Predicting the Origin of DNS Traffic Without Reference to Source Address

Joe Abley

1

***Abstract*—We propose a predictive system which is able to identify the originating system responsible for stateless Domain Name System (DNS) traffic received at a authoritative DNS server, without reference to the source address. The ability to predict whether particular DNS query traffic received at an authoritative server is legitimately sourced from a particular client system is useful in identifying some classes of malicious traffic in production DNS systems.**

***Index Terms*—Domain Name System, Machine learning, Predictive models**

I. Introduction

The Domain Name System (DNS) [1] includes a wire protocol [2] with which structured requests and responses are exchanged over a network. The DNS protocol is specified and widely used over IPv4 and IPv6 using the UDP transport protocol. Since UDP/IP is stateless there is no way for the receiver of a DNS request sent over UDP to verify the legitimacy of a source address. If it is possible for a system A to originate DNS requests towards a DNS server B with the source specified as the address of a third system C, then the response from B can be directed sent to C. Since DNS responses are usually larger in size than DNS requests, this represents an amplification attack on C by A with B acting as amplifier and is generally referred to as a reflection attack [3].

Some reflection attacks are relatively straightforward to identify and to mitigate since they follow naïve patterns. For example, DNS zones have a predictably-large bundle of resource records attached to the owner name corresponding to the zone apex: the base specification requires *SOA* and *NS* records; various commonly-used services require *TXT* records as part of their domain authentication processes and DNSSEC adds a *DNSKEY* resource-record set (RRSet) and *RRSIG* resource records (RRs) to every RRSet. A DNS request sent to an authoritative-server for a particular zone with *QTYPE=ANY* and *QNAME* set to the zone apex will result in a significant degree of amplification, and many early attacks followed this attack signature. Since queries with *QTYPE=ANY* are rarely required for reasons other than manual troubleshooting, however, a trivial mitigation is simply not to support them [4].

Despite the ease at which obvious amplification attacks can be anticipated and mitigated on authoritative DNS servers, the potential for a server to be used as an amplifier using patterns that are more difficult to predict remains, and the lower amplification potential may be offset by the difficulty in identifying the attack. By choosing query parameters that match legitimate, real-world use of the DNS, the attacks may seem impractical to block without collateral damage. This is especially true of amplification attacks against DNS resolvers.

The goal of this project is to construct a test can predict whether to a set of DNS queries received by an authoritative DNS server were sent by a particular resolver system based on relevant metrics derived from the queries but notably *not* the source address in the IP header. This capability would pave the way for a systematic, data-based approach to the identification of reflection attacks.

II. Infrastructure: Afilias, Google and INFO

Afilias operates authoritative DNS servers for the INFO top-level domain, as well as many others. These servers are made available to the Internet through commodity transit arrangements from multiple locations using anycast [5]. A significant originator of requests received by these servers is Google Public DNS[1]. Afilias and Google recently deployed private network interconnects (PNIs) between their

This project proposal was prepared for course 9603, Department of Electrical and Computer Engineering, Faculty of Engineering, Western University, Fall 2018, and includes data collection carried out on infrastructure operated by Afilias. The general approach of inferring ground truth from clean channels in DNS traffic in order to build predictive models is included in a patent for which application was made by Afilias to the USPO on 2018-03-06.

[1] https://developers.google.com/speed/public-dns/

networks in five locations globally which caused all query-response traffic between Google and Afilias to move off the Internet and onto the PNIs.

The nature of the PNIs is such that Afilias can have high confidence that all queries received over the PNIs are authentically sourced by Google. Correspondingly, traffic received at an Afilias node from the Internet, directed at anycast INFO servers which Afilias has made available over the PNI can be said with high confidence not to have been sent from Google. Both of these classifications are true without reference to the source address observed on any DNS query.

The PNI traffic and the Internet traffic in such a scenario can be used as ground truth to construct time-series datasets, and those derived datasets can then be used to train a classifier. This classifier can then be used against arbitrary DNS query sample sets to predict whether or not the query samples originate with Google or not, and can be tested against different traffic sets whose true origin is known with certainty.

III. Possible Applications

Traffic received over the Internet that purports to be from Google but which is predicted not to be from Google can be isolated and itself be subject to classification. Being able to classify known-bad traffic from packet captures after the fact is useful in forensic analysis of strange traffic patterns (e.g. noticeable spikes in traffic volume) and potentially also for use in real-time if the matching criteria can be distilled down into something with low cost to execute, like a Bloom filter.

Finding traffic from Google arriving from an unexpected direction also has the potential to inform operational management systems, e.g. causing an alarm in the network operations centre that could prompt an escalate to peers at Google to investigate, since it might be indicative of a routing problem.

IV. Related Work

Sebastian Castro and Jing Qiao at InternetNZ[2] used a transformation of discrete packet captures to time-series data based on 66 features and trained a classifier to distinguish between resolvers and other DNS clients (which they refer to as monitors) for traffic received at the .NZ nameservers [6].

The predictions made by the Castro-Qiao model have been verified and the model seems to exhibit reasonable accuracy. In particular there seem to be a sufficient number of features included in that model to distinguish between resolvers and other DNS clients in general, and hence there is some reason to believe this is a reasonable starting point.

V. Approach

A. Raw Dataset Collection

Packet captures to be obtained from Afilias in compressed pcap format over a suitable sample period, representing all DNS queries that arrived at a particular node (with destination address matching one particular anycast DNS server) over (a) the PNI and (b) the Internet. It seems possible (likely, even) that an aggregate set of queries from (presumably) multiple sources will not provide a reasonable source of ground truth for not-Google; it is possible, for example, that the classifier will in effect be distinguishing between single query sources and combined queries from multiple sources. In that case the Internet-sourced data can be reduced to isolate queries from particular sources.

B. Derivation of Time Series Dataset

Individual parameters in a DNS query like *QNAME*, *QCLASS*, *QTYPE*, *RD*, *DO*, can be counted over regular time intervals to produce a time series of vectors. The string quantity *QNAME* can be reduced to a numeric value by transforming it into a numeric measure of string similarity with a constant string (Castro-Qiao used Jaro-Winkler string distance). Entropic measures can also be included, such as those derived from the set of elapsed time between successive queries within the interval.

C. Feature Engineering

I propose to take the full Castro-Qiao set of features and use them as a starting point. The comparison being made in this study is different from theirs and hence we expect different redundancy between features, and other features to be required in order to distinguish reliably between Google's behaviour and that of other clients. Verification of the final feature set will be performed using a clustering algorithm and a suitable evaluation metric as described by Castro-Qiao.

Feature engineering [7] seems likely to be the most important component of this study, and it is expected that some iteration around feature sets will be required.

D. Construction of Supervised Classifier

I propose to use AUTO-SKLEARN [8] to find the right learning algorithm and to optimise its hyperparameters. The resulting classifier instantiated in Python can then be used against other datasets.

E. Testing of Supervised Classifier

The classifier produced can be tested against other query sets that are known to originate in Google Public DNS, and also against query sets that are known to originate elsewhere. The raw data for those comparison sets can be drawn from different time periods from the same infrastructure, or from different nodes in the Afilias network, or perhaps for query sets for nameservers other than those serving .INFO.

F. Assess Usefulness of Approach

It's by no means certain that this approach will yield a useful classifier, e.g. because the feature engineering requires more work, or because different resolvers are just too similar to distinguish between without using query source addresses. Google Public DNS is sufficiently different in scale, implementation and in its population of end-users that it's reasonable to imagine there might be some distinguishing characteristics, however.

References

[1] P. Mockapetris, "Domain names - concepts and facilities," Internet Requests for Comments, RFC Editor, STD 13, November 1987. [Online]. Available: http://www.rfc-editor.org/rfc/rfc1034.txt

[2] ——, "Domain names - implementation and specification," Internet Requests for Comments, RFC Editor, STD 13, November 1987. [Online]. Available: http://www.rfc-editor.org/rfc/rfc1035.txt

[3] J. Damas and F. Neves, "Preventing use of recursive nameservers in reflector attacks," Internet Requests for Comments, RFC Editor, BCP 140, October 2008. [Online]. Available: http://www.rfc-editor.org/rfc/rfc5358.txt

[4] J. Abley, O. Gudmundsson, M. Majkowski, and E. Hunt, "Providing minimal-sized responses to dns queries that have qtype=any," Working Draft, IETF Secretariat, Internet-Draft draft-ietf-dnsop-refuse-any-07, August 2018. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-dnsop-refuse-any-07.txt

[2] InternetNZ is the manager of the .NZ country-code top-level domain.

[5] J. Abley and K. Lindqvist, “Operation of anycast services,” Internet Requests for Comments, RFC Editor, BCP 126, December 2006. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4786.txt

[6] J. Qiao, “Source address classification - feature engineering,” accessed: 2018-09-16. [Online]. Available: https://blog.nzrs.net.nz/source-address-clustering-feature-engineering/

[7] J. Brownlee, “Discover feature engineering, how to engineer features and how to get good at it,” accessed: 2018-09-16. [Online]. Available: https://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/

[8] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter, “Efficient and robust automated machine learning,” in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2962–2970. [Online]. Available: http://papers.nips.cc/paper/5872-efficient-and-robust-automated-machine-learning.pdf