SIDN Labs

https://sidnlabs.nl

February 12, 2016

Peer-reviewed Publication

Title: ENTRADA: Enabling DNS Big Data Applications

Authors: Maarten Wullink, Moritz Müller, Marco Davids, Giovane C. M. Moura, and Cristian Hesselman

Venue: APWG Symposium on Electronic Crime Research (eCRIME 2016), Toronto, ON, Canada

Conference dates: June 1, 2, and 3, 2016.

Citation:

- M., Wullink, Müller, M., Davids, M., Moura, G.C.M., Hesselman, C.: ENTRADA: Enabling DNS Big Data Applications. In: APWG Symposium on Electronic Crime Research (eCRIME 2016), Toronto, ON, Canada. June 1, 2, and 3, 2016.
- Bibtex:

```
@inproceedings{sidn-ecrime-2016,
author = {{Maarten Wullink,  Moritz Muller, Marco Davids,
Giovane C. M. Moura,  and Cristian Hesselman}},
booktitle={{APWG Symposium on Electronic Crime Research
(eCRIME 2016), Toronto, ON, Canada.
June 1, 2, and 3, 2016}},
title={{ENTRADA: Enabling DNS Big Data Applications}},
year={2016},
month={June},
}
```

ENTRADA: Enabling DNS Big Data Applications

Maarten Wullink, Moritz Müller, Marco Davids, Giovane C. M. Moura, and Cristian Hesselman
SIDN Labs
Stichting Internet Domeinregistratie Nederland (SIDN)
Arnhem, The Netherlands
Email: {firstname.lastname}@sidn.nl

***Abstract*—DNS operators, TLD registries, hosting providers, and other Internet operators are frequently faced with the same question: how to draw insights and knowledge from their respective network traffic data in order to improve their services, security, and operations? "Big data" processing solutions play a major role as an enabling platform in this sense, especially with the increasing growth of the volume of Internet traffic.**

With this in mind, we have developed and presented in a previous work ENTRADA, an open-source high-performance Hadoop-based data streaming warehouse designed to both ingest continuous streams of data and deliver interactive response times over large datasets, even in a small cluster. Whereas in the previous study we focused on the architecture and performance evaluation, in this paper we present a series of use cases and applications that cover phishing, botnets, email security, and visualizations. These applications can be directly used by DNS operators, TLD registries, and researchers to quickly analyze their network data and improve their services, security, and operations.

I. Introduction

DNS operators, TLD registries, hosting providers, and other Internet operators are frequently faced with the same question: how to leverage their network traffic data in order to improve their security, services, and operations?

Before undertaking such task, a key element is to have the required computational resources to perform such analysis, especially with the constant increase in Internet traffic. For example, at SIDN, the domain name registry of the Netherlands (.nl), the traffic to and from our .nl DNS authoritative servers in `pcap` format comprises roughly 1 TB/day. Worse, in many data analysis cases, it is necessary to analyze *longitudinal data*, i.e., data collected over long periods of time (months and/or years), and it may be necessary also to test various hypothesis.

Performing *efficient* data analysis on such large datasets, with interactive response times (within seconds or minutes) is a major challenge. To cope with such scenarios, researchers and operators have resorted to computer cluster-based solutions as a way to achieve better performance, scalability, and dependability [1], [2], [3], [4], [5]. Such clusters are often based on Apache Hadoop [6] or other non-relational databases (NoSQL) [7].

For the specific case of the DNS traffic, other solutions exist (e.g.: [8], [9]), but they are either not open source or they do not meet our performance requirements. To cope with the performance requirements and lack of specific tools for big data analytics on DNS traffic, we have developed and made open source ENTRADA (ENhanced Top-level domain Resilience through Advanced Data Analysis). ENTRADA is capable of analyzing the equivalent of 53 TB of `pcap` files in under 3.5 minutes, even in a cluster of just 4 data processing nodes. It delivers such performance by converting `pcap` files to Apache Parquet [10], which is a query-optimized format, and by employing Impala, a multi-parallel SQL-like query engine [11], both of which are also open source. We make ENTRADA available at [12]. Moreover, we have evaluated its performance in [13].

At SIDN, we use ENTRADA to store and analyze the DNS traffic we receive at the authoritative DNS servers of .nl, which is the country-code Top-Level Domain (ccTLD) of the Netherlands. We use it as an *enabling platform* for supporting applications that further improve both security and stability of the .nl zone. ENTRADA has been operational for more than 1.5 years (as of January 2016), having stored more than 100 billion DNS query/response pairs.

This paper complements our previous studies [13], [14] by (i) providing a more detailed coverage on the deployment of ENTRADA and (ii) presenting a series of applications that developed atop ENTRADA, which can be adapted by DNS operators improve security and stability of their zone and by researchers and Internet operators to better understand their respective network traffic data.

The rest of this paper is divided as follows: first, in Section II, we cover the background on DNS and top-level domains. Then, in Section III, we cover the ENTRADA architecture as well as its deployment at SIDN Labs. The following two sections cover four types of applications that we have developed with ENTRADA: two applications to detect phishing and other malicious domains (Section IV) and two other applications for detecting botnets based on DNS query patterns (Section V). After that, we show how ENTRADA can be used to evaluate the adoption and usage of two DNS-based security standards for email security: DKIM and DMARC (Section VI). We show two other visualization applications – Phishing Campaigns and DNS Open Data – in Section VII. We summarize this work in Section IX.

II. Background

ENTRADA was developed at SIDN, which is the registry [1] of the Dutch ccTLD .nl. SIDN manages the authoritative DNS

[1] See a complete list of registries here https://www.iana.org/domains/root/db

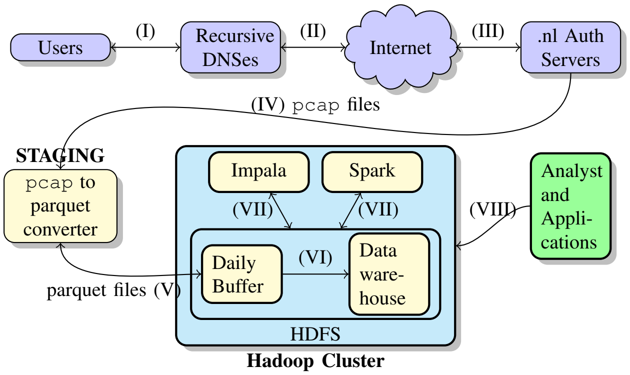


Fig. 1. ENTRADA data sequence flow

servers for .nl and handles the domain registration process of currently over 5.5 million domain names.

Authoritative DNS data and domain resolution: so far, ENTRADA is developed, but not limited, to store and process DNS related traffic and therefore, the applications presented in this paper rely on DNS traffic as well. For that reason, it is helpful to have a basic understanding of the domain name resolution process.

In Figure 1, we show, simplified, the resolution of a domain name followed by the steps carried out by ENTRADA. In this example, a user wants to reach a website (e.g. www.example.nl). Therefore, the user's computer must first resolve the domain name www.example.nl to an IP address. The computer includes a lightweight DNS stub resolver that connects to a recursive resolver (I in Figure 1), usually located in a ISP network, and asks which IP address is assigned to www.example.nl. The recursive resolver doesn't know the answer, and therefore initiates a recursive process first asking one of the root servers (the "." zone, II in Fig. 1, not shown) for the IP address of the servers authoritative for .nl.

As soon as the recursive resolver receives the response, it sends another query to one of the authoritative .nl servers and asks for the IP address of example.nl (III in the same figure). Again, the authoritative server responds with the IP address for another name server, which is authoritative for www.example.nl and which knows its IP address. After receiving the final IP address, the recursive resolver forwards this IP address back to the stub resolver and the user can connect to the website. We refer the reader to [13] for more details.

In order to reduce the number of queries sent by the recursive resolver, local caches are used, which store queried domain responses [15]. As a consequence, not every query initiated by a stub resolver gets forwarded to the authoritative name servers of .nl. As a result, only a share of the total number of queries for .nl domain names is stored in ENTRADA and processed by the applications introduced in this paper. This behavior is inherent to the way the DNS works.

The following section goes into more detail, how ENTRADA processes, stores and makes this data available.

III. ENTRADA Architecture

In a nutshell, ENTRADA consists of three main parts, which are also shown in Figure 1:

1) Network traffic converter (pcap to Parquet converter): converts pcap data to Parquet [10] file format, which we developed at SIDN Labs.
2) Apache Hadoop [16]: an open source distributed storage and processing framework. While Hadoop is divided into four main modules (common, Hadoop Distributed Filesystem (HDFS), YARN, and MapReduce), we only use the first two.
3) High performance query engine: Impala [11] and Spark [17], or any Parquet compatible engine that reads files directly from the HDFS.

We make a clear distinction between ENTRADA and its applications: ENTRADA is an enabling platform that provides functionalities for other applications (VII in Fig.1). These applications, in turn, are considered apart of ENTRADA.

Next we cover the main components of ENTRADA.

A. Network traffic converter

We have discussed in [13] our data model and data preprocessing. In this subsection, we summarize the main concepts, referring the reader to our previous work for an extensive discussion.

Network traffic data analysis with native pcap is a CPU and I/O intensive task. For example a simple count of distinct IP addresses for 100 TB of pcap data, would require reading and parsing all the data. To deal with this problem over other types of datasets, Google developed Dremel [18], a query system for analysis of read-only nested data that delivers aggregation queries (e.g., averages) for trillion-row tables in seconds. Dremel combines multi-level execution trees and columnar data storage [19].

Interactive response times are a must for any data streaming warehouse (DSW), such as ENTRADA. In our case, we convert pcap files to Apache Parquet [11] format, which is based on Dremel. Besides enabling fast aggregation query response times, Parquet employs encoding algorithms such as run length, dictionary and delta encoding on entire columns, since they have same-type values, reducing storage requirements. From the appr. 85 GB of daily pcap data per authoritative server, Parquet and Snappy, a compression algorithm we employ, compress it to appr. 6 GB (after also filtering some fields, as we discuss in [13].

Converting DNS Requests/Responses: To increase performance and to reduce the complexity when performing analysis of DNS data with ENTRADA, we created a data model in which every DNS request is matched with its respective response and stored as a single record. This solves the issue of having to match and join the DNS query and response during the analysis phase. We cover this data model in more detail in [12].

Besides that, we also enrich the respective model with metadata such as the autonomous system number and country of the IP packet source address.

We have developed a converter for DNS, UDP, TCP, IP and ICMP network data, which we make available at [12]. However, ENTRADA is not limited to these network protocol models: it can be easily extended to any structured data format, such as syslog files and Netflow data.

B. Apache Hadoop and HDFS

ENTRADA is designed to work on top of the Apache Hadoop distributed processing framework. Besides the common tools, we use only the HDFS component from Hadoop. HDFS allows to create a single logical distributed file system across all available data-nodes. This filesystem is automatically managed and distributed across the available data nodes, regardless of the number of nodes.

We store our converted Parquet files on HDFS and employ triple data redundancy (HDFS' default, 3 copies of each data block). Hadoop ensures the block copies are stored on distinct physical data nodes.

C. Query Engines

As we also covered in [13], we chose to use Cloudera Impala [11] as a query engine for several reasons. First, Impala is an open source massively parallel processing (MPP) SQL query engine, which enables analysts to make use of high-level SQL queries to analyze the data, while it handles all the data processing needed to retrieve the data from the respective data nodes.

To access these Parquet files stored on the HDFS, we create an SQL table with Impala that (i) represents the internal Parquet data model and (ii) provides a mechanisms to access all existing and the future added Parquet files as a single table, independently of the number of files.

The Impala tables have read-only access to the data and even deleting the table will not affect the Parquet data files. Having a data format which is independent of the query engine prevents an engine lock-in and allows us to use other query engines with Parquet support such as Apache Spark [17], which includes support for machine learning applications and SQL-like data access.

D. ENTRADA Deployment

Our ENTRADA deployment consists of a Cloudera [20] Hadoop on-premise cluster of 6 physical nodes (Figure 2). We employ three types of nodes:

- Data nodes: where data is actually stored and replicated using HDFS. Besides that, each data node runs an Impala daemon. Any Impala daemon can be used to submit SQL queries to, this daemon will then assume the role of orchestrator and distributes query fragments to available Impala daemons on the other data-nodes. We currently use four data nodes and two more will be added soon.
- Metadata node: it uses a PostgreSQL database for storing metadata about Impala tables and cluster configuration. We configured Cloudera Hadoop to store all cluster configuration details in PostgreSQL. Impala metadata such as table descriptions is stored in the PostgreSQL database.

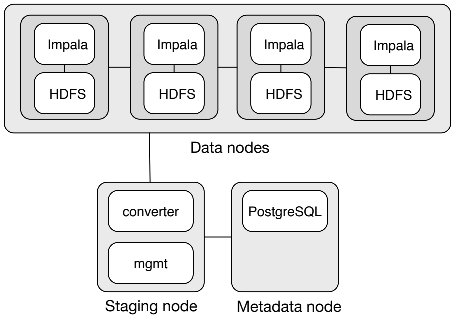


Fig. 2. ENTRADA deployment architecture at SIDN Labs.

- Staging node: receives the `pcap` data from the name servers and is responsible for converting this `pcap` data to Parquet and for Hadoop cluster management

We use the data nodes for storage and query execution. They have the following hardware configuration: a single 6-core 1.9GHz CPU with 128 GB of RAM and 6 TB of storage. Both metadata and staging nodes have the same CPU and memory resources as the data nodes, with 2 TB of storage instead. The combined cluster storage capacity is 24 TB. It is possible to store an estimated 150 billion database rows, where each row contains a DNS query and response pair.

Currently, our ENTRADA instance stores all the DNS traffic from two .nl authoritative servers, starting from May 2014. This corresponds to appr. 25% of the total DNS traffic to all the .nl authoritative servers (∼400 M daily queries stored). We refer the reader to [13] for more details and to [21] for a detailed discussion of our data privacy framework that conforms to both EU and Dutch laws.

Besides a deployment on physical hardware, ENTRADA can also be deployed using cloud services. It is however important to emphasize that Hadoop data replication assumes every hard disk is a physical disk, which may not hold for cloud services. We refer the reader to [22] for more on these specific concerns.

E. Performance

ENTRADA delivers interactive response times by (i) employing Parquet [11], an optimized column-based file format based on Google's Dremel [18], and (ii) by using Impala [11], an open source massively parallel processing SQL query engine, as can be seen in Figure 1. For aggregation type queries (e.g., calculating the average packet size), it takes 3.5 minutes to process the equivalent of 52 TB of `pcap` in a small 4 data node cluster [13]. This performance would be virtually impossible to achieve using the same hardware and the `pcap` format.

F. Source Code and Tutorial

We make the ENTRADA source code available at [12] and provide a tutorial, examples, and various guidelines for deploying an ENTRADA cluster.

IV. Detection of Malicious and Phishing Domains

To reduce the success of phishing attacks, the industry has developed a range of detection tools and provides blacklists that contain malicious URLs or domains. In many cases, these blacklists are compiled after a URL has been verified by a human operator or detected in an email message. For example, the Netcraft phishing blacklist [23] accepts user submissions through a browser toolbar.

ENTRADA enables DNS operators, TLD registries and security researchers to analyze query patterns in order to detect phishing attacks. In this section, we cover two applications we have developed for this use case: SIDekICk (Section IV-A) and nDEWS (Section IV-B). Both are used to improve the security of the .nl zone.

A. *SIDekICk*

Most phishing sites are hosted on compromised servers using existing domain names, but a significant and dangerous subset of all phishing domains consists of phishing domains registered solely for a malicious purpose [24]. For example, domain names imitating banks and credit card companies. These domains are known to have a larger number of DNS queries to authoritative servers in the very first hours after their creation [25].

We have observed the same behavior for the .nl zone using the ENTRADA platform. The graph on the right of Figure 3 shows the average number of DNS requests that we received at one of the .nl authoritative name servers for all phishing domains reported by Netcraft [23] (Jan–Aug 2015), while the graph on the left shows the same metric for a random sample of 20,000 newly created domain names (Jan-March 2015).

Based on this query pattern, we have developed SIDekICk (SuspIcious DomaIn Classification) [26], an ENTRADA application that attempts to automatically detect newly registered phishing domains. SIDekICk integrates two data sources: (i) registration data and (ii) DNS queries to the .nl authoritative name servers, which are stored in ENTRADA.

SIDekICk employs a supervised classifier, which requires a labeled training set during initialization. For this purpose, we have chosen a training set with phishing domains obtained from Netcraft, which consisted of a total of 1,900 domain names covering a period of 7 months (from December 2014 until May 2015). For each domain d, we retrieved the following features: geographic distribution of resolvers querying d, total number DNS queries for d, measurements of query growth of d over a period of three weeks, and whether we were able to observe a spike in queries on the day d got reported as a phish.

With the help of the training data, the SIDekICk classifier generates a decision tree classifier model [27], which automatically weighs the different features and selects the most significant ones on its own. SIDekICk observes newly created domain names for three consecutive days. On each day, the features are collected and classified by the trained decision tree.

After the training phase, we evaluated SIDekICk for a period of 31 days. In this period, we used it to analyze 61,100 newly registered domain names. In the same period, Netcraft reported 10 phishing domains for .nl. SIDekICk, on the other hand, reported 22 phishing domains and 11 domains hosting other malicious content such as concocted web shops selling fake products. The false positive rate of SIDekICk is 0.3 %, which accounts for only 200 of over 61,000 domains being mistakenly classified as malicious. This suggests that SIDekICk is able to support, or even replace, traditional phishing detection tools for detecting newly registered phishing domains. See [26] for more details on this research.

B. *nDEWS*

Another ENTRADA application we have developed is nDEWS (new Domains Early Warning System), which employs the same data sources as SIDekICk but with a different classification algorithm to detect suspicious domains. We have presented nDEWS in [14], which we summarize here.

nDEWS uses ENTRADA to retrieve the following features for each domain d added to the .nl zone: for each day the total number of DNS requests for d ($\sum Req$), total number of unique source IP addresses querying for d ($\sum IPs$), unique countries from which queries for d originate ($\sum CC$), and unique Autonomous Systems (ASes) from which queries for d originate ($\sum ASes$). As shown Figure 3, malicious domains are likely to have higher values for these features.

Unlicke the SIDekICk classifier, nDEWS employs the k-means clustering algorithm [28], which aims at partitioning the dataset into n clusters in a way that it minimizes the total distance between the data points and the cluster's corresponding centroid. The advantage of k-means is that it does not require any a-priori knowledge about the domains and does not require labeled datasets/training, which allows it to better cope with seasonal/diurnal patterns [29].

We classify every new domain into two clusters: "suspicious" and "normal". Suspicious ones are those domains that have higher values for the aforementioned features. On average, we detected 12.2 new suspicious domain names every day. Among the suspicious domains, there were phishing domains as reported by the Netcraft [23] anti-phishing feed, domains that distributed malware as reported by Google Safe Browsing and VirusTotal [30], [31], but also benign domains that received a large number of queries caused, for example as a result of viral content spread through social networks (false positives).

Currently, nDEWS is in a pilot phase. We notify two major .nl registrars, on a daily basis, of domains that have been classified as suspicious and are registered by their respective customers, so they can take appropriate action based on this information and provide feedback on the accuracy of the nDEWS notifications back to us.

C. *Discussion*

Both SIDekICk and nDEWS illustrate how ENTRADA enables us to easily build security applications to detect

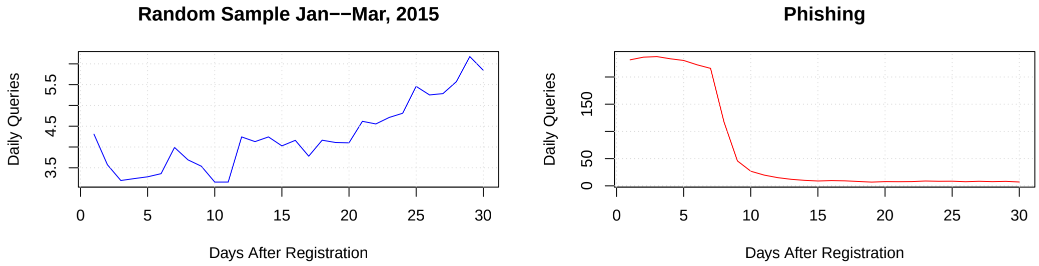


Fig. 3. .nl Random vs Phishing new domains average daily queries

malicious activity. Traffic patterns can be analyzed using real-time and historical data without human intervention, which enables the detection of newly registered phishing domains in near real-time. Also, by deploying these applications at the TLD-level, we can keep track of the whole zone – a vantage point that is not accessible for traditional phishing detection techniques. Additionally, the described classifiers nDEWS and SIDekICk can be extended easily, in order to detect not only newly registered domain names, but also domain names that are compromised, for example through a vulnerable content management system (CMS).

V. Botnet Detection

Botnets are used for a vast array of malicious activities: sending spam, distributed denial-of-service (DDoS) attacks, and hosting malware, among others. Independent of the botnet architecture, one common feature is the need for a command-and-control (C&C) server, which is a host from which the bots can retrieve instructions [32]. Some botnets use a C&C infrastructure based on domain names and before bots can communicate with the C&C server, they need to resolve its domain name. These DNS requests may be observed with ENTRADA as well. In this section, we introduce two ENTRADA applications for detecting botnet domain names based on (i) NXDOMAIN responses (Section V-A) and on (ii) specific resolver characteristics (Section V-B).

A. Detection through NXDOMAINs

To avoid detection, some botnets employ Domain Generating Algorithms (DGA) to generate thousands of random domains that each bot may use to contact a C&C server [33], where the algorithm may change every day. DGA-based botnets are designed to make it more difficult to take down the botnet. Earlier generations often used hard-coded domain names to enable bots to contact the C&C server. Now, the cost of registering thousands or millions domains of potential C&C domains generated with a DGA every day, make preemptive registration prohibitive.

As a consequence, DGA-based bots will issue many queries to randomly generated domains of which only a few will actually exist. Therefore, the vast majority of requests for DGA-based domain names will result in the DNS authoritative name server responding with a so-called ”non-existing domain answer” (NXDOMAIN) [34].

We developed an ENTRADA application that analyzes the DNS queries received at the .nl authoritative name servers to determine whether a domain name is a potential C&C server. For each domain name d not present in the .nl zone (NXDOMAIN), we compute, every day, the number n of requests. We sort the list of domain names by the number of requests in descending order and analyze the top domain names. By doing this, we were able to identify 17 DGA botnet C&C server domain names over the past 1.5 years. We registered them and configured the DNS records to point to our sinkhole, in order to prevent them from being registered by others and to be able to analyze the traffic between the bots and the C&C server.

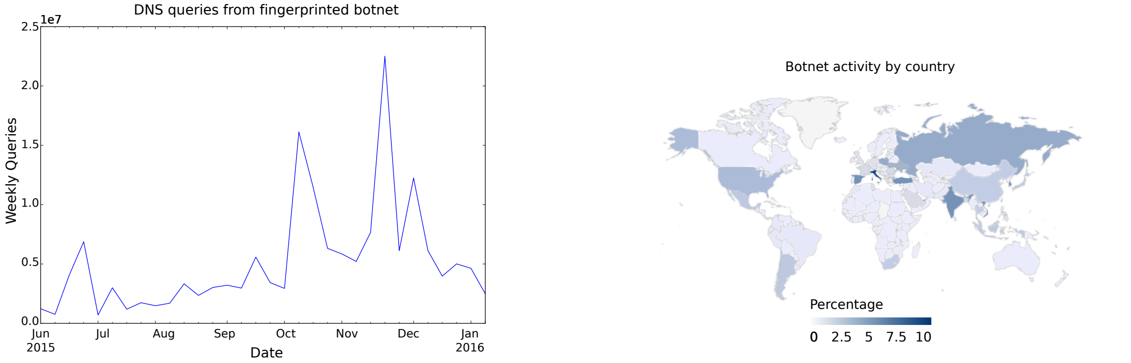
B. Fingerprint-based Botnet Detection

The .nl authoritative name servers receive queries from millions of distinct resolvers. We see resolvers that are operated by large ISPs sending thousands of queries each hour, small resolvers of home users which only occasionally send queries, or network probes querying our servers for measurement purposes. The behavioral patterns of these different classes of resolvers often differ from each other wildly. Whereas most resolvers show regular query behaviour adhering to the DNS standards as defined by the IETF, some resolvers display a rather unusual behavioral pattern. For example, some resolvers send a disproportional high volumes of queries followed by NXDOMAIN responses. Others query for domain names for which a name server is not authoritative, only use TCP for their queries, or use a limited range of source ports and query IDs. Some of this behavior is the result of misconfigured resolvers, others have their origin in malicious activity.

We use these characteristics to create a fingerprint for resolvers that are used for malicious activities. For example, botnets that send out spam, typically attempt to resolve the mail address of their victims beforehand.

We have identified the unique characteristics of resolvers used by a single botnet and use ENTRADA to continuously analyze their DNS queries [2]. We can detect this particular

[2] We cannot disclose those publicly or its respective botnet herders would change it.



(a) Weekly queries from fingerprinted botnet resolvers

(b) Geographic activity from fingerprinted botnet resolvers

Fig. 4. Characteristics of fingerprinted botnet resolvers

botnet activity almost immediately and as soon as we detect a resolver matching the botnet fingerprint we flag the corresponding IP address and save it in a database.

Then, we share information about IP addresses geolocated to the Netherlands with the Abuse Information Exchange (AbuseHUB) [35]. Members of this platform include large Dutch ISPs, which use the information to cleanup the botnet infections located within their network. With this system we are able to actively disrupt the distribution of spam-mail and other malicious activity.

Figure 4a shows the total number of DNS queries from bots of this particular botnet over the last 6 months. From this graph we can derive that the activity of the bots strongly varies over time and we can identify larger spam-runs in a timely manner. Also, we can see in Figure 4b that most botnet clients are located outside the Netherlands. This differs from what we would expect for the average .nl domain name, which is targeting mostly the Dutch market. This characteristic could additionally be used to identify infected machines.

Of course, it is important that due to DHCP churn effects – i.e., the ratio in which a user has his IP address changed by its ISP – , it is hard to estimate the *actual* number of bots [36]. However, since we provide both IP address and timestamp to AbuseHUB (which forwards this to the respective ISP), the ISP can use its DHCP logs to identify the infected user using the detected IP address.

C. Discussion

DNS-based botnet detection with ENTRADA allows for light-weight and quick detection and in comparison with other traditional botnet detection mechanisms like [37], it scales well with large DNS traffic data sets. We have shown two example applications, one that detect DGA-based domains and another that targets a specific botnet. It may also be possible to detected other DNS-based botnets by using their distinct DNS query characteristics.

VI. Email Security

In order to cope with email spoofing (e.g.: a rogue mail server impersonating legitimate ISP/email providers to carry out spam campaigns) and improve overall email security, the IETF standardized a series of DNS-based authentication and authorization mechanisms. As part of our registry role, we are interested in measure the adoption and usage of these standards in the .nl zone. We next summarize these standards.

DomainKeys Identified Mail (DKIM) [38] is a DNS-based email authentication method. It works as follows: whenever a mail server A receives an email message from `example.nl`, it can verify its authenticity by validating it with the sender public key published in the DNS server authoritative for `example.nl`. The public key for a domain is published as a TXT record. Using this key to validate the DKIM-signature in the email header, A can tell whether the message is indeed sent by the mail server of example.nl.

The second DNS-based email security standard is the Sender Policy Framework (SPF) [39]. SPF is used to specify which hosts are authorized to send email for a particular domain name. SPF uses a specially formatted TXT record in the DNS zone of the sender. For example, `192.168.0.0/24` for `example.nl`.

Finally, the third standard is Domain-based Message Authentication, Reporting, and Conformance (DMARC) [40], which builds on DKIM and SPF. A domain name owner creates one or more DMARC policies, which describe the action that should be taken when an email does not pass DKIM or SPF validation. For example, it can be a simple `p=reject` to indicate that all email that fails to conform to DKIM and SPF should be discarded. DMARC policies are also published using the DNS TXT record type.

Next we show how we can easily measure the adoption of DKIM and DMARC using ENTRADA and SQL.

A. Standards Usage from a TLD perspective

Given we have access to a global view for the zone we maintain (.nl), we can infer the adoption rates of these standards by analyzing the volume of DNS queries we receive for these particular types of records. This analysis requires multiple steps.

The first step consists of identifying what type of DNS queries and parameters are used by each of these standards. All three standards publish their information as a TXT record type in their respective authoritative servers. In specific:

- Both DKIM and DMARC use special subdomains: DKIM uses the `_domainkey` and DMARC uses the `_dmarc` subdomain format. Examples of DKIM and DMARC records are `mail._domainkey.example.nl` and `_dmarc.example.nl` respectively.
- SPF: for SPF, there is no standard format for subdomains as with DMARC and DKIM. Therefore, at the .nl authoritative server, we observe only a DNS query for TXT record types. Since we do not publish the SPF records for the domains (their authoritative name servers do), we cannot determine which TXT queries are specifically asking for SPF policies.

The second step consists of writing SQL queries that can be issued to Impala to retrieve the respective data. We cover this next.

B. DKIM and DMARC Usage in .nl

To determine the usage of DMARC for the domains in the .nl zone, we use the ENTRADA SQL query in Listing 1. The same query can be adapted to DKIM by replacing `_dmarc` with `_domainkey` in the `qname` parameter.

In this query, we select TXT records (`qtype=16`) and group the results by month. For this example, we use data from only one authoritative server (`server`), but the query can be trivially extended to a set of name servers. We make sure to filter out TXT queries for non existing domains and failed queries with the predicate `rcode=0` (NO ERROR).

```
select month, count(distinct domainname)
from dns.queries
where qtype=16
and qname like '_dmarc.%' and rcode=0
and year=_year
and month=_month
and server="ns1.dns.nl"
group by month
```

Listing 1. SQL query for distinct DMARC domain

We used data covering an 18 month period, with a SQL query for each month of data, which is equivalent to 2.5 TB of `pcap` data, which takes roughly 3 minutes for ENTRA to process. Performing this analysis on native `pcap` files it would take more than 3 minutes just to decompress the files.

Figure 5 shows the results. The first one, Figure. 5a, shows a timeseries of the number of distinct domains queried in the analyzed period. As can be seen, there has been an increase for this metric over the last 18 months, which suggest that DKIM and DMARC are increasingly being used.

However, it is also important to compare this growth with the growth of the .nl zone in terms of numbers of registrations, in order to tell whether the growth is related to the growth of the zone or not. To do that, we normalized the numbers from Fig. 5a with regards to the total number of domains in the .nl zone that have received DNS queries for email records (MX records). This is shown in Figure 5b. The Y axis, shows the percentage with regards to the .nl zone file (total number of domains). We see that, in fact, 27% of the .nl domains that receive queries for MX records, also receive DMARC queries, and 14.7% employ DKIM.

However, we need to keep in mind that these are numbers on the fraction of the domains that get queries for DKIM and DMARC records. This is an indication of the adoption of these technologies, but it does not tell how many email messages are ultimately protected by these standards. A DKIM validating email server will only retrieve a DKIM public key from the DNS if the DKIM header is present in a received email. These DNS queries can therefore be used to measure DKIM deployment for .nl domain names. DMARC however, does not use a special email header, a mail server with DMARC support will try to validate every received email. DMARC queries for a domain name therefore not to prove that the queried domain name supports DMARC. We can however use the queries to infer the desire of email recipients for performing DMARC email validation. Also, as discussed in Section II, caching at the resolver reduces the actual number of queries we observe.

It is also important to notice that not every domain with an MX record has DMARC, DKIM, and/or SPF records. To measure the percentage of the domains that have these records requires active scanning of the DNS, which is the scope of the OpenINTEL project of which we are also partners [41] and has been carried out by [42] as well.

C. DMARC and DKIM queries sources

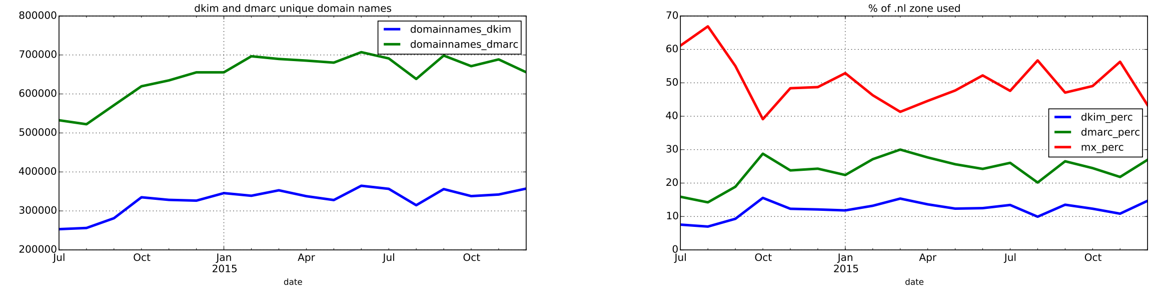
To determine from which countries most DKIM and DMARC queries originate, we employ the SQL query shown in Listing 2. This single query allows us to produce a single value for each country for the entire 18 month period dataset. As discussed in Section III, this information was added by geolocating the IP source address of the DNS request, using the Maxmind [43] IP geolocation database.

```
select country, count(1)
from dns.queries
where qtype=16
and (qname like '%_domainkey.%'
or qname like '_dmarc.%') and rcode=0
and ((year=2014 and month>6) or year=2015)
and server="ns1.dns.nl"
group by country
```

Listing 2. DMARC/DKIM queries grouped by countries

The query shown in Listing 2 took 23.5 minutes to be executed in our setup. We show the top 10 countries as source of DMARC and DKIM queries to .nl in Table I. As can be seen, 89,9% of all DNS queries originate from only the top 4 countries. Even though .nl is focusing on the Dutch market, the Netherlands surprisingly rank 3rd behind the United States and Ireland. By analyzing the US and IE queries we found that this is caused by the presence of large email providers such as Google, Microsoft and Yahoo in those countries.

To identify from which networks these queries originate, we employ the query shown in Listing 3, which groups the results by their respective Autonomous System Number (ASN). This query also took 23.5 minutes to execute, as we show in Table II. As can be seen, 85,4% of the DNS queries for DKIM and DMARC records are performed by large email providers such as Google, Microsoft, Yahoo, and AOL. The ”UNKN” AS is



(a) Unique domain names used for DMARC and DKIM queries.

(b) Usage relative to the .nl zone size.

Fig. 5. DKIM and DMARC usage in the .nl zone.

Country	# queries	percentage
US	208,533,790	42.60
IE	84,515,235	17.26
NL	79,052,717	16.15
BE	67,963,161	13.88
FI	9,112,053	1.86
RU	7,306,873	1.49
DE	7,119,556	1.45
GB	5,897,734	1.20
CN	5,446,895	1.11
DK	2,958,891	0.60

TABLE I
COUNTRY DISTRIBUTION

Provider	# ASN	# queries	percentage
Google	AS15169	302,465,578	61.79
Microsoft	AS8075	51,556,416	10.53
Unknown	UNKN	15,788,699	3.22
AOL	AS1668	12,971,456	2.65
Yahoo	AS36647	112,83,129	2.30
Yahoo	AS26101	101,24,857	2.07
Yahoo	AS36646	9,150,523	1.87
Yahoo	AS34010	45,22,388	0.92
IDC China Tel	AS23724	4,520,819	0.92
Mail.ru	AS47764	3,659,097	0.75

TABLE II
AUTONOMOUS SYSTEM DISTRIBUTION

used for situations where the Maxmind geolocation database contains no mapping between the IP address and autonomous system number.

```
select asn, count(1) as tot
from dns.queries
where qtype=16
and (qname like '%_domainkey.%'
or qname like '_dmarc.%') and rcode=0
and ((year=2014 and month>6) or year=2015)
and server="ns1.dns.nl"
group by asn
order by tot desc
```

Listing 3. SQL query for AS query count

D. Parallel Queries and Optimization

The dataset we used for this research consists of 18 months of name server data for one of the .nl authoritative name servers and contains 73 billion rows. Each row in the dataset is a composite of the original DNS query and its corresponding DNS response.

With Impala, each SQL query runs as a single thread on each data node. Since each data node has 6-cores, we would be underutilizing ENTRADA's capabilities this way. To improve performance, the SQL query shown in Listing 1, queries one month of data at a time. In this way, instead of having a single query for an 18 month period, we run multiple queries, each for for 1 month of data simultaneously, resulting in multiple parallel threads, one per month, on the cluster. This improves the performance significantly.

E. Discussion

Our results indicate that the usage of both DKIM and DMARC in the .nl zone has been increasing slowly over a period of 18 months. Especially large (free) mail providers seem to be leading in the adoption of these technologies. However, there is still plenty of room for more usage. For further research, it would be interesting to find out if other TLD operators observe the same behavior.

VII. VISUALIZATION

Many of the ENTRADA applications described above rely on the identification of specific patterns in DNS traffic and DNS packets. Visualizing this data is an important first step because it enables ENTRADA users such as researchers and network engineers to identify trends and suspicious activities.

This highlights a more general strength of ENTRADA, which is its capability to allow its users to easily explore large sets of data and extract interesting characteristics and statistics. Data stored in ENTRADA can be accessed and extracted through multiple APIs, such as the impala-shell, through Python, using Impyla [44], or via the Java Database Connectivity interface. As a result, ENTRADA works with any tool of choice.

In this section we first show the visualization of phishing domains. Then, we show how we use ENTRADA to produce open aggregated datasets on DNS traffic on the .nl zone, which are daily updated and can be found at [45].

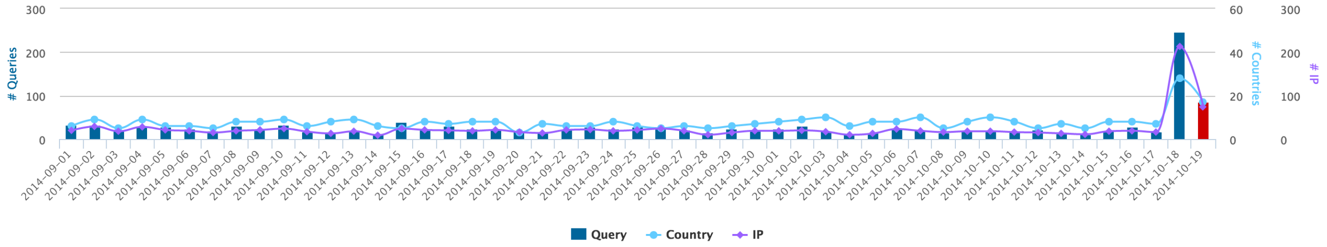


Fig. 6. DNS security scoreboard: query pattern of a domain involved in phishing. The red bar corresponds to the day it was detected by Netcraft.

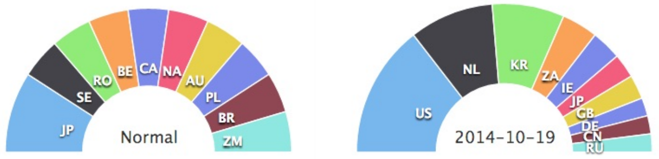


Fig. 7. DNS security scoreboard: shift on the geographical distribution of the countries querying for a phishing domain

A. Phishing Campaigns

For DNS operators, it is of the essence to be able to quickly visualize security events in their available datasets. We have developed a visualization web application that employs ENTRADA, which we refer to as the "DNS security scoreboard". This application produces visualizations of the query patterns of compromised domains utilized in phishing campaigns within the .nl zone.

The "DNS security scoreboard" works as follows: (i) first obtain two phishing blacklists: Netcraft's Anti-phishing feed [23] and PhishTank [46]. After that, we select only the .nl domains (since we only have DNS data for those) and retrieve, (ii) for each domain d involved in the phishing campaigns, the following features: total number of DNS requests ($\sum Req$), total number of unique source IP addresses ($\sum IPs$), unique countries ($\sum CC$), and unique Autonomous Systems (ASes) ($\sum ASes$), observed at our .nl authoritative servers. These metrics are then combined with the phishing report and stored in a PostgreSQL database. We then (iii) use a RESTful web-application built with Javascript and the Highcharts visualization library [47] which retrieves the data from PostgreSQL and produces the visualizations.

Figure 6 shows a time series for a compromised .nl domain. As can be seen, prior to the notification day (red bar), the domain has a relatively stable (diurnal/weekly [29]) pattern for the respective features. However, one day before the notification, we can observe an increase of the features – probably coinciding with the start of the "phishing" campaign.

Figure 7 shows the distribution of countries for the same domain. The left figure shows the distribution for the 30 day period before the domain was added to a blacklist, while the right figure shows the distribution on the notification day, there is a clear change in the distribution.

These visualizations help with identifying features that can be later utilized in algorithms aimed at detecting malicious domains, such as in [26].

B. DNS Open Data and Visualization

We also make a series of visualizations and aggregated datasets available to the public at our .nl stats site [45]. These stats include the number of registered domains in the .nl zone, graphs on the DNS query type, response code, DNSSEC, and many others. The site uses ENTRADA and updates itself automatically on a daily basis. All the datasets are open and are longitudinal (starting from May 2014).

Our stats site and data sets can be used by researchers, registries and other Internet operators to understand the .nl zone as well as to compare it with their own observations.

VIII. Related Work

There have been several research works aiming at improving the performance of analysis of large data sets. However, to the best of our knowledge ENTRADA is the first platform that uses off-the-shelf open source software to implement a Data Streaming Warehouse (DSW). While providing similar features, an existing solution such as DataDepot [1] only uses customized, closed-source software. The open source DSW DBStream [2] uses PostgreSQL as a query engine, whereas ENTRADA employs the off-the-shelf Impala query engine and Parquet file format based on Google's Dremel [18]. Even though a thorough benchmark still has to be carried out, a first comparison already indicates that ENTRADA outperforms DBStream: With DBStream, it takes more than 50 minutes to analyze a 640 GB raw dataset on a 10-node cluster. In contrast, with ENTRADA it takes less than 3.5 minutes to analyze almost 4 times more compressed data (2.2 TB) on a 6-node cluster. This work complements our previous work [13] by showing a series of applications that can build atop ENTRADA.

Analysis of off-line network traffic is presented by works such as [3], [4], [5]. They store snapshot data in a Hadoop cluster for short-term analysis. ENTRADA is designed to be append only, which means that new data is continuously appended but not updated.

Turing [8] is a commercially available closed-source solution for DNS big data analytics, developed by Nominet, the registry for .uk domain names. There is no publicly available information about the technical implementation of Turing. The Turing developers chose not to use Hadoop or any other open source NoSQL database but to develop custom storage and computation layers. A direct comparison between ENTRADA and Turing still needs to be carried out, but is difficult without publicly available information on Turing.

The `pcap` to Parquet converter software we developed for ENTRADA is partly based upon the Hadoop PCAP library developed by RIPE-NCC [9]. This library is used to analyze `pcap` data with Hadoop, using MapReduce [48] or MapReduce based Apache Hive [49] which are both designed for batch oriented processing. The start-up overhead of these batch oriented processes result in a high overhead compared to ENTRADA which uses Impala and Parquet.

IX. Summary and Future Work

This work complements our previous work [13], in which we focused on the data model, architecture, and performance evaluation of ENTRADA, a Hadoop-based data streaming warehouse that we have developed and have made available in [12].

In this paper, we present and discuss seven ENTRADA applications that we use on a daily basis. We have shown two applications to detect malicious and phishing domains [14], two applications to detected botnets using DNS authoritative data, and one application to measure the adoption of DNS-based standards to improve email security. Moreover, we have shown how visualization applications as well can be developed on top of the ENTRADA platform.

We have used ENTRADA for more than 1.5 years and used it with applications like the ones discussed here, to improve both the stability and security of the .nl zone. We hope the findings and insights presented here can help TLD registries, researchers, and other Internet operators to analyze their network traffic data. As future work, we will continue developing ENTRADA and new ENTRADA-based applications.

Acknowledgments: The authors would like Jelte Jansen for his valuable comments.

References

[1] L. Golab, T. Johnson, J. S. Seidel, and V. Shkapenyuk, “Stream Warehousing with DataDepot,” in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD ’09. New York, NY, USA: ACM, 2009, pp. 847–854.

[2] A. Bar, A. Finamore, P. Casas, L. Golab, and M. Mellia, “Large-scale network traffic monitoring with DBStream, a system for rolling big data analysis,” in *Big Data (Big Data), 2014 IEEE International Conference on*, Oct 2014, pp. 165–170.

[3] T. Vanhove, G. Van Seghbroeck, T. Wauters, F. De Turck, B. Vermeulen, and P. Demeester, “Tengu: An experimentation platform for big data applications,” in *Distributed Computing Systems Workshops (ICDCSW), 2015 IEEE 35th International Conference on*, June 2015, pp. 42–47.

[4] J. Liu, F. Liu, and N. Ansari, “Monitoring and analyzing big traffic data of a large-scale cellular network with Hadoop,” *Network, IEEE*, vol. 28, no. 4, pp. 32–39, July 2014.

[5] Y. Lee and Y. Lee, “Toward Scalable Internet Traffic Measurement and Analysis with Hadoop,” *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 1, pp. 5–13, Jan. 2012.

[6] T. White, *Hadoop: The Definitive Guide*. O’Reilly Media, Inc., 2009.

[7] N. Leavitt, “Will NoSQL Databases Live Up to Their Promise?” *Computer*, vol. 43, no. 2, pp. 12–14, Feb 2010.

[8] Nominet, “turing by Nominet - New Insights into DNS traffic,” http://www.nominet.uk/products-services/turing-by-nominet/, 2015.

[9] RIPE NCC, “Hadoop PCAP library,” https://github.com/RIPE-NCC/hadoop-pcap, 2015.

[10] Apache, “Apache Parquet,” https://parquet.apache.org/, 2015.

[11] M. Kornacker, A. Behm, V. Bittorf, T. Bobrovytsky, C. Ching, A. Choi, J. Erickson, M. Grund, D. Hecht, M. Jacobs *et al.*, “Impala: A modern, open-source SQL engine for Hadoop,” in *Proceedings of the Conference on Innovative Data Systems Research (CIDR’15)*, 2015.

[12] SIDN Labs, “ENTRADA homepage,” http://entrada.sidnlabs.nl/, 2015.

[13] Maarten Wullink, Giovane C. M. Moura, Müller, M, and Cristian Hesselman, “ENTRADA: a High Performance Network Traffic Data Streaming Warehouse,” in *Network Operations and Management Symposium (NOMS), 2016 IEEE (to appear)*, April 2016. [Online]. Available: https://www.sidnlabs.nl/downloads/sidn-noms2016_EN.pdf

[14] Giovane C. M. Moura, Moritz Muller, Maarten Wullink, and Cristian Hesselman, “nDEWS: a New Domains Early Warning System for TLDs,” in *IEEE/IFIP International Workshop on Analytics for Network and Service Management (AnNet 2016), co-located with IEEE/IFIP Network Operations and Management Symposium (NOMS 2016)*, April 2016. [Online]. Available: https://www.sidnlabs.nl/downloads/presentations/sidn-annet2016.pdf

[15] M. Andrews, “Negative Caching of DNS Queries (DNS NCACHE),” RFC 2308, Internet Engineering Task Force, Mar. 1998.

[16] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, “The Hadoop Distributed File System,” in *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, May 2010, pp. 1–10.

[17] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark: cluster computing with working sets,” in *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, 2010.

[18] S. Melnik, A. Gubarev, J. J. Long, G. Romer, S. Shivakumar, M. Tolton, and T. Vassilakis, “Dremel: Interactive Analysis of Web-scale Datasets,” *Proc. VLDB Endow.*, vol. 3, no. 1-2, pp. 330–339, Sep. 2010.

[19] M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. Madden, E. O’Neil, P. O’Neil, A. Rasin, N. Tran, and S. Zdonik, “C-store: A column-oriented dbms,” in *Proceedings of the 31st International Conference on Very Large Data Bases*, ser. VLDB ’05. VLDB Endowment, 2005, pp. 553–564.

[20] Cloudera, “Cloudera Hadoop,” http://cloudera.com/, 2015.

[21] C. Hesselman, J. Jansen, M. Wullink, K. Vink, and M. Simon, “A privacy framework for DNS big data applications,” Tech. Rep., 2014. [Online]. Available: https://www.sidnlabs.nl/downloads/whitepapers/SIDN_Labs_Privacyraamwerk_Position_Paper_V1.4_ENG.pdf

[22] Cloudera, “Cloudera Enterprise Reference Architecture for AWS deployments,” http://www.cloudera.com/content/www/en-us/documentation/other/reference-architecture/PDF/cloudera_ref_arch_aws.pdf, 2015.

[23] Netcraft, “Phishing Site Feed,” http://www.netcraft.com/anti-phishing/phishing-site-feed/, 2015.

[24] APWG, “Global Phishing Survey: Trends and Domain Name Use in 1H2014,” http://docs.apwg.org/reports/APWG_Global_Phishing_Report_1H_2014.pdf, 2014.

[25] Hao, Shuang and Feamster, Nick and Pandrangi, Ramakant, “Monitoring the initial dns behavior of malicious domains.”

[26] M. Müller, “SIDekICk: SuspIcious DomaIn Classication in the .nl Zone,” Master’s thesis, University of Twente, the Netherlands, 2015. [Online]. Available: http://eprints.eemcs.utwente.nl/26196/

[27] Scikit-learn, “Scikit-learn documentation 1.10: Decision trees,” http://scikit-learn.org/stable/modules/tree.html, 2015, retreived 2015-07-08.

[28] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[29] L. Quan, J. Heidemann, and Y. Pradkin, “When the Internet Sleeps: Correlating Diurnal Networks with External Factors,” in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ser. IMC ’14. New York, NY, USA: ACM, 2014, pp. 87–100.

[30] Google, “Google safe browsing api,” https://developers.google.com/safe-browsing/, January 2015.

[31] VirusTotal, “About VirusTotal,” https://www.virustotal.com/en/about/, 2015.

[32] E. Cooke, F. Jahanian, and D. McPherson, “The zombie roundup: understanding, detecting, and disrupting botnets,” in *Proceedings of the*

Steps to Reducing Unwanted Traffic on the Internet Workshop. Berkeley, CA, USA: USENIX Association, 2005, pp. 6–6.

[33] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou II, S. Abu-Nimeh, W. Lee, and D. Dagon, "From throw-away traffic to bots: Detecting the rise of dga-based malware." in *USENIX Security Symposium*, 2012, pp. 491–506.

[34] P. Mockapetris, "Domain names - implementation and specification," RFC 1035, Internet Engineering Task Force, Nov. 1987.

[35] Abuse Information Exchange, "Abuse Information Exchange," https://www.abuseinformationexchange.nl/english, Oct 2015.

[36] G. C. M. Moura, C. Gañán, Q. Lone, P. Poursaied, H. Asghari, and M. van Eeten, "How Dynamic is the ISPs Address Space? Towards Internet-Wide DHCP Churn Estimation," in *Networking Conference, 2015 IFIP*, 2015.

[37] R. Villamarín-Salomón and J. C. Brustoloni, "Identifying botnets using anomaly detection techniques applied to dns traffic," in *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*. IEEE, 2008, pp. 476–481.

[38] D. Crocker, T. Hansen, and M. Kucherawy, "DomainKeys Identified Mail (DKIM) Signatures," RFC 6376 (INTERNET STANDARD), Internet Engineering Task Force, Sep. 2011.

[39] S. Kitterman, "Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1," RFC 7208 (Proposed Standard), Internet Engineering Task Force, Apr. 2014, updated by RFC 7372.

[40] M. Kucherawy and E. Zwicky, "Domain-based Message Authentication, Reporting, and Conformance (DMARC)," RFC 7489 (Informational), Internet Engineering Task Force, Mar. 2015.

[41] OpenINTEL, "OpenINTEL Open Access," 2015. [Online]. Available: http://openintel.nl/

[42] Z. Durumeric, D. Adrian, A. Mirian, J. Kasten, E. Bursztein, N. Lidzborski, K. Thomas, V. Eranti, M. Bailey, and J. A. Halderman, "Neither snow nor rain nor mitm...: An empirical analysis of email delivery security," in *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*. ACM, 2015, pp. 27–39.

[43] Maxmind, "Maxmind," 2015. [Online]. Available: http://www.maxmind.com/

[44] Cloudera, "Python DB API 2.0 client for Impala and Hive," https://github.com/cloudera/impyla, 2015.

[45] SIDN Labs, ".nl stats and data: Insight into the use of .nl," http://stats.sidnlabs.nl/, 2016.

[46] PhishTank, "PhishTank: Join the Fight Against Phishing," 2016. [Online]. Available: http://www.phishtank.com

[47] Highcharts, "Highcharts Javascript librrary," http://www.highcharts.com/, 2015.

[48] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.

[49] Apache Hive, "Apache Hive," https://hive.apache.org/, 2015.