to spread across multiple countries, Internet Service Providers and ASNs.

3.3 Analysis of FFSN domains

IP addresses belonging to a Fast-Flux Service Network consist of compromised machines acting as a proxy for any device attempting a connection with its C&C server. Unlike the legitimate network resources belonging to a single organisation, an FFSN cannot contain itself within the bounds of a single ASN or network operator. As a result, we expect the number of distinct ASNs associated with an FFSN domain to be higher than that of a benign domain. (Previous work in this area [14] showed that with one DNS resolution per hour an FFSN could reach up to 800 distinct ASNs over a three month period.)

Our analysis of benign domains revealed a maximum of three ASNs associated with the IP addresses returned in any single DNS response. In contrast, a single DNS response for one FFSN domain, *hfgdgfhghfhd.net*, contained IP addresses spanning 10 distinct ASNs. Our own 24-hour trace of two FFSNs, *hfgdgfhghfhd.net* and *hjdhgsfdgfjdsd.net*, revealed that FFSN domains contain significantly higher counts of distinct ASNs compared to benign domains, with *hjdhgsfdgfjdsd.net* reaching 89 and *hfgdgfhghfhd.net* reaching 164 distinct ASNs over the trace period. In contrast to benign domains, due to constantly changing set of bots available to act as a proxy servers, the overall ASN count for an FFSN domain will inevitably expand over time. We therefore decide to make ASN-related features available to our classifier.

3.4 Other features

Bots employing DGA to establish a connection with their C&C server will algorithmically generate a set of domain names to connect to. Since relatively few of the generated domains will actually be registered or alive, the bot will attempt to resolve multiple non-existent domains before landing on a valid one. We can thus expect bots employing DGA to generate multiple *NXDomain* DNS responses. In our feature set we therefore include features related to the presence and frequency of NXDomain DNS responses.

A final malicious indicator, not listed or used in the previous work we have reviewed, is the querying by malware of an external DNS server directly. Internal hosts located on an enterprise or local area network generally direct their DNS queries to one of a set of predefined network-internal DNS servers. If the DNS server is unable to answer a host's query, it passes it to next appropriate server in the DNS hierarchy in a recursive fashion. However, if an internal host attempts to bypass the local DNS server and queries an external source directly, it might be an indication bot activity. This feature is added to our feature set.

4 Evaluation and results

4.1 Data collection

In order to collect authentic, malware-generated DNS traffic an enterprise network was simulated and infected with a selection of botnet malware. The evaluation set included malware samples belonging to known malware families, such as *Dreambot*, *Hancitor*, *Trickbot* and *Zeus*. The network was simulated on a *Windows Server 2012 R2 Datacenter*, with 24GB of RAM, running *Hyper-V Server Virtualisation Software*. The simulated network consisted of a DNS server, File and Printer Sharing Server, several end user machines running Windows 7 or Windows 10 and a monitoring machine running *Kali Linux*. The machines were connected via a simulated switched network and a mirroring port was configured so the Kali monitor received a copy of all traffic generated by end user machines. Intercepted traffic was logged using *tcpdump* and saved in *pcap* format for later analysis.

In contrast to other systems for dynamic analysis whose aim is to inspect large volumes of malware samples over a short period of time (usually few minutes) our environment monitored each malware instance for at least 20 minutes, in some cases up to 8 hours. This allowed for more in-depth analysis of malicious network traffic, increasing the probability of capturing C&C communication (as the initial connection might will not always occur in first minutes immediately following bot infection).

The advantage of using a virtualised environment is the ability to take *snapshots* of each machine on the network in a known good state. After infecting a machine and capturing generated network packets, each machine can be conveniently rolled back to its previous safe state. Rollback of the entire network can be completed within minutes.

The primary disadvantage of a virtualised environment however is the inability to run all malware samples. We observed that upon activation a malware sample often performed checks in order to detect whether it was executing in a virtual environment and altered its behaviour if so (halting execution or deleting itself). We were therefore limited to working with malware samples that executed in a virtual environment as well as third party malware packet traces. All malware samples used were acquired either from VxStream Sandbox[2] or from malicious links sent in spam campaigns to a large financial institution in Ireland.

While a virtual environment was necessary to safely collect malware traffic, for benign domains *tcpdump* was used to capture DNS answers returned in response to queries for Alexa Top 500 domains as described in Section 3.

4.2 Feature extraction

A total of 100 network traffic captures were generated. 50 packet captures were derived from the malware running on the virtual network and 50 from querying

[2] https://www.payload-security.com/products/vxstream-sandbox

the benign domains of the Alexa Top 500. From each packet capture the following DNS-related features were extracted:

- Maximum number of ASNs returned for a domain in a single DNS response
- Maximum number of ASNs returned for a single domain across all DNS responses
- NXDomain response frequency
- External DNS usage frequency

4.3 Results

A Naive Bayes classifier trained on the above features in the Weka machine learning framework [11] achieved a 65% classification accuracy. 44 of 50 benign traffic captures were correctly classified as benign with 6 incorrectly classified as malicious. 21 of 50 malicious traffic captures were correctly classified as malicious with 29 incorrectly classified as benign. Further analysis of the results and malware revealed that while some botnets made use of FFSN (leading to high ASN counts) others did not. Those botnets making use of FFSNs were correctly classified as malware by our classifier while other, arguably simpler, botnets were misclassified. Suggestions for addressing the resultant high false negative rate are presented below.

5 Conclusions and future work

We presented a prototype botnet detection system that leverages passive DNS traffic analysis to detect botnet-infected local area networks. Our primary aim was to explore the effectiveness of ASN-related features extracted from DNS traffic in characterising benign versus malicious domains. Experimental evaluation of a simple classifier trained on a minimal set of DNS-related features that included ASN-related data has successfully demonstrated the effectiveness of the approach. The significant role played by a domain's ASN count in malicious domain classification is an interesting finding.

The proposed approach, in relying on a minimal set of features, suffers from a high false negative rate. Future work should investigate whether the addition of supplementary features can address this shortcoming. Since a botherder is unable to control the availability or uptime of an infected machine, the Time-To-Live (TTL) value for each DNS record in an FFSN is expected to be low. Thus, adding TTL-related features to the model may reduce the false negative rate. A *PTR record* [17] is used for configuration of reverse DNS and resolves a given IP address to a domain name. Legitimate networks frequently have PTR records configured for all IP addresses in their pool. On the contrary, infected machines not part of a registered domain will typically have no associated PTR records. Failed reverse DNS lookup for a given IP address may be an indication of a bot. Two final interesting domain name properties, related to DNS traffic, are *WHOIS Registration Date* and *Modified Date*. Since domain names generated

by DGA-malware can be registered by a botherder even on the same day of an expected botnet communication, a recent *WHOIS Registration Date* could serve as an indicator of a potentially malicious domain. If a host on the network attempts to connect to numerous recently created or modified domains, such behaviour may be indicative of a bot presence.

References

1. Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou, N., Abu-Nimeh, S., Lee, W., Dagon, D.: From throw-away traffic to bots: Detecting the rise of DGA-based malware. In: USENIX security symposium. vol. 12 (2012)
2. Bilge, L., Kirda, E., Kruegel, C., Balduzzi, M.: Exposure: Finding malicious domains using passive DNS analysis. In: NDSS (2011)
3. Brisco, T.: DNS support for load balancing. RFC 1794, RFC Editor (April 1995), `https://tools.ietf.org/rfc/rfc1794.txt`
4. Choi, H., Lee, H.: Identifying botnets by capturing group activities in DNS traffic. Computer Networks 56(1), 20–33 (2012)
5. Choi, H., Lee, H., Kim, H.: Botgad: Detecting botnets by capturing group activities in network traffic. In: Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middlewaRE. pp. 2:1–2:8. COMSWARE '09, ACM, New York, NY, USA (2009)
6. Dagon, D., Gu, G., Lee, C.P., Lee, W.: A taxonomy of botnet structures. In: Twenty-Third Annual Computer Security Applications Conference (ACSAC 2007). pp. 325–339 (Dec 2007)
7. Farber, D.A., Greer, R.E., Swart, A.D., Balter, J.A.: Internet content delivery network (Nov 25 2003), US Patent 6,654,807
8. Govil, J., Govil, J.: Criminology of botnets and their detection and defense methods. In: 2007 IEEE International Conference on Electro/Information Technology. pp. 215–220 (May 2007)
9. Govil, J.: Examining the criminology of bot zoo. In: 2007 6th International Conference on Information, Communications Signal Processing. pp. 1–6 (Dec 2007)
10. Grill, M., Nikolaev, I., Valeros, V., Rehak, M.: Detecting DGA malware using NetFlow. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM). pp. 1304–1309 (May 2015)
11. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD explorations newsletter 11(1), 10–18 (2009)
12. Hawkinson, J.: Guidelines for creation, selection, and registration of an Autonomous System (AS). RFC 1930, RFC Editor (March 1996), `https://tools.ietf.org/rfc/rfc1930.txt`
13. Holz, T., Gorecki, C., Rieck, K., Freiling, F.C.: Measuring and detecting fast-flux service networks. In: NDSS (2008)
14. Hsu, C.H., Huang, C.Y., Chen, K.T.: Fast-flux bot detection in real time. In: RAID. pp. 464–483. Springer (2010)
15. Khattak, S., Ramay, N.R., Khan, K.R., Syed, A.A., Khayam, S.A.: A taxonomy of botnet behavior, detection, and defense. IEEE Communications Surveys Tutorials 16(2), 898–924 (Second 2014)
16. Kolias, C., Kambourakis, G., Stavrou, A., Voas, J.: DDoS in the IoT: Mirai and other botnets. Computer 50(7), 80–84 (2017)

17. Mockapetris, P.: Domain names - implementation and specification. RFC 1035, RFC Editor (November 1987), https://tools.ietf.org/rfc/rfc1035.txt
18. Nazario, J., Holz, T.: As the net churns: Fast-flux botnet observations. In: 2008 3rd International Conference on Malicious and Unwanted Software (MALWARE). pp. 24–31 (Oct 2008)
19. Ono, K., Kawaishi, I., Kamon, T.: Trend of botnet activities. In: 2007 41st Annual IEEE International Carnahan Conference on Security Technology. pp. 243–249 (Oct 2007)
20. Passerini, E., Paleari, R., Martignoni, L., Bruschi, D.: Fluxor: Detecting and monitoring fast-flux service networks. In: International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. pp. 186–206. Springer (2008)
21. Perdisci, R., Corona, I., Dagon, D., Lee, W.: Detecting malicious flux service networks through passive analysis of recursive DNS traces. In: 2009 Annual Computer Security Applications Conference. pp. 311–320 (Dec 2009)
22. Rossow, C., Dietrich, C.J., Bos, H., Cavallaro, L., Van Steen, M., Freiling, F.C., Pohlmann, N.: Sandnet: Network traffic analysis of malicious software. In: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. pp. 78–88. ACM (2011)
23. Salusky, W., Danford, R.: Know your enemy: Fast-flux service networks. The Honeynet Project pp. 1–24 (2007)
24. Silva, S.S., Silva, R.M., Pinto, R.C., Salles, R.M.: Botnets: A survey. Computer Networks 57(2), 378–403 (2013)
25. Villamarin-Salomon, R., Brustoloni, J.C.: Identifying botnets using anomaly detection techniques applied to DNS traffic. In: Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE. pp. 476–481. IEEE (2008)
26. Wu, J., Zhang, L., Liang, J., Qu, S., Ni, Z.: A comparative study for fast-flux service networks detection. In: The 6th International Conference on Networked Computing and Advanced Information Management. pp. 346–350 (Aug 2010)
27. Yadav, S., Reddy, A.K.K., Reddy, A.L.N., Ranjan, S.: Detecting algorithmically generated domain-flux attacks with DNS traffic analysis. IEEE/ACM Transactions on Networking 20(5), 1663–1677 (Oct 2012)
28. Yadav, S., Reddy, A.N.: Winning with DNS failures: Strategies for faster botnet detection. In: International Conference on Security and Privacy in Communication Systems. pp. 446–459. Springer (2011)